

Generalizable Representations for Cross-Lingual Transfer Learning

Zur Erlangung des akademischen Grades einer
Doktorin der Ingenieurwissenschaften

von der KIT-Fakultät für Informatik des
Karlsruher Instituts für Technologie (KIT)
genehmigte

Dissertation

von

Danni Liu

Tag der mündlichen Prüfung: 21. November 2025

1. Referent: Prof. Dr. Jan Niehues

2. Referent: Prof. Dr. Alexander Fraser

Abstract

Multilingual models are essential in natural language processing (NLP), the field of enabling computational models to process and generate human language. These models are crucial for expanding information access across linguistic boundaries, with machine translation models being particularly central to this effort. However, with approximately 7,000 languages spoken worldwide, it is infeasible to acquire sufficient training data for every language. This data scarcity challenge highlights the importance of cross-lingual generalization, the ability to transfer knowledge learned from some languages to settings with little or no training data. Such generalization depends on a model’s ability to learn internal representations that are shared and consistent across languages.

While the *classical* form of cross-lingual generalization involves training a model for a task in some languages and evaluating it on others, its application in machine translation is more complex as the input and output languages are inherently different. The directional nature of machine translation calls for *compositional* generalization, which requires the model to combine knowledge from trained directions (e.g., English-to-German and German-to-French) to directly perform translation for unseen directions (e.g., English-to-French). Moreover, the pre-training fine-tuning paradigm introduces a crucial *temporal* dimension to generalization. This perspective concerns how well a model’s pre-trained capabilities are preserved after specializing on selected tasks or languages. This thesis addresses these three aspects of generalization: classical, compositional, and temporal.

First, we investigate compositional generalization in translation models trained from scratch. We demonstrate that a primary reason why multilingual translation models fail to generalize compositionally is that their internal representations become highly entangled with information specific to the input language. We identify that this entanglement is reinforced by the residual connections within the model’s architecture. By proposing a method that relaxes this structural constraint, we achieve substantial improvements on unseen translation directions.

We then extend the investigation to the temporal dimension of generalization by incorporating pre-trained models. We observe that despite large-scale multilingual pre-training, a model’s internal representations often remain highly specific to the input language. Moreover, when these models are specialized for particular translation directions, they are prone to catastrophic forgetting, which substantially degrades their existing capabilities. To counteract this, our work evaluates several established mitigation strategies from the literature and introduces a novel approach. Our proposed method produces a more interpretable view of the model’s internal representations, although this benefit comes with a trade-off in translation quality.

Our investigation then extends to large language models (LLMs), which operate as general-purpose instruction-following systems. For these models, which typically use a decoder-only architecture, we first identify the middle layers as the most effective positions for intervention. Building on this finding, we propose a modular and computationally efficient post-hoc method to enhance language invariance in the model’s representations. This approach leads to substantial improvements in both classical cross-lingual transfer scenarios and the compositional generalization cases examined earlier.

Lastly, we revisit classical cross-lingual transfer with a novel application for attribute-controlled machine translation. This method trains a classifier to distinguish text attributes in languages with available labeled data and then transfers this capability to guide translations for other languages lacking such resources. The approach operates on the model’s internal activations rather than weights, making it suitable for scenarios where model weights must remain fully frozen. When compared to an oracle setup that updates the model weights, our activation-based method demonstrates stronger robustness against domain shifts.

Each of the contributions above is situated within a model modification scope that aligns with evolving practices in the field, ranging from training models from scratch to performing post-hoc interventions on large pre-trained models. This thesis advances the development of multilingual NLP models that not only achieve strong performance under trained conditions but also acquire more generalizable and adaptable underlying representations.

Zusammenfassung

Die Verarbeitung natürlicher Sprache (Natural Language Processing, NLP) befasst sich mit der Entwicklung von Methoden, die Computermodellen die Verarbeitung und Generierung menschlicher Sprache ermöglichen. Mehrsprachige Modelle, insbesondere solche für die maschinelle Übersetzung, spielen eine besondere Rolle, da sie den Informationszugang über sprachliche Grenzen hinweg erweitern. Bei weltweit etwa 7.000 gesprochenen Sprachen ist es jedoch nicht realisierbar, für jede Sprache ausreichende Trainingsdaten zu beschaffen. Diese Herausforderung der Datenknappheit betont die Relevanz der kreuzlingualen Generalisierung, der Fähigkeit, in einigen Sprachen erlerntes Wissen auf Kontexte mit wenigen oder keinen Trainingsdaten zu übertragen. Eine solche Generalisierung hängt von der Fähigkeit eines Modells ab, interne Repräsentationen zu lernen, die über Sprachen hinweg geteilt und konsistent sind.

Während die *klassische* Form der kreuzlingualen Generalisierung darin besteht, ein Modell für eine Aufgabe in einigen Sprachen zu trainieren und es auf anderen zu evaluieren, ist ihre Anwendung in der maschinellen Übersetzung komplexer, da die Eingabe- und Ausgabesprachen inhärent verschieden sind. Der gerichtete Charakter der maschinellen Übersetzung erfordert eine *kompositionelle* Generalisierung, welche vom Modell verlangt, Wissen aus trainierten Richtungen (z. B. Englisch-nach-Deutsch und Deutsch-nach-Französisch) zu kombinieren, um direkt Übersetzungen für ungesehene Richtungen (z. B. Englisch-nach-Französisch) durchzuführen. Darüber hinaus führt das Paradigma des Pre-Trainings und Fine-Tunings eine entscheidende *temporale* Dimension der Generalisierung ein. Diese Perspektive befasst sich damit, wie gut die vortrainierten Fähigkeiten eines Modells nach der Spezialisierung auf ausgewählte Aufgaben oder Sprachen erhalten bleiben. Diese Arbeit behandelt drei Aspekte der Generalisierung: den klassischen, den kompositionellen und den temporalen.

Zunächst untersuchen wir die kompositionelle Generalisierung in Übersetzungsmodellen, die von Grund auf trainiert wurden. Wir zeigen, dass ein Hauptgrund, warum mehrsprachige Übersetzungsmodelle nicht kompositionell generalisieren, darin liegt, dass ihre internen Repräsentationen stark mit Informationen verflochten sind, die spezifisch für die Eingabesprache sind. Wir identifizieren, dass diese Verflechtung durch die Residualverbindungen (residual connections) innerhalb der Modellarchitektur verstärkt wird. Indem wir eine Methode vorschlagen, die diese strukturelle Einschränkung lockert, erzielen wir erhebliche Verbesserungen bei ungesesehenen Übersetzungsrichtungen.

Anschließend erweitern wir die Untersuchung auf die temporale Dimension der Generalisierung, indem wir vortrainierte Modelle einbeziehen. Wir beobachten, dass die internen Repräsentationen eines Modells trotz umfangreichen mehrsprachigen Pre-Trainings oft hochspezifisch für die Eingabesprache bleiben. Wenn diese Modelle darüber

hinaus auf bestimmte Übersetzungsrichtungen spezialisiert werden, sind sie anfällig für katastrophales Vergessen, was ihre bestehenden Fähigkeiten erheblich beeinträchtigt. Um dem entgegenzuwirken, evaluiert unsere Arbeit mehrere etablierte Mitigationsstrategien aus der Literatur und stellt einen neuartigen Ansatz vor. Unsere vorgeschlagene Methode erzeugt eine interpretierbarere Ansicht der internen Repräsentationen des Modells, obwohl dieser Vorteil mit einem Kompromiss bei der Übersetzungsqualität einhergeht.

Unsere Untersuchung erstreckt sich dann auf große Sprachmodelle (Large Language Models, LLMs), die als universelle, anweisungsfolgende Systeme fungieren. Für diese Modelle, die typischerweise eine reine Decoder-Architektur verwenden, identifizieren wir zunächst die mittleren Schichten als die effektivsten Positionen für Interventionen. Aufbauend auf dieser Erkenntnis schlagen wir eine modulare und recheneffiziente Post-hoc-Methode vor, um die Sprachinvarianz in den Repräsentationen des Modells zu verbessern. Dieser Ansatz führt zu erheblichen Verbesserungen sowohl in klassischen Szenarien des kreuzlingualen Transfers als auch in den zuvor untersuchten Fällen der kompositionellen Generalisierung.

Zuletzt greifen wir den klassischen kreuzlingualen Transfer mit einer neuartigen Anwendung für die attributgesteuerte maschinelle Übersetzung erneut auf. Diese Methode trainiert einen Klassifikator, um Textattribute in Sprachen mit verfügbaren annotierten Daten zu unterscheiden, und überträgt diese Fähigkeit dann, um Übersetzungen für andere Sprachen zu steuern, denen solche Ressourcen fehlen. Der Ansatz operiert auf den internen Aktivierungen des Modells anstatt auf den Gewichten, was ihn für Szenarien geeignet macht, in denen die Modellgewichte vollständig unverändert bleiben müssen. Im Vergleich zu einem Orakel-Setup, das die Modellgewichte aktualisiert, zeigt unsere aktivierungsbasierte Methode eine stärkere Robustheit gegenüber Domänenverschiebungen (domain shifts).

Jeder der oben genannten Beiträge ist in einem Rahmen von Modellmodifikationen angesiedelt, der mit den sich entwickelnden Praktiken des Fachgebiets übereinstimmt und vom Training von Modellen von Grund auf bis hin zur Durchführung von Post-hoc-Interventionen bei großen vortrainierten Modellen reicht. Diese Arbeit treibt die Entwicklung mehrsprachiger NLP-Modelle voran, die nicht nur unter trainierten Bedingungen eine hohe Leistung erzielen, sondern auch generalisierbarere und anpassungsfähigere zugrundeliegende Repräsentationen erwerben.

Contents

Abstract	iii
Zusammenfassung	v
List of Tables	xiii
List of Figures	xv
1 Introduction	1
1.1 Motivation	1
1.2 Thesis Scope and Research Questions	3
1.3 Contributions	4
1.3.1 Methodological Contributions	4
1.3.2 Empirical Contributions	4
1.4 Thesis Outline	5
2 Background	7
2.1 Sequence-to-Sequence Models	7
2.1.1 Architecture Overview	8
2.1.2 Optimization	8
2.1.3 Transformer	9
2.1.4 Architecture Variants	9
2.2 Pre-Training and Fine-Tuning	10
2.3 Multilingual Models	11
2.3.1 Advantages and Challenges	11
2.3.2 Language Specialization Approaches	12
2.3.3 Language Alignment Approaches	13
2.4 Tasks	14
2.4.1 Machine Translation	14
2.4.2 Slot Filling	15
2.5 Evaluation	15
2.5.1 Classification Task Evaluation	16
2.5.2 Machine Translation Evaluation	16
3 Zero-Shot Translation without Pre-Training	19
3.1 Introduction	19
3.1.1 Task Definition	21

3.1.2	Contributions	21
3.2	Related Work	22
3.2.1	Alignment Losses	22
3.2.2	Language Token Strategies	22
3.2.3	Language-Specific Parameter Allocation	23
3.2.4	Monolingual Pre-Training, Data Augmentation and Mining	23
3.3	Disentangling Positional Information	23
3.3.1	Initial Analysis: Probing Language Invariance	24
3.3.2	Removing Middle-Layer Residual Connections	25
3.3.3	Position-Based Self-Attention Query	25
3.4	Experimental Setup	26
3.4.1	Datasets	26
3.4.2	Training and Evaluation	26
3.5	Results	28
3.5.1	Effect of Positional Disentanglement	28
3.5.2	Effect of Additional Regularization	29
3.5.3	Adaptation to Unseen Language	30
3.6	Analyses	30
3.6.1	Impact on Positional Correspondence	31
3.6.2	Impact on Language Invariance	32
3.6.3	Qualitative Analysis	33
3.7	Ablations	34
3.7.1	Where to Remove Residual Connections	34
3.7.2	Learned Positional Embeddings	35
3.8	Combination with Alignment Losses	35
3.8.1	Similarity Regularization	35
3.8.2	Adversarial Language Classification	36
3.8.3	Combining Residual Removal and Alignment Losses	37
3.9	Summary	38
4	Transfer with Translation-Specific Pre-Trained Models	39
4.1	Introduction	39
4.1.1	Task Definition	41
4.1.2	Contributions	42
4.2	Related Work	43
4.2.1	Robustness in Unseen Test Conditions	43
4.2.2	Discrete Representations	43
4.2.3	Analyzing Hidden Representations for Machine Translation	44
4.3	When does Pre-Training Help?	44
4.3.1	Setup	44
4.3.2	Pre-Training's Impact on Language-Specific Signals	45
4.3.3	Comparison to Random Initialization	45

4.3.4	Catastrophic Forgetting	46
4.3.5	Main Findings in Initial Analysis	48
4.4	Preserving Pre-Trained Knowledge by Discretization	48
4.4.1	Discretizing Encoder Latent Space	48
4.4.2	Soft Discrete Codes	50
4.5	Experimental Setup	51
4.5.1	Datasets	51
4.5.2	Training and Evaluation	51
4.6	Results	52
4.6.1	Impact of Bridge Languages	52
4.6.2	Impact of Using Codebooks	53
4.6.3	Using Discrete Codes to Interpret Learned Representations	54
4.7	Analyses on Learned Discrete Codes	56
4.7.1	How Well Clustered are the Hidden States?	56
4.7.2	Meaningfulness of Cluster Centers	57
4.8	Upper Bounds by Additional Training Data	57
4.8.1	Adding Parallel Data for Directions Unseen in Fine-Tuning	58
4.8.2	Multiple Bridge Languages	59
4.9	Summary	60
5	Transfer with Generalist Pre-Trained Models	61
5.1	Introduction	61
5.1.1	Task Definition	63
5.1.2	Contributions	63
5.2	Related Work	64
5.2.1	Multilingual Capabilities of Large Language Models	64
5.2.2	Multilingual Representation Alignment	64
5.2.3	Representation Analysis in Large Language Models	65
5.3	Analyzing Cross-Lingual Alignment in LLMs	65
5.4	Explicit Alignment in Fine-Tuning	67
5.4.1	Task Objective	67
5.4.2	Alignment Objective	67
5.4.3	Alternate Training	68
5.4.4	Data Requirement	69
5.5	Experimental Setup	69
5.5.1	Datasets	69
5.5.2	Evaluation	71
5.5.3	Model, Training, and Inference	71
5.6	Results	72
5.6.1	Necessity of Fine-Tuning and Multilingual Training Benefits	72
5.6.2	Overall Performance Comparison	72
5.6.3	Alignment Loss Placement	74

5.6.4	Impact on Representation Retrieval	75
5.7	Analyses	75
5.7.1	Larger-Scale Alignment	75
5.7.2	Resource Level of Alignment Languages	76
5.7.3	Generalization of Learned Alignment	77
5.7.4	Merging Alignment and Task Modules	77
5.7.5	Long Sequence Processing	78
5.8	Summary	79
6	Conditions for Catastrophic Forgetting in Multilingual Translation	81
6.1	Introduction	81
6.1.1	Task Definition	83
6.1.2	Contributions	83
6.2	Related Work	83
6.2.1	Catastrophic Language Forgetting in Machine Translation	83
6.2.2	Factors Influencing Catastrophic Forgetting	84
6.3	Controlled Setting to Study Catastrophic Forgetting	84
6.3.1	Base Models	85
6.3.2	Data	85
6.4	Gain-Forgetting Analyses	88
6.4.1	Model Scale and Fine-Tuning Data Size	89
6.4.2	Architecture and Language Control Mechanism	89
6.4.3	Analyses by Language Pair Types	91
6.4.4	Comparing LoRA and Full Fine-Tuning	92
6.5	Cross-Lingual Alignment for Forgetting Mitigation	92
6.5.1	Evaluated Methods	92
6.5.2	Translation-Specific Models	93
6.5.3	Instruction-Following Models	94
6.5.4	Why Gains Concentrate on Unseen Pairs	95
6.6	Summary	96
7	Inference-Time Transfer	97
7.1	Introduction	97
7.1.1	Task Definition	99
7.1.2	Contributions	99
7.2	Related Work	99
7.2.1	Controllable Generation	100
7.2.2	Multilingual Domain Adaptation	100
7.2.3	Activation Steering	101
7.3	Comparing Transferring Attribute Control Capabilities	101
7.3.1	Inference-Time Control	101
7.3.2	Fine-Tuning-Based Control	103
7.4	Experimental Setup	103

7.4.1	Formality Control	103
7.4.2	Grammatical Gender Control	105
7.4.3	Models and Evaluation	105
7.5	Supervised Conditions	106
7.6	Zero-Shot Conditions	107
7.6.1	New Target Languages	108
7.6.2	New Source and Target Languages	110
7.7	Summary	112
8	Conclusions	113
8.1	Answers to Research Questions	113
8.2	Limitations	116
8.3	Outlook	116
A	Supplementary Information for Chapter 3	119
A.1	Tokenization	119
A.2	Training Details	119
A.3	Evaluation Details	119
A.4	Translation Scores per Direction	120
A.5	Details on Implementation of Prior Approaches	125
B	Supplementary Information for Chapter 4	127
B.1	Dataset Details	127
B.2	Training, Inference, and Evaluation Details	127
C	Supplementary Information for Chapter 5	129
C.1	Dataset Details	129
C.2	List of Languages	129
C.3	Training and Inference Details	129
C.4	Results for Individual Languages	131
D	Supplementary Information for Chapter 7	133
D.1	Dataset Statistics	133
D.2	Training and Inference Details	133
D.3	Details on Human Evaluation	135
D.4	Comparison to Prior Works on Different Data Conditions	135
	Bibliography	137
	Acknowledgments	177
	Publications	179
	Curriculum Vitae	183

List of Tables

1.1	Overview of chapters	5
3.1	Classification accuracy when recovering input positional information . . .	25
3.2	Dataset overview of Chapter 3	26
3.3	Language overview of Chapter 3	27
3.4	Performance overview of proposed approaches to disentangle positional information	28
3.5	Impact of residual removal on zero-shot translation between related languages	29
3.6	Impact of variational dropout	30
3.7	Impact of low training data volume and lexical overlap	30
3.8	Performance of adapting to a new language before and after residual removal	31
3.9	Classification accuracy when recovering input positional information before and after residual removal	31
3.10	SVCCA scores and classification accuracy when recovering input language information	32
3.11	Example of direct translation outperforming pivot-based translation . . .	34
3.12	Ablation study on the location of residual removal	34
3.13	Ablation study on absolute positional encodings vs. learned positional embeddings	35
3.14	Performance of combining residual removal and alignment losses	38
4.1	Illustration of discrete codes associated with input tokens	40
4.2	Impact of pre-training on language-specific information	45
4.3	Impact of pre-training on translation performance of supervised directions	46
4.4	Impact of pre-training on translation performance of unseen directions during fine-tuning	47
4.5	Dataset overview of Chapter 4	51
4.6	Performance overview of techniques to prevent catastrophic forgetting during specialization	52
4.7	Impact of similar bridge languages on translation performance	53
4.8	Impact of using pre- and post-quantization encoder outputs during inference	58
4.9	Impact of adding oracle parallel data for previously unseen directions during fine-tuning	59
4.10	Impact of multi-bridge training	59
5.1	Dataset overview of Chapter 5	70

5.2	Impact of multilingual fine-tuning on LLM performance	72
5.3	Performance overview of middle-layer alignment on slot filling	73
5.4	Performance overview of middle-layer alignment on machine translation	73
5.5	Ablation study on the location of alignment loss	74
5.6	Impact of larger-scale alignment on transfer performance	76
5.7	Impact of alignment language selection on transfer performance	76
5.8	Impact of alignment language selection on translation retrieval accuracy	77
5.9	Performance of merging alignment modules post-hoc	78
5.10	Performance of middle-layer alignment on JSON generation	78
6.1	Model overview of Chapter 6	84
6.2	Dataset overview of Chapter 6	86
6.3	Performance of instruction- and token-based language control mechanisms on unseen target languages	90
6.4	Impact of in-language instructions	91
7.1	Dataset overview of Chapter 7	104
7.2	Performance overview on formality control with supervised translation directions	107
7.3	Performance overview on grammatical gender control with supervised translation directions	107
7.4	Performance overview on formality control with new target languages	108
7.5	Performance overview on grammatical gender control with new target languages	109
7.6	Human evaluation on formality-controlled translation	110
7.7	Performance overview on formality control with new source languages	111
7.8	Performance overview on grammatical gender control with new source and target languages	112
8.1	Summary of main findings	115

List of Figures

3.1	Illustration of positional correspondence to encoder input	22
3.2	Illustration of approaches for disentangling positional information	24
3.3	Classification accuracy per layer when recovering input positional information	31
3.4	Confusion matrices when recovering input language information	33
3.5	SVCCA scores per layer before and after residual removal	33
4.1	Illustration of soft discretization	42
4.2	Illustration of sliced discrete codes	49
4.3	Discrete code overlap results	55
4.4	Discrete code translation results	56
4.5	Analysis of clustering properties of encoder outputs	57
5.1	Translation retrieval accuracy over layers and the relation to transfer performance	66
5.2	Impact of multilingual task-specific fine-tuning on translation retrieval accuracy	66
5.3	Illustration of middle-layer alignment	67
5.4	Gradient similarity between task-specific and alignment training	69
5.5	Impact of alignment loss location on translation retrieval accuracy	75
6.1	Illustration of positive transfer vs. catastrophic forgetting	82
6.2	Gain-forgetting plots (small dataset)	88
6.3	Gain-forgetting plots (large dataset)	88
6.4	Gain-forgetting plots when gradually increasing training data volume	90
6.5	Impact of three cross-lingual alignment techniques on M2M-124	93
6.6	Impact of middle-layer alignment on Llama 3 and Qwen 2.5	94
7.1	Illustration of data scarcity in attribute-controlled translation	98
7.2	Illustration of classifier guidance and fine-tuning approaches for attribute-controlled translation	102

Chapter 1

Introduction

The story of the Tower of Babel describes humanity once sharing a single language before divine powers separated languages and scattered people. Despite its mythical nature, this story captures an insight about human languages. That is, underneath obvious differences such as vocabulary and grammar, there lies a shared conceptual foundation. The core meanings people intend to convey often remain consistent across languages, even when the ways of expressing them vary dramatically.

This observation has implications for computational models that process and generate human language. After training on many languages, these models may be able to transfer capabilities learned from one language to another. This phenomenon suggests generalization beyond language-specific patterns towards a common space where similar concepts from different languages are aligned.

However, to what extent does such a shared representation emerge automatically from exposure to multiple languages, and what techniques can better bridge the gap between languages and create more generalizable models? This thesis investigates these questions.

1.1 Motivation

The effectiveness of deep neural networks has been widely demonstrated in natural language processing (NLP) (Sutskever et al., 2014; Vaswani et al., 2017; Devlin et al., 2019) and other fields (Krizhevsky et al., 2012; Hinton et al., 2012). The success of these computational models is largely due to their ability to learn meaningful features that abstract away from raw data (Torralba et al., 2024). These abstract features, or *representations* (Bengio et al., 2013a), correspond to the activations in the intermediate layers of a neural network (Raghu et al., 2017; Vaswani et al., 2017; Kornblith et al., 2019). By leveraging such representations, models can apply what they have learned to unseen situations beyond their training examples (Zhang et al., 2017a). This process, however, relies heavily on sufficient training data (Kaplan et al., 2020).

The vast majority of training data in NLP is concentrated in a small number of languages with abundant linguistic resources, such as English and Chinese (Wenzek et al., 2020; Joshi et al., 2020; Schwenk et al., 2021a), representing a tiny fraction of the world’s over 7,000 languages (Ethnologue, 2025). For example, as of September 2025, data from Common Crawl (2025), a large-scale open web archive widely used to train many models

(Brown et al., 2020; NLLB Team, 2024; AI @ Meta et al., 2024), shows that more than 80% of its content belongs to the top 10 languages. For languages lacking training data, models must instead apply knowledge gained from other languages, operating in *zero-shot* conditions without having seen corresponding training data (Goodfellow et al., 2016; Johnson et al., 2017). The ability to generalize in this way, referred to as *cross-lingual generalization* (Ruder et al., 2019b), is therefore essential for extending the advances of NLP beyond the limited set of high-resource languages.

Generalization research in NLP has largely focused on domain shifts and robustness challenges, such as style variations (Oraby et al., 2018), topic drifts (Lazaridou et al., 2021), and the temporal evolution of language (Rosin and Radinsky, 2022; Su et al., 2022). In contrast, cross-lingual generalization has received less attention. The survey by Hupkes et al. (2023) finds that studies on cross-lingual generalization account for only 9% of the reviewed literature, while those on domain shifts and robustness constitute 34% and 21% respectively.

While cross-lingual aspects have been comparatively less explored within the NLP generalization literature, the concept itself is well grounded in theoretical foundations. Early works on word embeddings have demonstrated that words can be represented as vectors in a way that preserves geometric relationships across different languages (Mikolov et al., 2013; Lample et al., 2018), implying the presence of a shared semantic space. Such a shared space is a manifestation of a model’s cross-lingual generalization, which provides the basis for achieving *cross-lingual transfer* on specific downstream tasks.

The classical form of cross-lingual transfer involves training a model on a task in one or more languages and evaluating its performance on the same task in a language unseen during training (Pires et al., 2019; Wu and Dredze, 2019; Xu et al., 2020; Bigoulaeva et al., 2021) without involving different input and output languages. For example, a sentiment classifier trained only on English text may be tested on its ability to classify sentiment in German. For classification tasks, this process is relatively straightforward, since the possible output labels, e.g., “positive”, “negative”, are inherently independent of specific languages. In contrast, transfer is fundamentally more complex for generation or sequence-to-sequence tasks (Vu et al., 2022b; Li and Murray, 2023), where the model’s output must be a valid sequence in a designated language. This complexity arises because the model must be able to handle not only the semantics of the input but also the specific syntax and vocabulary of the output language.

Beyond the aforementioned classical form of cross-lingual transfer, the rise of large multilingual models that can process multiple languages within a single model (Johnson et al., 2017; Ha et al., 2017; Conneau et al., 2020) has expanded the notion of cross-lingual generalization. The highly multi-task nature of these models motivates an extension to *compositional* and *temporal* dimensions of generalization, as elaborated on next.

Compositional generalization describes the ability to perform new combinations of its learned concepts or capabilities (Schmidhuber, 1990). For instance, if someone knows what “hedgehog” and “riding a tricycle” are separately, by compositional generalization, they can understand “hedgehog riding a tricycle” without ever having encountered that

1.2. Thesis Scope and Research Questions

exact phrase before. This concept is also relevant in machine translation, where the basic task unit is a language pair instead of a single language. In this context, a model trained on translations from language A to B and from B to C exhibits compositional generalization if it can directly translate from A to C without intermediate steps. Demonstrating this capability indicates that the model has learned generalizable representations, rather than simply memorizing the translation pairs seen during training.

Moreover, the rise of large foundation models introduces a *temporal* dimension to generalization. These models typically first undergo a pre-training phase on vast multilingual data (Conneau et al., 2020; Xue et al., 2021; AI @ Meta et al., 2024). Afterwards, they may be continuously trained or *fine-tuned* on a subset of languages or tasks to improve performance (Zhu et al., 2023a; Gow-Smith and Sánchez Villegas, 2023; Gu et al., 2023) or add new capabilities (Wei et al., 2022a; Ouyang et al., 2022). This sequential process creates the risk of *catastrophic forgetting* (McCloskey and Cohen, 1989), where previously acquired capabilities are drastically degraded. Therefore, a central challenge is to ensure that representations remain generalizable over time, allowing models to expand their knowledge without overwriting existing capabilities.

In this context, machine translation provides an ideal testbed for studying these challenges for several reasons. First, translation inherently requires mapping between languages, making cross-lingual relationships explicit in the task definition. The models must learn to align semantic and syntactic structures across languages, creating strong pressure to develop representations that are agnostic of specific languages (Pires et al., 2019; Conneau et al., 2020). Second, the vast diversity of the world’s languages creates a natural environment for testing zero-shot generalization on unseen language pairs (Aharoni et al., 2019; Zhang et al., 2020a). Moreover, the widespread applications of machine translation (Och, 2005; Bentivogli et al., 2016; Barrault et al., 2019; NLLB Team, 2024) provide opportunities to transform research findings into practical impact.

1.2 Thesis Scope and Research Questions

This thesis investigates methods to develop multilingual representations that enhance cross-lingual transfer for sequence-to-sequence tasks, with a primary focus on machine translation. It addresses three facets of generalization:

1. Classical cross-lingual generalization, where a model is evaluated on its ability to perform tasks in languages unseen during task-specific training;
2. Compositional generalization, which assesses the ability to combine learned linguistic components in novel ways;
3. Temporal stability, which refers to the model’s ability to retain previously learned knowledge and mitigate catastrophic forgetting.

The overarching goal is to develop NLP models that not only perform well on trained tasks but also learn adaptable and generalizable underlying representations.

In this context, this thesis is guided by the following research questions:

Research Question 1: To what extent are representations in multilingual models language-invariant, and what factors contribute to their language-specific properties?

Research Question 2: How do current multilingual models perform in scenarios requiring classical cross-lingual, compositional, and temporal generalization, and how does the performance relate to the language-specific properties in Research Question 1?

Research Question 3: For practical scenarios with different constraints on the scope of model modification, what training procedures can encourage or leverage language-invariant representations for generalization across languages?

1.3 Contributions

This thesis makes the following methodological and empirical contributions to the field of natural language processing:

1.3.1 Methodological Contributions

1. We identify components in standard Transformer models that contribute to language-specific patterns and propose an alternative architecture that encourages language invariance while maintaining model expressiveness. ([Chapter 3](#))
2. We develop a soft discretization framework that addresses the expressiveness-invariance trade-off without modifying pre-trained model architectures. The approach maintains expressiveness by allowing access to both continuous and discrete representations through a soft discretization mechanism. ([Chapter 4](#))
3. We propose a middle-layer alignment approach to improve cross-lingual transfer for decoder-only language models, addressing the challenge of identifying optimal alignment positions in these models. The approach is modular and can be integrated with task-specific modules in a post-hoc manner. ([Chapter 5](#))
4. We propose an inference-time approach for attribute-controlled translation, which enables cross-lingual transfer of attribute control capabilities with higher robustness to domain shifts and language variety than standard fine-tuning approaches. ([Chapter 7](#))

1.3.2 Empirical Contributions

1. We demonstrate that standard Transformer models fail to produce language-invariant representations despite multilingual training data, and that input word order hinders the creation of language-invariant representations. ([Chapter 3](#))
2. We investigate how the selection of multilingual data affects the fine-tuning of pre-trained translation models, showing that using a linguistically similar bridge language,

1.4. Thesis Outline

Chapter	Modification Scope		Generalization Facet		
	Architecture Update	Weight Update	Classical Cross-Lingual	Compositional	Temporal
3	✓	✓		✓	
4	✗	✓		✓	✓
5	✗	✓	✓		✓
6	✗	✓		✓	✓
7	✗	✗	✓		✓

TABLE 1.1: Overview of core contribution chapters organized by modification scope (whether architecture or weights are changed) and by the facet of generalization they address, with checkmarks indicating the relevant dimensions for each chapter.

rather than English, is more resistant to catastrophic forgetting and leads to more aligned internal model representations. (Chapter 4)

3. We identify that the middle layer of decoder-only large language models has the highest degree of cross-lingual alignment compared to shallower or deeper layers. (Chapter 5)
4. We demonstrate that a multilingual model’s resistance to catastrophic forgetting of pre-trained languages depends on the relative scale of the pre-trained model and fine-tuning dataset. (Chapter 6)
5. We present a comprehensive comparison of cross-lingual alignment methods, including similarity-only, adversarial, and contrastive losses, finding that they achieve comparable effectiveness in mitigating catastrophic forgetting despite different mechanisms. (Chapter 6)
6. We show that across different transfer scenarios, machine translation models generalize less effectively to unseen target languages than to unseen source languages. (Chapter 6, Chapter 7)

1.4 Thesis Outline

The remainder of this thesis is organized as follows. After a review of relevant background in Chapter 2, the core contributions are presented in five chapters. Table 1.1 provides a summary that organizes these chapters along two key dimensions. The first, “modification scope”, specifies the constraints under which each approach operates: whether it modifies a pre-defined model architecture or updates its weights. The second dimension “generalization facet” connects each chapter to one of the facets of generalization introduced in §1.1.

Chapter 3 addresses the most flexible setting, where both the model architecture and its weights can be freely updated. In this context, we investigate compositional generalization, focusing on how a model can learn to combine trained translation directions to perform zero-shot translation.

[Chapter 4](#) transitions to a more constrained scenario using pre-trained translation-specific models with fixed architectures. The investigation extends to the temporal dimension, evaluating the model’s ability to retain knowledge of previously learned translation pairs after being fine-tuned on a selected set, thereby measuring its resistance to catastrophic forgetting.

[Chapter 5](#) broadens the scope to large general-purpose language models that follow instructions. In addition to the previously studied generalization scenarios, we also examine classical cross-lingual generalization in non-translation tasks where the input and output languages are the same, including slot filling and structured text generation.

[Chapter 6](#) presents a targeted analysis to reconcile inconsistent observations of catastrophic forgetting from the previous two chapters. It systematically investigates the conditions, such as model scale and fine-tuning data size, that determine when and why performance degradation occurs.

[Chapter 7](#) studies the most constrained setting, where both the model’s architecture and its weights are frozen. This chapter focuses on transferring attribute control (e.g., formality level) across languages at inference time, a task analogous to classical cross-lingual transfer. It also addresses the temporal dimension by examining whether the learned control mechanism is robust against domain shifts encountered during application.

[Chapter 8](#) concludes this thesis by presenting answers to the research questions and discussing the limitations and future work.

Chapter 2

Background

This chapter lays out the technical background for the work presented in this thesis. It begins by introducing the sequence-to-sequence framework (§2.1), the foundational architecture for tasks with variable-length outputs. The discussion then proceeds to the pre-training and fine-tuning paradigm (§2.2), which has become the de facto approach for high-performance models and a central workflow in this thesis. Afterwards, the chapter narrows its focus to multilingual models (§2.3), examining their advantages, challenges, and common approaches. This section is partly based on [Liu and Niehues \(2024b\)](#). The chapter concludes by defining the specific tasks investigated in this thesis (§2.4) and detailing the evaluation metrics for assessing model performance (§2.5).

2.1 Sequence-to-Sequence Models

Sequence-to-sequence learning is central to natural language processing (NLP), providing a common framework for a wide range of conditional text generation tasks such as machine translation, text summarization, as well as instruction following. As an application of the conditional language modeling task, sequence-to-sequence models learn the probability of generating an output sequence $Y = (y_1, \dots, y_T)$, given an input sequence $X = (x_1, \dots, x_N)$.

$$P(Y|X) = \prod_{t=1}^T P(y_t|y_{<t}, X), \quad (2.1)$$

where each term $P(y_t|y_{<t}, X)$ is the probability of generating y_t at timestep t based on the input sequence X and all previous outputs $y_{<t} = (y_1, \dots, y_{t-1})$.

This formulation differs from an unconditional language model, which models the probability of a sequence without conditioning on input X :

$$P(Y) = \prod_{t=1}^T P(y_t|y_{<t}). \quad (2.2)$$

In practice, the formulations in Equations 2.2 and 2.1 are often employed sequentially. Unconditional language modeling first establishes the fundamental knowledge of languages by learning from diverse text, while conditional modeling specializes this knowledge for specific tasks. This two-stage approach is further discussed in §2.2.

2.1.1 Architecture Overview

In most NLP applications, both the input sequence X and the output sequence Y are text. The initial step is to convert raw text into a sequence of discrete units referred to as tokens. These tokens can correspond to words, characters, or more often subword units (Sennrich et al., 2016b; Kudo and Richardson, 2018), as they offer a middle ground in granularity.

As modern NLP models are predominantly neural networks operating in continuous spaces, the discrete tokens must be mapped to numerical representations. This is realized by an embedding layer, which converts each token into a high-dimensional vector. This process transforms the input sequence of discrete tokens $X = (x_1, \dots, x_N)$, into a sequence of vectors $(\mathbf{e}_1, \dots, \mathbf{e}_N)$, where $\mathbf{e}_i \in \mathbb{R}^d$ and d is the embedding dimensionality. The embedding layer is often jointly optimized with the remaining model parameters by the process described later in §2.1.2.

The sequence of input embeddings $(\mathbf{e}_1, \dots, \mathbf{e}_N)$ is processed by a neural model to produce a sequence of contextualized vector representations $(\mathbf{h}_1^{(\text{in})}, \dots, \mathbf{h}_N^{(\text{in})})$, also referred to as hidden states. The currently dominant architecture for this purpose is the Transformer (Vaswani et al., 2017), which is explained further in §2.1.3. These contextualized representations serve as the input for the subsequent decoding phase, where the final output is generated.

To generate the output sequence, the model must project its internal vector representations back into the discrete token space. This process involves a linear projection from the final hidden state followed by a normalization step to produce a valid probability distribution over all possible tokens. Formally, at each generation step t , the model produces a final hidden state $\mathbf{h}_t^{\text{out}} \in \mathbb{R}^{d_{\text{hidden}}}$. This vector is projected to a logit vector:

$$\mathbf{z}_t = \mathbf{W}_{(\text{out})} \mathbf{h}_t^{(\text{out})}, \quad (2.3)$$

where $\mathbf{z}_t \in \mathbb{R}^{|V|}$ with $|V|$ being the cardinality of the set of possible tokens V and contains unnormalized scores for each possible token, and $\mathbf{W}_{\text{out}} \in \mathbb{R}^{|V| \times d_{\text{hidden}}}$ is the linear projection. To convert $\mathbf{z}_t$ into a valid probability distribution, the softmax function is applied. For the i -th token candidate w_i , its probability at time step t is:

$$P(w_i | y_{<t}, X) = \text{softmax}(\mathbf{z}_t)_i = \frac{\exp(z_{t,i})}{\sum_{j=1}^{|V|} \exp(z_{t,j})}, \quad (2.4)$$

where $z_{t,i}$ is the logit value for this token and $\exp(\cdot)$ is the exponential function. From this distribution, the next token can be selected to form the output sequence.

2.1.2 Optimization

The model's parameters, or weights, as denoted with θ , are learned from a dataset of input-output pairs $\mathcal{D} = \{(X^i, Y^i)\}$. Training the model is an optimization problem, where the objective is to find the parameters that maximize the conditional probability of the target sequences given their source sequences over the whole dataset. This is equivalent

2.1. Sequence-to-Sequence Models

to minimizing the negative log likelihood, which is formulated as the cross-entropy loss. Given a training sample (X, Y) , the loss is the sum of the negative log probabilities of each ground-truth token in the target sequence:

$$\mathcal{L}(\theta) = - \sum_{t=1}^{|Y|} \log P(y_t | y_{<t}, X; \theta). \quad (2.5)$$

In practice, this optimization is performed on batches of training samples using stochastic gradient descent with an optimizer like Adam (Kingma and Ba, 2015). The gradients of the loss with respect to θ are computed via backpropagation (Rumelhart et al., 1986).

2.1.3 Transformer

The Transformer architecture (Vaswani et al., 2017) has become the de facto approach for sequence modeling. A standard Transformer layer has two main sub-layers: a multi-head self-attention network and a position-wise feed-forward network. Each sub-layer is wrapped in residual connections (He et al., 2016) and combined with layer normalization (Ba et al., 2016). Residual connections create a shortcut connection by adding the sub-layer’s input to its output, allowing the original representation to be passed through the layer unmodified. This recurring pattern creates a residual stream, which enables a more direct information and gradient flow and is essential for mitigating the vanishing gradient problem (Hochreiter, 1998) when training models with a large number of layers. Moreover, since both the self-attention and the position-wise feed-forward networks are permutation-invariant, the original Transformer addressed this by adding fixed sinusoidal positional encodings to the input embeddings, thereby injecting information about each token’s absolute position.

The original Transformer architecture has been refined with several improvements that are now standard in modern large language models. Relative position encodings (Shaw et al., 2018; Dai et al., 2019), especially the Rotary Positional Encodings (Su et al., 2024), are widely used to address the poor length extrapolation of the original absolute positional encodings. Performance is further enhanced by replacing the feed-forward network’s Rectified Linear Unit (ReLU) (Fukushima, 1975) activation with Gated Linear Unit variants like SwiGLU (Shazeer, 2020). Moreover, the standard layer normalization (Ba et al., 2016) is often substituted with Root Mean Square Layer Normalization (Zhang and Sennrich, 2019) for computational efficiency.

2.1.4 Architecture Variants

The original Transformer (Vaswani et al., 2017) was introduced with an *encoder-decoder* architecture, where the encoder creates a contextual representation of the input sequence, which is then consumed by the decoder to autoregressively generate the output. However, as the Transformer’s core components are modular, they are also applicable for encoder-only and decoder-only architectures.

Encoder-only models, where BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019) is a representative, consist of a stack of Transformer encoder layers. They process the entire input sequence with bidirectional context, meaning the representation for each token is informed by all other tokens in the sequence. This makes them suitable for creating contextual representations of the inputs for classification and tagging tasks which assign labels to each token (Devlin et al., 2019; Sun et al., 2019), as well as for creating sentence representations for retrieval tasks (Feng et al., 2022). However, they are limited to tasks with fixed output lengths and are less suitable for text generation.

Decoder-only models, such as those from the GPT (Generative Pre-trained Transformer) series (Radford and Narasimhan, 2018; Radford et al., 2019; Ouyang et al., 2022), consist of a stack of autoregressive layers without a dedicated encoder. For conditional generation, the input sequence is provided as a prefix, and the model’s task is to generate a coherent continuation. This process is conceptually analogous to conditional generation in encoder-decoder models, but the implementation differs. The decoder conditions on the prefix using its own self-attention mechanism, while an encoder-decoder model uses a dedicated cross-attention mechanism to integrate context from the encoder.

2.2 Pre-Training and Fine-Tuning

Currently, the dominant paradigm in NLP is pre-training and fine-tuning, where the models first learn general knowledge from vast amounts of data and then specialize in specific tasks or behaviors.

During the pre-training phase, models are trained on extensive, often unlabeled data (Peters et al., 2018; Devlin et al., 2019; Radford et al., 2019). The training task depends on the model’s architecture. Models with decoders are typically trained for next-token prediction as described in Equation 2.2, where the model learns to predict the next token in a sequence given the previous context. Encoder-only models are commonly trained on an in-filling task, which predicts randomly masked tokens within a sequence based on both their left and right contexts (Devlin et al., 2019). The concept of a large language model (LLM) is defined by this massive pre-training data, as well as the size of the models in terms of parameter count. It was found that certain capabilities only emerge once a model surpasses a particular size threshold (Brown et al., 2020). These “emergent abilities” (Wei et al., 2022b) allow large models to perform new tasks based on a few examples provided in the input without parameter update.

Following pre-training, a fine-tuning stage is necessary to adapt the general-purpose model for specific applications or to instill desired behaviors. This process continues the training on a smaller and often labeled dataset relevant to the task of interest. The type of fine-tuning depends on the base model and the use case. For models pre-trained on a broad range of tasks within a certain use case, such as massively multilingual translation models M2M-100 (Fan et al., 2021) and NLLB-200 (NLLB Team, 2024), additional fine-tuning is often performed to focus on specific domains (Zhu et al., 2023a; Wassie et al., 2024) or language pairs (Liu and Niehues, 2022; Gow-Smith and Sánchez Villegas, 2023; Gu et al.,

2.3. Multilingual Models

2023). For generic pre-trained models, fine-tuning aligns the model with human intent, turning it from a simple text completion tool into a useful assistant. Approaches include supervised fine-tuning on instruction-response pairs (Wei et al., 2022a) and preference optimization with reinforcement learning (Ouyang et al., 2022; Rafailov et al., 2023).

2.3 Multilingual Models

Training a single model on data from multiple languages or language pairs (e.g., English-to-German and English-to-French translation) is an instance of multi-task learning (Caruana, 1997), where each language or language pair is a distinct but related task. The model learns a single set of parameters θ that is optimized to perform well across all of these tasks concurrently.

To realize the multi-task objective, the training datasets from different languages are aggregated. With a language-specific dataset for language L_i denoted as $\mathcal{D}_{L_i}$, the complete training corpus $\mathcal{D}_{\text{full}}$ consists of their union:

$$\mathcal{D}_{\text{full}} = \bigcup_{i=1}^N \mathcal{D}_{L_i}, \quad (2.6)$$

where N is the number of languages or language pairs. Although the composition of this aggregated dataset, particularly the relative proportions of language-specific data, can be controlled by specific sampling strategies (Arivazhagan et al., 2019b; Conneau and Lample, 2019), the fundamental principle of learning from the combined data remains unchanged. The model’s shared parameters are then optimized on this aggregated dataset using the process detailed in §2.1.2.

2.3.1 Advantages and Challenges

Multilingual models offer several advantages, particularly in the context of data scarcity. By training jointly on high-resource and low-resource languages, these models may facilitate transfer learning (Pan and Yang, 2010; Ruder et al., 2019a), where patterns learned from data-rich languages can improve performance on others. Meanwhile, the substantially larger and more diverse training data stabilizes the optimization process, which is often difficult under low-data conditions. The practical result of these combined benefits is evident in tasks like machine translation, where systems trained on multiple source languages for a single target language have been shown to consistently outperform their bilingual counterparts (Aharoni et al., 2019; Arivazhagan et al., 2019b; Fan et al., 2021; Chiang et al., 2022; Meng and Monz, 2024).

From a modeling perspective, an essential property of multilingual models is their capability of zero-shot transfer, enabling them to generalize to tasks for which they have not received explicit training. For instance, a model trained on German-to-English and English-to-French translation may acquire the ability to translate from German to French

without direct training on that specific pair. This phenomenon suggests the development of abstract cross-lingual representations rather than language-specific mappings.

From an operational perspective, unified multilingual models are highly parameter-efficient. Deploying and maintaining a single consolidated model for multiple languages is considerably more cost-effective in terms of computational infrastructure than managing many individual specialized models.

Despite these benefits, the multi-task nature of multilingual models introduces challenges. A primary issue is inter-language interference, where languages compete for the model’s finite capacity, especially in dense architectures (Aharoni et al., 2019). This competition can lead to performance degradation, often affecting low-resource languages that are overshadowed by the data volume of high-resource languages in the training data (Bapna and Firat, 2019).

2.3.2 Language Specialization Approaches

The challenge of inter-language interference and capacity bottleneck has motivated approaches that incorporate language-specific components. While early work relied on hand-crafted sharing strategies, such as using language-specific decoders (Dong et al., 2015), attention heads (Zhu et al., 2020b), or normalization layers (Zhang et al., 2020a), more recent research has focused on methods that learn to balance shared and specialized capacity. These approaches can be broadly categorized by how they modify a dense model’s architecture, by adding new parameters or by specializing existing parameters.

Parameter Addition Methods

Mixture-of-Experts (MoE) networks (Eigen et al., 2014; Shazeer et al., 2017; Lepikhin et al., 2021) address the capacity bottleneck by incorporating multiple “expert” sub-networks and a gating mechanism that selectively activates them for a given input. This architecture not only increases total model capacity but also provides a natural framework for balancing task-specific and task-agnostic computations. For multilingual applications, a common MoE configuration reserves some experts that are shared across all languages while dedicating others to specific languages. The gating mechanism learns the optimal combination of shared and language-specific experts for each input. This approach has seen various successful applications, including multilingual speech recognition (Gaur et al., 2021; Kwon and Chung, 2023; Hu et al., 2023; Wang et al., 2023a) and machine translation (Zhang et al., 2021; NLLB Team, 2024; Pires et al., 2023).

Like MoEs, adapters (Rebuffi et al., 2017; Houlsby et al., 2019; Bapna and Firat, 2019) are another form of conditionally activated network. They can be conceptualized as a restricted case of MoE with a crucial difference: the routing is fixed and determined a priori, not learned dynamically. Instead of a soft gating mechanism, adapters employ a hard-gating scheme where specific modules are assigned to predefined tasks. A variety of allocation schemes have been explored, such as assigning adapters to language pairs (Bapna and Firat, 2019), individual languages (Philip et al., 2020), or language families

2.3. Multilingual Models

(Chronopoulou et al., 2023b). Efforts have also been made to improve their parameter efficiency through techniques like hyper-adapters (Baziotis et al., 2022). In multilingual speech-to-text translation, for instance, language-specific adapters have improved upon monolithic models and achieved results comparable to full fine-tuning with far fewer parameters (Le et al., 2021).

Parameter Specialization Methods

In contrast to MoEs and adapters that add specialized components, pruning achieves specialization through a subtractive process. Based on a trained dense model, pruning identifies and removes unnecessary weights to create sparse, specialized sub-networks for each language or task. This approach creates a partially shared network where the distribution of language-specific and shared capacity is learned through weight removal. This method has demonstrated effectiveness in various multilingual tasks (Lu et al., 2022; Yang et al., 2023; Lin et al., 2021; Koishekenov et al., 2023; He et al., 2023).

An alternative line of work uses matrix factorization to disentangle language-specific and shared knowledge within the existing parameter space. Instead of adding new modules, this approach decomposes a model’s weight matrices into low-rank factors, where some factors are shared across all languages and others are specialized. This creates a learned allocation of capacity without increasing the parameter count. This technique has been applied in multilingual speech recognition (Pham et al., 2021b) and translation (Xu et al., 2023). Compared to MoEs and adapters, a key advantage of factorization is its parameter efficiency, which becomes particularly relevant as the number of supported languages grows (Pham et al., 2021b; Xu et al., 2023).

2.3.3 Language Alignment Approaches

While the language specialization approaches discussed in §2.3.2 mitigate interference in supervised settings, a different set of techniques are required for applications that rely on strong cross-lingual generalization, the model’s ability to perform tasks involving languages for which it has not received direct training. In these scenarios, the focus shifts from creating specialized parameters for each language to forming a unified representational space. The primary goal becomes learning aligned representations, where inputs with equivalent meaning from different languages are mapped to nearby points within the model’s internal representational space. The work in this thesis continues and expands this line of research.

While pre-trained multilingual models naturally develop some degree of unified multilingual representations (Pires et al., 2019; Conneau et al., 2020; Muller et al., 2021), the representation spaces of different languages, although overlapping, are not fully aligned by standard multilingual training alone (Libovický et al., 2020; Liu et al., 2021b; Gong et al., 2021). Explicitly strengthening the relationships between semantically equivalent content has shown benefits in various applications, including cross-lingual retrieval (Yu et al., 2018), mining for parallel sentences from monolingual sources (Schwenk et al.,

2021a; Duquenne et al., 2023a,b; Okabe et al., 2025), zero-shot classification (Hu et al., 2021; Gritta and Iacobacci, 2021) and translation (Arivazhagan et al., 2019a; Pham et al., 2019; Duquenne et al., 2022). Despite different approaches, these works share a common objective of aligning representations of semantically equivalent content across languages while preserving overall expressiveness.

2.4 Tasks

While this thesis primarily focuses on machine translation tasks (§2.4.1), our evaluations also extend to tasks traditionally framed as sequence tagging, such as slot filling (§2.4.2). The instruction-following capabilities of large language models make it possible to address these distinct tasks within the same next-token generation framework.

2.4.1 Machine Translation

Machine Translation (MT) is a canonical application of the sequence-to-sequence framework described in §2.1. In this task, the input sequence X represents a sentence in a *source language*, and the model’s objective is to generate the corresponding output sequence Y , which is the translated sentence in a *target language*. The translation process typically operates on a sentence-by-sentence basis. To learn this mapping, sequence-to-sequence models are trained on parallel data, which consists of sentence pairs where each source sentence is matched with its translation. This parallel data provides the supervision signal for the model to learn the conditional probability distribution $P(Y|X)$.

When a single model is trained to translate into multiple target languages, it requires an explicit signal to determine which target language to generate for a given source sentence, as the conditional probability $P(Y|X)$ is otherwise ambiguous. Traditionally, this was realized by prepending a special token corresponding to the target language, either to the source sequence X (Johnson et al., 2017) or the target sequence Y (Ha et al., 2017). When prepended to X , the encoding of the entire input is informed by the desired output language. On the other hand, using it as the first token of Y guides the autoregressive generation process from its first step. More advanced approaches incorporate embedding vectors specific to the target language at each position of the output sequence (Ha et al., 2017), providing a durable signal throughout generation.

More recently, with the rise of instruction-following models, the target language identity is often incorporated in natural language instructions or prompts that precede the source sentences. Specifically, the input X is broken down into two components: an instruction, such as “translate this from English to German”, and the source sentence itself. This reframes multilingual translation as a general instruction-following task, unifying it with other conditional text generation problems handled by large language models.

2.5. Evaluation

2.4.2 Slot Filling

Slot filling is a core NLP task that aims to extract specific pieces of information from text and assign them to predefined semantic categories, or “slots”. For instance, given the sentence “Book a flight to Berlin for tomorrow”, the objective is to extract “Berlin” as the value for the “destination” slot and “tomorrow” as the value for the “date” slot. This task has traditionally been framed as a sequence tagging problem. In this formulation, the model assigns a label to each token in an input sequence $X = (x_1, \dots, x_N)$ to identify text spans that correspond to slot values. Tokens that do not belong to any slot are typically assigned a null or “outside” label. This approach constrains the output space, distinguishing it from more open-ended generation tasks such as machine translation, as discussed in §2.4.1.

With the rise of large language models, many sequence tagging tasks have been reformulated as generation problems. In this formulation, rather than assigning a label to every token, the model generates a string that describes the filled slots and their corresponding values. For the previous example sentence, the model may produce a structured text such as “[destination]: Berlin; [date]: tomorrow”. In this output, only the words corresponding to slot values are present, while other words are omitted because they do not fill any slot. This generated string is then post-processed and parsed to reconstruct a slot-value dictionary.

This generation approach has high requirements on the model’s ability to adhere to a strict output format. Any deviation from the predefined template, such as a missing separator or a slightly altered label name, can cause failures in parsing and thereby extracting the structured information. The challenge of maintaining format consistency is exacerbated when the number of potential labels is large, as this increases the variability of possible outputs and the possibility of producing ill-formed structures.

2.5 Evaluation

Evaluation metrics are crucial for assessing model quality. While human evaluation is the most direct way to measure how helpful models are to users, it faces significant challenges in cost and scalability. Conducting human evaluations requires substantial resources, including recruiting qualified evaluators, establishing clear rating criteria, and allocating time for annotation, all of which quickly become prohibitive as model development accelerates.

These limitations necessitate automatic evaluation metrics. These metrics operate by comparing model outputs against reference outputs on curated test sets. The difficulty of automatic evaluation varies with the nature of the task, particularly with the degree of acceptable variation in the expected output. This section describes the standard metrics for structured tasks with deterministic solutions, as well as for machine translation.

2.5.1 Classification Task Evaluation

Some NLP tasks have a single correct output for each input, for instance text classification, part-of-speech tagging, and slot filling. In these cases, the ground truth for any given input is unambiguous, making the evaluation process relatively straightforward. A system's output can be directly compared against the annotated ground-truth reference. In this context, pure accuracy is not suitable when dealing with imbalanced datasets, where the score could be misleading when some classes significantly outnumber others. Instead, performance is commonly measured by metrics based on the number of true positives (TP), false positives (FP), and false negatives (FN).

Precision measures the fraction of predicted positive instances that are correct.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (2.7)$$

Recall measures the fraction of actual positive instances that the model correctly identifies.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (2.8)$$

The F_1 score is the harmonic mean of precision and recall and is particularly useful when there is an uneven class distribution.

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (2.9)$$

For multiclass classification problems, these metrics are calculated on a per-class basis and then aggregated.

2.5.2 Machine Translation Evaluation

In contrast to tasks with a single correct output, many NLP applications such as machine translation and summarization are open-ended, meaning multiple viable outputs can exist for a single input. This property makes evaluation more complex, as a generated output may differ lexically from reference texts while still being semantically correct. This section reviews two types of automatic evaluation metrics, with a focus on their application to machine translation.

Match-Based Metric: BLEU

BLEU (Papineni et al., 2002), standing for BiLingual Evaluation Understudy, is a long-standing automatic metric for MT evaluation. Given a candidate translation, BLEU measures the overlap of its n -grams (sequences of n adjacent tokens) with one or more reference translations. The score is formally defined as a brevity-penalized geometric mean of n -gram precisions:

$$\text{BLEU} = \text{BP} \cdot \exp \left(\sum_{n=1}^N w_n \log p_n \right), \quad (2.10)$$

2.5. Evaluation

where BP is the brevity penalty that penalizes translations shorter than the references, p_n is the n -gram precision itself, and w_n are the weights for each n -gram precision. In the standard configuration by Papineni et al. (2002), N is set to 4 and w_n is set uniformly to $1/N$. BLEU scores range from 0 to 1 but are conventionally multiplied by 100 for readability. This thesis follows this convention.

Despite its wide adoption, BLEU has well-documented limitations. First, it is highly sensitive to the chosen text tokenization method, which leads to poor comparability across studies. This issue is particularly relevant for languages that do not use whitespace to delimit words, such as Chinese or Japanese, as they require external tokenizers that may introduce further inconsistencies. Moreover, BLEU’s reliance on exact substring matching makes it ill-suited for morphologically rich languages (where words extensively change form to represent grammatical features), as it penalizes valid grammatical variations of words as complete mismatches.

To address these shortcomings, the sacreBLEU (Post, 2018) toolkit was introduced to standardize the evaluation pipeline and ensure comparable scores. To better account for morphological diversity, more fine-grained metrics such as chrF (character n -gram F-score) (Popović, 2015) and subword-level BLEU (Goyal et al., 2022) have emerged as alternatives.

Neural Metric: COMET

Neural evaluation metrics address many of the limitations of match-based approaches. These metrics score translations by their semantic similarity to references rather than lexical overlap. COMET (Rei et al., 2020, 2022), standing for Crosslingual Optimized Metric for Evaluation of Translation, is a representative approach. This metric has been widely adopted after demonstrating much stronger correlation with human judgment compared to BLEU (Freitag et al., 2022).

The COMET architecture is built on a pre-trained multilingual encoder such as XLM-RoBERTa (Conneau et al., 2020), which is a modular component that can be replaced with other models. This encoder generates contextualized embeddings for the source text, the candidate translation, and the reference translation. The embeddings are then passed to a feed-forward regression network. This model is trained to predict quality scores assigned by human annotators, which are typically normalized using a z-score transformation to account for inter-annotator variance. Similar to BLEU, the COMET scores are conventionally scaled by 100 for readability.

By using contextual embeddings, COMET can handle semantically equivalent expressions that would otherwise be penalized under match-based evaluation. However, it also has limitations, including insensitivity to incorrect output languages (Zouhar et al., 2024a) and biases toward certain text styles (Zouhar et al., 2024b).

Metrics in this Thesis

Given these complementary strengths and weaknesses, current best practices recommend reporting both in multilingual evaluation (Zouhar et al., 2024a). Therefore, we use BLEU as a transparent surface-level measure and COMET to capture deeper semantic correspondence. For BLEU, we use the sacreBLEU toolkit (Post, 2018) to ensure reproducibility and report the specific evaluation signatures. For COMET, we use the standard model variant named `Unbabel/wmt22-comet-da`, which was the default at the time of writing. An exception is Chapter 3, where only BLEU scores are reported, as the experiments were conducted before the publication of COMET (Rei et al., 2020).

Interpreting Scores

A critical caveat for automatic metrics, whether string-match-based or neural, is the difficulty of interpreting their scores in terms of absolute quality (Kocmi et al., 2024b). Match-based scores like BLEU are highly dependent on the linguistic properties of the target language and the difficulty of the test set. For instance, a BLEU score of 30 might indicate high quality for a morphologically rich language but only mediocre quality for English. For COMET scores, there is also not a direct interpretation in terms of absolute quality. Receiving a COMET score of 80 does not mean that a translation is 80% correct. The primary use of this metric lies in ranking the relative performance of different systems. While recent work has developed more interpretable metrics that provide fine-grained error analyses (Fernandes et al., 2023; Guerreiro et al., 2024), fully interpretable evaluation remains an active research area beyond the scope of this dissertation.

Chapter 3

Zero-Shot Translation without Pre-Training

As motivated in the introduction (§1.1), an essential generalization ability we study is combining knowledge from trained tasks to tackle unseen ones, i.e., compositional generalization. Multilingual machine translation provides an ideal testbed for evaluating this ability. Having learned to translate language pairs $A \rightarrow B$ and $B \rightarrow C$, a model that generalizes effectively should be able to directly perform $A \rightarrow C$ translation. This ability to translate between language pairs without explicit training on parallel data is known as zero-shot translation.

Despite its conceptual appeal, zero-shot translation remains challenging in practice. In this chapter, we first investigate why current multilingual models struggle with this task by analyzing their internal representations. Our analysis reveals that these representations retain substantial language-specific information, which hinders generalization across language pairs. Building on this insight, we propose a technique that improves zero-shot translation by reducing language-specific information. Lastly, we demonstrate that our approach provides synergistic gains when combined with existing methods.

Of all chapters in this thesis, this one assumes the highest architectural flexibility, as the models are unconstrained by pre-trained components. This chapter is partly based on [Liu et al. \(2021b\)](#).

3.1 Introduction

Multilingual neural machine translation (MNMT) systems integrate multiple translation directions within a single model ([Dong et al., 2015](#); [Firat et al., 2016](#); [Ha et al., 2016](#); [Johnson et al., 2017](#)), offering advantages including simplified deployment and potential for cross-lingual knowledge transfer. One particularly valuable form of this transfer is to combine learned translation directions to handle new translation pairs without parallel training data.

Since their inception, MNMT models have demonstrated the ability to directly translate between language pairs unseen in their training data, i.e., zero-shot translation ([Johnson et al., 2017](#); [Ha et al., 2017](#)). However, initial studies were limited in language coverage, typically involving a small number of languages and focusing on closely related language pairs. This falls short when considering the approximately 7,000 languages in the world. In real-world scenarios, most languages have available English translations, suggesting

that an ideal system would train on English translations and enable direct translation between the non-English languages. This offers a significant advantage in data efficiency, as training on a linear number of language pairs could potentially serve a quadratic number of language pairs.

An alternative for addressing zero-shot conditions is to pivot through an intermediate language, typically English. In *pivot-based translation*, text is first translated from the source language into the pivot language, then from the pivot language into the target language. Despite its simplicity, pivoting introduces both theoretical and practical limitations. First, this approach suffers from an inherent information loss by conversion to intermediate languages, as certain concepts or linguistic structures may lack direct equivalents in the pivot language. For example, the German word “*gemütlich*” describes a warm, cozy atmosphere with a sense of belonging, a concept that requires lengthy explanation in English but translates directly to the Dutch equivalent “*gezellig*”. Additionally, pivoting doubles inference latency, increasing computation overhead and serving costs.

Despite the theoretical and practical appeal, zero-shot translation has consistently underperformed compared to pivot-based methods. Prior works (Arivazhagan et al., 2019a; Zhang et al., 2020a; Rios et al., 2020) suggest that standard systems tend to generate poor outputs, often in an incorrect target language. A central issue causing incorrect target language generation is that models capture spurious correlations between source-target language pairs with supervised training data (Gu et al., 2019), thereby limiting generalization to new source-target language combinations.

We argue that language-invariant representations are a prerequisite for zero-shot translation, and that their absence hinders zero-shot performance. Ideally, language-invariant representations would function as a universal intermediate format that captures semantic content while abstracting away source language characteristics. This would enable the decoder to apply its learned target language generation capabilities regardless of the source language input. Specifically, during training, the model learns to encode inputs in forms that facilitate translation for supervised directions. If encoded representations are not language-invariant, decoders must handle inputs distributed differently from training data when prompted for zero-shot translation to different target languages, which naturally degrades performance.

This invariance mirrors other concepts in computer science, including *intermediate representations* in compilers and *interfaces* in software engineering, as both serve as abstraction layers that decouple input specifics from output generation. In compiler design, intermediate representations allow the same backend to generate code for different target architectures regardless of the source language. In software engineering, interfaces allow different implementations to work together without explicitly pairing each possible combination. Crucially, these approaches eliminate the need to create a quadratic number of direct mappings between input and output types, reducing the quadratic number of pairwise connections to linear.

3.1. Introduction

3.1.1 Task Definition

In this chapter, we study the following task:

Given a model trained on $N - 1$ languages and their English translations, how to improve direct translation for the remaining $\underbrace{N(N - 1)}_{\text{all language pairs}} - \underbrace{2(N - 1)}_{\text{trained pairs}}$ zero-shot directions?

This task is fundamentally a challenge of compositional generalization. To translate between a pair of non-English languages, the model cannot rely on existing training examples. Instead, it must learn to compose two skills acquired from training, translating from the source language into English and then into the target language. Achieving direct translation instead of relying on two-step pivot translation has advantages in reducing inference latency and the risk of error propagation.

3.1.2 Contributions

In this chapter, we show that typical Transformer encoders (Vaswani et al., 2017) produce outputs with strong positional correspondence to input tokens, which hinders the creation of language-invariant representations. For example, in the English sentence in Figure 3.1, encoder outputs $h_{1,2,3}$ correspond to “a”, “big”, “cat” respectively. While this property is essential for tasks such as sequence tagging, it becomes harmful in tasks requiring language-invariant representations. Even assuming perfect lexical mapping of input embeddings (e.g., “cat” and “gato” sharing identical embedding vectors), the resulting encoder outputs remain language-specific due to word order differences across languages. We therefore propose relaxing this structural constraint and offer the model flexibility in word reordering within the encoder.

Specifically, we make the following contributions:

- We show that the positional correspondence to input tokens hinders zero-shot translation. We achieve considerable gains on zero-shot translation quality by only removing residual connections once in a middle encoder layer.
- Based on detailed analyses of model intermediate representations, we show that our approach creates more language-invariant representations both on the token and sequence level.
- Our proposed model allows easy integration of new languages, which enables zero-shot translation between the new language and all other languages previously trained on.
- We show that our architectural modification is complementary to previous methods that explicitly align representations. When combined with these methods, our approach leads to further improvements.

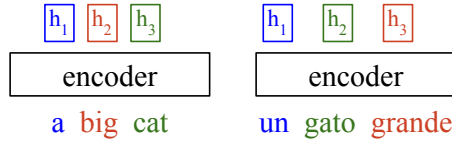


FIGURE 3.1: An example of language-specific encoder outputs as a result of the strong positional correspondence to input tokens (even assuming the word embeddings are cross-lingually mapped).

3.2 Related Work

While early multilingual translation systems exhibited some zero-shot translation capabilities (Johnson et al., 2017; Ha et al., 2017), the performance fell far behind pivot-based methods, even under favorable testing conditions with limited language diversity. Subsequent research has explored several directions for improving zero-shot translation performance, which we organize into the following subsections.

3.2.1 Alignment Losses

Fully parameter-shared models have gained popularity, with massively multilingual systems showing encouraging results (Aharoni et al., 2019; Zhang et al., 2020a; Fan et al., 2021). Besides advantages such as compactness and ease of deployment, the tightly coupled model components also raise new questions. One key question is how to form language-agnostic representations at a suitable abstraction level. In this context, one approach introduces auxiliary training objectives to align the representations of different languages (Arivazhagan et al., 2019a; Al-Shedivat and Parikh, 2019; Pham et al., 2019), with semantically similar contents having higher similarity. In this work, we take a different perspective. Instead of introducing additional objectives, we relax some of the pre-defined structural constraints to facilitate language-invariant representations. We also demonstrate that our approach is complementary to alignment losses proposed in previous works.

3.2.2 Language Token Strategies

Controlling output languages is a fundamental challenge in multilingual translation. This is typically addressed by trainable special placeholders, known as language tokens, which signal the relevant language identities to the model (Ha et al., 2016; Johnson et al., 2017). This mechanism is based on identical principles to subsequently developed prompt tuning techniques (Lester et al., 2021). In zero-shot translation, as incorrect output language generation has been identified as a major limitation, prior works have proposed methods to improve target language control through additional embeddings along decoding steps (Ha et al., 2017) and reinforcement of target language identity in early layers (Qu et al., 2024). Zero-shot performance is highly sensitive to language token strategies (Wu et al., 2021; ElNokrashy et al., 2022). In this chapter, we adopt the target language embeddings

3.3. Disentangling Positional Information

approach (Ha et al., 2017; Pham et al., 2019), which has demonstrated strong zero-shot performance.

3.2.3 Language-Specific Parameter Allocation

While our work focuses on fully parameter-shared models, language-specific parameter allocation is an orthogonal direction that exchanges model compactness for architectural modularity and reduced parameter interference. In zero-shot translation, one line of work focuses on controlling or learning the level of parameter sharing between languages. One approach employs language-specific encoders and decoders for individual languages while maintaining shared intermediate layers (Lu et al., 2018; Vázquez et al., 2019; Escolano et al., 2020; Purason and Tättar, 2022). Another approach dynamically generates parameters for language-specific components (Platanios et al., 2018; Baziotis et al., 2022).

3.2.4 Monolingual Pre-Training, Data Augmentation and Mining

Besides modeling efforts, zero-shot translation can be enhanced through data-centric approaches that mitigate the scarcity of direct parallel data. Monolingual pre-training strengthens the decoder’s generation capabilities for languages in zero-shot pairs by leveraging large monolingual corpora (Gu et al., 2019; Ji et al., 2020). Synthetic data generated by back-translation (Gu et al., 2019; Lakew et al., 2019b; Zhang et al., 2020a) or pivot-based translation (Currey and Heafield, 2019) effectively converts the zero-shot problem into a supervised one by creating pseudo-parallel corpora. A similar approach is to mine parallel data for non-English languages from existing training data (Freitag and Firat, 2020) or monolingual resources (Schwenk, 2018; Hangya and Fraser, 2019) to enrich the existing English-centric data. While acknowledging the effectiveness of these data-centric approaches, our work is deliberately scoped to models without them, thereby creating a controlled setting to evaluate generalization capabilities under stricter, English-centric data conditions.

3.3 Disentangling Positional Information

Zero-shot inference relies on a model’s generalizability to conditions unseen in training. As motivated earlier, in the context of zero-shot translation, the input should ideally be encoded into a language-invariant representation, based on which the decoder can translate into any target language required, similar to the notion of an interlingua. However, the ideal of “any input language, same representation” cannot be easily fulfilled with a standard encoder, as we have shown in the motivating example in Figure 3.1.

Our work identifies a key contributor to this issue: the strong positional correspondence between the encoder’s input tokens and its output hidden states. Formally, for a sequence of input token embeddings $(\mathbf{e}_1, \dots, \mathbf{e}_n)$, the corresponding encoder output $(\mathbf{h}_1, \dots, \mathbf{h}_n)$ retains a structure where the i -th hidden state $\mathbf{h}_i$ primarily contains information about $\mathbf{e}_i$. While this property is essential for tasks such as contextual embedding and

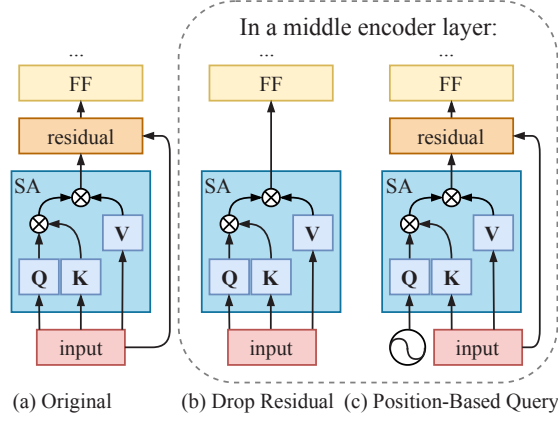


FIGURE 3.2: Illustrations of our proposed modifications to an original encoder layer: dropping residual connections once (§3.3.2); making attention query based on position encoding (§3.3.3). Before each self-attention (SA) and feed forward (FF) layer we apply layer normalization, which is not visualized here for brevity.

sequence tagging, it is less suitable for language-invariant representations. As sentences with identical meaning in different languages often have different lengths and word orders, this positional dependency ensures they are encoded into structurally dissimilar hidden state sequences.

We hypothesize that this positional correspondence is enforced by two architecture features: residual connections and the alignment patterns in encoder self-attention. By modifying these two components accordingly, we aim to alleviate the positional correspondence. Specifically, we propose setting a specialized encoder layer free from these positional constraints, allowing it to reorder information rather than strictly following the input token order.

Prior work suggests that the choice of positional encoding affects how the Transformer model preserves word order information, where absolute positional encodings, as used in the original Transformer (Vaswani et al., 2017), strengthen this information (Ahmad et al., 2019). In contrast, we show that this issue is not caused exclusively by absolute positional encodings, as the same problem also impacts learned positional embeddings (§3.7.2).

3.3.1 Initial Analysis: Probing Language Invariance

To quantify the degree to which positional information is preserved in the encoder outputs, we conduct a probing analysis, a common technique for inspecting model internal representations (Adi et al., 2017; Libovický et al., 2020). After training a baseline multilingual Transformer (details in §3.4.2), we attach an auxiliary classifier to its encoder outputs. The classifier’s objective is to predict the original input token identities¹ or their absolute position identities² from the corresponding encoder outputs. As shown in Table 3.1, the classifier can recover token identities from the baseline model’s encoder output with near-perfect accuracy. This result demonstrates a strong, nearly one-to-one mapping between

¹in $\{1, \dots, |V|\}$ where $|V|$ is the number of possible tokens

²in $\{1, \dots, \text{max_pos}\}$ where max_pos is the maximum number of tokens accepted by the model

3.3. Disentangling Positional Information

input tokens and output hidden states, verifying our hypothesis on the strong positional correspondence.

Dataset	Token ID	Position ID
IWSLT	99.9	93.3
Europarl	99.5	85.1
PMIndia	99.6	90.1

TABLE 3.1: Classification accuracy (%) when recovering input positional information (token and position identities) from encoder outputs. High accuracy indicates strong positional correspondence to input tokens.

3.3.2 Removing Middle-Layer Residual Connections

In the original Transformer architecture from Vaswani et al. (2017), residual connections (He et al., 2016) are applied in every layer, for both the multi-head attention and the feed-forward layer. By adding each layer’s inputs to its outputs, the residual connections facilitate gradient flow to bottom layers of the network. However, since the residual connections are present throughout all layers, they strictly impose a one-to-one alignment between the inputs and outputs. For the encoder, this causes the outputs to be positionally corresponding to the input tokens.

We propose to relax this condition, such that the encoder outputs become less position- and hence language-specific. Meanwhile, to minimize the impact on the model architecture and ensure gradient flow, we limit this change to only one encoder layer, and only its multi-head attention layer. Figure 3.2(b) gives a visualization of this change in comparison to the original encoder in Figure 3.2(a).

3.3.3 Position-Based Self-Attention Query

Besides the residual connections, another potential reason for the positional correspondence is the encoder self-attention alignment. Via the self-attention transform, each position is a weighted sum from all input positions. While the weights theoretically can distribute over all input positions, they are often concentrated locally, particularly with output position i focusing on input position i . Previous works on various sequence tasks (Yang et al., 2020; Zhang et al., 2020b) have shown heavy weights on the diagonal of the encoder self-attention matrices.

In this light, the motivation of our method starts with the formation of the self-attention weight matrix: $\text{score}(\mathbf{Q}, \mathbf{K}) = \mathbf{Q}\mathbf{K}^T$, where $\mathbf{Q}$ and $\mathbf{K}$ are the query and key matrices. This $N \times N$ matrix encapsulates dot product at each position against all N positions. Since the dot product is used as a similarity measure, we hypothesize that when $\mathbf{Q}$ and $\mathbf{K}$ are similar, the matrix will have heavy weights on the diagonal, thereby causing the positional correspondence. Indeed, $\mathbf{Q}$ and $\mathbf{K}$ are likely similar since they are projections from the same input. We therefore propose to reduce this similarity by replacing the projection base of the self-attention query by a set of sinusoidal positional encodings. Moreover, to avoid

potential interaction with positional information retained in $\mathbf{K}$, we use a wave length for this set of sinusoidal encodings that is different from what is added onto encoder input embeddings. Figure 3.2(c) contrasts our position-based attention query with the original model in Figure 3.2(a), where the key, query, and value are all projected from the input to the self-attention layer.

3.4 Experimental Setup

Our experiments cover high- and low-resource languages and different data conditions. We choose an English-centered setup, mirroring real-world data availability of English translations. We train on $X \leftrightarrow \text{en}$ parallel data, and test the zero-shot translation between all non-English languages. This scenario is particularly difficult for zero-shot translation, as half of the target-side training data is in English.

3.4.1 Datasets

Our datasets originate from three sources: IWSLT 2017 (Cettolo et al., 2017), Europarl v7 (Koehn, 2005), and PMIndia (Haddow and Kirefu, 2020). The IWSLT and Europarl data are taken from the MMCR4NLP corpus (Dabre and Kurohashi, 2017). An overview of the datasets is in Table 3.2.

To investigate the role of training data diversity, we construct two conditions for Europarl, where one is fully multiway aligned, and the other has no multiway alignment at all. Both are subsets of the full dataset with 1M parallel sentences per direction. In the main experiments, we report results on Europarl disjoint and multiway. For later experiments (§3.8.3), we also include Europarl full. Moreover, we study the challenging case of PMIndia with very limited training data, distinct writing systems, and a large number of agglutinative languages that are particularly difficult to translate into. Table 3.3 outlines the languages in our experiments.

Dataset	Language Coverage ($X \leftrightarrow \text{en}$)	# zero-shot directions	# sent. per direction
IWSLT	{it, nl, ro}	6	145K
PMIndia	{bn, gu, hi, ml, mr, kn, or, te, pa}	72	26-53K
Europarl:			
- disjoint			119K
- multiway	{da, de, es, fi, fr, it, nl, pt}	56	119K
- full			1M

TABLE 3.2: Overview of the datasets.

3.4.2 Training and Evaluation

Training Details for Baselines

By default we use Transformer (Vaswani et al., 2017) with 5 encoder and decoder layers. For the Europarl datasets, we enlarge the model to 8 encoder and decoder layers to

3.4. Experimental Setup

Code	Name	Family	Script
it	Italian	Romance	Latin
nl	Dutch	Germanic	
ro	Romanian	Romance	
da	Danish	Germanic	
de	German	Germanic	
es	Spanish	Romance	
fi	Finnish	Uralic	
fr	French	Romance	
pt	Portuguese	Romance	
bn	Bengali	Indo-Aryan	Bengali
gu	Gujarati	Indo-Aryan	Gujarati
hi	Hindi	Indo-Aryan	Devanagari
kn	Kannada	Dravidian	Kannada
ml	Malayalam	Dravidian	Malayalam
mr	Marathi	Indo-Aryan	Devanagari
or	Odia	Indo-Aryan	Odia
pa	Punjabi	Indo-Aryan	Gurmukhi
te	Telugu	Dravidian	Telugu

TABLE 3.3: Overview of the languages in our experiments.

account for the larger amount of training data. To control the output language, we use a target-language-specific begin-token as well as language embeddings concatenated with decoder word embeddings³, following [Pham et al. \(2019\)](#). No source language token is used, following the motivation of creating language-invariant encoder representation. Moreover, we include variational dropout ([Gal and Ghahramani, 2016](#)) as a comparison since it was used in a previous work on zero-shot translation ([Pham et al., 2019](#)) instead of the standard element-wise dropout. With variational dropout, all timesteps in a layer output share the same mask. This differs from the standard dropout, where each element in each timestep is dropped according to the same dropout rate. We hypothesize that this technique helps reduce the positional correspondence with input tokens by preventing the model from relying on specific word orders. Additional training details are available in [Appendix A.2](#).

Training Details for Proposed Models

As motivated in [§3.3](#), we modify a middle encoder layer, specifically, the third of five layers and the fifth of eight layers in our respective models. To analyze the effects of our interventions, we establish two experimental conditions, which we refer to by the labels “Residual” and “Query” in our results. The “Residual” condition denotes the removal of the residual connection, while the “Query” condition involves replacing the standard self-attention query with a position-based query. This query’s projection basis is a positional encoding with a wave length of 100.

³The concatenation of language embedding and decoder word embedding is then projected down to the embedding dimension to form the input embedding to the decoder.

Dataset	Supervised Directions			Zero-Shot Directions			
	Baseline	Residual	+Query	Baseline	Residual	+Query	Pivot
(1) IWSLT 2017	29.8	29.4	29.4	10.8	17.7 (+6.9)	17.8	19.1
(2) Europarl multiway	34.2	33.9	33.1	11.3	26.1 (+14.8)	25.1	25.9
(3) Europarl disjoint	35.6	35.4	34.9	8.2	26.7 (+18.5)	25.8	27.1
(4) PMIndia	30.4	29.9	29.2	0.8	2.3 (+1.5)	1.1	22.1

TABLE 3.4: BLEU scores on supervised and zero-shot directions. On IWSLT and Europarl (rows (1) – (3)), removing residual connections once substantially improves zero-shot translation while retaining performance on supervised directions. On PMIndia from row (4), our approach can be improved further by additional regularization (Table 3.6). Detailed scores per language pair are in Appendix A.4.

Evaluation

For IWSLT, we use the official tst2017 set. For PMIndia, as the corpus does not come with dev and test sets, we partition the dataset ourselves by creating a multiway subset of all languages, resulting in 1,695 sentences in the dev and test set each. For Europarl, we use the test sets from the MMCR4NLP corpus (Dabre and Kurohashi, 2017).

3.5 Results

3.5.1 Effect of Positional Disentanglement

Our approach substantially improves zero-shot translation quality, as summarized in Table 3.4. The first observation is that modification in residual connections is essential for zero-shot performance⁴. We gain 6.9 and up to 18.5 BLEU points over the baseline on IWSLT and Europarl (rows (1) to (3)) respectively. When inspecting the model outputs, we see that the baseline often generates off-target translation in English, in line with observations from prior works (Arivazhagan et al., 2019a; Zhang et al., 2020a). Our proposed models are not only consistent in generating the required target languages in zero-shot conditions, but also show competitive performance to pivoting via English. The effects are particularly prominent between related languages. As shown in Table 3.5, on Europarl, zero-shot outperforms the pivoting when translating between languages from the same families.

In the highly challenging case of the PMIndia dataset (row (4)), while removing the residual connection does improve the zero-shot performance, the score of 2.3 indicates that the outputs are still far from being useful. Manual inspection confirms that generated translations are mostly English instead of the desired output language. Nonetheless, we are able to remedy this by further regularization in §3.5.2.

Contrary to the large gains by removing residual connections, the attention query modification is not effective when combined with residual removal. This suggests that the primary source of position-specific representation is the residual connections.

⁴We also experimented with: 1) removing the residual in more layers, but observed large negative impact on convergence; 2) replacing the residual connections by meanpooled sentence embeddings, but the gains on zero-shot directions were less than removing the residual connections.

3.5. Results

Dataset	Family	Baseline	Pivot	Residual
Europarl multiway	Germanic	6.9	25.9	26.2 (+0.3)
	Romance	10.2	32.8	33.1 (+0.3)
Europarl disjoint	Germanic	11.8	24.8	25.5 (+0.7)
	Romance	13.5	31.0	32.3 (+1.3)

TABLE 3.5: Zero-shot BLEU scores between languages of the same families on Europarl multiway and disjoint (rows (2) and (3) from Table 3.4). Our approach outperforms pivoting via English.

Moreover, by contrasting rows (2) and (3) of Table 3.4, we show the effect of training data diversity. In practice, the parallel data from different language pairs are often to some degree multiway. Multiway data could provide an implicit bridging that facilitates zero-shot translation. With disjoint data, gains can come from training with a larger variety of sentences. Given these two opposing hypotheses, our results suggest that the diverse training data is more important for both supervised and zero-shot performance. With disjoint data, we first observe improved supervised translation performance by around 1.5 points for all three model configurations (Baseline, Residual, Residual+Query). Meanwhile, the zero-shot score also increases from 26.1 to 26.7 points with our model (Residual). The baseline, on the contrary, loses from 11.3 to 8.2 points. This suggests that our model can better utilize the diverse training data than the baseline under zero-shot conditions.

3.5.2 Effect of Additional Regularization

In §3.4.2, we hypothesized that variational dropout helps reduce position-specific representation. Table 3.6 shows the outcome of replacing the standard dropout by this technique. First, variational dropout also improves zero-shot performance over the baseline, yet not as strongly as residual removal. On IWSLT and Europarl, there is no additive gain by combining both techniques. On PMIndia, however, combining our model and variational dropout is essential for achieving reasonable zero-shot performance, as shown by the increase from 2.3 to 14.3 points. Why is the picture different on PMIndia? We identify two potential reasons: 1) the low lexical overlap⁵ among the languages (8 different scripts in the 9 Indian languages); 2) the extreme low-resource condition (30K sentences per translation direction on average).

To understand this phenomenon, we create an artificial setup based on IWSLT with 1) no lexical overlap by appending a language tag before each token; 2) extremely low resource by taking a subset of 30K sentences per translation direction. The scores in Table 3.7 show the increasing benefit of variational dropout given very low amount of training data and shared lexicon. We interpret this through the lens of generalizable representations: With low data amount or lexical overlap, the model tends to represent its input in a highly language-specific way, hence hurting zero-shot performance.

⁵We also tried mapping the 9 Indian languages into the Devanagari script, but got worse zero-shot performance compared to the current setup.

Dataset	Zero-Shot Directions			
	Baseline	+vardrop	Residual	+vardrop
IWSLT	10.8	14.9	17.7	17.7
Europarl	8.2	25.1	26.7	26.4
PMIndia	0.8	2.4	2.3	14.3

TABLE 3.6: Zero-shot BLEU scores by variational dropout (“+vardrop”) on IWSLT, Europarl disjoint, and PMIndia. On the first two datasets, combining residual removal and variational dropout has no synergy. On PMIndia with little data and low lexical overlap, the combination of the two is essential.

	Condition	Residual	+ vardrop
(1)	Normal	17.7	17.7 (+0.0)
(2)	(1)+little data	11.9	12.9 (+1.0)
(3)	(2)+no lexical overlap	9.7	12.2 (+2.5)

TABLE 3.7: Zero-shot BLEU scores on a subset of IWSLT artificially constructed with little training data and no shared lexicon. The benefit of regularizing by variational dropout becomes prominent as the amount of training data and shared lexicon decreases.

3.5.3 Adaptation to Unseen Language

So far our model has shown promising zero-shot performance. Here we extend the challenge of zero-shot translation by integrating a new language. Specifically, we fine-tune a trained English-centered many-to-many system with a new language using a small amount of $X_{\text{new}} \leftrightarrow \text{English}$ parallel data. At test time, we perform zero-shot translation between X_{new} and all non-English languages previously involved in training. This practically simulates the scenario of later acquiring parallel data between a low-resource language and the central bridging language in an existing system. After fine-tuning with the new data, we can potentially increase translation coverage by $2N$ directions, with N being the number of languages originally in training. We fine-tune a trained system on IWSLT (Row 1 in Table 3.4) using a minimal amount of $\text{de} \leftrightarrow \text{en}$ data with 14K sentences. When fine-tuning we include the original $X_{\text{old}} \leftrightarrow \text{en}$ training data, as a form of rehearsal (Robins, 1995) to avoid losing performance on previously learned directions. This procedure is lightweight, since the model has already converged on the original training data.

In Table 3.8, our model outperforms the baseline on zero-shot translation, especially when translating from the new language ($X_{\text{new}} \rightarrow$). When inspecting the outputs, we see the baseline almost always translates into the wrong language (English), causing the low score of 1.8 BLEU. We hypothesize that the baseline overfits more on the supervised direction ($X_{\text{new}} \rightarrow \text{en}$), where it achieves the higher score of 18.5 BLEU. In contrast, our model is less susceptible to this issue and consistently stronger under zero-shot conditions.

3.6 Analyses

To see beyond BLEU scores, we first analyze how much position- and language-specific information is retained in the encoder hidden representations before and after applying our

3.6. Analyses

	Supervised		Zero-Shot	
	Baseline	Residual	Baseline	Residual
$X_{\text{new}} \rightarrow$	18.5	17.3	1.8	6.7
$\rightarrow X_{\text{new}}$	13.6	13.4	8.3	10.7

TABLE 3.8: Effects of adaptation to new language (de $\leftrightarrow$ en on IWSLT). Zero-shot translation directions are de $\leftrightarrow$ {it, nl, ro}. Our model has significantly stronger zero-shot performance.

approaches. We then study circumstances where zero-shot translation tends to outperform its pivoting-based counterpart. Lastly, we discuss the robustness of our approach to the impact of different implementation choices.

3.6.1 Impact on Positional Correspondence

We first revisit the probing experiments in §3.3.1, and investigate the impact of our proposed approach. Table 3.9 compares the classification accuracy of the baseline and our model. First, the baseline encoder output has an exact one-to-one correspondence to the input tokens, as evidenced by the nearly perfect accuracy when recovering token IDs. This task becomes much more difficult under our model. We see a similar picture when recovering the position IDs.

Dataset	Model	Token ID	Position ID
IWSLT	Baseline	99.9	93.3
	Residual	48.5	51.4
Europarl disjoint	Baseline	99.5	85.1
	Residual	71.6	22.5
PMIndia	Baseline	99.6	90.1
	Residual	63.3	26.9

TABLE 3.9: Accuracy (%) of classifiers trained to recover input positional information (**token** ID or **position** ID) based on encoder outputs. **Lower** values indicate higher difficulty of recovering the information, and therefore **less** positional correspondence to the input tokens.

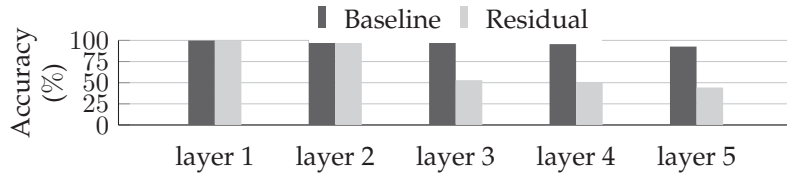


FIGURE 3.3: Accuracy of recovering position IDs after each encoder layer on IWSLT. When we remove the residual connection in the 3rd encoder layer, classification is much more difficult.

We also try to recover the position IDs based on the outputs from each layer. As demonstrated in Figure 3.3, the accuracy drops sharply at the third layer, where the residual connection is removed. This indicates that the devised transition point at a middle encoder layer is effective.

3.6.2 Impact on Language Invariance

To validate whether our approach indeed leads to more language-invariant representations as hypothesized, we assess the similarity of encoder outputs on the sentence and token level using the two following methods:

SVCCA Score

The singular vector canonical correlation analysis (SVCCA) (Raghu et al., 2017) measures similarity of neural network outputs, and has been used to assess representational similarity in multilingual translation (Kudugunta et al., 2019). SVCCA scores range from 0 to 1. As SVCCA operates on fixed-size inputs, we meanpool the encoder outputs and measure similarity on a sentence level.

Language Recovery Accuracy

Since more similar representations are more difficult to distinguish, poor performance of a language classifier indicates high similarity. Based on a trained model, we train a token-level linear projection from the encoder outputs to the number of classes (languages).

Findings

As shown in Table 3.10, our model consistently achieves higher SVCCA scores and lower classification accuracy than the baseline, indicating more language-invariant representations. When zooming into the difficulty of classifying the languages, we further notice much higher confusion (therefore similarity) between related languages. For instance, Figure 3.4 shows the confusion matrix when classifying the 8 non-English languages in Europarl. After residual removal, the similarity is much higher within the Germanic and Romance family. This also corresponds to cases where our model outperforms pivoting (Table 3.5).

Dataset	Model	SVCCA Score (higher: more similar) Range: [0, 1]	Language Recovery Accuracy (%) (lower: more similar)
IWSLT (4 languages)	Baseline	0.682	95.9
	Residual	0.703	87.6
Europarl disjoint (9 languages)	Baseline	0.652	87.0
	Residual	0.680	69.9
PMIndia (10 languages)	Baseline	0.621	74.1
	Residual	0.650	62.0

TABLE 3.10: Average pairwise similarity of encoder outputs between all languages in each dataset. Higher SVCCA scores and lower classification accuracy indicate higher similarity. We note that the SVCCA score between random vectors is around 0.57.

Moreover, we compare the SVCCA scores after each encoder layer, as shown in Figure 3.5. Confirming our hypotheses, the model outputs are much more similar after the

3.6. Analyses

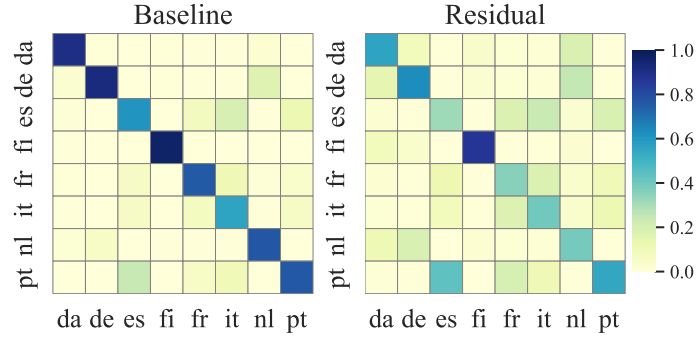


FIGURE 3.4: Confusion matrices when classifying languages in Europarl disjoint (x: true, y: predicted). Encoder outputs of related languages (Romance, Germanic families) are more similar after residual removal.

transition layer, as shown by the sharp increase at layer 3. This contrasts with the baseline, where similarity increases nearly linearly.

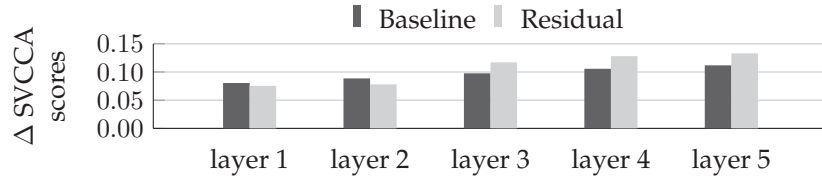


FIGURE 3.5: Average SVCCA scores after each encoder layer between all language pairs in IWSLT ({it, nl, ro, en}). Scores are reported additive to scores between random vectors. Similarity significantly increases after the 3rd layer where we apply residual removal.

Given these findings and previous analyses in §3.6.1, we conclude that our devised changes in a middle encoder layer allow higher cross-lingual generalizability in top layers while retaining the language-specific bottom layers.

3.6.3 Qualitative Analysis

In §3.5 we have shown that between related languages zero-shot translation surpasses pivoting performance. Here we manually inspect some pivoting translation outputs (nl→en→de) and compare them to zero-shot outputs (nl→de). In general, we observe that the translations without pivoting are much more similar to the original sentences. For instance in Table 3.11, when pivoting, the Dutch sentence “*geven het voorbeeld* (give the example)” is first translated to “set the example”, then to “*setzen das Beispiel* (set the example)” in German, which is incorrect as the verb “*setzen* (set)” cannot go together with the noun “*Beispiel* (example)”. The zero-shot outputs, on the other hand, directly translate “*geven* (give; Dutch)” to “*geben* (give; German)”, resulting in a more natural pairing with “*Beispiel* (example)”. With this example, we intend to showcase the potential of bypassing the pivoting step and better exploiting language similarity.

Stage	Text
Input (nl)	... geven in dit verband het verkeerde voorbeeld, maar anderen helaas ook.
Pivot-in (nl→en)	... are setting the wrong example here, but others are unfortunately also.
Pivot-out (en→de)	... setzen hier das falsche Beispiel ein, andere sind leider auch.
Zero-shot (nl→de)	... geben in diesem Zusammenhang das falsche Beispiel, aber leider auch andere.

TABLE 3.11: Comparison of pivoting (nl→en→de) versus zero-shot translation (nl→de). Pivoting via English produces the incorrect verb-noun collocation *setzen das Beispiel* (literal: “set the example”) in German, whereas zero-shot translation leverages cross-lingual similarity to generate the correct idiomatic expression *geben das Beispiel* (“give the example”).

3.7 Ablations

3.7.1 Where to Remove Residual Connections

In our main experiments, all proposed modifications take place in a middle encoder layer. To validate this choice, we conduct an ablation study on the IWSLT dataset, individually removing the residual connection from each of the five encoder layers. As shown in Table 3.12, removing the residual connection from the first encoder layer degraded zero-shot performance by 2.6 BLEU, which suggests that the model requires strong positional awareness in its lowest layers to process the input sequence effectively. In contrast, applying the removal to the top layers (the fourth or fifth) slowed convergence and resulted in a 0.4 BLEU loss on supervised translation tasks. We attribute this to impaired gradient flow to the lower layers, which hinders learning. These two observations support our strategy of modifying a middle encoder layer. The initial layers appear crucial for encoding positional information, while the final layers are necessary for stable training. A middle layer therefore serves as a transition point from position-aware to more abstract representations. For deeper models with tens of encoder layers, the modifications should instead focus on the lower-to-middle block of layers to ensure gradient flow.

Residual Removal Location	Supervised		Zero-Shot	
	Baseline	Residual	Baseline	Residual
Layer 3 of 5 (row (1) of Table 3.4)		29.4		17.7
Layer 1 of 5		29.1		15.1
Layer 2 of 5	29.8	29.1	10.8	17.7
Layer 4 of 5		29.0		17.6
Layer 5 of 5		29.0		17.4

TABLE 3.12: BLEU scores on the IWSLT dataset, comparing different residual removal locations out of the 5 encoder layers. Bottom layer removal degrades performance, while top layer removal leads to slower convergence.

3.8. Combination with Alignment Losses

Approach	Supervised		Zero-Shot	
	Baseline	Residual	Baseline	Residual
(1) Absolute pos. encodings (row (1) of Table 3.4)	29.8	29.4	10.8	17.7 (+6.9)
(2) Learned pos. embeddings	29.6	29.1	4.8	17.1 (+12.3)

TABLE 3.13: BLEU scores on the IWSLT dataset validating our approach with learned positional embeddings, comparing with our modified model with a standard Transformer baseline with 5 encoder and decoder layers each. Our approach remains effective with learned positional embeddings.

3.7.2 Learned Positional Embeddings

To validate that our findings are independent of the specific positional encoding scheme, we replicate our experiments on the IWSLT dataset using learned positional embeddings instead of the fixed sinusoidal encodings from our main experiments. The results in Table 3.13 are consistent with our previous observations. The baseline model using learned embeddings still suffers from poor zero-shot translation performance, achieving 4.8 BLEU on average. In contrast, our residual removal approach is still effective, substantially improving the zero-shot score to 17.1 BLEU while maintaining a comparable supervised score of 29.1 BLEU. These findings demonstrate that our approach is robust across different forms of positional embeddings. We hypothesize that even if learned embeddings are more language-agnostic, the residual connections throughout the full network still enforce a strong positional correspondence to the input sequence. Since our approach is designed to break this correspondence, its effectiveness should not depend on the specific type of positional encoding.

3.8 Combination with Alignment Losses

Among the prior work on zero-shot translation (§3.2), a prominent direction is to actively encourage similar representations for sentences with the same meaning. This contrasts with our approach, which aims for a similar outcome implicitly via architectural modification. In this section, we investigate whether these two strategies are complementary, focusing on a similarity regularization (§3.8.1) and adversarial language classification (§3.8.2). We further describe our implementation improvements of the latter for stable training.

3.8.1 Similarity Regularization

The principle behind this approach is that source and target sentences in a parallel corpus are semantically equivalent, and therefore should be mapped to nearby points in the model’s representation space. As demonstrated by prior works (Arivazhagan et al., 2019a; Pham et al., 2019), introducing an auxiliary loss that promotes such similarity can significantly improve zero-shot translation.

Given a source sentence X and a target sentence Y from a parallel pair, we first compute their sentence-level representations by applying a meanpooling operation over the encoder’s output hidden states. This step produces fixed-size vectors, accommodating sentences of different lengths. We then define the similarity loss using a chosen distance metric $\text{dist}(\cdot)$ between these two sentence representations:

$$\mathcal{L}_{\text{sim}} = \text{dist}(\text{meanpool}(\text{ENCODER}(X)), \text{meanpool}(\text{ENCODER}(Y))). \quad (3.1)$$

The full model is then trained to minimize a combined objective function that includes both the standard machine translation loss and this new similarity term:

$$\mathcal{L}_{\text{full}} = \mathcal{L}_{\text{MT}} + \lambda \cdot \mathcal{L}_{\text{sim}}, \quad (3.2)$$

where $\mathcal{L}_{\text{MT}}$ is the standard cross-entropy loss, and λ is a hyperparameter that scales the contribution of the auxiliary loss.

3.8.2 Adversarial Language Classification

A drawback of the similarity regularizer is the need to manually select pooling methods and distance metrics. A more data-driven method is to remove source language signals by adversarial training (Ganin and Lempitsky, 2015). We extend the adversarial language classification approach from Arivazhagan et al. (2019a) with two modifications. First, we introduce a modified adversarial loss function that addresses gradient flow issues in the original formulation. Second, we apply language classification at the token level rather than the sentence level, eliminating the need to manually select a pooling method.

Standard Adversarial Loss

The adversarial approach trains a language classifier that attempts to predict the source language from encoder representations, while simultaneously training the encoder to produce representations that deceive this classifier. The language classifier minimizes the standard cross-entropy loss:

$$\mathcal{L}_{\text{classifier}} = - \sum_{c=1}^L y_c \log(p_c), \quad (3.3)$$

where L is the number of classes to predict, y_c is a binary indicator whether the true language label is c , and p_c is the predicted probability for the instance belonging to language c . The encoder is updated adversarially using a gradient reversal layer (Ganin and Lempitsky, 2015), which reverses the gradients from the language classifier during backpropagation. This encourages the encoder to learn representations that are invariant to source language identity. We perform a binary classification to distinguish between English and non-English. This approach follows Arivazhagan et al. (2019a) and is further supported by our initial experiments showing that a binary setup outperforms multiclass classification.

3.8. Combination with Alignment Losses

Modified Adversarial Loss

An issue of the standard approach in Equation 3.3 is that when the language classifier performs well (high accuracy), the loss landscape becomes relatively flat, resulting in small gradients flowing back to the encoder. This is problematic because high classifier accuracy indicates that the encoder representations contain strong language-specific signals, precisely when we need large gradients to drive the encoder toward language-invariant representations. To address this issue, we propose a modified adversarial loss function:

$$\mathcal{L}_{\text{adv_classifier}} = \sum_{c=1}^L y_c \log(1 - p_c). \quad (3.4)$$

This formulation effectively mirrors the standard loss in Equation 3.3 across both the horizontal axis and the vertical line defined by $x = 0.5$. Crucially, this modification preserves the optimization direction while amplifying gradient magnitudes when the classifier performs well, ensuring stronger adversarial updates when they are most needed.

Training Procedure

The full training objective combines the original translation loss with our modified adversarial loss (Equation 3.4). For training stability, we alternate optimization steps between the two losses. Optimizing both jointly would lead to co-adaptation of the parameters of the translation and classification modules and empirically led to training instability.

Further implementation and training details of both the similarity regularization and adversarial approaches are provided in Appendix A.5.

3.8.3 Combining Residual Removal and Alignment Losses

Table 3.14 shows the results of combining our residual removal approach with previously proposed alignment losses. The results indicate that these strategies are complementary for improving zero-shot translation. Across all configurations, models that incorporate our method (rows (2), (4), (6), (8)) consistently outperform their counterparts without it (rows (1), (3), (5), (7)). The most significant gains are observed on PMIndia, which is the most challenging dataset due to its small data size and low lexical overlap.

For supervised translation directions, applying our residual removal method results in a slight degradation of up to 0.7 BLEU, again most notably on PMIndia. This suggests a potential trade-off between learning the language-invariant representations beneficial for zero-shot tasks and retaining the language-specific features that aid supervised translation.

It is worth noting that on the two Europarl datasets, our best-performing model performs on par with the strong pivot-based methods. However, on the more challenging IWSLT and PMIndia datasets, a considerable gap remains compared to pivoting (1.1 and 5.0 BLEU respectively).

In addition, the combination of residual removal and adversarial losses has been shown effective in cross-lingual summarization (Solovyev et al., 2024), a task that combines

Dataset	Supervised Directions			Zero-Shot Directions			
	Baseline	Similarity	Adversarial	Baseline	Similarity	Adversarial	Pivot
(1) IWSLT 2017	30.0	29.5	29.5	10.9	16.7	16.8	19.1
(2) + Residual	29.6	29.1	29.1	17.9	17.1	18.0	
(3) Europarl disjoint	34.5	34.4	34.4	6.9	25.1	25.2	25.7
(4) + Residual	34.1	34.1	34.1	23.2	25.4	25.5	
(5) Europarl full	34.7	35.3	34.5	13.4	24.5	25.3	26.0
(6) + Residual	34.5	35.0	34.7	25.2	26.5	26.1	
(7) PMIndia	30.5	30.0	30.4	2.4	8.9	7.3	22.1
(8) + Residual	29.8	29.5	29.6	14.3	15.7	17.1	

TABLE 3.14: BLEU scores before and after combining residual removal with the baseline, similarity regularization, adversarial classification. Residual removal consistently improves zero-shot performance with minimal impact on supervised performance.

machine translation and text summarization. The language identity probing experiments in that work have shown that combining the adversarial language classifier with residual removal increases language invariance.

3.9 Summary

This chapter investigates an essential generalization ability of neural machine translation models, namely their ability to combine knowledge from trained languages to perform zero-shot translation. As our initial endeavor into representations that generalize across languages, this chapter focuses on models trained from random initialization, as it provides the maximum degree of architectural freedom. Unlike using pre-trained models where the architecture is predefined, training from scratch allows analyzing the impact of architectural design choices on generalization.

We reveal that the strong positional correspondence to input tokens in standard models hinders zero-shot translation by enforcing the creation of language-specific representations. Specifically, we demonstrate that: **1)** the encoder outputs retain word orders of source languages; **2)** this positional information reduces cross-lingual generalizability and therefore zero-shot translation quality; **3)** these issues can be simply yet effectively alleviated by removing the residual connection in a single middle encoder layer. With this simple modification, we achieve substantial improvements on zero-shot translation. On related languages, our proposed model is able to surpass pivot-based translation, a long-standing performance ceiling for zero-shot translation performance.

With extensive analyses of model internal representations, we confirm that the encoder outputs with our proposed approach are more language-invariant. Furthermore, we demonstrate a clear synergy between our architectural modification and prior methods with explicit alignment objectives. When combined, our residual removal method and an explicit similarity loss consistently outperform each approach applied individually. These alignment objectives will be further investigated in the next chapter on adapting pre-trained models.

Chapter 4

Transfer with Translation-Specific Pre-Trained Models

Multilingual machine translation research has been transformed by large-scale data collection and pre-training efforts, resulting in abundant openly available training data (Schwenk et al., 2021a,b) and pre-trained models (Fan et al., 2021; NLLB Team, 2024). This development alters two assumptions that guided our previous investigation of cross-lingual transfer. First, the classical notion of “zero-shot” translation, where a model has zero exposure to a language pair, has become increasingly rare. Second, the fixed architectures of the pre-trained models preclude modifications during subsequent adaptation stages.

While the context has changed, the core challenge of learning generalizable representations persists in a new form. Pre-trained models typically require additional specialized training to achieve optimal performance on specific tasks, which introduces tensions between adapting a model to a subset of languages and preserving its performance on all others. This challenge is compounded by the inability to alter the model’s architecture, which calls for solutions that operate within these constraints.

The central question therefore becomes: how can we adapt a pre-trained multilingual translation model to a smaller set of languages while preserving, or ideally improving, performance across all language pairs? This problem is analogous to the zero-shot translation scenarios investigated in the previous chapter.

Inspired by the established success of pivot-based translation which relies on an intermediate language to bridge pairs in zero-shot settings, we propose an approach that simulates this mechanism by learning a pseudo-pivoting language. This intermediate pseudo-language serves a dual purpose, as a mechanism to enhance transfer to unseen language pairs and as an analytical probe into the model’s internal representations. This chapter is partly based on Liu and Niehues (2022).

4.1 Introduction

Models pre-trained on large general-purpose datasets have become fundamental components for improving performance on downstream tasks, including machine translation. These pre-trained models are subsequently further trained, or *fine-tuned*, to optimize performance on tasks of interest. Compared to random initialization, pre-training enhances

source sentence	learning	a	new	language
(English)	↓	↓	↓	↓
discrete codes	3	609	57	1042
source sentence	belajar	bahasa	baru	
(Indonesian)	↓	↓	↓	
discrete codes	3	57	258	

TABLE 4.1: We learn a sequence of discrete codes to represent source sentences, aiming to improve inference-time robustness and interpretability in multilingual translation models. The resulting codes do not necessarily maintain a one-to-one correspondence with input words or tokens.

robustness against adversarial inputs (Hendrycks et al., 2019) and accelerates convergence during fine-tuning (He et al., 2019).

Two categories of pre-trained models can be used for initialization for machine translation models. *Task-agnostic* pre-trained models, such as mBART (Liu et al., 2020) or multilingual variants of large language models (AI @ Meta et al., 2024), encode general linguistic knowledge but lack task-specific biases. In contrast, *task-specific* pre-trained models, such as M2M-100 (Fan et al., 2021) and NLLB-200 (NLLB Team, 2024), are explicitly trained on tens of thousands of language pairs.

While powerful, even massive task-specific models require further fine-tuning to achieve optimal performance on specific language pairs. This specialization, however, introduces the risk of catastrophic forgetting (McCloskey and Cohen, 1989), a general machine learning challenge not only present in NLP. When a model is fine-tuned on a limited set of language pairs, it may lose its pre-trained abilities on languages that are not present in the fine-tuning data. This issue is an extension of the zero-shot translation problem explored in the previous chapter, and is the focus of this chapter. While the previous chapter addressed language directions completely absent from training data, this chapter investigates how pre-existing knowledge on language pairs unseen in fine-tuning can be forgotten.

The use of models pre-trained on massive multilingual datasets presents a particular challenge for mitigating catastrophic forgetting. Although established techniques such as rehearsal (Robins, 1995) and Elastic Weight Consolidation (Kirkpatrick et al., 2017) can alleviate this issue, they typically rely on access to at least a subset of the original training data. However, given the scale and language coverage of the pre-training data, retaining and reusing them is often infeasible.

Our proposed approach is motivated by a consistent finding in the literature: the superior performance of pivot-based translation over direct translation for unseen language pairs (Al-Shedivat and Parikh, 2019; Arivazhagan et al., 2019a; Zhu et al., 2020a; Zhang et al., 2020a; Yang et al., 2021). We hypothesize that the advantage of pivot-based systems results from the nature of their intermediate representation. Direct translation models map source text to a continuous vector space, which can be brittle and fail to generalize to unseen data. In contrast, pivot-based systems force information through a discrete

4.1. Introduction

bottleneck, i.e., the token sequence in the pivot language. This discretization may act as a regularizer, constraining the model to form more robust and generalizable representations, thereby mitigating performance degradation on language pairs absent in fine-tuning.

Another motivation is from the long-standing concept of constructed languages, such as *Interlingua* and *Esperanto*. These languages exemplify human-designed linguistic structures based upon commonalities across related languages. These languages demonstrate the feasibility of a universal linguistic system designed to capture commonalities across multiple languages, effectively serving as a human-engineered intermediate representation. While conceptually elegant, such languages are impractical for data-driven models due to the severe scarcity of parallel data, which would require extensive expert curation.

Therefore, instead of relying on manually-created linguistic resources, we propose to learn an artificial intermediate language in an unsupervised manner. This leads to two primary objectives: The theoretical objective is to learn sequences of discrete tokens capable of representing source sentences across diverse languages, which subsequently serve as context for the machine translation decoder, as illustrated in [Table 4.1](#). The practical objective is to leverage this approach to enhance performance on language pairs absent from the fine-tuning data of pre-trained translation models.

Both objectives align with our overarching goal of learning generalizable representations. By forcing information through a discrete bottleneck, we restrict the model’s representational capacity, which encourages greater sharing of representations across different source languages. A model that fosters such generalization is inherently more robust to the exclusion of language pairs during fine-tuning, making it suitable for combating catastrophic forgetting.

4.1.1 Task Definition

In this chapter, we study the following task:

Given a pre-trained translation model supporting $N(N - 1)$ translation directions across N languages, and parallel data on a subset of m languages each paired with a common bridge language (therefore $m + 1$ languages in total), how to fine-tune the pre-trained model on these $2m$ directions while retaining pre-trained knowledge on the remaining directions, particularly on those $\underbrace{m(m + 1)}_{\text{all pairs for } m + 1 \text{ languages}} - \underbrace{2m}_{\text{fine-tuned pairs}}$ directions where both the source and target languages were part of the fine-tuning data, but not directly paired for training?

This problem definition addresses both compositional and temporal generalization. Compositional generalization is tested by evaluating the $m(m + 1) - 2m$ directions, for instance, translating directly from language A to B after being fine-tuned only on A-to-bridge and bridge-to-B translation. Meanwhile, the temporal dimension is addressed by assessing the model’s ability to retain its performance on pre-trained directions not included in the fine-tuning data.

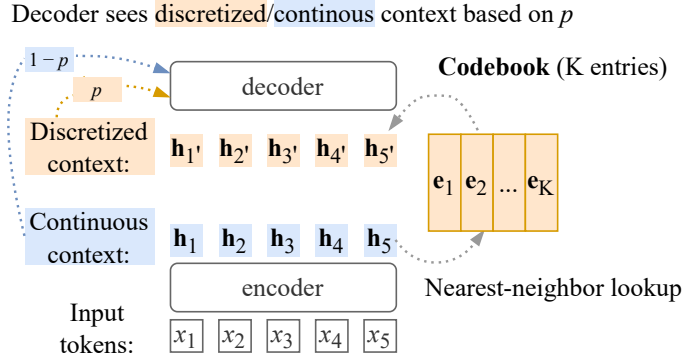


FIGURE 4.1: Our method discretizes encoder outputs by a discrete codebook. Training alternates between discretized and continuous context (probability p), while inference uses the resulting clustered continuous representations.

4.1.2 Contributions

In this chapter, we address a manifestation of catastrophic forgetting, where standard English-centric fine-tuning of pre-trained multilingual models degrades performance on language pairs absent from the adaptation data. To mitigate this, we propose using discrete intermediate representations as a regularizer to preserve pre-trained knowledge while the model adapts. Our method maps each encoder hidden state to an entry in a fixed-size codebook, which effectively clusters the continuous representations around a finite number of centroids in the latent space. The goal is to strike a balance between representational expressiveness and the regularizing effect of discretization, as illustrated in Figure 4.1.

Specifically, we make the following contributions:

- We propose a framework to learn discrete tokens as intermediate representations of Transformer-based models (§4.4). On large-scale multilingual translation experiments, our approach surpasses direct fine-tuning in preventing catastrophic forgetting while offering more interpretable intermediate representations (§4.6.2).
- We demonstrate that using linguistically similar bridge languages yields stronger performance and improved data efficiency compared to English as a bridge language (§4.6.1).
- We leverage the learned discrete codes to analyze the role of bridging languages through two novel diagnostic tasks: discrete code overlap and discrete code translation. These analyses reveal that linguistically similar bridge languages facilitate knowledge-sharing across all languages covered by the model (§4.6.3).

4.2 Related Work

Several lines of research are relevant to this chapter, including robustness to unseen test conditions (§4.2.1), discrete representations (§4.2.2), and analyses of model internal representations (§4.2.3).

4.2.1 Robustness in Unseen Test Conditions

Generalization to unseen conditions is a central challenge in machine learning (Socher et al., 2013; Norouzi et al., 2014; Romera-Paredes and Torr, 2015; Xian et al., 2017). In machine translation, this primarily manifests as the task of zero-shot translation. Early multilingual models demonstrated a surprising capability to translate between language pairs they were never explicitly trained on (Johnson et al., 2017). However, this zero-shot performance has proven to be notoriously brittle and sensitive to a wide range of factors, including the diversity of training data (Rios et al., 2020), language token strategies (Wu et al., 2021; ElNokrashy et al., 2022), and dropout configurations (Arivazhagan et al., 2019a; Liu et al., 2021b).

A primary failure mode for these models is off-target translation (Zhang et al., 2020a), where the model generates output in an entirely incorrect language. The unreliability of zero-shot inference has led practitioners to favor more robust methods. In competitive shared tasks (Anastasopoulos et al., 2021; Libovický and Fraser, 2021a), for example, the dominant strategy has been to eliminate zero-shot conditions by generating synthetic parallel data through back-translation (Pham et al., 2021a; Zhang and Sennrich, 2021; Liu and Niehues, 2021; Knowles and Larkin, 2021; Libovický and Fraser, 2021b).

While converting the problem from a zero-shot to a semi-supervised setting is a pragmatic solution, it re-frames the problem rather than addressing the model’s intrinsic generalization capabilities. Fully realizing the promise of knowledge sharing in multilingual systems depends on developing methods that can reliably generalize without relying on such workarounds.

4.2.2 Discrete Representations

Quantization techniques are commonly applied to discretize continuous latent spaces by constraining the number of representation options. Vector Quantized Variational Autoencoder (VQ-VAE) (van den Oord et al., 2017) is a representative approach that learns discrete tokens for continuous inputs such as images and audio, demonstrating effectiveness in creating discrete representations for audio processing (Tjandra et al., 2020; Baevski et al., 2020).

Kaiser et al. (2018) proposed an improvement to VQ-VAE, which decomposes quantization input and output into several subspaces. Termed *slicing*, this technique splits the feature dimension into multiple groups and creates a discrete code for each of the groups. This technique is effective in safeguarding against representation collapse, where most representations get mapped to single points. Kaiser et al. (2018) applied this slicing

variant to auto-encoding for learning more compact sequence representations, enabling accelerated output generation in auto-regressive decoders.

The most related work to our approach is that of [Escolano et al. \(2021\)](#), who applied sliced VQ-VAE ([Kaiser et al., 2018](#)) on machine translation. Our work has several major differences: 1) Our focus is on fully parameter-shared multilingual systems, while [Escolano et al. \(2021\)](#) concentrated on auto-encoding and bilingual systems using language-specific encoders and decoders. 2) In [Escolano et al. \(2021\)](#), translation of unseen language pairs requires a subsequent training step with a dedicated encoder for the new language, while our work supports this directly after one step of training. 3) We extend sliced VQ-VAE ([Kaiser et al., 2018](#)) by soft codes that utilize both continuous and quantized encoder hidden states.

4.2.3 Analyzing Hidden Representations for Machine Translation

A wide range of methods exist to analyze knowledge sharing in multilingual machine translation ([Aji et al., 2020](#); [Mueller et al., 2020](#); [Chiang et al., 2022](#)). Most of these approaches either use translation performance as an indirect proxy or require complex post-processing after training, such as computing correlation scores between model hidden states ([Kudugunta et al., 2019](#); [Chiang et al., 2022](#)), training classifiers to probe linguistic features ([Liu et al., 2021b](#)), or pruning submodules of the model ([Kim et al., 2021](#)). In contrast, models whose hidden states are directly associated with discrete tokens offer greater interpretability. This property is particularly beneficial in unseen testing scenarios, where pinpointing the causes of model behavior is crucial.

4.3 When does Pre-Training Help?

As this is the first chapter to incorporate pre-trained models, we start with an analysis of the impact of initializing models with pre-trained weights, contrasting it with random initialization. We first examine the conditions under which pre-training is beneficial, evaluating its impacts on learning language-invariant representations (§4.3.2) and for improving translation performance (§4.3.3). We then identify the issue of catastrophic forgetting, where models lose pre-trained knowledge during fine-tuning (§4.3.4).

4.3.1 Setup

To evaluate the effect of pre-training, our experiments compare models initialized with weights from M2M-124¹ ([Goyal et al., 2022](#)) to models with random initialization. The M2M-124 model is an extension of M2M-100 ([Fan et al., 2021](#)), a many-to-many translation model, and increases the number of languages supported for direct translation from 100 to 124. Our experiments are conducted in two configurations, each centered on a different bridge language: English (en) or Indonesian (id). We focus on translation directions between the bridge language and three other languages: Javanese (jv), Malay (ms), and

¹In this chapter, we use the 175M parameter variant; the larger variant is studied later in [Chapter 6](#).

4.3. When does Pre-Training Help?

Tagalog (tl). This setup results in training data for the pairs $\text{en} \leftrightarrow \text{yv}$, $\text{en} \leftrightarrow \text{ms}$, and $\text{en} \leftrightarrow \text{tl}$ in the first configuration, and $\text{id} \leftrightarrow \text{yv}$, $\text{id} \leftrightarrow \text{ms}$, and $\text{id} \leftrightarrow \text{tl}$ in the second. Further details on the training data are presented in §4.5.1.

4.3.2 Pre-Training’s Impact on Language-Specific Signals

The previous chapter demonstrated that multilingual models retain strong language-specific signals in their intermediate representations. We now investigate whether this finding holds for models pre-trained on massively multilingual datasets. Specifically, we replicate the language identity probing experiments from the previous chapter’s Table 3.10, aiming to recover the source language from encoder representations. The results are presented in Table 4.2. First, the off-the-shelf pre-trained model still allows for easy identification of the source language as shown in row (1). The classification accuracy is well above the chance level of approximately 25%, indicating that significant language-specific information persists despite pre-training on over 100 languages. Continual training on our selected translation data does not consistently reduce this language specificity as shown in row (2). However, there is a clear distinction when comparing to training from scratch. The randomly initialized model develops far stronger language-specific signals, with classification accuracies exceeding 90% as shown in row (3). This comparison suggests that while large-scale pre-training does lead to more language-invariant representations than training from scratch, it only does so to a limited degree.

ID	Model	English-Bridge	Indonesian-Bridge
(1)	Pre-trained M2M-124	81.4	67.6
(2)	$\hookrightarrow$ Continue training on supervised data	77.1	85.5
(3)	Train from scratch with the same data	92.0	96.7

TABLE 4.2: Input language recovery accuracy (%), evaluating how well source language identity can be recovered from the encoder representations of (1) an off-the-shelf pre-trained M2M-124 model; (2) the same model after continual training on selected translation pairs; and (3) a randomly initialized model with the same architecture, trained from scratch on the same data. Each setup has 4 possible classes.

4.3.3 Comparison to Random Initialization

After examining pre-training’s impact on model internal representations, we analyze its impact on translation performance. We evaluate translation directions present in fine-tuning (Table 4.3) separately from those present in pre-training but absent in fine-tuning (Table 4.4).

In rows (1) and (4) of Table 4.3, we report results from directly running inference on the pre-trained M2M-124 model. This represents an extreme scenario where pre-trained models are available but no additional training on parallel data is possible. At the opposite extreme, rows (3) and (6) show performance of training with random initialization, representing a setting where parallel data is available but no pre-trained resources exist. Rows (2) and (5) demonstrate the conventional fine-tuning approach, where the pre-trained

ID	Model	$\{jv, ms, tl\} \rightarrow X$		$X \rightarrow \{jv, ms, tl\}$	
		BLEU	COMET	BLEU	COMET
English-Bridge (X=en)					
(1)	Pre-trained M2M-124	19.6	66.9	14.0	64.9
(2)	$\hookrightarrow$ Continue training on supervised data	27.8	78.4	27.1	72.7
(3)	Train from scratch with the same data	26.9	78.2	28.0	72.2
Indonesian-Bridge (X=id)					
(4)	Pre-trained M2M-124	20.0	72.5	14.7	71.5
(5)	$\hookrightarrow$ Continue training on supervised data	27.7	84.5	24.3	80.2
(6)	Train from scratch with the same data	27.6	83.9	24.4	79.4

TABLE 4.3: Average translation performance in BLEU($\uparrow$) and COMET($\uparrow$) on language directions involved in fine-tuning. Fine-tuning consistently improves performance over out-of-the-box baseline scores. Pre-training does not bring universal gains across all directions.

model is initialized and subsequently trained on the parallel data, thereby leveraging both resources. These fine-tuned models substantially outperform out-of-the-box usage of the pre-trained model (+9.6 BLEU on average), highlighting the importance of focusing on a subset of languages of interest.

Moreover, pre-training does not bring universal gains across all directions. Under the English-bridge condition, pre-training improves 0.9 BLEU over random initialization (26.9 $\rightarrow$ 27.8 BLEU). However, COMET improvements are more modest, with only one of three directions showing statistically significant improvements at $\alpha = 0.05$. Out-of-English directions show no improvement from pre-training. For the Indonesian-centric configuration, despite minimal BLEU changes, pre-training significantly improves COMET in 4 out of the 6 directions. The discrepancy between BLEU and COMET scores suggests that these translations may deviate more from the references while preserving semantics.

4.3.4 Catastrophic Forgetting

To systematically investigate catastrophic forgetting after fine-tuning, as shown in Table 4.4, we evaluate performance on three categories of language directions not involved in fine-tuning:

1. Unseen pairs: Both source and target languages appear individually in fine-tuning data but not in combination. This includes all translation directions between Javanese (jv), Malay (ms), and Tagalog (tl), e.g., Javanese to Malay.
2. Unseen source: Only the source language is absent from fine-tuning data, while the target language is present.
3. Unseen target: Only the target language is absent from fine-tuning data, while the source language is present.

For the two bridge language configurations, the unseen language is Indonesian (id) in the English-bridge setup and English in the Indonesian-bridge setup.

4.3. When does Pre-Training Help?

ID Model	Unseen Pair		Unseen Source		Unseen Target	
	BLEU	COMET	BLEU	COMET	BLEU	COMET
ENGLISH-BRIDGE						
(1) Pre-trained M2M-124	9.9	60.5	14.7	71.5	20.0	72.5
(2) $\hookrightarrow$ Continue training on supervised data	4.7	53.0	20.9	79.6	1.2	44.1
(3) Train from scratch with the same data	6.4	54.6	—	—	—	—
INDONESIAN-BRIDGE						
(4) Pre-trained M2M-124	9.9	60.5	14.0	64.9	19.6	66.9
(5) $\hookrightarrow$ Continue training on supervised data	17.9	74.6	13.1	63.1	4.9	57.9
(6) Train from scratch with the same data	16.0	73.0	—	—	—	—

TABLE 4.4: Average translation performance in BLEU($\uparrow$) and COMET($\uparrow$) on language directions not included in fine-tuning data. English-bridge fine-tuning degrades performance on unseen pairs. In both data conditions, catastrophic forgetting occurs on unseen target languages. Cells with “—” indicate non-applicable conditions, as models trained from scratch do not support unseen source or target languages.

Unseen Language Pairs

Performance on unseen language pairs varies substantially between data configurations. As shown in row (5) of Table 4.4, the Indonesian-bridge setup shows improvement from 9.9 to 17.9 BLEU after fine-tuning, while the English-bridge configuration exhibits degradation from 9.9 to 4.7 BLEU. This divergence indicates that bridge language selection influences the model’s ability to preserve translation capabilities for unseen language combinations. In addition to results in the table, pivoting through the bridge language (source $\rightarrow$ bridge $\rightarrow$ target) achieves 19.7 and 17.5 BLEU for Indonesian- and English-bridge configurations respectively, outperforming direct translation. This difference further motivates methods that preserve pivoting-like behavior in direct translation systems.

Unseen Source Languages

Both bridge configurations maintain performance on unseen source languages without significant degradation. The English-bridge setup even improves on unseen source languages relative to the baseline. This preservation of input processing capabilities indicates that the model’s encoder representations remain sufficiently generalizable across languages, even after specialized fine-tuning.

Unseen Target Languages

Fine-tuned models exhibit severe performance degradation for unseen target languages, achieving BLEU scores of 1.2 and 4.9. This degradation occurs consistently across both bridge configurations, indicating substantial loss of generation capabilities for languages absent from fine-tuning data. The consistent difficulty in generating unseen target languages suggests that this is a fundamental limitation rather than a data-specific artifact.

4.3.5 Main Findings in Initial Analysis

In this initial analysis, we demonstrate that the benefits of pre-trained initialization are conditional, with primary improvements when translating into English. We also identify catastrophic forgetting as a significant challenge after fine-tuning, where a promising area of improvement is unseen language pairs where both source and target languages occurred individually during training but not in combination. [Chapter 6](#) is devoted to a more comprehensive study of the different unseen conditions analyzed here.

4.4 Preserving Pre-Trained Knowledge by Discretization

As motivated in §4.1, our approach is to learn a discrete representation for source sentences. We achieve this by augmenting the standard encoder-decoder architecture with a trainable codebook containing k cluster centers, or “codes”. Alongside the primary translation objective, the model is trained to map each continuous hidden state from the encoder to its nearest code in this codebook. For this discretization to be meaningful, the process must overcome two challenges:

1. Avoiding representational collapse: The model must utilize a diverse set of codes, preventing the trivial solution where all encoder outputs are mapped to a single or very few cluster centers.
2. Preserving sufficient information: The resulting sequence of discrete codes, which forms an information bottleneck, must retain enough contextual information for the decoder to reconstruct the meaning of the source sentence and perform high-quality translation.

4.4.1 Discretizing Encoder Latent Space

Our model architecture extends the standard Transformer by incorporating a vector quantization (VQ) module between the encoder and the decoder. This module, denoted by the quantization operation $q(\cdot)$, transforms the continuous hidden states produced by the encoder, $\text{enc}(X) \in \mathbb{R}^{N \times D}$ where $X = (x_1, \dots, x_N)$, into a sequence of discrete representations before they are passed to the decoder.

The core of this process is a learnable codebook, $\mathbf{E} \in \mathbb{R}^{K \times D}$, which contains K unique embedding vectors, $\{\mathbf{e}_j\}_{j=1}^K$. Each vector $\mathbf{e}_j$ has a dimensionality D that matches the model’s hidden dimension. For each encoder hidden state corresponding to an input token, $\text{enc}(x_i)$, the quantization step performs a nearest-neighbor search to find the codebook vector that is closest in Euclidean distance. The index k of this vector is found by:

$$k = \arg \min_{j \in \{1, \dots, K\}} \|\text{enc}(x_i) - \mathbf{e}_j\|_2. \quad (4.1)$$

The original hidden state $\text{enc}(x_i)$ is then replaced by its corresponding quantized vector, $\mathbf{e}_k$. Because this operation is applied to every token in the input sequence X , the final

4.4. Preserving Pre-Trained Knowledge by Discretization

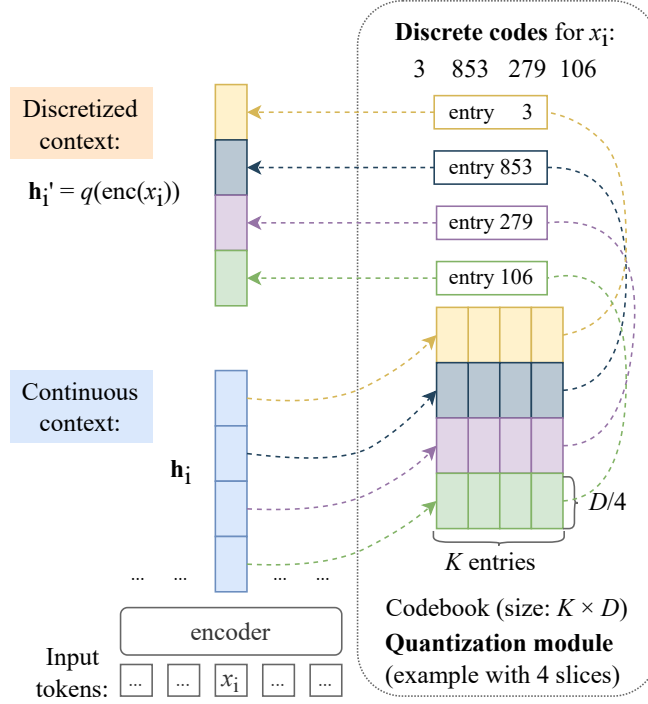


FIGURE 4.2: Illustration of the generation of the discrete codes based on a sliced codebook.

quantized representation, $q(\text{enc}(X))$, preserves the same shape $\mathbb{R}^{N \times D}$ as the original encoder output.

This quantization step is susceptible to index collapse (Kaiser et al., 2018), where only a few entries from the embedding table are actively used, thereby limiting the model’s expressiveness. To mitigate this issue, we adopt the multi-slice quantization approach of Kaiser et al. (2018). Specifically, for an input token x_i , we first partition its corresponding encoder hidden state, $\text{enc}(x_i) \in \mathbb{R}^D$, into S non-overlapping slices:

$$\text{enc}(x_i) = [\mathbf{h}_i^{(1)}; \mathbf{h}_i^{(2)}; \dots; \mathbf{h}_i^{(S)}], \quad (4.2)$$

where each slice $\mathbf{h}_i^{(s)}$ has a dimensionality of D/S , and the notation $[\cdot; \cdot]$ denotes concatenation. A separate nearest-neighbor search is then performed for each slice against a corresponding codebook. The resulting quantized slices are subsequently concatenated to form the final representation for the token:

$$q(\text{enc}(x_i)) = [q(\mathbf{h}_i^{(1)}); q(\mathbf{h}_i^{(2)}); \dots; q(\mathbf{h}_i^{(S)})] \quad (4.3)$$

This composite vector is then passed to the decoder as context. This multi-slice quantization process is illustrated in Figure 4.2.

The slicing mechanism resembles multi-head attention (Vaswani et al., 2017) in that both split the embedding dimension into subspaces for richer representation.

4.4.2 Soft Discrete Codes

Training

The quantization module inherently functions as an information bottleneck, as it maps the continuous space of encoder outputs to a finite set of discrete representations. While this process is beneficial for learning structured and shared representations, it risks a loss of fine-grained information that can degrade model performance. To balance this trade-off between discretization and performance, we introduce a *soft* quantization scheme. In this approach, we stochastically determine whether to pass the quantized or the original continuous representations to the decoder during training. Specifically, the decoder receives the quantized context, $q(\text{enc}(X))$, with a probability of p , and the original unquantized context, $\text{enc}(X)$, with a probability of $1 - p$. This gating mechanism, as illustrated in Figure 4.1 earlier, allows the model to leverage the rich information from the continuous space while still being guided to learn a discrete representational layer.

In Equation 4.1, the lookup of index k is a non-differentiable operation, which blocks the flow of gradients to the encoder during backpropagation. To allow end-to-end training when the quantized context is selected, we employ the straight-through estimator (Bengio et al., 2013b), which copies the gradients from the post-quantization representation, $q(\text{enc}(X_i))$, to the pre-quantization encoder output, $\text{enc}(X_i)$. The effectiveness of this gradient approximation depends on the proximity between the encoder’s continuous output and its corresponding discrete codebook vector. If $\text{enc}(X_i)$ and $q(\text{enc}(X_i))$ are far apart, the copied gradient may be a poor estimate. To ensure these representations remain aligned, we incorporate two auxiliary loss terms from the VQ-VAE framework (van den Oord et al., 2017), namely a codebook loss and a commitment loss as defined by:

$$\mathcal{L}_{\text{codebook}} = \|\text{sg}[\text{enc}(X)] - q(\text{enc}(X))\|_2 \quad (4.4)$$

and

$$\mathcal{L}_{\text{commitment}} = \|\text{enc}(X) - \text{sg}[q(\text{enc}(X))]\|_2, \quad (4.5)$$

where $\text{sg}[\cdot]$ denotes the stop gradient operation. Intuitively, Equation 4.4 pushes the codebook entries closer to the points assigned to them, while Equation 4.5 limits the growth of the encoder hidden states by clipping them to the codebook entries. Each of the terms has weights α_{codebook} and $\alpha_{\text{commitment}}$ to control their importance relative to the main translation objective.

Inference

The above training procedure is intended to ensure that the encoder’s continuous hidden states become well-aligned with the discrete codebook entries, effectively organizing the representation space into distinct clusters around the codebook vectors. During inference, we exclusively use the continuous context, $\text{enc}(X)$. This allows the model to benefit from the more fine-grained information present in the continuous representations. We will provide an empirical validation of this clustering property in §4.7.

4.5. Experimental Setup

	jv	ms	tl	id	en
jv		340K	662K	644K	2,556K
ms	2M		1,174K	4,060K	12,023K
tl	3M	16M		2,356K	12,348K
en	18M	230M	158M		
id	5M	65M	30M		

TABLE 4.5: Number of sentence pairs (above diagonal) and target tokens (below diagonal) from bitext for each language pair after preprocessing. Data marked with light gray are used in the main experiments.

4.5 Experimental Setup

To study model robustness in unseen conditions and the influence of language relatedness, we select parallel data between the two high-resource languages: Indonesian (id) and English (en) and three other languages in the Austronesian family: Javanese (jv), Malay (ms), and Filipino/Tagalog (tl). This leads to two conditions:

- **English-Bridge** (EN-BRIDGE): parallel data for $\text{en} \leftrightarrow \{\text{jv}, \text{ms}, \text{tl}\}$
- **Indonesian-Bridge** (ID-BRIDGE): parallel data for $\text{id} \leftrightarrow \{\text{jv}, \text{ms}, \text{tl}\}$

4.5.1 Datasets

We conduct experiments using parallel data from the Large-Scale Multilingual Machine Translation Shared Task (Wenzek et al., 2021) at the 2021 Conference on Machine Translation (WMT21) (Akhbardeh et al., 2021), specifically the track small-task-2. The training parallel data are compiled from the OPUS platform (Tiedemann, 2012), where datasets are detailed in Appendix B.1. We apply the preprocessing pipeline described by Fan et al. (2021), which includes length filtering, bitext deduplication, and histogram filtering for language identification. An overview of the preprocessed training data is presented in Table 4.5. The validation data is the FLoRes-101 (Goyal et al., 2022) dev set. We include all tested language pairs in the validation data for early stopping. All models are evaluated on the FLoRes-101 devtest set.

4.5.2 Training and Evaluation

Pre-Trained Model We initialize all models with the pre-trained M2M-124 model provided in the WMT shared task (Wenzek et al., 2021). This model has a vocabulary size of 256K, 6 layers in both the encoder and decoder, 16 attention heads, embedding dimension of 512 and inner dimension of 2048.

Training As the training data for different languages are very unbalanced, we use temperature-based sampling (Arivazhagan et al., 2019b) with coefficient 5.0, which heavily upsamples low-resource directions and is recommended for unbalanced data conditions (Arivazhagan et al., 2019b; Tang et al., 2021). For our codebook approach, we use 10K codebook entries. Initial trials with a size of 1K degraded performance, while 40K

heavily reduced training speed. We choose 16 slices² for the codebook, the same value as the number of attention heads. We keep these two values identical as both slicing and multi-head attention break the embedding dimension into multiple subspaces of lower dimensionality. Additional training details are in Appendix B.2.

Comparisons Like in the previous chapter (§3.8), we compare to two alignment losses that have demonstrated effectiveness in handling unseen language pairs.

- Similarity regularization (Arivazhagan et al., 2019a; Pham et al., 2019): enforces the representations for the source and target sentences to be close.
- Adversarial language classification (Arivazhagan et al., 2019a): removes source language signals from the encoder hidden states.

Evaluation We report BLEU (Papineni et al., 2002) and COMET (Rei et al., 2022) scores. We use the language identifier from NLLB Team (2024) for measuring the proportion of the output in the correct languages on a sentence basis.

ID Model	{jv, ms, tl} → X		X → {jv, ms, tl}		Unseen Pair	
	BLEU	COMET	BLEU	COMET	BLEU	COMET
ENGLISH-BRIDGE (X=en)						
(1) Direct fine-tuning (row (2) of Table 4.3, 4.4)	27.8	72.7	27.1	78.4	4.7	53.0
(2) + Similarity regularization	28.0	73.1	27.4	78.5	17.0	70.9
(3) + Adversarial language classification	27.5	72.5	26.6	78.2	17.1	70.9
(4) + Codebook (ours)	26.8	71.9	26.2	77.8	15.5	66.8
INDONESIAN-BRIDGE (X=id)						
(5) Direct fine-tuning (row (5) of Table 4.3, 4.4)	27.7	80.2	24.3	84.5	17.9	74.6
(6) + Similarity regularization	27.7	80.1	24.3	84.4	18.9	74.7
(7) + Adversarial language classification	26.9	79.3	23.4	84.1	18.3	73.6
(8) + Codebook (ours)	27.1	79.2	23.6	83.8	18.3	73.7

TABLE 4.6: Average translation performance in BLEU(↑) and COMET(↑) on language directions involved (first two columns) and unseen (last column) in fine-tuning. Additional regularization approaches are beneficial on the English-bridge setup, but has minimal impact on the Indonesian-bridge setup.

4.6 Results

We first discuss the translation performance of our multilingual systems (§4.6.1, §4.6.2), and then use the learned discrete codes to investigate cross-lingual knowledge-sharing of the trained models (§4.6.3).

4.6.1 Impact of Bridge Languages

In Table 4.6, the most noteworthy difference is the impact of additional regularization approaches on unseen language pair performance. For the English-bridge configuration, all three approaches in rows (2) – (4) demonstrate substantial improvements on unseen pairs relative to the baseline in row 1. In contrast, for the Indonesian-bridge configuration,

²Initial experiments on smaller datasets showed weaker translation performance with 2 and 4 slices.

4.6. Results

the additional approaches in rows (6) – (8) show minimal improvements compared to standard fine-tuning in row (5).

These results indicate that incorporating a linguistically similar bridge language may facilitate the reuse of pre-trained knowledge. Moreover, the English-bridge condition utilizes approximately four times more parallel data than the Indonesian-bridge condition, suggesting that employing a similar bridge language may offer greater data efficiency. This efficiency could result from reduced linguistic diversity across translation directions, potentially simplifying the multilingual learning task.

A confounding factor in these findings is that the Indonesian fine-tuning data may contain samples from the unseen language pairs due to difficulties in distinguishing closely related languages. Such contamination would render the “unseen” pairs no longer truly unseen, potentially inflating the scores. Admittedly, web-crawled data as used in our experiments does carry inherent risks of incorrect language labeling (Kreutzer et al., 2022). While our preprocessing pipeline incorporates language identification (§4.5.1), mislabeled samples may remain in the dataset due to errors classifying linguistically similar languages.

To mitigate these concerns, we conduct a controlled experiment using the Wikimedia dataset derived from Wikipedia texts, where language identities are known. Results in Table 4.7 provide additional evidence that employing a similar bridge language facilitates translation performance on unseen language pairs.

ID	Training Pairs	Bridge	→ Bridge		Bridge →		Unseen	
			BLEU	COMET	BLEU	COMET	BLEU	COMET
(1)	{gl, ca}↔	en	35.7	81.8	36.2	79.3	6.3	59.9
(2)		es	27.5	84.9	27.9	85.5	19.0	81.0
(3)	{jv, ms, tl}↔	en	32.7	78.0	34.0	81.3	9.1	60.2
(4)		id	23.1	77.4	21.9	81.3	15.9	70.5

TABLE 4.7: Translation quality (BLEU(↑), COMET(↑)) on the Wikimedia dataset when using similar versus dissimilar bridging languages. The minimal risk of incorrect language data guarantees that unseen language pairs are excluded from the training data. Results confirm that using similar bridging languages consistently leads to stronger performance on language pairs not encountered during fine-tuning.

4.6.2 Impact of Using Codebooks

In Table 4.6, the proposed codebook approach improves performance on unseen pairs relative to direct fine-tuning (4.7→15.5 BLEU, 53.0→66.8 COMET). However, this improvement comes at the expense of supervised directions and underperforms the two continuous representation approaches by about 1.5 BLEU. The performance degradation on supervised directions may result from the information bottleneck inherent in the discretization process. The underperformance relative to continuous approaches stems from persistent off-target translation errors, where test sentences are translated into incorrect target languages. While the codebook approach reduces off-target translations from 87.4%

to 13.1% compared to the fine-tuning baseline in row (1), this remains substantially higher than the 4.7% achieved by the alternative models in rows (2) and (3). Despite this limitation, the codebook approach offers advantages for analyzing learned representations, which we leverage to investigate the distinct behaviors observed between the two data conditions when processing unseen language pairs.

4.6.3 Using Discrete Codes to Interpret Learned Representations

An advantage of our codebook approach is that it allows easier interpretation of the model’s internal representations. We leverage this property to investigate the following question: why is the ID-BRIDGE data condition more performant despite using less data?

Formalization

To address the question above, we analyze the discrete codes for semantically equivalent sentences across multiple languages. We encode sentences from the test sets of all languages used in our main experiments, English (en), Indonesian (id), Javanese (jv), Malay (ms), and Tagalog (tl), again using the FLoRes-101 devtest set.³ For an input sentence with N tokens, we encode them with the trained model’s encoder and then extract a sequence of discrete codes $T_{1,\dots,N}^{(s)}$ for each of the model’s S slices. Each $T_i^{(s)}$ represents a discrete code from the codebook that corresponds to a token’s representation within slice s .

We then compare the codes of semantically equivalent sentences from two different languages by evaluating two aspects: 1) the degree of overlap in the codes used, and 2) the transformation required to map one sequence of codes to the other. These metrics provide a quantitative measure of similarity between the code sets, which in turn serves as a proxy for the alignment of the model’s internal representations across these languages.

Discrete Code Distribution

The first analysis is based on the frequency of individual discrete codes without considering their orders or co-occurrence statistics. For each language and each model slice, we construct a probability distribution over the entire codebook vocabulary. This distribution P is defined by the probability of observing each discrete code c_i :

$$p(c_i) = \frac{\text{frequency}(c_i)}{\sum_{c_j \in \mathcal{C}} \text{frequency}(c_j)}, \quad (4.6)$$

where c_i is a discrete code from the codebook $\mathcal{C}$.

To compare a pair of languages i and j , we compute the Kullback-Leibler (KL) divergence between their respective code distributions, P_i and P_j :

$$D_{\text{KL}}(P_i || P_j), \quad (4.7)$$

³The FLoRes-101 sets are multiway, meaning the sentences across languages are direct translations of each other.

4.6. Results

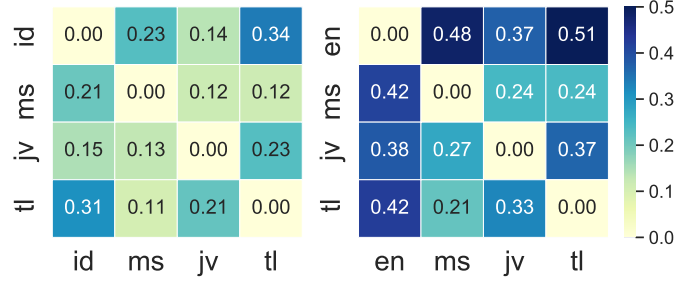


FIGURE 4.3: KL divergence($\downarrow$) of code distribution for the ID-BRIDGE (left) and EN-BRIDGE (right) setup. *Lower* scores indicate greater representational sharing. ID-BRIDGE results in more sharing not only between itself and {ms, jv, tl} but also among {ms, jv, tl}.

where a lower value indicates that the two languages exhibit more similar patterns of code usage, which in turn implies a higher degree of representational alignment within that specific model slice.

Figure 4.3 depicts the KL divergence of code distribution averaged over all slices. A comparison between the EN-BRIDGE and ID-BRIDGE setups reveals several differences in how the model aligns language representations. First, the EN-BRIDGE model exhibits a strong divergence between the representations for the bridge language (English) and those for the other languages ({ms, jv, tl}). This indicates a large representational gap. In the ID-BRIDGE counterpart, however, the distinction between the bridge language and the other languages is far less. Second, and more importantly, the degree of representational overlap *among* the non-bridge languages ({ms, jv, tl}) is substantially higher in the ID-BRIDGE setup. This is crucial because the input tokens for these languages were identical in both configurations. The difference in alignment can therefore be attributed directly to the model’s learned representational strategy rather than to variations in the training data. Overall, these results demonstrate that the choice of a bridge language not only determines the representational relationship between the bridge language and the target languages but also fundamentally influences the degree of alignment *among* the target languages themselves.

Discrete Code Translation

The code distribution analysis above makes a simplified assumption by considering the discrete codes as a *bag of words*. To assess the structural (dis)similarity between the code representations for different languages, we consider the task of translating the discrete codes of a language to another.

While a constructed language like Interlingua would create the same representations for the source sentences with identical meanings, our discrete code representation is not yet invariant to the source language. Nevertheless, we do expect them to be more abstracted from the source sentences, making the translation task easier than directly between the raw tokens. Here we train a translation model on the discrete codes and use the test performance to quantify how similarly the source languages are represented.

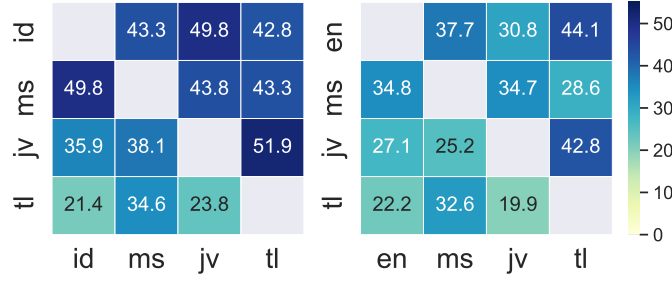


FIGURE 4.4: BLEU(↑) scores of translating between discrete codes for the ID-BRIDGE (left) and EN-BRIDGE (right) setup. *Higher* scores indicate greater representational sharing. The superior performance in the ID-BRIDGE setup suggests its codes have achieved a more unified structure across languages.

When the representations are more different from each other, i.e. language-specific, the translation quality on the discrete codes is expected to be lower.

Specifically, we randomly sample 100K sentence pairs⁴ for each translation direction in the experiments of Table 4.6 extract their discrete codes assigned by the trained models (rows (4) and (8) of Table 4.6), and train a new Transformer-base (Vaswani et al., 2017) to translate between the extracted codes of different languages. We flatten the slices, therefore making each source token represented by 16 discrete codes. After training for 200K steps, we report BLEU scores on the test set, which is also converted to discrete codes. The results are shown in Figure 4.4. First, the translation task is clearly easier on the discrete codes derived from the ID-BRIDGE system. Second, the score differences are especially prominent when translating out of Malay (ms) and Javanese (jv), which are more related to Indonesian than Filipino/Tagalog (tl). Along with the results from the code overlap, our results show that using a similar bridge language results in higher knowledge-sharing not only syntactically but also structurally, especially between related languages.

4.7 Analyses on Learned Discrete Codes

We further investigate the discrete codes regarding their usefulness for the learned representations (§4.7.1) as well as the translation task (§4.7.2).

4.7.1 How Well Clustered are the Hidden States?

As motivated in §4.4, although at inference time we use the continuous encoder hidden states instead of the cluster centers, the soft discrete codes will still enforce encoder hidden states into clusters, thereby resembling a discrete structure. To verify whether the encoder latent space indeed becomes more discretized with our approach, we analyze the encoder hidden states on the test set using Principal Component Analysis (PCA). If the data points representing the encoder outputs are well-clustered, a larger percentage of their variance should be explained by the learned principal components. As shown in Figure 4.5, our

⁴The training data (Table 4.5) allow us to use 340K sentences. We sampled 100K for faster experiment iteration.

4.8. Upper Bounds by Additional Training Data

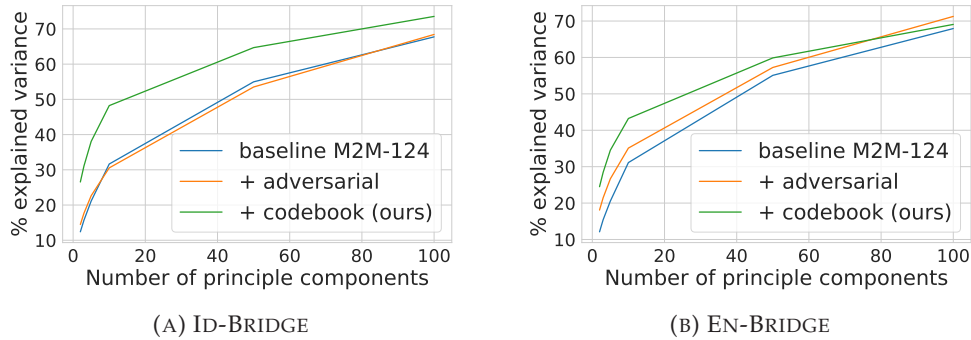


FIGURE 4.5: Our codebook approach creates better-clustered encoder hidden states, as shown by a much higher percentage of variance explained by PCA compared to both the baseline and a strong alternative approach, the adversarial language classifier.

approach consistently leads to higher proportion of explained variances compared to the baseline M2M-124, as well as the strong alternative approach with the adversarial language classifier. These results therefore confirm the effectiveness of our soft discrete code approach in enforcing discrete structures in the encoder latent space.

4.7.2 Meaningfulness of Cluster Centers

Recall that at inference time our soft discrete code model uses the encoder hidden states prior to discretization, although it does use both pre- and post-discretization encoder context in training. The main reason behind this is that discretizing the encoder hidden states to cluster centers creates an information bottleneck that limits model expressiveness. Despite the expected performance degradation, we are nonetheless interested in quantifying how much information is lost by using the cluster centers as context instead. In other words, the question is *how meaningful are the cluster centers for the translation task?* In Table 4.8, we report the results of using the cluster centers as context for the decoder at inference time. Compared to using the encoder hidden states, we see a degradation of 4.1 and 1.7 BLEU for ID-BRIDGE and EN-BRIDGE respectively. This indicates that the cluster centers are still relevant for the translation task, although much less powerful than the encoder hidden states prior to discretization. It also rules out the possibility of the learned codes being trivial repetitions, which would otherwise have been detrimental to the translation performance.

4.8 Upper Bounds by Additional Training Data

Our experiments so far use single-bridge languages and consider directions fully unseen during fine-tuning. We now study the impact when either of the two conditions changes: 1) when parallel data is available for directions unseen in fine-tuning (§4.8.1) and 2) when using multiple bridge languages (§4.8.2). In these experiments, scores for systems shown in Table 4.6 show minor deviations from their previous values due to experiments run on different hardware.

Encoder States at Inference	Avg. BLEU($\uparrow$)			
	$\rightarrow X$	$X \rightarrow$	$Y \leftrightarrow Z$	Avg.
EN-BRIDGE ($X=en$)				
Encoder states (Table 4.6 row (4))	26.8	26.3	15.2	20.9
Cluster centers	24.3	24.6	13.9	19.2
ID-BRIDGE ($X=id$)				
Encoder states (Table 4.6 row (8))	27.2	23.6	18.3	21.9
Cluster centers	22.8	20.0	14.3	17.8

TABLE 4.8: At inference time, using cluster centers instead of the clustered encoder states degrades performance by 1.7 to 4.1 BLEU. Despite the degradation, the scores show that translation from the cluster centers is still meaningful. This also rules out the possibility of the learned codes collapsing to trivial repetitions.

4.8.1 Adding Parallel Data for Directions Unseen in Fine-Tuning

To determine how much parallel data is needed to surpass direct translation, we create an oracle condition by supplementing the fine-tuning data with parallel data for language pairs previously unseen during training (§4.5.1). As detailed in Table 4.5, the oracle parallel data amounts to 2.2M sentences in total (340K for *jp-ms*, 662K for *jp-tl*, and 1.2M sentences for *ms-tl*). We then sample 100%, 10%, and 1% of this oracle parallel data, combining each subset with the original training data. For consistency, we use the same multilingual system configuration as employed in rows (3) and (6) of Table 4.6. The results are presented in Table 4.9.

Surprisingly, adding 1% oracle bitext (22K sentence pairs in total) for the previously unseen directions already yields performance comparable to the best direct systems, achieving 18.4 and 17.3 BLEU for ID-BRIDGE and EN-BRIDGE, respectively. However, this improvement comes at the cost of a slight decrease in supervised direction performance, with drops of 0.4 BLEU for ID-BRIDGE and 0.7 BLEU for EN-BRIDGE. This trade-off is likely related to temperature-based sampling, which aggressively up-samples the extremely low-resource directions and causes the model to de-prioritize higher-resource directions.

When the amount of oracle bitext is increased to 10% (220K sentence pairs), the system surpasses direct performance on the unseen directions. Further increasing the amount from 10% to the full oracle dataset leads to only marginal additional improvements. For ID-BRIDGE, performance appears to saturate at 10%, with no further gain from the remaining 90% of data. In contrast, the EN-BRIDGE system continues to benefit, particularly on $Y \leftrightarrow Z$ directions, gaining an additional 0.5 BLEU.

Despite these improvements, EN-BRIDGE continues to lag behind ID-BRIDGE on these directions by 0.8 BLEU (18.9 vs. 19.7 BLEU). This may be explained by the greater amount of data required for effective training when the bridge language is distantly related to the other languages, thereby increasing the overall translation challenge. This finding aligns with the earlier observation that using a related bridge language eases the multilingual translation task and enhances knowledge sharing (§4.6.1).

4.8. Upper Bounds by Additional Training Data

Oracle Bibtex Amount	Avg. BLEU($\uparrow$)			
	$\rightarrow X$	$X \rightarrow$	$Y \leftrightarrow Z$	Avg.
EN-BRIDGE ($X=en$)				
Best system without $Y \leftrightarrow Z$ fine-tuning data	27.9	27.2	17.3	22.4
1%	27.0	26.8	17.1	22.0
10%	27.5	27.4	18.4	22.9
100% (2.2M bibtex)	27.7	27.5	18.9	23.2
ID-BRIDGE ($X=id$)				
Best system without $Y \leftrightarrow Z$ fine-tuning data	27.5	24.1	18.4	22.1
1%	26.9	23.9	18.4	21.9
10%	27.3	24.5	19.7	22.8
100% (2.2M bibtex)	26.6	24.8	19.7	22.7

TABLE 4.9: Impact of adding oracle parallel data for the previously unseen directions for training the baseline models. Adding 10% parallel data (roughly 220K sentence pairs in this case) surpasses the best system trained without $Y \leftrightarrow Z$ fine-tuning data.

Data Condition			Avg. BLEU($\uparrow$)			
			$\rightarrow X$	$X \rightarrow$	$Y \leftrightarrow Z$	Avg.
MULTI-BRIDGE	$X=id$		27.0	24.3	18.3	21.9
	$X=en$		27.8	27.7		23.0
Only ID-BRIDGE			27.1	24.2	17.7	21.7
Only EN-BRIDGE			28.1	27.6	5.1	16.5

TABLE 4.10: Results of using multiple bridges (combining ID-BRIDGE and EN-BRIDGE). Despite substantial gains over EN-BRIDGE, the multi-bridge system only gives a mild improvement in zero-shot performance ($Y \leftrightarrow Z$) over the ID-BRIDGE system.

4.8.2 Multiple Bridge Languages

While our experiments so far have focused on single bridge languages, in practice multi-bridge parallel data is often available. While multi-bridge setups have been shown to outperform English-only bridging (Rios et al., 2020), what remains unclear, however, is whether combining data from multiple single-bridge conditions leads to additional synergy.

To investigate this, we train a multi-bridge system by combining data from the ID-BRIDGE and EN-BRIDGE setups. As shown in Table 4.10, the multi-bridge system performs on par with the best single-bridge models for supervised directions ($\rightarrow X$ and $X \rightarrow$). For unseen directions ($Y \leftrightarrow Z$), while the multi-bridge setup yields substantial gains over EN-BRIDGE (rising from 5.1 to 18.3 BLEU), it provides only a marginal improvement over ID-BRIDGE. Considering that training the multi-bridge system more than doubles the cost compared to ID-BRIDGE, these results suggest that choosing a bridge language closely related to the remaining languages is a more data-efficient strategy for achieving strong direct translation performance.

4.9 Summary

This chapter investigates an important challenge in the adaptation of pre-trained multilingual translation models, namely their susceptibility to catastrophic forgetting. When pre-trained models are fine-tuned on a limited set of language pairs, they often develop language-specific biases that degrade or eliminate their pre-trained capabilities on languages unseen in fine-tuning. To counteract this, we propose a method based on discrete intermediate representations. The core hypothesis is that by forcing the model to channel information through a discrete bottleneck, we can preserve pre-trained knowledge more effectively while still allowing for specialization.

Our investigation leads to several key findings: **1)** We first establish that standard fine-tuning on a narrow, English-centric dataset leads to significant catastrophic forgetting of language pairs that were present during pre-training. **2)** Our proposed soft discretization approach mitigates this forgetting, although it underperforms continuous regularization approaches due to information loss from discretization. **3)** Using a linguistically similar bridge language during fine-tuning is a highly effective strategy. This approach not only combats forgetting and improves data efficiency, but also facilitates the learning of more generalizable representations.

For scenarios where computational resources are less constrained, we show that multi-bridge systems can further enhance performance. We also conduct oracle experiments to quantify the data requirements for recovering performance on forgotten language pairs, finding that adding back as little as 10% of the original parallel data can surpass the performance of strong direct translation baselines.

Finally, this chapter is situated within our broader exploration of generalizable representations. We focus on task-specific pre-trained models here, deferring the analysis of generic models to the next chapter. Unlike training from scratch, adapting pre-trained models constrains us to their fixed architectures, rendering the structural modifications from the previous chapter unsuitable. However, we confirm that the underlying principle of regularization remains highly beneficial in this new context. These findings pave the way for our subsequent investigation into whether similar principles apply to large-scale generalist language models.

Chapter 5

Transfer with Generalist Pre-Trained Models

The field of natural language processing has seen a paradigm shift since late 2022, moving from task-specific models towards generalist models that handle diverse tasks through *instruction following*. This shift is largely driven by scaling, where models with billions of parameters, commonly referred to as large language models (LLMs), demonstrate capabilities that were not explicitly covered in training. Building on the findings from the previous chapter on task-specific models, a natural and crucial next direction is to investigate whether similar cross-lingual transfer mechanisms apply to the new generation of generalist models.

The investigation in this chapter differs from the previous one in two aspects. First, we move from task-specific models to generalist ones, allowing our experiments to extend beyond machine translation to a broader range of tasks. Second, the architecture changes from encoder-decoder models, which offer a natural alignment target in the encoder, to decoder-only LLMs. This shift brings a higher degree of freedom in selecting where to enforce language invariance within the monolithic model.

Given the computational costs of training these large models, this chapter introduces a parameter-efficient, modular approach to cross-lingual transfer. The alignment modules integrate with pre-existing task-specific components to enable an efficient and performant cross-lingual transfer. This chapter is partly based on [Liu and Niehues \(2025b\)](#).

5.1 Introduction

While LLMs have demonstrated remarkable abilities to perform tasks based on instructions alone or only a few examples ([Wei et al., 2022a](#); [Chowdhery et al., 2023](#)), achieving state-of-the-art performance often requires additional supervised fine-tuning on specific tasks ([Shen et al., 2024](#); [Xu et al., 2024](#); [Alves et al., 2024](#)). However, the scarcity of task-specific training data across languages and the inherent LLM performance gaps across languages present critical challenges. This motivates our investigation of extending the benefits of task-specific training beyond languages with supervised data.

As motivated in earlier chapters, although cross-lingual transfer is a well-established field ([Wang and Zheng, 2015](#); [Ruder et al., 2019a](#); [Artetxe and Schwenk, 2019b](#); [Pfeiffer et al., 2020](#)), its application to generation tasks with variable-length outputs is considerably more

difficult than classification tasks (Vu et al., 2022b; Li and Murray, 2023). This challenge is especially relevant for LLMs, which formulate all tasks as next-token generation problems.

A primary strategy for enabling transfer is cross-lingual alignment, which includes methods like the adversarial language classification and similarity regularization techniques discussed in previous chapters. However, applying these methods to modern LLMs is not straightforward due to architectural shifts. In previous architectures, the location for applying alignment was more clearly defined. For *encoder-only* models, representations were aligned using objectives like cross-lingual masked language modeling (Conneau and Lample, 2019) or by minimizing the distance between parallel sentence representations (Feng et al., 2022). These techniques also extended to *encoder-decoder* models, where alignment was typically enforced on the encoder’s outputs (Arivazhagan et al., 2019a; Pham et al., 2019; Duquenne et al., 2023b). In contrast, the integrated architecture of *decoder-only* models lacks such clear separation between input processing and output generation. This ambiguity makes it difficult to determine where and how to apply alignment objectives, as also highlighted in the survey by Hämmerl et al. (2024).

Beyond the architectural challenge, making the alignment step modular and post-hoc is another challenge that motivates our work. Given the resource-intensive nature of fine-tuning LLMs, an ideal cross-lingual alignment method should be disentangled from task-specific fine-tuning. This modularity would allow cross-lingual capabilities to be instilled into an already fine-tuned model without repeating the original computationally expensive training process.

The previous chapters explored two strategies, one driven by explicit similarity optimization and another based on an adversarial language classifier. Despite the demonstrated performance gains, these methods are less suitable for the modular, post-hoc properties required here. When decoupled from a primary task objective, these methods are prone to representation collapse, a failure mode where the model learns a trivial solution by mapping all inputs to a single point. To counteract this, in this chapter, we adopt a contrastive learning approach, an established technique for preventing representation collapse. This method pulls similar representations together while pushing dissimilar ones apart. By enforcing this relational structure on the representation space, the model can learn meaningful alignments instead of collapsing into trivial solutions.

Besides the methodological shift to contrastive learning, this chapter also involves an extended evaluation setup. As we move beyond translation tasks, a practical challenge emerges regarding the alignment data. Non-translation tasks no longer provide the parallel data required for cross-lingual alignment. Therefore, we must source parallel data from external datasets, which may differ in domain and language coverage from the fine-tuning data of the task at hand.

This experimental setup creates two dimensions of generalization. The first is task generalization, which, consistent with the previous chapter, measures the model’s task performance on conditions unseen during fine-tuning. The second is language generalization, which assesses whether the learned alignment itself extends to languages that were absent from the alignment training phase. We examine both dimensions in this chapter.

5.1. Introduction

5.1.1 Task Definition

In this chapter, we study the following task:

Given a large language model that supports N languages and task-specific data on k ($k \ll N$) languages, after task-specific fine-tuning on these k languages, how to improve task performance on remaining $N - k$ languages that are absent during fine-tuning?

This problem formulation encompasses two challenges. The first is the classical form of cross-lingual generalization, which aims to leverage task-specific fine-tuning on k languages to improve performance on the remaining $N - k$ languages. The second, similar to the previous chapter, is the temporal challenge involving knowledge retention, as the fine-tuning process must avoid catastrophic forgetting and preserve the model’s general-purpose abilities from pre-training. For specific applications on machine translation, an additional aspect of compositional generalization also emerges. To maintain the generality of this chapter, this specific aspect is reserved for [Chapter 6](#).

5.1.2 Contributions

In this chapter, we start by quantifying the degree of cross-lingual alignment present in two prominent LLMs, Llama 3 ([AI @ Meta et al., 2024](#)) and Qwen 2.5 ([Qwen Team et al., 2025](#)). The analyses suggest that the middle layer provides the strongest potential for cross-lingual transfer. We then apply these insights to improve cross-lingual transfer in task-specific fine-tuning. By alternately training on alignment and task-specific data, we aim to improve the task performance on languages without fine-tuning data.

Specifically, we make the following contributions:

- Our analysis across a broad range of languages reveals that, in pre-trained decoder-only large language models, the representations from the middle layers exhibit the highest degree of cross-lingual alignment compared to other layers in the network (§5.3).
- We propose an alignment procedure focused on LLM middle layers during task-specific fine-tuning. This approach improves cross-lingual transfer (§5.6.2) and enhances alignment across all network depths (§5.6.3) compared to other placement configurations.
- We show that the cross-lingual transfer improvements extend beyond the specific languages involved in alignment (§5.6.2) to a wider range of languages supported by the underlying LLMs.
- We show that task-specific and alignment modules trained separately can be combined post-hoc to improve transfer performance (§5.7.4). This is particularly suitable to practical scenarios with existing task-specific modules that are later instilled with additional cross-lingual capabilities.

5.2 Related Work

We first review related work on multilingual LLMs (§5.2.1), alignment techniques that encourage similar representations across languages (§5.2.2), and analyses of LLM representations (§5.2.3).

5.2.1 Multilingual Capabilities of Large Language Models

The performance of LLMs often varies across languages, a disparity largely attributed to imbalances in their pre-training data. However, even models trained predominantly on English exhibit some degree of multilingual capability (Aycok and Bawden, 2024; Yuan et al., 2024), potentially due to the unintentional ingestion of multilingual content during pre-training (Briakou et al., 2023). Meanwhile, there are significant efforts to expand the language coverage, either incorporating more languages during the initial pre-training phase (AI @ Meta et al., 2024; Qwen Team et al., 2025), or adapting existing models post-hoc by adding new languages through continued training or other extension methods (Yong et al., 2023; Lai et al., 2024b; Mahdizadeh Sani et al., 2025). Despite these inherent or intentionally added multilingual capabilities, effectively applying them to downstream tasks in low-resource settings remains challenging (Adelani et al., 2024; Iyer et al., 2024).

5.2.2 Multilingual Representation Alignment

Enhancing meaningful cross-lingual relationships between model representations has been a well-studied area. The goal is to create a language-agnostic representation space that enables knowledge transfer across languages. This is particularly relevant when bridging dissimilar languages, as successful alignment in these cases would demonstrate that the model has learned abstract semantic concepts independent of linguistic surface forms. This capability is essential for many tasks, including intermediate tasks such as bilingual lexicon induction (Zhang et al., 2017b) and sentence embeddings (Feng et al., 2022; Li et al., 2023b), as well as more direct applications like information retrieval (Izacard et al., 2022) and translation (Pham et al., 2019).

Multilingual representation alignment can be achieved by various mechanisms, such as similarity losses that push translations toward each other (Pham et al., 2019), contrastive losses (Hadsell et al., 2006) that additionally incorporate non-translation pairs, and adversarial losses (Ganin and Lempitsky, 2015) that remove language-specific signals. The cross-lingual transfer capabilities of these approaches are extensively documented in the literature. In particular, contrastive learning methods have shown promising results (Pan et al., 2021; Chi et al., 2021; Qi et al., 2022). Our contribution is not applying contrastive learning itself, but rather investigating how to effectively align multilingual spaces specifically in decoder-only models.

In the context of LLMs, Wang et al. (2025) use linear projections learned offline to align non-English representations with English ones during decoding. Our work differs in that our alignment objective is parameterized by the same weights as task-specific fine-tuning,

5.3. Analyzing Cross-Lingual Alignment in LLMs

and is directly applicable to multilingual fine-tuning. [Wu et al. \(2024a\)](#) align LLM top-layer representations specifically for the task of semantic textual similarity. Different from this work, they do not consider cross-lingual transfer in downstream tasks or explore intermediate LLM layers for alignment.

5.2.3 Representation Analysis in Large Language Models

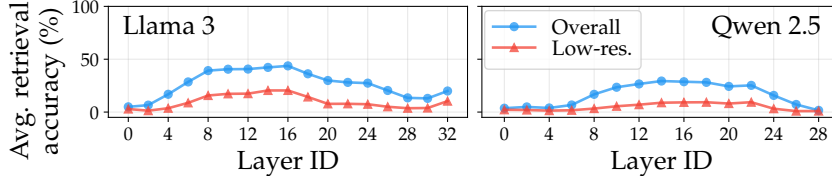
Several recent works have analyzed LLM internal representations with geometric analysis of representation spaces ([Razzhigaev et al., 2024](#); [Lee et al., 2025](#)), probing classifiers ([Wang et al., 2024](#); [Li et al., 2025a](#)), or logit lens analysis ([Wu et al., 2025](#)). Multiple studies ([Wu et al., 2025](#); [Mao and Yu, 2024](#); [Zhong et al., 2024](#)) reported higher representational similarity in middle layers in various evaluation settings, complementing our findings. [Wu et al. \(2025\)](#) identify “semantic hubs” in LLM middle layers that integrate information from various data types, while we focus specifically on cross-lingual representations rather than multi-modality. [Mao and Yu \(2024\)](#) show that SFT on machine translation increases similarity between parallel sentences from the same MT corpus, while we show that SFT on a non-translation task does *not* increase representation similarity, thereby motivating explicit alignment during SFT. [Zhong et al. \(2024\)](#) measure pairwise similarity to English representations (“latent language”) on high resource languages, while we focus on pairwise similarity across a more diverse set of language pairs. [Kargaran et al. \(2025\)](#) use similarity between parallel sentences to estimate cross-lingual transfer capabilities. Our analysis in §5.3 shares the same motivation, and we additionally show that actively enforcing alignment can improve transfer performance.

5.3 Analyzing Cross-Lingual Alignment in LLMs

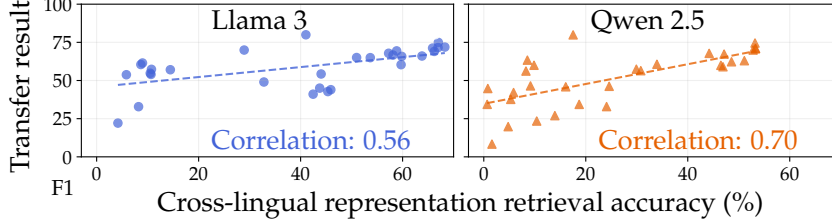
To understand how well LLM representations capture semantic equivalence across languages, we use translation retrieval as a diagnostic task. We choose this retrieval task over other metrics like cosine similarity or SVCCA score ([Raghu et al., 2017](#)) because it better captures relative relationships: if a model’s representations enable us to identify a sentence’s translation from a set of candidates, the exact numerical distance between the query and the retrieved translation is less important than the ability to rank translations as the most semantically similar.

Specifically, we first extract model activations at each network layer for all language variants of the input text. To handle variable-length sequences, we create fixed-size sentence embeddings by mean-pooling the activations over the sequence length dimension. For translation retrieval, given a query sentence in one language, we compare its embedding to the embeddings of candidate sentences in the target language using ratio-based margin similarity ([Artetxe and Schwenk, 2019a](#))¹. For N languages, we evaluate retrieval accuracy across all $N(N - 1)$ possible language pairs. We use the FLORES-200 dataset ([NLLB Team, 2024](#)), which provides high-quality multiway parallel texts across diverse languages (detailed setup in §5.5.2).

¹shown to outperform cosine similarity for cross-lingual retrieval tasks ([Artetxe and Schwenk, 2019a](#))



(A) Translation retrieval accuracy (averaged over 35 languages and 1190 language directions) varies by layer, with the middle layer showing the highest score. Lower-resource languages are poorly aligned.



(B) Positive correlation between base model cross-lingual semantic alignment measured by translation retrieval accuracy and downstream transfer performance.

FIGURE 5.1: Two observations (§5.3) motivating our approach of aligning multilingual representations (§5.4).

Our investigation of Llama 3 and Qwen 2.5 models² reveals three key findings:

Overall weak semantic alignment, with peak in middle layers: As shown in Figure 5.1a, the average translation retrieval accuracy across 1,190 language pairs remains below 50%, with Llama 3 outperforming Qwen 2.5. Low-resource languages³ show especially weak alignment, achieving less than half of the overall average accuracy. In particular, the *middle* layers of both models demonstrate the strongest retrieval performance. This suggests stronger potential for cross-lingual transfer at these intermediate representations.

Strong correlation between base LLM semantic alignment and downstream task transfer: To what extent can the semantic alignment present in the base LLM predict cross-lingual transfer performance after supervised fine-tuning? Using multilingual slot filling as a case study, we train models on 5 high-resource languages jointly and evaluate transfer performance on 25 additional languages (detailed setup in §5.5.1). As shown in Figure 5.1b, for both Llama 3 and Qwen 2.5, we observe statistically significant positive correlations ($p < 0.01$) between middle-layer retrieval accuracy and downstream task performance. This correlation suggests that increasing cross-lingual alignment in LLM intermediate representations may improve cross-lingual transfer.

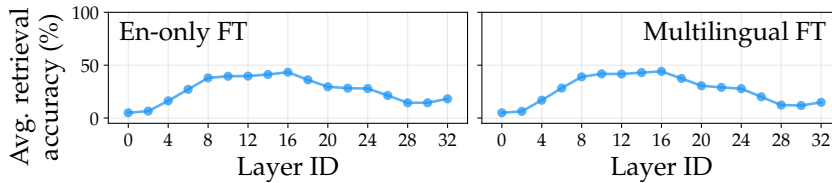


FIGURE 5.2: Task-specific fine-tuning shows minimal impact on translation retrieval accuracy.

²specifically the 8B-Instruct and 7B-Instruct variants

³resource levels as defined by [NLLB Team \(2024\)](#)

5.4. Explicit Alignment in Fine-Tuning

Task-specific fine-tuning preserves but does not enhance semantic alignment: After analyzing the base LLMs, we examine how supervised fine-tuning affects the models’ internal semantic alignment. Using the same multilingual slot filling task as before, we study both English-only and multilingual fine-tuning. Despite multilingual fine-tuning being an established method for improving cross-lingual transfer (Li and Murray, 2023; Chirkova and Nikoulina, 2024), we observe that neither training configuration alters the models’ cross-lingual semantic alignment (Figure 5.2). This preservation of baseline alignment patterns, even under multilingual training, indicates that pure fine-tuning does not sufficiently strengthen cross-lingual alignment. This further motivates us towards explicit cross-lingual alignment during fine-tuning.

5.4 Explicit Alignment in Fine-Tuning

As illustrated in Figure 5.3, we propose alternately training with the original task objective and an alignment objective that explicitly encourages similar representations for semantically related contents.

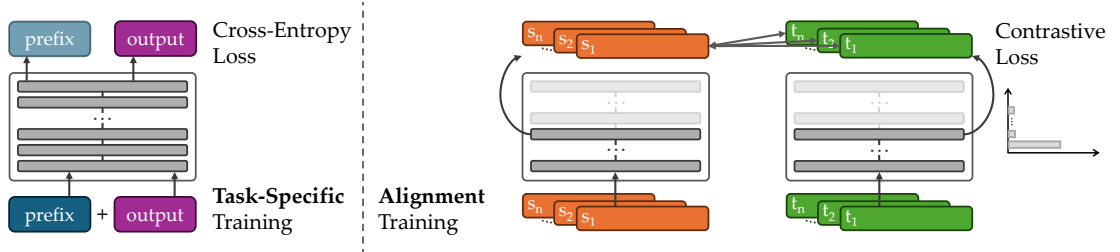


FIGURE 5.3: Illustration of our approach which alternates training between task-specific and alignment objectives. The alignment objective operates on middle-layer representations.

5.4.1 Task Objective

For task-specific training, we follow the standard conditional language modeling setup, using a cross-entropy loss over the predicted text $Y = (y_1, \dots, y_T)$ conditioned on the input prefix $X = (x_1, \dots, x_N)$:

$$\mathcal{L}_{\text{task}} = -\frac{1}{T} \sum_{t=1}^T \log P(y_t | y_{<t}, X). \quad (5.1)$$

The input prefix X combines the task instruction and input text, such as “Translate the sentence from English to German: Hello world.”.

5.4.2 Alignment Objective

Representation alignment can be achieved by various methods, including similarity regularizers and adversarial losses recurring in the previous chapters (§3.8, §4.5). We adopt contrastive loss for modularized alignment training, where only the alignment objective is active during some training steps. This precludes similarity-only approaches, which

lack mechanisms to preserve dissimilarity and consequently suffer from representation collapse, a phenomenon where all representations converge to identical points. Contrastive loss mitigates this collapse by incorporating both positive and negative pairs, in this case maximizing similarity between translation pairs and minimizing similarity between non-translation pairs. Specifically, given a batch $\mathcal{B}$ of n pairs of parallel sentences, the alignment loss for a sentence pair (s, t) is:

$$\mathcal{L}_{\text{align}} = -\log \frac{\exp(\text{sim}(\mathbf{h}_s^i, \mathbf{h}_t^i))}{\sum_{v \in \mathcal{B}} \exp(\text{sim}(\mathbf{h}_s^i, \mathbf{h}_v^i))} \quad (5.2)$$

where $\mathbf{h}_s^i$ is the mean-pooled⁴ hidden states at the i^{th} LLM layer for input s and $\text{sim}(\cdot, \cdot)$ is a similarity function. Motivated by our finding that middle layers have the strongest cross-lingual alignment potential, we select i as the middle layer and compare its performance to other layer positions. We use cosine similarity following prior works (Gao et al., 2021; Ye et al., 2022). The similarity score is optionally scaled by a temperature parameter τ , which controls the peakiness of the softmax distribution and in turn determines the relative importance of non-translation pairs. This temperature parameter is tuned on the development sets.

5.4.3 Alternate Training

As shown in Figure 5.3, we alternate between optimizing the task-specific objective and the alignment objective across training steps. Compared to joint optimization that computes a combined loss for both objectives and performs a single backward pass, this approach does not involve manually balancing objective weights and mitigates potential gradient conflicts between the two training objectives.

To illustrate these gradient conflicts, we sampled 1,280 training instances for each loss function at training initialization and repeated this procedure 10 times. We computed gradients for all trainable parameters and measured gradient alignment using cosine similarity, where positive values indicate aligned gradients and negative values indicate conflicts. Figure 5.4 shows the average cosine similarities between: (1) different batches within the task objective, (2) different batches within the alignment objective, and (3) batches across different objectives. The positive intra-objective similarities contrasted with negative inter-objective similarities confirm the gradient conflicts between the two training objectives.

Note that the task and alignment losses can be activated separately. Deactivating the alignment loss degenerates to standard task-only training. Conversely, deactivating the task loss trains the model only for alignment. The modularity allows combining separately-trained task and alignment models.

⁴Initial experiments with attention pooling degraded performance. We also tried a stop-gradient operator on English representations to align non-English representations towards English, but it did not give consistent gains.

5.5. Experimental Setup

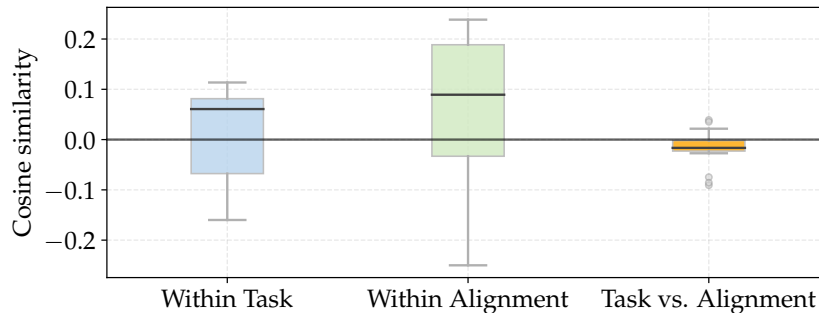


FIGURE 5.4: Gradient similarity, measured as the cosine similarity between the gradients of all trainable parameters (flattened into vectors), is generally positive within both task-specific and alignment objectives. It becomes negative when comparing task objectives against alignment objectives, indicating conflicts.

5.4.4 Data Requirement

Our approach requires minimal parallel data. Later experiments show that for lower-resource languages, a few hundreds of sentences of parallel data is sufficient to improve transfer. This data requirement is feasible to satisfy with existing multilingual datasets such as Tatoeba (Tiedemann, 2020), which covers over 300 languages. Our approach also offers a practical advantage over alternatives that require monolingual language modeling training for each transfer target language (Ansell et al., 2022; Vu et al., 2022b; Chronopoulou et al., 2024).

5.5 Experimental Setup

5.5.1 Datasets

In general, we fine-tune on several high-resource languages and then evaluate transfer performance on additional languages. We do not focus on English-only fine-tuning, since our initial experiments demonstrated that multilingual fine-tuning substantially outperforms English-only fine-tuning (§5.6.1), thus establishing it as a stronger baseline. Table 5.1 presents a dataset overview. Descriptions of the language codes are available in Appendix C.2.

Main Task Data

We evaluate our approach on slot filling and machine translation, both modeled as generation tasks with templates shown in Appendix C.3. For slot filling, we use the MASSIVE dataset (FitzGerald et al., 2023). We train on 5 high-resource languages, and evaluate transfer performance on 15 additional diverse languages, 5 of which have non-Latin writing systems. This task presents a challenge due to the 60 possible slots, requiring strictly following the output format for correct parsing. For machine translation, we use ALMA (Xu et al., 2024)’s training and test data, and additionally test on 6 zero-shot directions from WMT 23 (Kocmi et al., 2023).

	Dataset	Languages
Slot Filling		
Task - train	MASSIVE	{ar, en, es, ru, zh}
Task - test	MASSIVE	supervised + {af, az, cy, de, el, fr, hi, is, ja, jv, sw, th, tl, tr, ur}
Alignment	Tatoeba	low-resource: {cy, jv, jp, sw, tl}-en mid-resource: {el, hi, th, tr}-en high-resource: {ar, es, ru, zh}-en
Machine Translation		
Task - train	ALMA	{cs, de, is, ru, zh} ↔ en
Task - test	WMT 23	supervised + {he, ja, uk} ↔ en
Alignment		(same as “Task - train”)
JSON Generation (challenge task)		
Task - train	UNER	{en, pt, zh}
Task - test	UNER	supervised + {da, hr, sk, sr, sv}
Alignment	Tatoeba	{da, sv}-en
Semantic Alignment Evaluation (diagnostic task)		
Alignment	FLoRes-200	$N(N - 1)$ pairs for N languages

TABLE 5.1: Dataset statistics for three downstream tasks and one diagnostic task. “Train” refers to languages involved in SFT, and “test” includes SFT languages and additional transfer languages unseen during training. See Appendix C.1 for more details.

Challenge Task Data

To assess performance on long-sequence processing and structured text generation, we include JSON generation as a challenge task. We use the UNER dataset (Mayhew et al., 2024) from the Aya collection (Singh et al., 2024), which requires following example instructions and extracting named entities into JSON format. A challenge not present in the previous tasks is the longer inputs, with an average input length exceeding 150 tokens in English. For this task, we train on 3 high-resource languages (en, pt, zh) and transfer to the 5 remaining languages.

Alignment Data

For alignment, we mainly use parallel data to English from Tatoeba (Tiedemann, 2020), except for machine translation, where the training sentences are inherently parallel. For slot filling, our main experiments align the five languages with the weakest baseline⁵ transfer performance (cy, jv, jp, sw, tl) reported by the dataset creators (FitzGerald et al., 2023). We choose them because their weak baseline performance suggests a lack of effective transfer, providing a strong testbed for evaluating the potential benefits of our alignment approach. For ablation, we alter the following factors of the alignment data:

- Resource level (low, medium, high-resource)
- Language coverage
- Domain (oracle data, different, very distant)

⁵their baseline is an XLM-R model trained on English

5.5. Experimental Setup

For machine translation, given the inherent semantic equivalence of translation pairs, we directly leverage the translation data for alignment. For JSON generation, we align the two lowest-resourced in UNER (da and sv)⁶ to English. The lowest-resource alignment languages have a few hundred parallel sentences: Javanese (264), Swahili (371), Welsh (823). Further details on the alignment data are in Appendix C.1.

5.5.2 Evaluation

Semantic Alignment Evaluation

As described in §5.3, we evaluate cross-lingual semantic alignment by retrieval accuracy. Given N languages, we perform many-to-many retrieval and average the accuracy over the $N(N - 1)$ language pairs. For the initial analyses (§5.3), the 35 languages are listed in Appendix C.2. We use the FLoRes-200 (NLLB Team, 2024) development set with 997 parallel sentences. While FLoRes partially overlaps with ALMA’s training data, it remains the only reliable massively multilingual multiway corpus to the best of our knowledge. Alternatives such as Tatoeba have been advised against due to data imbalance and noise (Heffernan et al., 2022; Janeiro et al., 2025). We also demonstrate that this overlap does not result in memorization effects (§5.7.3). When reporting an aggregated retrieval accuracy for a model, we average over all language pairs at even-numbered layers’ retrieval accuracy, excluding the input embedding layer.

Task Performance Evaluation

For slot filling, we report F_1 scores using the original evaluation script by FitzGerald et al. (2023). For machine translation, we report BLEU (Papineni et al., 2002) and COMET-22 (Rei et al., 2022) scores. For JSON generation, we parse the generated outputs back to named entity tuples and then evaluate F_1 scores.

5.5.3 Model, Training, and Inference

We build upon Llama (AI @ Meta et al., 2024) and Qwen (Qwen Team et al., 2025), specifically Meta-Llama-3-8B-Instruct and Qwen2.5-7B-Instruct. We use LoRA (Hu et al., 2022) adapters with a rank of 8 for all attention components and linear projections. The effective batch size is 128 for both objectives, with mini-batches of 32 examples considered for the contrastive objective⁷. Alignment data from different languages are re-sampled to an approximately uniform distribution. More details are in Appendix C.3.

⁶While Serbian (sr) is also low-resourced in UNER, we exclude it from alignment due to data quality. Running language identification reveals that many sentences in the Serbian alignment data are not actually in Serbian.

⁷While contrastive learning typically benefits from larger batch sizes (Chen et al., 2022), our initial experiments with increased batch sizes did not give consistent improvements.

5.6 Results

5.6.1 Necessity of Fine-Tuning and Multilingual Training Benefits

Before assessing our proposed approach, we establish supervised multilingual fine-tuning as an appropriate, strong baseline. We support this in two steps: first, by demonstrating the necessity of fine-tuning over zero-shot inference, and second, by showing the superiority of multilingual training over an English-only setup for cross-lingual transfer.

Zero-Shot vs. Task-Specific Fine-Tuning We first confirm that supervised fine-tuning (SFT) is critical, as zero-shot performance is inadequate. On slot filling, Llama 3 achieves only 6.6% F_1 on English as it struggles to adhere to the required output format. Given this poor performance, we omit zero-shot evaluation for other languages. For machine translation, supervised FT provides substantial gains of 4–6 COMET points over the zero-shot baseline, as shown in rows (2) and (5) vs. (1) and (4) in Table 5.4.

	ar	en	es	ru	zh	cy	ja	jv	sw	tl
English-only	59.8	82.5	82.4	65.8	61.6	60.3	39.7	37.8	39.8	57.5
Multilingual	75.5	81.7	74.5	77.6	73.8	44.0	65.8	41.0	42.8	65.0
	af	az	de	el	fr	hi	is	th	tr	ur
English-only	60.3	39.6	71.1	64.8	68.2	62.1	39.2	75.3	52.9	49.9
Multilingual	66.0	49.0	75.0	69.4	71.9	70.0	45.0	79.9	60.4	57.1

TABLE 5.2: F_1 scores for slot filling comparing English-only vs. multilingual fine-tuning. Gray background indicates languages used in supervised fine-tuning. On 14 of 15 languages unseen in fine-tuning, multilingual fine-tuning achieves stronger performance.

English-Only vs. Multilingual Fine-Tuning We show that multilingual training provides a stronger baseline for cross-lingual transfer than English-only. As shown in Table 5.2, for the MASSIVE dataset, the multilingual model substantially outperforms its English-only counterpart on languages unseen in fine-tuning. This result justifies our use of multilingual fine-tuning as the baseline in subsequent experiments.

5.6.2 Overall Performance Comparison

The main results are shown in Table 5.3 for slot filling and Table 5.4 for machine translation.

Gains in cross-lingual transfer with supervised performance preserved: Our approach improves cross-lingual transfer across different tasks and models. For slot filling, we observe gains in both supervised and transfer (F_1 +0.4 and +1.5 respectively) settings on Llama fine-tuning, with similar improvements on Qwen (F_1 +0.7 supervised, +2.0 transfer). In machine translation with Llama in row (3), our approach brings substantial gains when transferring to out-of-English translation directions (+1.5 BLEU, +1.1 COMET).⁸ For into-English directions, there is a modest improvement in +0.5 BLEU and +0.2 COMET. The larger gains on out-of-English directions suggest the approach is more

⁸The observation of alignment not improving supervised directions is in line with findings of Pan et al. (2021), where the purely contrastive learning setup also does not improve supervised scores over their baseline (“m-Transformer”).

5.6. Results

ID	Model	Slot Filling (MASSIVE)			
		Supervised (5 lang.)	Transfer (15 lang.)	Transfer (aligned)	Retrieval (all 20 lang.)
		F ₁	F ₁	F ₁	Acc.
(1)	LLAMA 3	–	–	–	39.1
(2)	+ SFT	76.6	60.2	51.7	39.4
(3)	+ alignment	77.0	61.7	55.5	73.2
(4)	QWEN 2.5	–	–	–	21.4
(5)	+ SFT	76.3	53.3	41.6	20.9
(6)	+ alignment	77.0	55.3	46.5	20.5

TABLE 5.3: Slot Filling results on MASSIVE dataset. Retrieval accuracy is averaged over all language pairs and layers. **Bold:** highest task scores which outperform the other setups. Language-specific results are in Appendix C.4.

ID	Model	Machine Translation (WMT23)						
		Supervised (5 lang.↔En)		Transfer (3 lang.→En) (En→3 lang.)				Retrieval (all 9 lang.)
		BLEU	COMET	BLEU	COMET	BLEU	COMET	Acc.
(1)	LLAMA 3	25.8	75.5	27.8	75.8	14.8	71.3	51.5
(2)	+ SFT	30.0	81.5	31.8	82.8	15.5	79.6	(55.3)
(3)	+ alignment	29.9	81.5	32.3	83.0	17.0	80.7	(84.5)
(4)	QWEN 2.5	23.0	74.5	28.5	81.3	12.6	71.2	36.5
(5)	+ SFT	27.4	78.4	29.7	82.7	14.6	76.9	(38.8)
(6)	+ alignment	27.2	77.6	30.8	82.7	14.7	76.9	(75.6)

TABLE 5.4: Machine Translation results on the WMT23 test set. Symbols ↔ and → and ← indicate translation directions. Retrieval accuracy is averaged over all language pairs and layers. **Bold:** highest task scores which outperform the other setups. (Results in brackets): potentially inflated scores due to partial overlap between retrieval and translation data. Language-specific results are in Appendix C.4.

beneficial for non-English generation in this case. For Qwen in row (6), our approach shows minor gains in into-English translation (+1.1 BLEU but no change in COMET), and does not influence out-of-English scores. It also leads to a degradation (−0.8 COMET) on supervised directions. This is potentially due to Qwen’s non-English-centric pre-training combined with our English-centric alignment data. With this exception, our approach maintains or improves supervised performance while enhancing transfer.

Aligned languages improve the most, but gains extend to other languages: The diverse language coverage in the slot filling dataset allows us to compare how the alignment objective benefits transfer to both aligned and non-aligned languages. While aligned languages show the strongest improvements (F₁ +3.8 and +4.9 for Llama and Qwen respectively), the benefits extend to other languages. Over the remaining 10 non-aligned languages, there is an average F₁ improvement of 0.4 (per-language results in Appendix C.4). This suggests that the alignment step enhances the model’s general cross-lingual transfer capabilities rather than optimizing for specific language pairs.

Smaller gains on non-Latin script languages: Beyond overall performance improvements, we observe smaller gains on languages with diverse writing systems. Specifically, for the

	Supervised $\uparrow$	Transfer $\uparrow$	Std dev gain over baseline
Slot filling (MASSIVE): F_1			
SFT baseline	76.6	60.2	—
Middle (layer 16)	77.0	61.7	2.6
Top (layer 32)	76.6	62.0	3.3
Bottom (layer 8)	76.8	58.0	3.0
Middle + Top	77.6	62.5	3.4
Machine translation (WMT23): COMET			
SFT baseline	81.5	79.6	—
Middle (layer 16)	81.5	80.7	3.9
Top (layer 32)	82.0	80.2	4.2
Bottom (layer 8)	81.2	80.1	5.6
Middle + Top	81.5	80.6	6.1

TABLE 5.5: Impact of alignment loss placement on supervised and transfer performance on Llama 3 for slot filling and machine translation.

non-Latin script transfer languages in the slot filling task (Greek, Hindi, Japanese, Thai, Urdu), the average improvement is only 0.5 F_1 in contrast to the overall average gain of 1.5. This reduced gain is likely related to suboptimal tokenization for these languages in multilingual models (Rust et al., 2021; Petrov et al., 2023; Hong et al., 2024). When tokens poorly align with linguistic units, the mean-pooled sentence representations may poorly capture semantics, thereby impacting our alignment objective.

No catastrophic forgetting observed: An interesting observation in Table 5.4 is the absence of catastrophic forgetting. Fine-tuning on selected language pairs for translation enhances performance on languages unseen during this fine-tuning stage. This finding stands in contrast to observations in previous chapters, where standard fine-tuning often led to performance degradation on previously learned languages. The underlying reasons for this discrepancy will be investigated in detail in the next chapter.

5.6.3 Alignment Loss Placement

To validate our choice of middle-layer alignment motivated by the analysis in §5.3, we compare performance when applying the alignment loss at different network depths: bottom (8th), middle (16th), and top (32nd) layers of Llama. We also combine middle and top alignment.

Middle layer placement achieves more balanced improvements in transfer languages: As shown in Table 5.5, compared to the “middle” configuration, the “bottom” configuration clearly leads to poor overall performance in both supervised and transfer settings, with a particularly strong degradation on the slot filling task. While top-layer alignment maintains overall strong performance, it shows more unbalanced gains across transfer languages, as evidenced by the higher standard deviation of performance changes on transfer languages.

Middle layer placement achieves better alignment across network depths: To better understand the effects of different loss placements, we run the translation retrieval task over model activations from different intermediate layers. As shown in Figure 5.5, When

5.7. Analyses

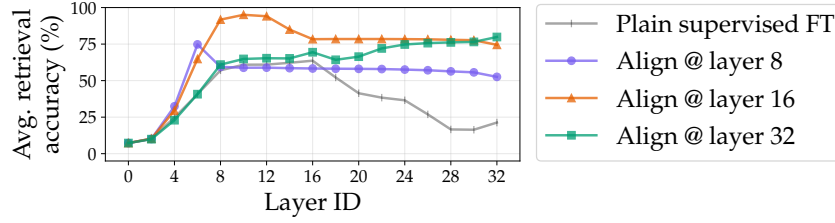


FIGURE 5.5: Retrieval accuracy over model depths when aligning different layers of Llama 3. Middle layer placement leads to overall better alignment.

the alignment loss is applied at the middle (16th) layer, semantic alignment is enhanced not only at that layer but also in multiple preceding layers. In contrast, top-layer alignment primarily affects only the final layer, and bottom-layer alignment shows limited improvement in alignment quality across all layers. This is likely because the lower layers are occupied with processing more fundamental text features (Belinkov et al., 2017; Peters et al., 2018) rather than abstract meanings.

Aligning several layers does not show consistent gains: Results in Table 5.5 suggest that aligning at multiple layers may be complementary. However, adding alignment losses at both middle and top layers brings further improvements on slot filling, but does not on machine translation. This task-dependent behavior indicates that how to best align multiple layers still requires further investigation.

5.6.4 Impact on Representation Retrieval

To assess the impact of the alignment loss on the learned model representations, we also report the retrieval accuracy for all languages involved in each task (20 for slot filling and 9 for machine translation) after fine-tuning (Table 5.3 and Table 5.4). For Llama on the slot filling task, the alignment loss substantially improves retrieval accuracy from 39.4% to 73.2%. For Qwen, the alignment loss does not improve retrieval among the 20 slot filling languages, possibly due to the lower accuracy of the base model with many low-resource languages with 0% accuracy, making improvement more challenging. For machine translation, as noted earlier in §5.5.2, the retrieval test data overlaps with part of the task training data, potentially inflating accuracy (marked in brackets in Table 5.4). However, we verify that this overlap does *not* lead to perfect retrieval accuracy: Specifically, at the 16th layer where the alignment loss is applied, English-source retrieval accuracies for supervised languages show varying accuracy: cs (98.1%), de (96.5%), is (66.9%), ru (90.6%), and zh (94.8%). This suggests that the overlap does not make the retrieval diagnostic task trivial.

5.7 Analyses

5.7.1 Larger-Scale Alignment

While our main configuration for slot filling (Table 5.1) allows studying performance on languages not involved in alignment, we also explore larger-scale alignment scenarios as

Alignment Setup	Avg supervised F1 _↑ (5 lang.)	Avg transfer F1 _↑ (15 lang.)
5 lang $\leftrightarrow$ en (row (3) of Table 5.3)	77.0	61.7
All $\leftrightarrow$ English (38 pairs)	77.5	63.6 (+1.9)
All $\leftrightarrow$ all (238 pairs)	77.7	63.6 (+1.9)

TABLE 5.6: Effects of larger-scale alignment configurations on slot filling. Aligning all non-English languages to English (+1.9 F1) outperforms our base configuration, while fully connecting all 20 languages offers no additional gain beyond English-only alignment.

Alignment Language Resource Level	Super. (5 lang.)	Transfer (15 lang.)	Gain on Aligned (4/5 lang.)
SFT (row (2) of Table 5.3)	76.6	60.2	–
Low (row (3) of Table 5.3)	77.0	61.7	+3.8
Medium	77.8	61.4	+1.1
High	77.6	60.4	+0.7

TABLE 5.7: Effect of alignment language resource levels on slot filling F1. In three groups of alignment languages: low ({cy, jv, ja, sw, tl}), medium ({el, hi, th, tr}), and high-resource ({ar, es, ru, zh}), languages involved in alignment consistently show improvements, with the strongest gains (+3.8 F1) in the low-resource group.

oracle setups where all languages have parallel data. We conduct additional experiments on slot filling with two expanded configurations:

- All 19 non-English languages aligned to English (38 directional pairs)
- All 20 languages aligned to each other (238 pairs with alignment data from all 380 possible pairs).

The results in Table 5.6 show that expanding alignment to all languages further improves transfer performance (F1 +1.9) in an oracle setup where every transfer language has alignment data. However, multiway alignment data does not further improve transfer, suggesting that aligning to English implicitly creates multiway alignment effects.

5.7.2 Resource Level of Alignment Languages

In our main experiments, we selected the 5 languages with the weakest performance from the MASSIVE baseline (FitzGerald et al., 2023) for alignment. We now vary the resource level of the alignment languages using a medium-resource group with {el, hi, th, tr}–en and a high-resource group with {ar, es, ru, zh}–en, which also have supervised task training data. As shown in Table 5.7, all three configurations improve F1 scores for the languages involved in alignment. However, the low-resource group exhibits the largest gains (+3.8 F1), indicating that our approach is most beneficial to languages with weaker initial performance. Moreover, overall transfer gains relative to the SFT baseline diminish when using high-resource languages for alignment, likely because these languages already have well-aligned representations and aligning them provides little benefit to lower-resource languages in the transfer set. Overall, the results show that our approach is robust to the choice of alignment languages, but selecting initially poorly aligned languages could provide broader benefits across different languages.

5.7. Analyses

Alignment Data	Overall (20 lang.)
Multilingual {ar,es,ru,zh,sw}-en	80.2
Only de-en	71.9
Only es-en	72.9
Only zh-en	72.7
de-en FLoRes (oracle)	77.7
Tatoeba (different)	71.9
IWSLT (very distant)	68.5

TABLE 5.8: Alignment generalization across languages and domains. *Upper*: Multilingual training improves overall alignment. *Lower*: Impacts of alignment transfer reasonably across domains, with performance drops when training data differs from test domain.

5.7.3 Generalization of Learned Alignment

Table 5.8 examines the language and domain generalization of our alignment component. To isolate the effects of task-specific joint training, we train the models using only the alignment loss, following the same setup as our previous experiments but without optimizing on task-specific data. We then evaluate retrieval accuracy as described in §5.5.2.

Language Generalization

To better understand the role of learning the alignment modules with multilingual data, we now use single languages paired with English for alignment, unlike in our main experiments that align multiple language pairs. As shown in Table 5.8 (upper portion), single-language-to-English alignment training leads to diminished performance compared to multilingual training. Interestingly, we see comparable accuracy drops regardless of which individual language is used for alignment, suggesting that the gains of multilingual alignment come from the diversity of the training data rather than characteristics of individual languages.

Domain Generalization

To isolate the effects of multilinguality, we focus on alignment between a single language pair (English-German). In Table 5.8 (lower portion), we first establish an oracle setup using models trained on FLoRes data (Wikipedia domain, overlapping with retrieval data). We then compare to two setups where the alignment data come from other domains: Tatoeba (short sentences for language learning; different) and IWSLT 2017 (public speaking transcriptions; very distant). While we observe a decrease in retrieval accuracy compared to the oracle setup, the results suggest that, to enforce alignment into the model, it is not strictly necessary to source alignment data from the same domain as the task-specific data.

5.7.4 Merging Alignment and Task Modules

Our previous experiments focused on models jointly trained on both task and alignment objectives. However, in practice, it may be necessary to enhance existing task-specific

Setup	Supervised	Transfer
Slot filling (MASSIVE): $F_1\uparrow$		
SFT (row (2) of Table 5.3)	76.6	60.2
Joint (row (3) of Table 5.3)	77.0 (+0.4)	61.7 (+1.5)
Merge	76.9 (+0.3)	61.3 (+1.1)
Machine translation (WMT23): COMET $\uparrow$		
SFT (row (2) of Table 5.4)	81.5	79.6
Joint (row (3) of Table 5.4)	81.5 (+0.0)	80.7 (+1.1)
Merge	82.0 (+0.5)	80.2 (+0.6)

TABLE 5.9: Performance comparison of merged alignment and task modules versus joint training. Post-hoc merging of separately-trained LoRA adapters achieves comparable improvements to joint training.

	Supervised (en, pt, zh)	Transfer (da, sv)	Transfer (5 lang.)
Llama SFT	83.4	82.1	79.3
+ alignment	82.4	83.1	79.8

TABLE 5.10: Results on JSON generation evaluated with F_1 , showing modest gains for aligned languages but decreased performance for supervised languages.

models with cross-lingual capabilities, where joint re-training is infeasible due to computational constraints or unavailability of the original task training data. Inspired by recent advances in model merging (Matena and Raffel, 2022; Ilharco et al., 2023; Chronopoulou et al., 2023a), we explore the feasibility of combining separately-trained task and alignment modules. We merge two sets of trained LoRA adapters by averaging their weights⁹: the alignment module trained in isolation (§5.7.3), and task-specific modules (rows (2) and (5) in Table 5.3).

Table 5.9 shows that this post-hoc merging brings improvements comparable to joint training. Moreover, the improvements are more evenly distributed across languages compared to the larger gains observed on languages used directly in alignment. These results demonstrate that our alignment approach is modular and can be combined with existing task-specific models.

5.7.5 Long Sequence Processing

We investigate a more challenging task requiring longer input and output generation using UNER (§5.5.1). As shown in Table 5.10, while aligned languages still show improvements, the gains are more modest compared to previous experiments, with an F_1 increase of 1.0 on aligned languages and 0.5 across all transfer languages. Moreover, there is an average degradation of 1.0 F_1 on supervised languages, mainly due to the decline in Chinese ($-2.2 F_1$). We suspect that this is due to our sentence-level alignment objective operating on fixed-length representations, which creates conflicts with processing longer sequences. As Chinese is the only character-based language in the JSON generation dataset, which

⁹We use a weighted average tuned on the development set (details in Appendix C.3)

has roughly twice the number of tokens compared to English of equivalent content, the conflict could be more influential for Chinese.

5.8 Summary

This chapter investigates how to effectively integrate cross-lingual alignment into the task-specific fine-tuning of decoder-only LLMs, with the goal of improving transfer to languages unseen during fine-tuning. We propose a modular middle-layer alignment objective that can either be trained jointly with the task objective or be integrated post-hoc by merging separately trained alignment parameters. Building upon the generalist nature of the underlying models, our evaluation extends beyond translation to a variety of downstream tasks. The experiments demonstrate that our approach improves transfer performance and enhances alignment across all network depths.

Our results also lead to several practical recommendations for improving transfer performance after LLM task-specific fine-tuning: **1)** Aligning a few weakly-performing languages yields broad transfer benefits. A few hundreds of parallel sentences as alignment data are sufficient. **2)** Alignment data can be sourced from different domains than the task. **3)** Existing task-specific models can be retrofitted with cross-lingual capabilities via parameter merging, avoiding the cost of full retraining.

This chapter extends our investigation from translation-specific to generalist models. While the results support the benefits of cross-lingual alignment, they also reveal a crucial difference from our previous findings. That is, the catastrophic forgetting that occurred during fine-tuning in the last chapter was not observed here. This discrepancy likely results from different factors, including the underlying model architecture, the use of parameter-efficient fine-tuning, and the amount of training data. A systematic investigation into the influence of these factors will be the focus of the next chapter.

Chapter 6

Conditions for Catastrophic Forgetting in Multilingual Translation

The previous two chapters demonstrated the effectiveness of fine-tuning pre-trained models for specialized use cases. This specialization process introduces the risk of catastrophic forgetting (McCloskey and Cohen, 1989), where a model’s capabilities significantly degrade on tasks or languages not present in its fine-tuning data. However, this risk did not emerge uniformly in the two chapters. The fine-tuning process in Chapter 4 led to substantial forgetting, while the same process in Chapter 5 did not, despite both following state-of-the-art fine-tuning practices in their contexts.

This discrepancy suggests that the occurrence of catastrophic forgetting is contingent on several variables, such as the underlying model architecture, the fine-tuning strategy, and the scale of the training data. To isolate the influence of these variables, this chapter presents a series of controlled experiments to disentangle their effects and identify the conditions that lead to forgetting. We also examine how cross-lingual alignment techniques investigated previously impact models that exhibit different degrees of forgetting, connecting this analysis to the overarching goal of learning generalizable representations. This chapter is partly based on Liu and Niehues (2025a).

6.1 Introduction

A common application for multilingual pre-trained models is to fine-tune them on specific languages or language pairs (Liu and Niehues, 2022; Gow-Smith and Sánchez Villegas, 2023; Gu et al., 2023). Ideally, this specialization would not degrade, and could potentially enhance, performance on unseen languages through positive transfer, as illustrated on the left of Figure 6.1. However, empirical evidence frequently demonstrates the contrary. Models often lose proficiency in languages excluded from the fine-tuning data (Vu et al., 2022b; Sun et al., 2023; Winata et al., 2023), as shown on the right of Figure 6.1.

Machine translation (MT) serves as a compelling testbed for studying multilingual catastrophic forgetting. First, a model supporting N languages encompasses $N(N - 1)$ directed translation pairs, creating a large and structured space to analyze forgetting patterns. Second, languages can be “unseen” in different roles. For example, a language may be present only as a source language, only as a target language, or in specific source-target pairs that were never explicitly trained. This enables fine-grained analysis of how

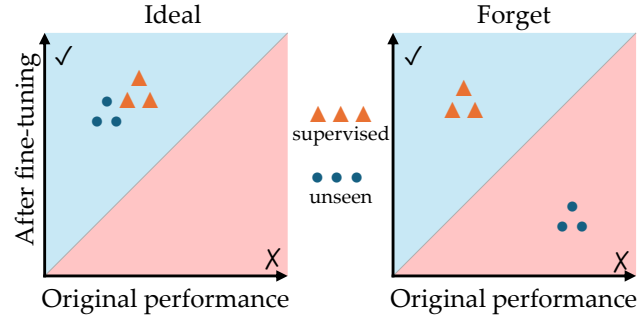


FIGURE 6.1: Selectively fine-tuning on some tasks or languages may lead to positive transfer (left) or catastrophic forgetting (right).

different types of exposure during fine-tuning affect retention. Moreover, forgetting can occur asymmetrically, where a model may retain the ability to translate from language A to B while losing the reverse direction.

Besides our own results from the previous two chapters, the broader literature also presents a fragmented and sometimes contradictory picture of when catastrophic forgetting occurs in MT. On one hand, studies on traditional encoder-decoder MT models trained from scratch (Berard, 2021) and some large pre-trained models (Vu et al., 2022a; Liu and Niehues, 2022; Liu et al., 2023; Lai et al., 2023) report severe forgetting after standard fine-tuning, where the ability to translate unseen directions is almost entirely lost. These findings suggest that catastrophic forgetting is an inevitable consequence of selective specialization. On the other hand, recent works on large language models (LLMs) provided mixed evidence. Richburg and Carpuat (2024) demonstrated that fine-tuning Llama 2 (Touvron et al., 2023) and Tower (Alves et al., 2024) models on specific language pairs could improve performance on unseen pairs, indicating positive transfer. Conversely, Zan et al. (2024) found that their fine-tuned Llama 2 models performed very poorly on unseen directions, again indicating issues with forgetting.

These conflicting results raise fundamental questions about the factors leading to catastrophic forgetting. To what extent do model architecture (encoder-decoder versus decoder-only), scale, or fine-tuning methodology determine whether a model forgets or generalizes? How do factors like the volume of fine-tuning data, the use of parameter-efficient fine-tuning (PEFT), or instruction-following capabilities influence the retention of multilingual abilities? To resolve these ambiguities, we conduct a systematic study to identify the conditions that trigger catastrophic forgetting in multilingual MT. We systematically control for key variables, including model architecture and size, fine-tuning data composition and scale, full-parameter vs. parameter-efficient fine-tuning, and instruction-following versus standard fine-tuning approaches.

Building on this analysis, we then evaluate how the cross-lingual alignment techniques studied previously, namely adversarial, similarity-based, and contrastive methods, perform under these different forgetting conditions. Our goal is to determine whether these techniques primarily reverse performance degradation caused by fine-tuning, or if they can also facilitate positive transfer, leading to capabilities that surpass those of the original

6.2. Related Work

pre-trained model.

6.1.1 Task Definition

In this chapter, we study the following problem:

Given a pre-trained model (either translation-specific or general-purpose) with initial capabilities to translate $N(N - 1)$ directions across N languages, and a fine-tuning dataset on a subset of m languages, each paired with a common bridge language, under what conditions does specialization on these $2m$ directions lead to catastrophic forgetting on the remaining translation directions unseen during fine-tuning?

Since an exhaustive evaluation of all $N(N - 1) - 2m$ directions is infeasible, we focus on a representative set of these unseen directions: those with an unseen source language, an unseen target language, or translation directions where the source and target languages appear independently in the training data but are never paired together. This investigation tackles both the compositional and temporal aspects of generalization.

6.1.2 Contributions

This chapter provides a systematic investigation into the conditions that induce catastrophic forgetting in multilingual models. To quantify these effects, we introduce gain-forgetting analyses that contextualize model performance before and after fine-tuning. Through a series of controlled experiments, we make the following key contributions:

- The relative scale between pre-trained model parameters and fine-tuning data volume is a critical factor in catastrophic forgetting, with smaller models fine-tuned on larger datasets being most vulnerable (§6.4.1).
- A model’s instruction-following ability, rather than its underlying architecture (encoder-decoder versus decoder-only), is a primary factor in resisting forgetting (§6.4.2).
- Contrary to common assumptions, parameter-efficient fine-tuning using LoRA (Hu et al., 2022) provides no significant advantage over full fine-tuning in preventing catastrophic forgetting under our experimental conditions (§6.4.4).
- Besides mitigating forgetting, cross-lingual alignment methods may facilitate positive transfer, with improvements observed on translation directions with unseen target languages (§6.5).

6.2 Related Work

6.2.1 Catastrophic Language Forgetting in Machine Translation

Catastrophic forgetting in machine translation has been extensively studied. Dakwale and Monz (2017); Thompson et al. (2018, 2019) established that domain-specific fine-tuning degrades performance on previously learned domains, where a domain refers

Model	Parameters	Previous Usage	Model Type
M2M-124-0.2B	175M	Chapter 4	Translation-specific
M2M-124-0.6B	615M	–	
Qwen2.5-0.5B-Instruct	494M	–	Generic instruction-following
Qwen2.5-7B-Instruct	7B	Chapter 5	
Llama-3-8B-Instruct	8B	Chapter 5	

TABLE 6.1: Overview of base models used in this chapter, including their parameter counts, previous usage in earlier chapters, and model type, where translation-specific models are pre-trained on massively multilingual machine translation, and generic instruction-following models are general-purpose language models that follow instructions.

to a subject area or text style with its specialized vocabulary and linguistic conventions. Many subsequent works have investigated the underlying mechanisms and mitigation strategies for domain forgetting in MT, e.g., [Gu and Feng \(2020\)](#); [Saunders and DeNeefe \(2024\)](#); [Eschbach-Dymanus et al. \(2024\)](#); [Wu et al. \(2024b\)](#); [Hu et al. \(2024\)](#). Compared to domain forgetting, the multilingual aspect of forgetting has received less attention. Our findings in [Chapter 4](#) are consistent with the literature highlighting this problem. [Berard \(2021\)](#) demonstrated severe language forgetting in conventional encoder-decoder-based MT models during standard fine-tuning on selected languages, while [Vu et al. \(2022a\)](#) showed that domain-specific fine-tuning compounds forgetting across both domains and languages. [Liu et al. \(2023\)](#) confirmed that standard fine-tuning consistently triggers catastrophic forgetting of unseen language pairs, even in large pre-trained models. Besides language and domain forgetting, models also lose in-context learning abilities after fine-tuning ([Alves et al., 2023](#)).

6.2.2 Factors Influencing Catastrophic Forgetting

The scale of both the model and its pre-training data has been identified as a key factor in mitigating catastrophic forgetting ([Ramasesh et al., 2022](#)). Our study extends this analysis by examining the relative scale between the model and the fine-tuning data. The choice of fine-tuning methodology, particularly parameter-efficient fine-tuning with Low-Rank Adaptation (LoRA) ([Hu et al., 2022](#)) being a representative technique, is another contested factor. [Kalajdziewski \(2024\)](#) suggests that LoRA does not resolve catastrophic forgetting, while [Biderman et al. \(2024\)](#) demonstrate that LoRA “learns less and forgets less”. The finding by [Zhang et al. \(2024\)](#) that the optimal fine-tuning method is highly task-dependent warrants a specific investigation for the task of multilingual MT.

6.3 Controlled Setting to Study Catastrophic Forgetting

As introduced at the beginning of this chapter, the experimental setups in [Chapter 4](#) and [Chapter 5](#) differ in several ways, including model architecture (task-specific vs. instruction-following), fine-tuning strategy (full vs. parameter-efficient with LoRA), and data scale (large vs. small). While both configurations represent standard approaches in their respective contexts, these different factors prevent a clear attribution of the observed

6.3. Controlled Setting to Study Catastrophic Forgetting

differences in catastrophic forgetting to any single variable. To disentangle these effects, this chapter introduces a controlled experimental setting. By systematically isolating each factor, we conduct an analysis of its specific impact on model generalization and knowledge retention. This investigation is structured along two dimensions, namely the choice of base model and the characteristics of the training dataset.

6.3.1 Base Models

The selection of models is designed to facilitate direct comparisons across the dimensions of model architecture, scale, and fine-tuning method. An overview of all base models and their configurations is provided in [Table 6.1](#).

Translation-Specific Models

Extending the setup from [Chapter 4](#), we use the M2M-124 ([Goyal et al., 2022](#)) in two sizes. In addition to the M2M-124-0.2B variant, we include the M2M-124-0.6B version to study the impact of model scale and to provide a comparison point against instruction-following models of a similar size.

Instruction-Following Models

Building on the approach from [Chapter 5](#), we focus on models from the Qwen 2.5 ([Qwen Team et al., 2025](#)) and Llama 3 ([AI @ Meta et al., 2024](#)) families, extending the analysis to more scales and fine-tuning configurations. The Qwen2.5-0.5B-Instruct model, with a parameter count similar to M2M-124-0.6B, facilitates a direct comparison between translation-specific and instruction-following architectures. The larger Qwen2.5-7B-Instruct serves as a large-scale baseline, which we evaluate under both full fine-tuning and LoRA to isolate the impact of the fine-tuning method. We further include the similarly-sized Llama-3-8B-Instruct model, also fine-tuned using LoRA.

6.3.2 Data

Dataset Overview

To assess the effects of training data volume and composition, we replicate and expand the setups from [Chapter 4](#) and [Chapter 5](#). A data overview is in [Table 6.2](#).

The first configuration, denoted `SMALL`, builds upon the setup detailed in [Chapter 5](#) by introducing additional evaluation scenarios. For training, we use data from the ALMA collection ([Xu et al., 2024](#)), which consists of five languages paired with English: Czech (cs), German (de), Icelandic (is), Russian (ru), and Chinese (zh). To evaluate generalization to unseen languages, we employ the same WMT 2023 test set ([Kocmi et al., 2023](#)) from the previous chapter, which covers Hebrew (he), Japanese (ja), and Ukrainian (uk). This test set provides 3 unseen source-to-English and 3 English-to-unseen-target translation directions. We further extend the evaluation to assess translation capabilities between the

Dataset	Details
SMALL	Training Data: ALMA (117K sentence pairs) Training directions: {cs, de, is, ru, zh} ↔ en Testing directions: - Unseen pair (20 directions): {cs, de, is, ru, zh} ↔ {cs, de, is, ru, zh} - Unseen source (3 directions): {he, ja, uk} → en - Unseen target (3 directions): en → {he, ja, uk}
LARGE	Training Data: WMT21 large-scale multilingual track (54M sentence pairs) Training directions: {jv, ms, tl} ↔ en Unseen testing directions: - Unseen pair (6 directions): {jv, ms, tl} ↔ {jv, ms, tl} - Unseen source (4 directions): id → {en, jv, ms, tl} - Unseen target (4 directions): {en, jv, ms, tl} → id

TABLE 6.2: Dataset overview for training and testing configurations for both small and large-scale experiments.

5 non-English training languages, creating 20 language pairs using multiway test data from WMT 2024 (Kocmi et al., 2024a).

The second configuration, denoted LARGE, adopts the setup from Chapter 4 and uses data from the WMT 2021 Large-Scale Multilingual Shared Task (Wenzek et al., 2021). The training data comprises three Austronesian languages paired with English: Javanese (jv), Malay (ms), and Tagalog (tl). For evaluation, we introduce Indonesian (id) as an unseen but related language, assessing translation to and from all 4 languages in training. This results in 4 unseen source language and 4 unseen target language directions. No additional languages are introduced to maintain consistency with the language coverage in Chapter 4. All evaluation data are from the FLoRes-101 test set (Goyal et al., 2022).

Moreover, to isolate the effect of data quantity, we also create subsampled versions of the LARGE dataset, each with 12K, 120K, and 1.2M sentences per language pair.

Unseen Conditions

We evaluate catastrophic forgetting on three types of unseen language pairs. Our analysis focuses on pairs where at least one language was seen during fine-tuning, as pairs with two unseen languages consistently showed severe performance degradation in preliminary experiments. The three categories are:

- **Unseen Pair:** Both the source and target languages are present in the fine-tuning data, but not in combination. This is the most challenging category as explained next.
- **Unseen Source:** The source language has not been seen during fine-tuning, but the target language has.
- **Unseen Target:** The target language has not been seen during fine-tuning, but the source language has.

Difficulty of the Three Unseen Conditions

Among the three evaluated categories, the “unseen pair” scenario presents a unique challenge. Although it may seem counterintuitive, this case is often more difficult than

6.3. Controlled Setting to Study Catastrophic Forgetting

scenarios involving languages completely unseen during fine-tuning. The primary reason for this difficulty lies in the English-centric nature of the fine-tuning dataset. Because all training examples are paired with English, the model learns an implicit association that a specific source language uniquely predicts English as the target language, which represents a spurious correlation (Gu et al., 2019). When presented with an unseen pair, e.g., from German to Czech after fine-tuning on English-Czech and German-English translation, the model has been implicitly trained to associate German inputs with English outputs. Direct German-Czech translation requires the model to override this spurious correlation.

In contrast, the other two conditions do not present this conflict to the same level. For an unseen source language, the model has not formed any directional association during fine-tuning. Therefore, there is no learned association that needs to be overridden. The unseen target language scenario is also comparatively less difficult. Specifically, as long as the source language has to translate into multiple different target languages during training or has not been seen in training, the model does not learn a one-to-one mapping to a single output. This condition applies to four of the seven unseen target language scenarios ($\text{en} \rightarrow \{\text{he}, \text{ja}, \text{uk}, \text{id}\}$), where the source language was part of a multi-target translation setup. This configuration discourages over-specialization toward a single output language, thereby reducing the overall difficulty of translating into a novel target language for this category.

Language Control Mechanisms

We specify the target language according to the conventions of each base model. For M2M-124, we employ its native token-based control, prepending the source and target sentences with their respective language tokens. For instruction-following models, we use the instruction: “Translate the given sentence from [source language] to [target language]”, followed by the source sentence. In ablation studies, we also test the effect of providing these instructions in the target language itself.¹

Training, Inference, and Evaluation

We compare full fine-tuning, where all model parameters are updated, with parameter-efficient fine-tuning using LoRA. The experimental setups follow Chapter 4 and Chapter 5. We note a minor correction to the implementation in Chapter 5. The experiments there contained an unintended behavior where the alternation between task and alignment objectives occurs across gradient accumulation steps instead of minibatches, the correct behavior. We have verified that this difference in implementation does not alter the conclusions reported in Chapter 5. Following the configuration in Chapter 5, our LoRA setup uses a rank of 8 and an α of 16, applying adapters to all self-attention components (Query, Key, Value, Output, Gate) and linear projections. This LoRA configuration was chosen after initial experiments applying LoRA to fewer components showed weaker

¹Instructions were translated using DeepL. For languages not supported by DeepL, we used Google Translate.

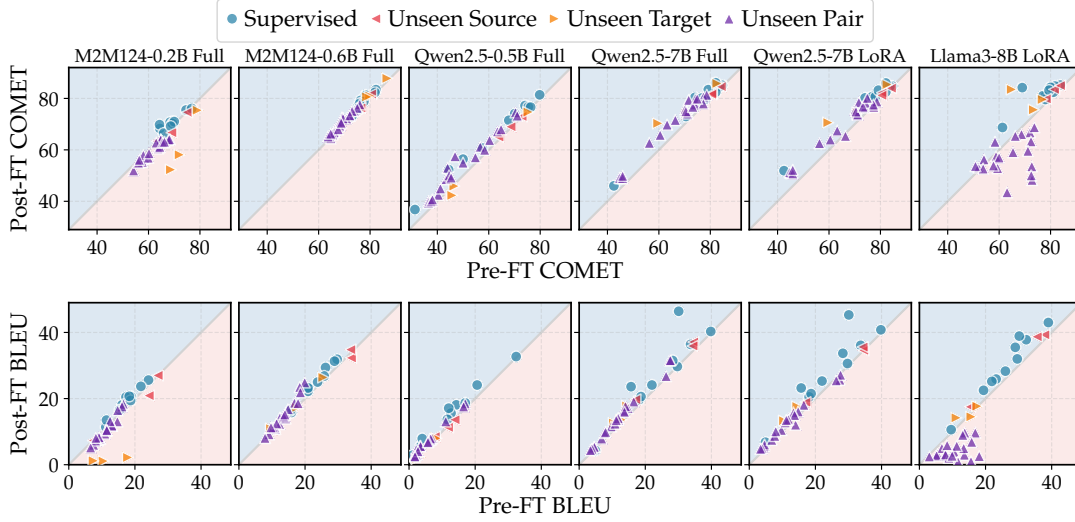


FIGURE 6.2: Gain-forgetting plots on the SMALL dataset (117K sentence pairs). Catastrophic forgetting is minimal, except unseen language pairs on Llama (addressed later in Table 6.4).

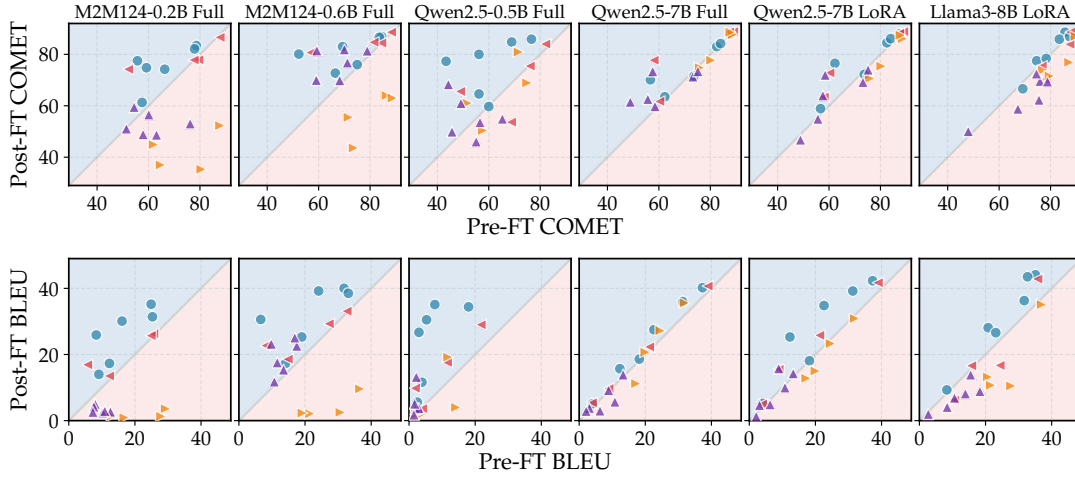


FIGURE 6.3: Gain-forgetting plots on the LARGE dataset (54M sentence pairs). Catastrophic forgetting is more severe, especially with translation-specific models where they show performance collapse approaching 0 BLEU.

supervised performance. It also creates conditions more analogous to full fine-tuning than selective adapter application, minimizing potential confounding factors related to parameter coverage.

For evaluation, we report translation quality using both BLEU (Papineni et al., 2002) and COMET (Rei et al., 2022) as in previous chapters. For more targeted assessment of whether the model generates output in the correct language, we also measure language identification accuracy using the tool by Lui and Baldwin (2011).

6.4 Gain-Forgetting Analyses

We investigate the trade-off between performance gains on fine-tuned language pairs and potential catastrophic forgetting on those unseen during fine-tuning. To visualize this

6.4. Gain-Forgetting Analyses

relationship, we create scatter plots (Figure 6.2 and Figure 6.3) where each point represents a language pair’s performance before (x -axis) and after (y -axis) fine-tuning. The diagonal line ($y = x$) is a reference boundary, where points below indicate catastrophic forgetting, while those above indicate performance improvement.

6.4.1 Model Scale and Fine-Tuning Data Size

Impact of Model Size Larger model variants consistently exhibit greater resistance to catastrophic forgetting. For M2M-124 models, the 0.6B parameter variant shows fewer language pairs in the forgetting zone compared to its 0.2B counterpart. Similarly for Qwen2.5, the 7B model demonstrates substantially less forgetting than the 0.5B model across all language pairs. This confirms the finding from Ramasesh et al. (2022) that the base model scale helps mitigate forgetting.

Impact of Fine-Tuning Data Volume We additionally observe that the amount of fine-tuning data plays a crucial role in forgetting. By contrasting Figure 6.2 ($\sim 100\text{K}$ sentences FT data) and Figure 6.3 ($\sim 54\text{M}$ sentences FT data), it becomes clear that higher-data-volume fine-tuning leads to stronger forgetting across all model variants. This observation extends the findings of Ramasesh et al. (2022), by demonstrating that catastrophic forgetting is impacted not only by base model scale, but also by the intensity of task-specific training.

Controlled Analysis of Data Volume Effects To isolate the impact of data volume from dataset-specific factors (e.g., the ALMA dataset has higher-quality data), we conduct controlled experiments using subsampled portions of the WMT21 dataset. We systematically vary the amount of fine-tuning data while maintaining a consistent data source. Starting with 12K sentences per language pair (matching ALMA), we increase the volume by an order of magnitude at each step: $12\text{K} \rightarrow 120\text{K} \rightarrow 1.2\text{M}$ sentences per language pair. Figure 6.4 demonstrates the progressive increase in catastrophic forgetting as training volume grows. At 12K sentences per language pair, the gain-forgetting pattern resembles ALMA results, with most language pairs clustered near the diagonal line and minimal performance changes. At 120K sentences, a shift toward forgetting emerges, particularly for target languages unseen in fine-tuning. At 1.2M sentences, severe catastrophic forgetting occurs, though not to the full extent observed in the complete dataset. The progressive degradation suggests that the intensity of fine-tuning on specific language pairs impacts the forgetting patterns.

6.4.2 Architecture and Language Control Mechanism

To isolate the impact of model architecture and pre-training objectives on catastrophic forgetting, we compare two models of comparable scale but different designs: M2M-124-0.6B, a translation-specific encoder-decoder model, and Qwen2.5-0.5B, a general-purpose decoder-only model pre-trained for instruction following.

Forgetting Patterns Across Architectures In Figure 6.3 on the LARGE dataset, where forgetting effects are strongest due to large-scale fine-tuning data, both models exhibit

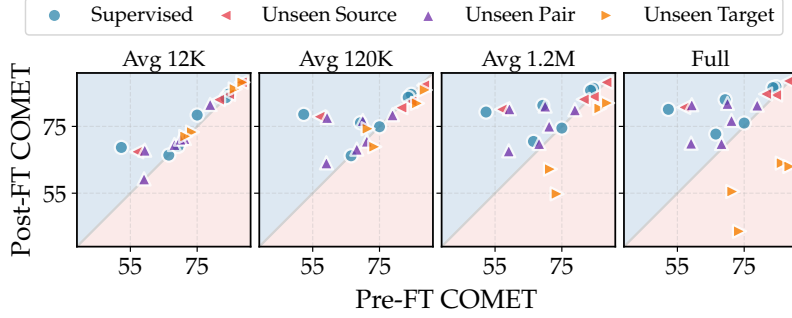


FIGURE 6.4: Controlled analysis of data volume effects by subsampled portions of the LARGE dataset. Forgetting becomes severe as fine-tuning data amount increases.

Model	Average		en→id	
	COMET	BLEU	COMET	BLEU
Qwen2.5-0.5B-Instruct original	64.0	8.0	71.6	11.8
Instruction-based fine-tuning	65.3	6.8	80.9	19.2
Language-token-based fine-tuning	60.9	5.2	72.1	14.0

TABLE 6.3: Effectiveness of different language control mechanisms on unseen target languages compared to the base model without fine-tuning. Instruction-based language control outperforms token-based control.

catastrophic forgetting with multiple language pairs falling below the diagonal. However, they differ in their forgetting patterns: M2M-124-0.6B exhibits severe performance degradation on unseen target languages, while Qwen2.5-0.5B shows modest forgetting. We hypothesize that this is due to the target language control mechanisms used by the models. As discussed in §6.3.2, M2M-124 relies on language-specific tokens prepended to both source and target sentences, with the target-side token determining the output language. In contrast, with Qwen, we use natural language instructions to specify the target language, leveraging its existing instruction-following capabilities. This instruction-based mechanism may support more generalizable language control and help mitigate catastrophic forgetting on unseen target languages. We examine this hypothesis next.

Isolating Language Control Mechanisms To test the previous hypothesis that natural language instructions facilitate language control, we conduct a controlled experiment by fine-tuning Qwen2.5-0.5B using the same token-based language specification format as M2M-124, as described in §6.3.2. This format eliminates natural language instructions entirely, allowing fair comparisons between models while holding the language control method unchanged. The results support our hypothesis that instruction-following paradigms provide superior language control. As shown in Table 6.3, when trained with token-based language control, Qwen2.5’s performance on unseen target languages drops substantially from 65.3 to 60.9 COMET averaged over 4 unseen target language pairs. To account for low initial performance in some non-English language pairs, we specifically examine the English-Indonesian pair, which has a stronger baseline. In this case, performance still degrades substantially from 80.9 to 72.1 COMET and from 19.2 to 14.0 BLEU. These results demonstrate that it is the instruction-following paradigm, rather than the

6.4. Gain-Forgetting Analyses

decoder-only architecture itself, that provides stronger protection against the catastrophic forgetting of target language.

Impact of In-Language Instructions Building on our previous findings regarding instruction following for language control, we investigate whether using instructions in the target language (in-language instructions) can mitigate catastrophic forgetting on unseen language pairs. While prior work on in-language instructions for multilingual LLMs shows mixed results (Marchisio et al., 2024; Mondshine et al., 2025; Liu et al., 2025; Enomoto et al., 2025), these studies primarily evaluate models out-of-the-box. In contrast, we focus specifically on the training effects of in-language instructions.

We focus on the Llama3-8B trained on the SMALL dataset, which exhibits strong catastrophic forgetting (rightmost plots in Figure 6.2). As the results in Table 6.4 suggest, for unseen language pairs affected by forgetting, in-language instructions substantially outperform English instructions. Specifically, average language accuracy improves dramatically from 22.1% to 82.0%, with corresponding translation quality gains as measured by COMET increasing from 57.2 to 70.9. Notably, this does not impact performance on supervised language pairs, and slightly improves performance on unseen target languages (COMET 79.6→80.3).

	Metric	Supervised	Unseen Pair	Unseen Source	Unseen Target
Original	COMET	76.9	63.8	81.1	71.8
	BLEU	26.0	10.9	29.5	14.8
	LangID	97.9	85.1	97.9	93.5
English instruction	COMET	81.4	57.2	82.8	79.6
	BLEU	30.0	4.4	31.8	15.5
	LangID	96.7	22.1	98.4	94.1
In-language instruction	COMET	81.7	70.9	82.9	80.3
	BLEU	30.3	14.4	32.4	16.1
	LangID	97.5	82.0	98.5	95.1

TABLE 6.4: In-language instructions recover catastrophic forgetting on unseen pairs, reversing a 6.6 COMET loss (63.8→57.2) into a 7.1 COMET gain (63.8→70.9) with Llama under the SMALL data configuration.

6.4.3 Analyses by Language Pair Types

The results in Figure 6.2 and Figure 6.3 also suggest that catastrophic forgetting patterns are strongly dependent on the language pair type.

Token-Based Control and Target Language Forgetting For translation-specific models (M2M-124 variants) which use specialized tokens for language control, performance degradation is most acute for unseen target languages. This is expected, as if the language token for a target language is never encountered during fine-tuning, the model’s ability to interpret it and generate the correct language catastrophically degrades.

Unseen Pairs as Main Vulnerability for Instruction-Following Models In contrast to translation-specific models, instruction-following models demonstrate greater resilience

on unseen target languages, a capability we attribute to the generalizable nature of natural language prompts (§6.4.2). However, these models are not immune to forgetting and are most susceptible when handling unseen language pairs. This is particularly evident with the Llama3-8B model. We hypothesize this vulnerability is compounded by the fact that these unseen pairs are often non-English-centric. Base models typically possess weaker zero-shot capabilities for such translation directions due to the prevalence of English in their pre-training data. Fine-tuning on a different, often English-centric, set of pairs appears to accelerate the forgetting of these already fragile, non-English-centric translation abilities.

6.4.4 Comparing LoRA and Full Fine-Tuning

From the fourth and fifth columns of [Figure 6.2](#) and [Figure 6.3](#), we observe that LoRA and full fine-tuning result in comparable levels of catastrophic forgetting. Note that we applied LoRA adapters to all components of self-attention and linear projections, thereby minimizing differences in parameter coverage as a confounding factor. Our finding differs from that of [Biderman et al. \(2024\)](#), who observed that LoRA mitigates forgetting when adapting models to dissimilar domains like code and math. We hypothesize that this difference is because our fine-tuning task (translation) requires a smaller domain shift for the base models, which already exhibit strong zero-shot translation capabilities, whereas adapting to code or math requires a larger deviation.

6.5 Cross-Lingual Alignment for Forgetting Mitigation

After identifying architectural and training factors that influence catastrophic forgetting, this section situates the analysis within the broader context of methods for creating generalizable representations. To this end, we conduct a comprehensive comparison, evaluating three previously introduced techniques: the similarity regularization and adversarial language classification loss that appeared in both [Chapter 3](#) and [4](#), and the contrastive loss from [Chapter 5](#). The performance of these methods is assessed across translation-specific and instruction-following model architectures, thereby consolidating the findings from the previous experiments. This evaluation also expands the scope of experiments to a more comprehensive set of unseen conditions, namely unseen language pairs, unseen source languages, and unseen target languages.

6.5.1 Evaluated Methods

We briefly re-introduce the three cross-lingual alignment methods evaluated in this analysis. Each technique aims to foster the creation of language-invariant representations through different mechanisms.

- Adversarial language classification (§3.8.2, §4.5.2): includes an adversarial language classifier that encourages language-agnostic representations by penalizing the model’s ability to predict the source language from hidden states.

6.5. Cross-Lingual Alignment for Forgetting Mitigation

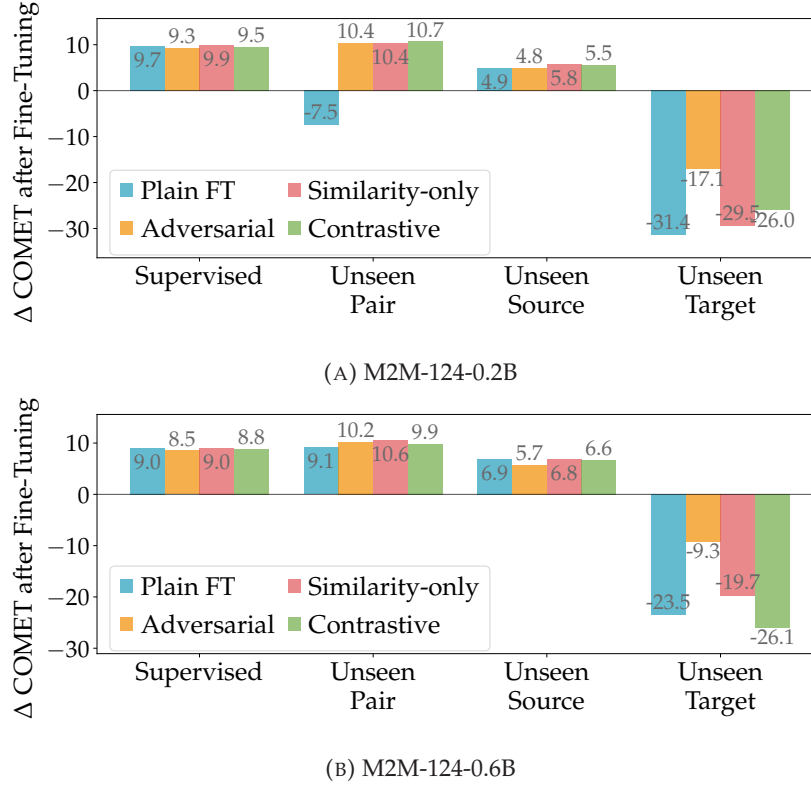


FIGURE 6.5: Following the experimental setup from [Chapter 4](#), we fine-tune translation-specific pre-trained models on the LARGE dataset, comparing standard fine-tuning (plain FT) against three cross-lingual alignment methods: adversarial, similarity-only, and contrastive. The alignment methods bring consistent improvements for unseen language pairs, but exhibit persistent forgetting in unseen target language scenarios.

- Similarity regularization (§3.8.1, §4.5.2): pulls together translation pairs without negative examples. While a naive implementation would lead to representation collapse, joint training with the translation loss mitigates this by maintaining discriminative power for the primary task.
- Contrastive loss (§5.4.2): uses a contrastive objective that pulls together representations of translation pairs while pushing apart representations of non-translation pairs.

These losses are applied on the encoder for encoder-decoder models, and on the middle layers of decoder-only model following the procedure in [Chapter 5](#).

6.5.2 Translation-Specific Models

We first evaluate the three alignment methods on translation-specific models: the 0.2B and 0.6B variants of M2M-124. The results are shown in [Figure 6.5](#), displaying change in translation quality for various language categories, comparing each alignment method against the plain fine-tuning baseline.

Gains in Unseen Language Pairs Among the three unseen categories, alignment methods primarily improve performance on unseen language pairs. These improvements are observed when plain fine-tuning causes forgetting ([Figure 6.5a](#)) and when it brings improvements ([Figure 6.5b](#)). For the 0.2B model, these methods reverse a -7.5 COMET

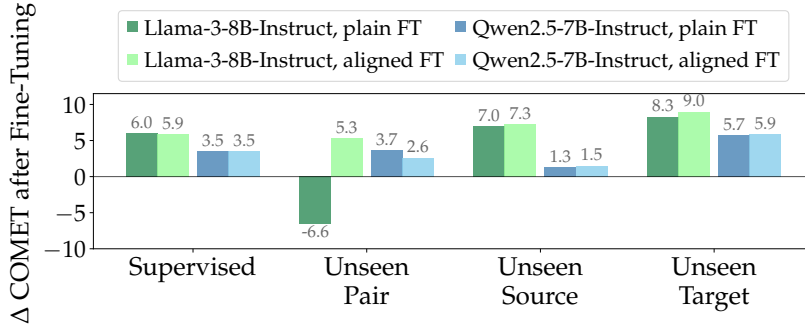


FIGURE 6.6: Following the experimental setup from [Chapter 5](#), we fine-tune general-purpose pre-trained models on the SMALL dataset with the middle-layer alignment approach from [Chapter 5](#). The alignment approach (aligned FT) reverses catastrophic forgetting compared to standard fine-tuning (plain FT). The impact on other conditions is milder.

loss by plain fine-tuning (60.5→53.0) into a gain of over 10 COMET. On the larger 0.6B model, the gains are more modest but consistent, ranging from +0.8 to +1.5 COMET over the plain fine-tuning baseline. In contrast, pairs with unseen source languages show minimal benefit from alignment methods, with performance remaining comparable to plain fine-tuning.

Persistent Forgetting in Unseen Target Languages The last column of [Figure 6.5](#) shows that all three approaches still result in drastic, double-digit COMET degradation for this category. This suggests an inherent weakness of the token-based language control mechanism discussed in §6.4.3. Cross-lingual alignment, while beneficial for transfer, struggles to overcome this fundamental limitation.

Similar Performance Patterns Across Alignment Methods The three evaluated alignment methods exhibit highly similar performance patterns. While the adversarial approach shows an advantage for unseen target languages ([Figure 6.5](#)), the improvement is insufficient to overcome the severe forgetting in this category. We argue this difference is of limited practical relevance, as the degradation results from an inherent limitation in the token-based language control that none of the methods fully resolve. Moreover, the instruction-based language control already demonstrates superior baseline performance in this setting (§6.4.2). Therefore, given their comparable overall effectiveness, we select a single representative alignment method for the subsequent analysis of instruction-following models.

6.5.3 Instruction-Following Models

For instruction-following models, following the configuration in [Chapter 5](#), we proceed with the contrastive approach due to its generality, as the other two approaches (adversarial language classification and similarity regularization) require joint training with task-specific loss to avoid collapse. In [Figure 6.6](#), results are shown for both Llama3-8B and Qwen2.5-7B with LoRA fine-tuning on both the SMALL and LARGE data configurations.

Impact on Unseen Pairs The results suggest that the effects of cross-lingual alignment in preventing forgetting are conditional on the fine-tuning intensity. When plain fine-tuning

6.5. Cross-Lingual Alignment for Forgetting Mitigation

on Llama leads to a 6.6 COMET loss (63.8→57.2) on unseen pairs, cross-lingual alignment converts forgetting into performance gain, reaching 69.1 COMET. This improvement is consistent across 17 of 20 translation directions, with the remaining three showing only minor degradation of 0.4 to 0.9 COMET. However, the gain is not as large as the 70.9 achieved by in-language instructions (Table 6.4). This suggests that when the practical setup allows for incorporating in-language instructions, that strategy is a more effective alternative to cross-lingual alignment.

Impact on Unseen Source and Target Languages On unseen source languages, cross-lingual alignment generally leads to performance comparable to standard LoRA fine-tuning, in line with previous observations on task-specific models (§6.5.2). On unseen target languages, cross-lingual alignment provides a modest gain of 0.7 COMET (79.6→80.3) for Llama, whereas it offers no significant improvement for the Qwen model. These results suggest that the primary advantage of cross-lingual alignment is its ability to reverse forgetting on unseen language pairs. In conditions where standard fine-tuning already yields improvements, the additional gains from alignment are much milder. This forgetting pattern here, especially in the unseen target category, differs from those observed on translation-specific models in §6.5.2. The persistent strong forgetting observed previously is substantially reduced, with alignment occasionally surpassing the performance of standard fine-tuning. This suggests that as models move forward in their instruction-following capabilities, their potential for cross-lingual transfer is also enhanced.

6.5.4 Why Gains Concentrate on Unseen Pairs

The most significant performance improvements are observed on unseen pairs, where both the source and target languages were included in the training data but never appeared together. As discussed in §6.3.2, this evaluation scenario is particularly challenging because fine-tuning can cause the model to overfit to spurious source-target associations, leading the model to generate outputs in an incorrect target language.

We interpret these results as evidence that cross-lingual alignment methods directly counteract this degradation. Encouraging more language-invariant representations leads to disentangling semantic content from language-specific features. By breaking the spurious associations learned during training, alignment mitigates the effects of forgetting and restores the model’s ability to generate the correct target language. Consequently, the performance gains are most substantial on these unseen pairs. Considering that the number of translation directions in a multilingual system scales quadratically, and that many languages may only have parallel data to English, breaking the spurious correlations that affect unseen pairs is of high practical importance for scalable translation models.

In contrast, for translation directions involving entirely unseen languages, the central challenge is a general lack of exposure rather than spurious correlations. Therefore, the impact of this alignment mechanism is much milder in those scenarios.

6.6 Summary

In this chapter, we aim to resolve conflicting observations, both in our previous results from [Chapter 4](#) and [Chapter 5](#), as well as the broader literature, regarding the conditions that cause catastrophic forgetting during multilingual fine-tuning for machine translation. With a series of controlled experiments, we can now reconcile these findings.

Our analysis reveals that the primary determinants of catastrophic forgetting are the base model’s scale and the volume of the fine-tuning data. Larger models consistently demonstrated greater resilience to forgetting, while larger fine-tuning datasets tend to wash away pre-trained knowledge. Furthermore, the instruction-following capability of the base model is a critical factor. This ability appears to provide a mechanism that helps the model generalize its translation skills to unseen language pairs. In contrast, other factors often thought to be relevant play a less critical role. First, the choice between full fine-tuning and parameter-efficient fine-tuning (LoRA) did not substantially alter the degree of catastrophic forgetting. Moreover, the underlying model architecture, i.e., whether an encoder-decoder like M2M-124 or a decoder-only LLM, was not a primary driver of forgetting.

Based on our findings, we provide the following practical recommendations: **1)** Consider the relative scale between model size and fine-tuning data. Larger datasets may require larger base models to prevent forgetting. **2)** Prioritize models with strong instruction-following abilities over specific architectural choices. **3)** Do not rely solely on parameter-efficient fine-tuning methods as a forgetting mitigation strategy. **4)** For models exhibiting forgetting, cross-lingual alignment such as the middle-layer approach from [Chapter 5](#) is promising for unseen pairs where both source and target languages have been separately seen in fine-tuning. For instruction-following models, we recommend first training with in-language instructions before cross-lingual alignment approaches. This initial data-oriented approach is preferable due to its simplicity and its potential to achieve equivalent or superior performance.

This chapter and the two previous ones form a continuous line of investigation into methods for fostering and preserving generalizable representations within multilingual pre-trained models. This chapter concludes this inquiry with systematic analyses that reconcile some seemingly inconsistent observations from the previous chapters. Building upon this foundation, the subsequent chapter transitions from the preservation of multilingual knowledge to its application. It explores how the transferability of multilingual representations can be leveraged to achieve fine-grained stylistic control over model outputs at inference time.

Chapter 7

Inference-Time Transfer

As pre-trained models become more advanced, they are also growing in size and becoming less transparent. Many state-of-the-art models are deployed behind restricted interfaces that render their parameters hidden and immutable. This creates a “grey box” scenario, where we may observe internal activations but cannot access or update the model’s weights due to restricted access or computation constraints. In these scenarios, adaptation strategies that depend on further training model parameters are no longer feasible.

In contrast to the previous chapters’ focus on updating model parameters, this chapter investigates techniques applied during the inference stage. This shift is accompanied by a more fine-grained objective. Rather than evaluating general output quality, we aim to control specific categorical attributes of the translation, such as formality. The primary goal is to enable the cross-lingual transfer of this control capability at inference time, without relying on model fine-tuning.

We use a classifier’s gradient to steer a pre-trained multilingual model that does not natively support attribute-controlled translation. We then systematically compare this inference-time approach to traditional training-based methods to identify the trade-offs. Our findings indicate that while training-based methods are dominant in supervised settings, the performance gap diminishes in zero-shot scenarios. The inference-time method is particularly effective for new and linguistically distant target languages and demonstrates greater robustness across different domains. This chapter is partly based on [Liu and Niehues \(2024a\)](#).

7.1 Introduction

As machine translation quality improves, research interest is expanding from general semantic accuracy to more fine-grained attributes of the output. This chapter moves beyond general quality to the contextual appropriateness of translations. While often subtle, attributes like formality and grammatical gender significantly influence the perceived quality of translations. For instance, an incorrect formality level can make an email seem unprofessional, and errors in grammatical gender may appear confusing or even offensive. This use case is particularly challenging when such attributes are not explicitly encoded in the source text, for example, when translating from a language like English which lacks grammatical gender and explicit formality levels.

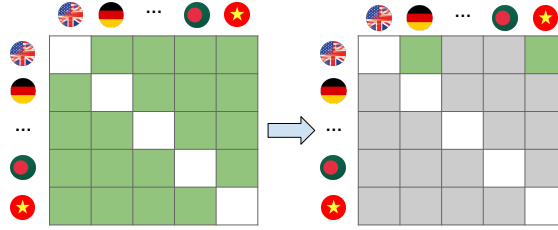


FIGURE 7.1: The number of translation directions with attribute-annotated data (right) is far less than that of what massively pre-trained models serve (left). Vertical and horizontal axes indicate source and target languages respectively.

The standard approach to instilling attribute control in generic translation models is to train them on explicitly annotated data. However, acquiring this data is a substantial bottleneck, as it requires language-specific knowledge and manual curation. This challenge is magnified in the context of massively multilingual models, where the data acquisition effort quickly becomes impractical. As the number of required language pairs grows quadratically with the number of supported languages, data acquisition becomes a constraint that supervised methods struggle to overcome. For example, a model supporting 100 languages would theoretically need annotated data for 9,900 language pairs to control a single attribute like formality across all of them, as illustrated in Figure 7.1.

While prior work (Michel and Neubig, 2018; Saunders et al., 2020; Nadejde et al., 2022) demonstrated success by fine-tuning on limited annotated data, the cross-lingual transferability of these attribute control mechanisms remains largely unexplored. To extend fine-grained customization to languages that lack supervised data, a systematic investigation into the transferability of attribute controllers is essential.

Fine-tuning on attribute-specific data is a straightforward method to achieve attribute control, and studies have shown that even a few hundred attribute-specific sentences can be sufficient for doing so (Rippeth et al., 2022; Wu et al., 2023). However, this approach risks overfitting and catastrophic forgetting (Freitag and Al-Onaizan, 2016; Thompson et al., 2019). As investigated in Chapter 6, this risk is particularly relevant in multilingual contexts, where fine-tuning on a subset of languages can erase pre-trained knowledge of others (Garcia et al., 2021; Cooper Stickland et al., 2021; Liu and Niehues, 2022). Even when the model does not catastrophically forget certain languages, translation quality may still degrade if a domain mismatch exists between the fine-tuning data and the test domain. These challenges will be empirically validated in this chapter across various zero-shot conditions, language pairs, and domains.

An alternative paradigm is inference-time customization, which keeps the pre-trained model’s parameters fixed. Instead, the generation process is steered toward desired attributes at inference, for example, by re-weighting the output vocabulary distribution (Saboo and Baumann, 2019; Yang and Klein, 2021; Landsman et al., 2022) or editing model activations (Dathathri et al., 2020). For cross-lingual transfer to be effective, the control mechanism must operate on features that are shared across languages. This requirement makes methods that rely on surface-level vocabulary unsuitable, as vocabularies are language-specific. Therefore, this chapter focuses on activation-steering methods, which

7.2. Related Work

modify internal model representations that are more likely to be shared across languages.

Given these distinct paradigms, it is essential to understand the trade-off between fine-tuning and inference-time control methods. The main objective of this chapter, therefore, is to systematically evaluate their strengths and weaknesses in cross-lingual transfer, and to explore potential synergies.

7.1.1 Task Definition

In this chapter, we study the following task:

Given a pre-trained multilingual translation model supporting $N(N - 1)$ translation directions across N languages, and attribute-annotated data for k of the $N(N - 1)$ translation directions (where $k \ll N(N - 1)$), how can we enable attribute-controlled translation for language pairs beyond this small supervised set?

This problem definition addresses both classical cross-lingual and temporal generalization. The cross-lingual aspect concerns the ability to apply attribute control to language pairs beyond the supervised set. The temporal dimension measures how well the model preserves its original translation performance, thereby evaluating the extent of catastrophic forgetting after the control mechanism is introduced.

7.1.2 Contributions

In this chapter, we investigate the cross-lingual transferability of attribute-control mechanisms for multilingual translation. We conduct a comprehensive analysis comparing traditional fine-tuning with a training-free, inference-time control method based on activation steering, as shown in [Figure 7.2](#). Our primary contributions are:

- We propose an inference-time technique that enables cross-lingual control over specific translation attributes, such as formality and grammatical gender, within pre-trained multilingual machine translation models.
- We demonstrate that inference-time control becomes increasingly competitive with fine-tuning-based approaches when transferring to distant languages and out-of-domain conditions. While fine-tuning dominates in supervised settings, classifier guidance shows stronger domain robustness and closes the performance gap in challenging zero-shot scenarios.
- We reveal a trade-off between control strength and overall translation quality. Our findings show that while stronger guidance improves attribute accuracy, it can negatively impact translation quality.

7.2 Related Work

This chapter is situated at the intersection of several broader research areas, namely controllable generation ([§7.2.1](#)), domain adaptation ([§7.2.2](#)), and activation steering ([§7.2.3](#)).

7.2.1 Controllable Generation

The task studied in this chapter, attribute-controlled translation, is a specific instance of controllable text generation. While controllable generation is a well-established research area with steady progress (Keskar et al., 2019; Krause et al., 2021; Yang and Klein, 2021; Liu et al., 2021a), how well control mechanisms generalize across languages has been less explored.

Conventional machine translation models have investigated control over a wide array of attributes, including politeness (Sennrich et al., 2016a; Niu et al., 2018; Feely et al., 2019), grammatical gender (Vanmassenhove et al., 2018; Saunders et al., 2020), length (Takase and Okazaki, 2019; Lakew et al., 2019a; Marchisio et al., 2019; Niehues, 2020), and general style (Michel and Neubig, 2018; Schioppa et al., 2021; Vincent et al., 2023; Wang et al., 2023b). The predominant paradigm in these works is supervised, relying on the availability of attribute-annotated data. This limits their applicability to new languages where such labeled resources are scarce or non-existent. A common strategy to overcome data scarcity is the use of synthetic data by pseudo-labeling (Rippeth et al., 2022; Lee et al., 2023), using a classifier to filter for supervised data per attribute. Our work departs from these assumptions. Building upon massively multilingual translation models, we do not assume the scalability of creating synthetic data for all languages served by the backend model. Furthermore, we do not assume a classifier that can a priori distinguish attribute classes for zero-shot languages without attribute-annotated data.

The rise of LLMs has re-framed attribute-controlled translation as an instruction-following task (Raunak et al., 2024), where the desired translation attribute is explicitly specified in the prompt. This has led to remarkable results in zero- and few-shot control (Sarti et al., 2023; Garcia et al., 2023). In cross-lingual scenarios, multilingual LLMs can be prompted with examples from different languages (Sarti et al., 2023). Despite these advances, specialized translation models remain crucial, especially for low-resource languages. At the time of writing, dedicated models often demonstrate stronger performance in low-resource languages than general-purpose LLMs (Zhu et al., 2023b; Sarti et al., 2023; Robinson et al., 2023), underscoring the continued importance of studying them.

7.2.2 Multilingual Domain Adaptation

Attribute-controlled translation can be viewed as a lightweight domain adaptation task. Prior works on adapting pre-trained multilingual models have reported catastrophic forgetting of languages not included during fine-tuning, resulting in loss of the ability to generate these languages (Cooper Stickland et al., 2021; Vu et al., 2022a). However, our experiments on fine-tuning for zero-shot attribute control (§7.6.1) present a different picture, showing no observable forgetting of language generation abilities. We attribute this difference to two potential factors. First, our fine-tuning datasets are relatively small. This is consistent with our findings in Chapter 6 that limited training data does not overwrite pre-trained knowledge. Second, attribute control is a subtler form of adaptation compared to domain shifts with specialized vocabularies and writing styles

7.3. Comparing Transferring Attribute Control Capabilities

(e.g., medical or legal domains). The lower fine-tuning intensity needed for attribute control consequently reduces the risk of catastrophic forgetting.

7.2.3 Activation Steering

Activation steering methods guide model behavior by manipulating their internal activations during the generation process. A prominent approach is classifier-guided steering, as proposed by [Dathathri et al. \(2020\)](#) with the Plug and Play Language Model (PPLM). In this method, a classifier is trained to distinguish specific labels. During generation, the model’s activations are iteratively updated to steer outputs toward the desired class. In diffusion models, classifier guidance is used for controllable image synthesis ([Dhariwal and Nichol, 2021](#)) and controlled text generation ([Li et al., 2022](#)).

Another category of activation steering is contrastive steering, which guides model representations using contrastive pairs. These pairs typically contrast desired versus undesired output properties ([Lai et al., 2024a](#)), such as factual versus hallucinatory ([Li et al., 2023a](#); [Rimsky et al., 2024](#)), biased versus non-biased ([Li et al., 2025b](#)), or English versus non-English ([Wang et al., 2025](#)). By steering the model’s internal representations to reduce the difference between the positive and negative pairs, contrastive steering encourages the model to exhibit desired properties during generation.

In this chapter, we adopt a classifier-guided approach due to its interpretability ([Laguna et al., 2024](#)). The attribute classifier provides explicit probability scores that support clearer analysis of steering effectiveness compared to representation-based approaches.

7.3 Comparing Transferring Attribute Control Capabilities

This section discusses two techniques for the cross-lingual transfer of attribute control, comparing an inference-time approach without updating model weights (§7.3.1) with a conventional fine-tuning baseline (§7.3.2).

7.3.1 Inference-Time Control

We build on the established finding that the internal representations (activations) of pre-trained multilingual models capture cross-lingual commonalities ([Pires et al., 2019](#); [Liu et al., 2020](#)). While previous chapters demonstrated that these representations are not fully language-agnostic, models with multilingual pre-training did show increased levels of representational sharing compared to those without this procedure (§4.3.2). By training an attribute classifier on these shared activations, this method aims to achieve cross-lingual transfer, enabling the classifier to be used at inference time to steer generation for languages that lack attribute-annotated data. To leverage this shared representational space, the control mechanism must operate on model activations rather than on model weights. This method involves two stages. First, an attribute classifier is trained using the frozen pre-trained model’s internal representations. Then, at inference time, this classifier guides the manipulation of the model’s activations to steer generation towards the desired

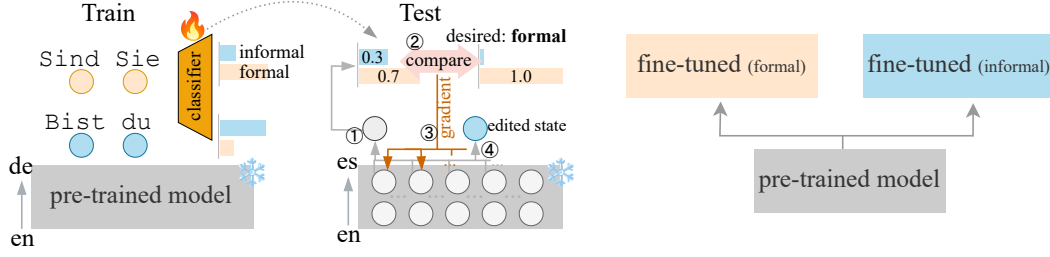


FIGURE 7.2: Left: **Inference**-time control by gradient-based classifier guidance: training classifiers for attributes on decoder activations, and using its predictions to edit inference-time model *activations* towards desired attributes. Right: Standard **training**-time control by fine-tuning on attribute-specific data.

attribute. To the best of our knowledge, no prior work has explored this activation steering technique for the purpose of cross-lingual transfer.

Specifically, we extend the approach by Dathathri et al. (2020) to encoder-decoder models. For machine translation, given a frozen pre-trained model, we run forward passes with attribute-annotated¹ parallel data $(\mathbf{X}, \mathbf{Y})^c$ for $c \in [C]$, where $\mathbf{X}$ and $\mathbf{Y}$ are the source and target sentences with individual sentence pairs $(\mathbf{x}, \mathbf{y})^i \in (\mathbf{X}, \mathbf{Y})$, and C is the number of attribute labels.

While freezing the translation model’s parameters, we train a classifier that maximizes $P(c \mid \mathbf{h})$, where c is the ground-truth attribute label and $\mathbf{h}$ is the last decoder layer’s hidden states after forced-decoding parallel data $(\mathbf{x}, \mathbf{y})$:

$$\mathbf{h} = \text{decoder}(\mathbf{y}, \text{encoder}(\mathbf{x})). \quad (7.1)$$

Like with a standard model, the output distribution is then $\text{softmax}(\mathbf{W}\mathbf{h})$, where $\mathbf{W}$ maps the hidden states $\mathbf{h}$ to the vocabulary distribution.

At inference time step t , the hidden state is:

$$\mathbf{h}_t = \text{decoder}(y_{t-1}, \mathbf{A}_{t-1}), \quad (7.2)$$

where y_{t-1} is the token from the previous step, and $\mathbf{A}_{t-1}$ is the model activations. $\mathbf{A}_{t-1}$ contains activation key-value pairs² from the decoder self-attention and cross-attention for steps 1 to $t - 1$. Based on all available decoder states up until $t - 1$, we predict an attribute label: $\arg\max_c P(c \mid \mathbf{h}_{1,\dots,t-1})$ (step ① in Figure 7.2). We meanpool the states from timestep 1 to $t - 1$ for the prediction as it empirically showed better performance than 1) using a token-level classifier without pooling and 2) operating on the cumulative sum of hidden states from all time steps so far.³

As $\mathbf{h}_{1,\dots,t-1}$ is only determined by $\mathbf{A}_{t-1}$, we can rewrite $P(c \mid \mathbf{h}_{1,\dots,t-1})$ as $P(c \mid \mathbf{A}_{t-1})$. Comparing the prediction to the desired attribute c^* (step ② in Figure 7.2), we can derive gradients measuring how much the current activations satisfy the desired c^* (step ③

¹Only the target side needs attribute labels.

²Note these are not the key/value projection weights, but the activations after applying these projections.

³In initial experiments training an English-German formality classifier, the accuracy on the dev set was 86.7% (meanpool), 66.1% (token-level) and 73.0% (cumulative sum).

7.4. Experimental Setup

in Figure 7.2). The gradients, $\nabla_{\mathbf{A}_{t-1}} P(c^* | \mathbf{A}_{t-1})$, are then back-propagated for several iterations with given step sizes, resulting in updated activations $\mathbf{A}'_{t-1}$, which further leads to modified decoder hidden state:

$$\mathbf{h}'_t = \text{decoder}(y_{t-1}, \mathbf{A}'_{t-1}). \quad (7.3)$$

A new output token y_t that more likely satisfies the control is generated from $\mathbf{h}'_t$ by $\text{softmax}(\mathbf{W}\mathbf{h}'_t)$ (step ④ in Figure 7.2).

7.3.2 Fine-Tuning-Based Control

A more conventional method for attribute control involves fine-tuning the pre-trained model on attribute-specific parallel data. For this approach to successfully transfer to language pairs without annotated data, the adaptation process must learn the general features of the desired attributes rather than overfit to the specific languages present in the fine-tuning data. This is crucial to prevent catastrophic forgetting of the model’s capabilities in other languages.

We fine-tune the entire model on the data for each attribute, resulting in one specialized model per attribute, as illustrated in Figure 7.2. We note that simpler methods, such as prepending attribute tags to the source sentences (Chu et al., 2017; Kobus et al., 2017), were insufficient to make the model attribute-aware, likely because the limited fine-tuning data could not prevent the model from treating the tags as noise.

While parameter-efficient fine-tuning (PEFT) (Bapna and Firat, 2019; Philip et al., 2020; Hu et al., 2022) is a more computation-efficient alternative, we do not explore it in this chapter for two reasons: our primary focus is on the transferability of the control mechanism itself, and, as demonstrated in the previous chapter, the choice of PEFT versus full fine-tuning does not fundamentally alter the catastrophic forgetting behavior.

7.4 Experimental Setup

We experiment on two attribute control tasks: formality and grammatical gender control. The experiments cover the following languages: German (de), Spanish (es), French (fr), Hindi (hi), Italian (it), Korean (ko), Portuguese (pt), Russian (ru). As outlined in Table 7.1, the training data has English as input. The low data volume not only reflects the practical challenge of data scarcity, but is also an established condition in existing benchmarks (Nadejde et al., 2022).

7.4.1 Formality Control

The training data is sourced from the CoCoA-MT dataset (Nadejde et al., 2022), which features an overlap between its training and test domains.⁴ For zero-shot conditions,

⁴We excluded Japanese due to the pre-trained model’s low translation accuracy (<40%) on formality-annotated words, in contrast to scores exceeding 60% for the other five languages.

Task	Translation Directions	# Sentences per language per attribute
Formality control (formal vs. informal)		
Train	en→{de, es, fr, hi, it}	400
Test (supervised)	en→{de, es, fr, hi, it}	600
Test (new target)	en→{pt, ru, ko}	600
Test (new source)	{de, fr, hi, it}→es	366-572
Grammatical gender control (feminine vs. masculine)		
Train	en→es	194
Test (supervised)	en→es	552-556
Test (new target)	en→{it, fr}	515-546
Test (new source and target)	{es, fr}→it, {es, it}→fr	271-365

TABLE 7.1: Overview of training and test data, where the last column shows data amount for *each attribute*. The number of sentences may vary per language pair, a property inherited from the original datasets.

we transfer controllers trained on different language pairs to new translation directions. Specifically, we investigate the following two cases:

Transfer to New Target Languages To assess the impact of multilingual training data on transferability, we compare attribute controllers trained under two settings:

- **Single-direction:** Controllers are trained on individual language pairs, specifically en→es and en→de, representing Romance and Germanic languages.
- **Multilingual:** A single controller is trained on all available source languages simultaneously: en→de, es, fr, hi, it.

The transfer capabilities of these controllers are evaluated on three new target languages from the IWSLT 2023 formality control shared task (Agarwal et al., 2023). These language pairs were selected to represent varying degrees of linguistic distance from the training languages: 1) Close: en→pt; 2) Related: en→ru; 3) Distant: en→ko.

Finally, the 400 sentences of supervised data available for the en→ko direction are used to establish an oracle performance baseline. This baseline represents an upper bound for control accuracy in a supervised setting.

Transfer to New Source Languages To evaluate the transfer of control to new source languages, we constructed a new test set by realigning the CoCoA-MT data. The original test sets only provide English source sentences. By matching sentences from different languages that share a common English translation, we create new test pairs with non-English source and target languages. In this configuration, the source sentences themselves contain formality information, a key distinction from the original setup. This allows us to assess the model’s ability to not only preserve the source sentence’s formality level but also to alter it when explicitly steered by the controller.

Due to the lack of suitable training data, with formality control we only consider same domain for training and testing conditions. Domain shifts are investigated next in grammatical gender control.

7.4. Experimental Setup

7.4.2 Grammatical Gender Control

For the formality control setup above, the data for training the attribute controller come from the same domain as the test set. To evaluate domain generalization, for grammatical gender control, we train the controller on texts with very different styles from the test data. For *training* the attribute controller, we use the en-es set from [Saunders et al. \(2020\)](#)⁵ with artificial sentences of very simple grammatical structure up to 7 words. In contrast, for the *test* set we use MuST-SHE ([Bentivogli et al., 2020](#)), which consists of TED talks with much longer sentences and more versatile styles. More dataset details are in Appendix D.1. Besides transfer to new target languages like previously (§7.4.1), we also explore the following setting:

Transfer to New Source & Target Languages The MuST-SHE test set comes in en-{es, fr, it}. Like previously, we re-align them using English as pivot, creating non-English source and target sentences. In this case, both the source and target sentences have the same gender. As the attribute training data is in en→es, we evaluate {es, fr}→it and {es, it}→fr for the transfer to new translation directions where both the source and target languages differ from training.

7.4.3 Models and Evaluation

Models We use two types of backend models. For the main experiments, we use the pre-trained NLLB-200 distilled 600M model ([NLLB Team, 2024](#)), which covers 200 languages for many-to-many translation. We also train a Transformer-base ([Vaswani et al., 2017](#)) from scratch to verify if observed phenomena are specific to models with massive multilingual pre-training. The Transformer-base model covers all languages in our experiments and is trained on OPUS-100 ([Zhang et al., 2020a](#)). Details of these data, as well as the training and inference details, are presented in Appendix D.1 and D.2. When reporting the results, we use “CG” to denote systems that steer the output by the classifier guidance approach in §7.3.1, and “FT” for fine-tuning as described in §7.3.2.

Control Evaluation For *formality* control, we report matched accuracy (M-Acc; %) following [Nadejde et al. \(2022\)](#). For *gender* control, we use the official evaluation script ([Bentivogli et al., 2020](#)) for accuracy (%). For formality, as the test set is the same for both formalities, the baseline M-Acc for the two formality labels add up to 100%. This is not the case for gender control since the attribute-specific sets are not parallel.

Quality Evaluation We use COMET ([Rei et al., 2020](#)) as the main translation quality metric, and additionally report BLEU ([Papineni et al., 2002](#)) to compare to prior works. Note that BLEU is impacted by n -gram matches on the correct formality or gendered words, while COMET is less susceptible to the artifact. For COMET score comparisons, we run paired T-tests and bootstrap resampling using comet-compare. We use “*” or “+” to mark systems better or worse than the base pre-trained model at significance level $\alpha = 0.05$.

⁵<https://github.com/DCSaunders/tagged-gender-coref#adaptation-sets>

Human Evaluation To test the transfer to real low-resource languages, we conduct a human evaluation on Bengali, which was marked as low-resource in the NLLB-200 training data (NLLB Team, 2024). Details on the evaluation are in Appendix D.3.

Baselines Few existing works experimented on the same data conditions as ours. An exception is the “mBART-large Gold Fine-tuned” model by Rippeth et al. (2022), who fine-tuned mBART (Liu et al., 2020) on parts of CoCoA-MT (Nadejde et al., 2022) for formality control. Their results overlap with our supervised results on $\text{en} \rightarrow \{\text{de}, \text{es}, \text{hi}\}$ and zero-shot results on $\text{en} \rightarrow \text{ru}$. Other than this, the majority of prior works used *more relaxed* data conditions than ours, e.g., using an existing attribute classifier that covers *zero-shot* languages for pseudo-labeling (Lee et al., 2023) or hypothesis reranking (Wu et al., 2023). We report these results in Appendix D.4. Overall, our model’s performance is comparable to the leading systems.

7.5 Supervised Conditions

Table 7.2 and Table 7.3 show the results for formality and grammatical gender control using supervised controllers with the NLLB-200 model. The results indicate that both fine-tuning (FT) and classifier guidance (CG) are able to steer the output towards the desired attributes while either maintaining the original translation quality or incurring only a slight degradation. It is important to note that the overall translation quality does not increase. This is expected, as the pre-trained model is already capable of these language pairs, and the limited volume of training data is sufficient only to instill attribute control capabilities, not to further specialize the model’s translation capabilities as in Chapter 4 and Chapter 6.

Fine-tuning more effective than classifier guidance in supervised conditions: A comparison of scores in Table 7.2 and Table 7.3 clearly shows FT is more effective than CG. For formality control, FT consistently scores nearly 100% M-Acc. It also substantially improves the quality scores due to adapting towards the specific domain of the attribute-annotated data, which is the same as the test domain in this case. On the other hand for CG, while it also improves the formality accuracy, the scores lag behind fine-tuning in both accuracy and quality. The gap is especially prominent on hi and it, where the underlying NLLB model has a strong bias towards a single formality: the accuracy for the rare formality is nearly zero (3.3% and 3.2% respectively). This is likely to do with NLLB’s training data, which might be skewed towards one single formality for some languages. In this case, CG can only partly recover the ability to generate translation in the formality NLLB is unfamiliar with. These results indicate that CG is only effective when the underlying model does not suffer from an absolute bias towards one attribute.

Classifier guidance more robust to domain mismatch: As motivated in §7.4.2, the gender control results in Table 7.3 allow us to assess the impact of domain mismatch between the controller training data and the test data, a very realistic scenario in practice. Here, while fine-tuning achieves higher accuracy for gendered words, it also degrades translation quality by 0.9 COMET. This provides further evidence that the previously improved

7.6. Zero-Shot Conditions

	Model	F _{formal}	I _{informal}	Avg.	BLEU	COMET
en→de	base	45.6	54.4	—	35.7±1.0	82.1
	+CG	95.0	89.6	92.3	38.4±1.1	81.6†
	+FT	100.0	100.0	100.0	43.6±1.2	83.8*
	Rippeth et al.	93.6	77.4	85.5	37.4	—
en→es	base	29.7	70.3	—	40.0±1.1	83.9
	+CG	72.9	92.4	82.7	41.2±1.2	84.4*
	+FT	100.0	95.9	98.0	46.0±1.2	85.5*
	Rippeth et al.	96.7	82.7	89.7	38.3	—
en→fr	base	76.8	23.2	—	36.0±1.1	80.8
	+CG	99.8	77.2	88.5	38.8±1.2	80.9
	+FT	100.0	99.3	99.7	43.0 ±1.1	83.0*
en→hi	base	96.7	3.3	—	24.0±0.9	75.5
	+CG	99.3	30.7	65.0	24.3±0.9	75.0†
	+FT	99.6	99.2	99.4	36.4±1.0	81.7*
	Rippeth et al.	98.5	64.7	81.6	28.7	—
en→it	base	3.2	96.8	—	41.3±1.1	84.9
	+CG	18.7	99.5	59.1	40.6±1.1	84.1†
	+FT	98.6	99.3	99.0	49.6±1.1	86.0*

TABLE 7.2: Formality control results in *supervised* condition (controllers trained on formality-annotated data), with base (pre-trained out of the box), CG (steering by classifier guidance as described in §7.3.1), FT (fine-tuning as described in §7.3.2)

	Model	F _{feminine}	M _{masculine}	Global	BLEU	COMET
en→es	base	58.8	86.7	73.6	45.0±1.2	84.9
	+CG	75.0	89.7	82.8	44.7±1.2	84.7
	+FT	90.2	89.7	86.9	43.7±1.2	84.0†

TABLE 7.3: Grammatical gender control results in *supervised* condition (cross-domain: controller trained on gender-annotated data from a different domain).

COMET scores (Table 7.2) are results of fine-tuning on in-domain data. In contrast, the translation quality with CG does not significantly differ from NLLB by the T-tests, suggesting its stronger domain robustness. We hypothesize it is because CG operates on the last decoder layer’s hidden states, which are just one projection away from the output vocabulary. These representations likely contain more word-level than domain information, which is precisely needed in the task of attribute control.

7.6 Zero-Shot Conditions

Having established the supervised performance, we now turn to the central question of this chapter to evaluate the zero-shot transfer of attribute control. We evaluate three scenarios: transfer to new target languages (§7.6.1), new source languages and previously unseen source-target language pairs (§7.6.2).

7.6.1 New Target Languages

Now we transfer the trained controllers to target languages unseen when training the controllers, i.e., those without attribute annotation. In Table 7.4 and Table 7.5, we report the results on formality and gender control respectively. In Table 7.4, we also compare the single-direction and multilingual controllers as motivated in §7.4.1.

		Pre-trained NLLB					Transformer-Base				
Model		F	I	Avg.	BLEU	COMET	F	I	Avg.	BLEU	COMET
en→pt	base	47.7	52.3	—	41.7±1.1	85.1	35.8	64.2	—	38.7±1.1	82.2
	+CG (de)	75.6	74.2	74.9	43.0±1.1	84.9	50.0	72.8	61.4	38.7±1.1	81.8†
	+FT (de)	99.0	45.5	72.3	40.4±1.0	85.3	79.2	71.2	75.2	39.6±1.1	82.7*
	+CG (es)	85.4	83.6	<u>84.5</u>	43.8±1.0	85.0	53.3	79.8	66.6	38.8±1.1	81.7†
	+FT (es)	99.8	28.7	64.3	40.3±1.0	85.2	93.9	80.1	87.0	40.5±1.0	82.5*
	+CG (multi)	84.8	80.0	82.4	43.7±1.1	84.9	55.9	80.8	68.4	39.0±1.1	81.8†
	+FT (multi)	99.5	51.0	75.3	42.3±1.0	85.9*	95.8	81.9	<u>88.9</u>	41.4±1.0	83.1*
	+CG +FT (multi)	100.0	83.2	91.6	42.1±1.0	85.7*	97.8	93.7	95.8	41.0±1.0	82.4
en→ru	base	55.0	45.0	—	30.3±1.0	83.7	43.9	56.1	—	24.2±1.0	75.9
	+CG (de)	87.3	77.7	82.5	32.2±1.0	83.1	67.2	71.8	69.5	24.6±0.9	75.0†
	+FT (de)	99.5	84.7	<u>92.1</u>	33.0±1.1	84.2*	84.0	69.3	76.7	25.0±1.0	75.8
	+CG (es)	86.8	73.9	80.4	32.4±1.0	83.2	61.7	76.8	69.5	24.8±1.0	75.0†
	+FT (es)	98.3	60.6	79.5	32.8±1.1	84.1*	83.5	68.6	76.1	26.1±1.0	76.6*
	+CG (multi)	87.3	78.2	82.8	32.2±1.0	83.2	72.2	80.9	76.6	25.0±1.0	75.0†
	+FT (multi)	99.8	79.6	89.7	33.0±1.1	84.2*	87.5	69.8	<u>78.7</u>	25.9±1.0	77.0*
	+CG +FT (multi)	100.0	93.0	96.5	33.1±1.0	84.4*	96.2	91.3	93.8	26.2±1.0	76.2
	<u>Rippeth et al.</u>	100.0	13.8	56.9	23.5	—	—	—	—	—	—
en→ko	base	50.9	49.1	—	15.7±0.7	82.6	32.0	68.0	—	10.6±0.6	74.0
	+CG (de)	67.0	64.6	<u>65.8</u>	15.7±0.7	82.1†	45.2	78.2	61.7	10.4±0.6	73.4†
	+FT (de)	67.8	54.2	61.0	12.8±0.6	84.1*	42.7	66.4	54.6	10.7±0.6	74.0
	+CG (es)	68.9	61.6	65.3	15.1±0.8	82.1†	46.3	77.6	62.0	10.7±0.6	74.1
	+FT (es)	64.4	47.3	55.9	14.0±0.7	84.4*	47.4	62.7	55.1	11.7±0.6	75.2*
	+CG (multi)	67.0	61.7	64.4	15.5±0.8	82.2	46.0	78.1	<u>62.1</u>	10.6±0.6	74.1
	+FT (multi)	68.5	46.2	57.4	13.4±0.7	84.7*	48.3	68.4	<u>58.4</u>	11.0±0.6	74.4
	+CG +FT (multi)	70.0	63.5	66.8	13.2±0.7	84.2*	58.9	81.8	70.4	10.8±0.6	73.4†
	+oracle CG (ko)	70.3	62.6	66.5	15.2±0.7	81.7†	58.9	82.3	70.6	11.2±0.6	74.5*
	+oracle FT (ko)	79.4	93.5	86.5	22.2±0.9	86.2*	86.7	97.9	92.3	19.1±0.9	74.0*

TABLE 7.4: Zero-shot formality control results. **Best** and second best results under the same data condition are marked.

Gap between fine-tuning and classifier guidance shrinks in zero-shot conditions: While fine-tuning was consistently leading in supervised conditions (§7.5), now under zero-shot conditions with unseen target languages, the gap shrinks. For formality control, on Korean, the most distant language, CG consistently achieves stronger control results than fine-tuning, indicating more robustness when transferring to unfamiliar settings. Overall in Table 7.4, for the main experiments on NLLB-200, CG outperforms FT in 7 of the 9 pairwise comparisons ($\{\text{de, es, multi}\} \times 3$ target languages). With gender control results in Table 7.5, fine-tuning achieves stronger control accuracy (avg. +4.9% abs.) but degrades translation quality (−0.6 COMET) due to domain mismatch. On the other hand,

7.6. Zero-Shot Conditions

	Model	F _{feminine}	M _{masculine}	Global	BLEU	COMET
en→it	base	53.8	88.9	73.1	35.1±1.0	84.1
	+CG	72.3	92.8	83.6	35.4±1.1	83.7
	+FT	83.6	91.2	87.8	34.4±1.0	83.5+
	+CG +FT	88.6	94.5	91.8	33.4±1.0	82.6+
en→fr	base	55.3	88.4	72.4	38.3±1.3	82.6
	+CG	67.8	90.3	79.4	38.7±1.2	82.5
	+FT	78.9	90.8	85.0	38.2±1.2	82.0+
	+CG +FT	87.0	91.9	89.5	37.4±1.2	81.9+

TABLE 7.5: Zero-shot grammatical gender control results on *new target* languages with *domain mismatch*.

CG retains the translation quality. This confirms the previous finding (§7.5) on its stronger domain robustness.

Multilingual controllers help when the base model is not massively multilingual: In Table 7.4, controllers trained on multiple translation directions (multi) are compared to those trained on single directions (en→es or de). On Transformer-base, multi consistently outperforms its single-direction counterparts, regardless of whether the controller is fine-tuning- or CG-based. In contrast, for the pre-trained NLLB, there is no clear distinction between the multilingual systems and the rest. This indicates that NLLB does not further benefit from multilinguality in the controller training stage, likely because it already underwent a massively multilingual pre-training stage. This shows that massively multilingual models are a useful basis for attribute control especially when annotated resources are limited to single languages.

Classifier guidance is complementary with fine-tuning: When applying CG on top of the fine-tuned models, we see the *strongest* control accuracy for both formality and gender control. This observation is consistent whether the base model is the pre-trained NLLB or the normal Transformer-base. Compared to fine-tuning alone, the addition of CG also does not degrade translation quality on NLLB. On the more challenging case of gender control which involves domain mismatch, adding CG to fine-tuning does not impact translation quality on fr and causes a slight degradation on it. This is likely linked to poor hyperparameter choices in CG: due to time constraints we directly used the hyperparameters when applying CG alone, which are too strong for models already fine-tuned for attribute control. We are optimistic for improved scores under more fitting hyperparameters.

Fine-tuning did not erase knowledge on other languages: To our surprise and different from results in domain adaptation (Cooper Stickland et al., 2021; Vu et al., 2022a), fine-tuning did not erase the pre-trained model’s knowledge on the target languages absent in supervised fine-tuning, as reflected by the translation quality scores (Table 7.4, 7.5). This is not specific to NLLB, but also observed on the Transformer-base trained with random initialization on a few translation directions. Therefore, this phenomenon is not a result of massively multilingual pre-training, but more likely linked to the light fine-tuning strength with limited number of updates and small learning rates.

	Model	Quality (1-5)	Formality (1-3)	Win (%)	Win & Tie (%)
(1)	NLLB-200	4.25 $\pm$ 0.75	2.69 $\pm$ 0.46	—	—
(2)	CG (multi) formal	4.00 $\pm$ 0.79	2.63 $\pm$ 0.48	56.3	81.3
(3)	CG (multi) informal	4.44 $\pm$ 0.70	2.38 $\pm$ 0.69	62.5	93.8
(4)	FT (multi) formal	4.31 $\pm$ 0.85	2.63 $\pm$ 0.48	43.8	68.8
(5)	FT (multi) informal	4.13 $\pm$ 1.05	2.44 $\pm$ 0.49	62.5	93.8

TABLE 7.6: Human evaluation on Bengali, with quality on a 5-point scale $\uparrow$ and formality on a 3-point scale ($\uparrow$: formal) with standard deviations. Last two columns show pairwise comparison of formality scores to baseline NLLB-200 given the same source sentences (winning: scoring more in the direction of the desired formality).

Comparison to oracle data condition: In the bottom rows of Table 7.4, we report the oracle performance of using 400 sentences as supervised data for training the controllers. Our strongest zero-shot results match the performance of oracle CG, but still lag far behind the upper-bound of fine-tuning on in-domain data with attribute annotation (oracle FT). We believe this gap is magnified as Korean is not only linguistically distant from the languages used in training, it also differs in the notion of formality: Korean involves multiple levels of formality instead of a binary informal-formal distinction. For the zero-shot transfer, this means transferring a controller trained for binary control to a multi-class problem with an unknown class mapping, which is naturally more challenging.

Human Evaluation on Bengali: The human evaluation results are presented in Table 7.6. First, adding attribute control does not appear to impact translation quality. Second, pairwise comparisons with the baseline show both CG and fine-tuning are effective in formality control, where CG has slightly higher win ratio than FT against the baseline. Third, the impact on formality scores is more prominent when steering towards informal translation. This is likely because the baseline translations already have a high level of formality. Moreover, the rare usage of the lowest formality level in Bengali (Appendix D.3) could explain the relatively high formality scores for the systems steered towards “informal” (rows (3) and (5)).

7.6.2 New Source and Target Languages

New source languages easier than new target languages: In Table 7.7, we report the results of transferring controllers trained with English source to new source languages. Contrasting these scores with the target-side zero-shot results in Table 7.4, it is clear that transferring to new source languages is a much easier task. This is expected, as attribute-controlled translation primarily places lexical constraints on the target side. Once the controller can generate translations with the correct attribute, swapping the source language does not pose a large challenge. Even when the source formality disagrees with the desired output formality (uncolored columns in Table 7.7), the controllers are able to steer the translations toward the required attributes.

NLLB struggles to preserve source attributes: Contrasting the colored “base” cell in Table 7.7 with its uncolored counterpart, we see that NLLB does have some notion of

7.6. Zero-Shot Conditions

	Model	F _{formal}	I _{informal}	Avg.	BLEU	COMET
Source Formal	base	77.8	22.2	—	23.9±0.5	79.0
	X→de +CG	98.6	71.5	85.1	25.9±0.5	78.7
	+FT	100.0	100.0	100.0	30.1±0.7	80.7*
	base	57.8	42.2	—	29.8±0.5	82.7
	X→es +CG	86.7	73.3	80.0	30.5±0.8	82.3
	+FT	99.6	77.4	88.5	32.8±0.7	83.9*
	base	97.0	3.0	—	29.5±0.6	79.1
	X→fr +CG	99.8	40.4	70.1	30.6±0.6	78.9
	+FT	99.9	99.4	99.7	34.2±0.7	81.0*
	base	98.2	1.8	—	20.2±0.4	73.2
	X→hi +CG	99.2	9.8	54.5	20.3±0.4	73.0
	+FT	99.4	99.3	99.4	26.5±0.6	75.3*
	base	23.0	77.0	—	27.6±0.6	83.5
	X→it +CG	45.8	88.1	67.0	28.1±0.6	82.9
	+FT	99.2	88.1	93.7	32.4±0.7	84.4*
Source Informal	base	48.5	51.5	—	24.7±0.5	79.3
	X→de +CG	94.0	87.7	90.9	27.0±0.5	79.0
	+FT	100.0	99.7	99.9	30.0±0.6	80.7*
	base	20.3	79.7	—	29.9±0.5	82.7
	X→es +CG	67.5	93.7	80.6	31.1±0.6	82.3
	+FT	99.8	97.8	98.8	33.2±0.7	83.9*
	base	87.7	12.3	—	30.3±0.6	79.6
	X→fr +CG	99.9	59.5	79.7	32.5±0.6	79.6
	+FT	100.0	100.0	100.0	35.6±0.6	81.5*
	base	98.4	1.6	—	20.8±0.4	73.6
	X→hi +CG	99.2	12.3	55.7	20.8±0.4	73.4
	+FT	99.7	99.5	99.6	27.7±0.6	75.8*
	base	1.5	98.5	—	28.0±0.6	83.6
	X→it +CG	17.1	99.4	58.3	28.0±0.6	82.9
	+FT	98.2	99.2	98.7	32.8±0.7	84.5*

TABLE 7.7: Zero-shot formality control results on *new source* languages, using controllers trained on English as source. Sources are {de, es, fr, hi, it}. Colored columns indicate source formality agreeing with desired target formality.

formality in the source sentences, as source sentences with the correct formality improve accuracy on the desired formality (57.8 vs. 42.2% and 79.7 vs. 20.3%). However, the signals in the input alone are insufficient for generating the correct formality. This is confirmed by another zero-shot experiment when both the source and target languages are new (Table 7.8). Here the sources already contain the correct grammatical genders. Despite this, NLLB cannot fully utilize the signals in the source, especially on the feminine gender. Its accuracy (75.4-83.2%) still lags behind the masculine class (87.0-90.4%). Both CG and fine-tuning substantially improve the accuracy and mostly close the gap between the two grammatical classes. This shows both approaches strengthen the source signals that are otherwise neglected.

	Model	F _{feminine}	M _{masculine}	Global	BLEU	COMET
es→it	base	79.4	89.3	85.2	30.0±1.5	83.3
	+CG	87.6	92.3	90.4	29.5±1.4	82.9+
	+FT	90.9	90.5	90.7	30.0±1.3	82.9+
fr→it	base	75.4	90.4	84.2	28.1±1.4	82.6
	+CG	85.1	94.1	90.4	27.7±1.4	82.3
	+FT	90.4	93.6	92.3	28.6±1.4	82.5
es→fr	base	83.2	87.0	85.3	31.2±1.4	79.9
	+CG	86.8	88.8	87.9	31.3±1.4	79.8
	+FT	89.2	88.5	88.8	31.4±1.5	79.7
it→fr	base	76.1	87.2	84.3	31.5±1.3	80.4
	+CG	86.4	89.1	87.9	31.5±1.3	80.5
	+FT	90.6	88.9	89.6	31.8±1.4	80.4

TABLE 7.8: Zero-shot grammatical gender control results on *new source and target* languages.

7.7 Summary

To facilitate attribute-controlled translation in low-resource scenarios, we examine how effectively control mechanisms can be transferred across languages in pre-trained multilingual models. We employ activation steering, a non-invasive, inference-time method, to equip a pre-trained multilingual machine translation model with controllable generation capabilities and contrast its performance with that of standard model fine-tuning.

Our results suggest a clear set of trade-offs, leading to the following recommendations for adding attribute control to existing systems: **1)** Given in-domain data, even only several hundred annotated sentences, fine-tuning provides the most reliable performance. **2)** The optimal method for zero-shot transfer to a new language depends on language relatedness and domain. In case of *distant* new target languages or strong *domain* mismatches between the attribute-annotated data and test data, decoding with classifier guidance is more promising. Otherwise fine-tuning is recommended. **3)** In case specific resource constraints preclude fine-tuning or hosting multiple specialized variants of the underlying model, we then recommend inference-time control by classifier guidance. **4)** In case the underlying translation model is not massively multilingual, fine-tuning the model or training the controller on multiple target languages is beneficial.

The activation steering approach in this chapter is distinct within this thesis for its non-invasive nature. It is the only method that leaves the underlying model’s parameters and architecture entirely unmodified, in contrast to the architectural modifications explored in Chapter 3 and the parameter updates central to Chapters 4-6. While this inference-time approach does not explicitly enforce the cross-lingual alignment objectives of previous chapters, our findings demonstrate its value in specific use cases where conventional fine-tuning proves suboptimal. A logical next step is to investigate the transferability of such inference-time control mechanisms to general-purpose foundation models, a direction we leave for future work.

Chapter 8

Conclusions

This chapter concludes the thesis by summarizing its key contributions and outlining potential directions for future research. We first address the research questions posed in the introduction (§8.1), thereby synthesizing the findings presented throughout the preceding chapters. This is followed by a discussion of the limitations to contextualize these findings (§8.2). The thesis concludes by suggesting avenues for future research that build upon the findings from this work (§8.3).

8.1 Answers to Research Questions

This thesis investigates methods for developing multilingual representations that enhance cross-lingual transfer for sequence-to-sequence tasks, with a primary focus on machine translation, as established in [Chapter 1](#). Based on the findings presented in this work, this section addresses the research questions.

Research Question 1: To what extent are representations in multilingual models language-invariant, and what factors contribute to their language-specific properties?

The internal representations of multilingual models are substantially language-specific and do not converge toward a language-invariant space, even with large-scale multilingual pre-training. Probing analyses in [Chapter 3](#) on encoder-decoder models trained across 6 to 18 translation directions reveal that surface-level language-specific information is heavily preserved within the encoder outputs. Across various datasets, it is possible to recover each token’s original identity and absolute position with over 85% accuracy, while the input language itself is identified with accuracy ranging from 74.1% to 95.9%.

Large-scale multilingual pre-training does not alter this observation. As demonstrated in [Chapter 4](#), translation models pre-trained on over 100 languages exhibit a similar degree of language specificity. These language-specific properties persist even with a more lenient evaluation in [Chapter 5](#), which, as opposed to the exact information recovery accuracy previously, employs a cross-lingual retrieval diagnostic task that only requires a correct translation to be the nearest neighbor in the representation space. Even with this more lenient evaluation, retrieval accuracy remains below 50% with recent decoder-only large language models.

In response to the second part of Research Question 1, this work identifies several factors that contribute to these language-specific patterns, extending beyond inherent properties like linguistic relatedness (Chapter 3) and language resource levels (Chapter 5). The analysis in Chapter 3 reveals that residual connections recurring in model architectures reinforce word order information specific to each input language. Moreover, Chapter 4 demonstrates that, holding other factors unchanged, the selection of a bridge language during multilingual fine-tuning significantly impacts the resulting representations.

Research Question 2: How do current multilingual models perform in scenarios requiring classical cross-lingual, compositional, and temporal generalization, and how does the performance relate to the language-specific properties in Research Question 1?

This thesis evaluates classical cross-lingual generalization on non-translation tasks, including slot filling (Chapter 5) and attribute classification for inference-time steering (Chapter 7). The findings indicate that direct transfer with pre-trained multilingual models is effective. For instance, models can readily perform the task in languages unseen in task-specific training, with F_1 scores between 41.0% and 79.9% on a variety of low- and high-resource languages. The analysis in Chapter 5 further establishes a positive correlation between downstream transfer performance and the language invariance of model representations, as measured by cross-lingual retrieval accuracy. Consistent with this, Chapter 7 shows that attribute classifiers trained on multilingual representations also transfer effectively across diverse languages. These findings suggest that this form of generalization is relatively straightforward to achieve with multilingual pre-trained models, although explicit cross-lingual alignment approaches further improve the transfer performance, as shown in Chapter 5.

In contrast, compositional generalization, assessed with a model’s ability to combine learned translation directions to handle new language pairs in a single pass, remains a significant challenge. As demonstrated in Chapter 3 and Chapter 4, conventional encoder-decoder models trained on English-centric translation data fail to generalize compositionally to non-English pairs where both languages were seen separately during training, often generating an incorrect output language. This failure mode is directly attributed to the source-language-specific representations identified in the answers to Research Question 1, which hinder the model from developing generalizable representations disentangled from language identities. The analysis in Chapter 6 shows that choices of model architecture (encoder-decoder versus decoder-only) do not mitigate this weakness. Instead, instruction-following capabilities are more crucial, where providing instructions in the desired output language is an effective strategy for models supporting this functionality.

Temporal generalization, which concerns a model’s ability to retain pre-trained knowledge after fine-tuning, is evaluated by measuring catastrophic forgetting. The investigation in Chapter 4 highlights that specializing a model on a subset of language pairs can degrade its capabilities in other languages. However, this effect is conditional. As demonstrated in

[Chapter 6](#), a model’s susceptibility to forgetting is determined by the interplay between the scale of the base model and the volume of the fine-tuning data. [Chapter 7](#) presents a more fine-grained use case, showing that forgetting can also appear at the domain level when a model is adapted for a specific style attribute, such as grammatical gender.

Research Question 3: For practical scenarios with different constraints on the scope of model modification, what training procedures can encourage or leverage language-invariant representations for generalization across languages?

The suitable training procedure is highly dependent on the underlying model architecture and resource constraints. This work presents approaches required for different scenarios, from high-flexibility use cases of training models from scratch to the restricted cases of steering frozen pre-trained models. The core findings across these scenarios are summarized in [Table 8.1](#).

Scenario	Approach	Key Findings
Encoder-decoder models trained from scratch	Removing residual connections in a middle encoder layer and complement this with explicit cross-lingual alignment objectives.	Residual connections in the encoder enforce input word orders in models’ internal representations, thereby hindering compositional generalization required for zero-shot translation.
Task-specific pre-trained models	Fine-tune with cross-lingual alignment losses. For applications requiring discrete tokens associated with encoder outputs, a soft discretization approach can be used at the cost of translation quality.	Alignment losses mitigate catastrophic forgetting particularly on translation pairs where source and target are separately seen in fine-tuning. Using adversarial, similarity-only, and contrastive losses for alignment leads to comparable performance.
Decoder-only generic pre-trained models	Apply alignment objectives to the middle layers of the model. If alignment is performed as a separate step, use a contrastive loss to prevent collapse.	The middle layers of pre-trained decoder-only models exhibit the highest level of language-invariant representations. Alignment modules can be merged with task-specific modules in a post-hoc manner.
Frozen models	Use inference-time steering to guide model outputs toward desired attributes without modifying the underlying weights.	Translation attributes such as formality can be controlled dynamically during inference without attribute-specific training data for the output language.

TABLE 8.1: Overview of training procedures for enhancing transfer under different model modification constraints.

8.2 Limitations

English-Centric Translation Directions

A large part of the experiments in this thesis evaluate translation between non-English languages after English-centric translation training, a setup where models often fail. While we also demonstrate that data-centric interventions, such as replacing English with a bridge language more similar to the remaining languages, can mitigate many of these failure modes, we frame this issue as a fundamental modeling challenge rather than a “data problem”. The primary goal is therefore to diagnose why models struggle with compositional generalization. While acknowledging the practical utility of data-centric strategies, this thesis prioritizes the analysis of the architectural and representational bottlenecks that hinder this form of generalization.

Truly Low-Resource Languages

Another limitation of this work is its reliance on simulated low-resource settings for evaluation. Most of the languages investigated have existing supervised data, which was intentionally held out to create experimental conditions without supervised data. This choice is scientifically motivated, as it creates performance upper bounds and allows clear measurement of model performance against a supervised oracle. However, as research extends toward the long tail of the world’s approximately 7,000 languages, many challenges not addressed in this work will become relevant, such as severe data quality issues, the absence of standardized writing systems, and the difficulty of converting non-digital resources into a machine-readable format. Bridging the gap between the model-centric approaches presented in this work and these practical data constraints remains an essential open problem.

8.3 Outlook

Multimodality

The idea of cross-lingual alignment central to this work could be extended to the multimodal domain by treating each modality as a distinct language. While there has been a substantial amount of work on aligning common modalities, such as speech-to-text (Ye et al., 2022; Wang et al., 2022; Ouyang et al., 2023; Duquenne et al., 2023b; Züfle and Niehues, 2025) and image-to-text (Radford et al., 2021; Girdhar et al., 2023), alignment techniques such as explored here could be applied to more specialized modalities like time series or graph-structured data. A challenge in multimodal learning is the absence of multiway parallel data across all modalities. This mirrors the zero-shot problem in translation. By using one or more high-resource anchor modalities, e.g., text, to bridge those that are never seen together, it may be possible to establish meaningful alignments and facilitate knowledge transfer between them.

More Fine-Grained Cross-Lingual Signals

Several key approaches in this work operate on sentence-level representations, where individual tokens are aggregated into single vectors for analyses or cross-lingual alignment. This method is limited by the inherent difficulty of encoding the complete meaning of a sentence into a single fixed-size vector, as Ray Mooney famously pointed out in his critique that one “can’t cram the meaning of a whole [...] sentence into a single [...] vector”¹. Recent works on sentence embeddings (Li et al., 2023b; Janeiro et al., 2025) have demonstrated improvements by complementing sentence-level objectives with more fine-grained token-level training signals. A promising avenue for future research, therefore, is to explore these finer-grained objectives within the cross-lingual alignment for transfer. This direction is particularly relevant given the tokenization disparities across languages (Rust et al., 2021). For low-resource languages, where tokenization is often suboptimal, token-level alignment may capture semantic relationships that sentence-level aggregation might otherwise miss.

Multilingual Consistency in LLMs

The strategies for achieving language invariance, developed within this thesis for machine translation, can be extended to other multilingual applications where consistent model behavior is critical. In instruction-following systems, encouraging language-invariant representations could help a model execute commands in a consistent way regardless of the input language. This consistency is particularly crucial in safety-related scenarios, where harmful behaviors might circumvent safeguards when presented in different languages (Deng et al., 2024), or private information intended for removal in one language may remain (Choi et al., 2024).

Languages Fully Unseen in Pre-Training

In the chapters based on pre-trained models, we have concentrated on languages for which the underlying pre-trained models possess at least some existing knowledge, with the goal of strengthening or recovering multilingual capabilities for specific tasks. An important avenue for future work is the extension of these methods to languages entirely absent from the models’ pre-training data. In these scenarios, the primary challenge shifts from the bidirectional alignment explored in this work to a unidirectional one, which embeds the new language’s representations into the established representational space. Potential approaches include mapping the new vocabulary at a token level to the existing representational space, as demonstrated in the proof of concept by Mullov et al. (2024) on higher-resource languages.

¹<https://www.cs.utexas.edu/~mooney/cramming.html>

Appendix A

Supplementary Information for Chapter 3

A.1 Tokenization

For the languages with Latin script, we first apply the Moses tokenizer and truecaser, and then learn byte pair encoding (BPE) using the subword-nmt toolkit (Sennrich et al., 2016b). For the Indic languages, we use the IndicNLP toolkit¹ and SentencePiece (Kudo and Richardson, 2018) for tokenization and BPE respectively. We choose 40K merge operations and only use tokens with minimum frequency of 50 in the training set.

A.2 Training Details

Our implementation is based on the NMTGMinor toolkit². For all models, we use 8 attention heads, embedding size of 512, inner size of 2048, dropout rate of 0.2, label smoothing rate of 0.1. We follow the learning rate schedule from Vaswani et al. (2017) with 8,000 warmup steps. The model uses a single shared word embedding matrix for both the input and output. The decoder’s final output layer is tied to this same embedding matrix.

We train for 64 epochs and average the weights of the 5 best checkpoints ordered by development loss. By default, we only include the supervised translation directions in the development set. The only exception is the Europarl-full case, where we also include the zero-shot directions in development set for early stopping.

When analyzing model hidden representations through classification performance (§3.6.1 and 3.6.2), we freeze the trained encoder-decoder weights and train the classifier for 5 epochs. The classifier is a linear projection from the encoder hidden dimension to the number of classes, followed by softmax activation. As the classification task is lightweight and convergence is fast, we reduce the warmup steps to 400 while keeping the learning rate schedule unchanged.

A.3 Evaluation Details

For evaluation, we use the sacreBLEU toolkit (Post, 2018). The default evaluation signature is `BLEU+case.mixed+numrefs.1+smooth.exp+tok.13a+version.1.4.12`. On

¹https://github.com/anoopkunchukuttan/indic_nlp_library

²<https://github.com/quanpn90/NMTGMinor>

the PMIndia dataset, following Goyal et al. (2022), we use the subword-based tokenizer `tok.flores101` instead of `tok.13a` for better tokenization of the Indic languages.

For unseen translation directions ($X \rightarrow Y$), we compare against a pivot-based approach that uses English as an intermediary ($X \rightarrow \text{English} \rightarrow Y$). The pivot translations are generated by our baseline multilingual model.

To simulate the case of later adding a new language, we learn a new BPE model for the new language and keep the previous model unchanged. Due to the increased number of unique tokens, the vocabulary of the previously-trained model is expanded. In this case, for the model weights related to the word lookup table size, we initialize them as the average of existing embeddings perturbed by random noise.

A.4 Translation Scores per Direction

Multiway IWSLT 2017

	Direction	Pivot ($X \rightarrow \text{en} \rightarrow Y$)	Baseline	+vardrop	Residual	+vardrop	Residual+Query	+vardrop
Supervised	en-it	-	30.4	30.1	29.8	29.8	30.0	29.7
	en-nl	-	27.9	27.7	27.4	27.6	27.4	26.9
	en-ro	-	23.2	22.9	22.9	22.4	23.3	22.2
	it-en	-	35.8	35.7	35.5	35.0	35.2	33.8
	nl-en	-	31.0	31.0	30.4	30.5	30.5	29.9
	ro-en	-	30.6	30.4	30.1	29.4	29.8	29.3
	Average		29.8	29.6	29.4	29.1	29.4	28.6
Zero-shot	it-nl	19.8	11.5	15.0	18.5	18.6	18.7	18.0
	it-ro	18.3	11.8	15.8	17.8	17.7	18.2	17.3
	nl-it	20.1	10.0	15.2	17.9	18.1	17.9	18.0
	nl-ro	16.4	9.2	14.1	15.5	15.5	15.6	15.5
	ro-it	21.1	12.2	15.8	19.6	19.8	19.5	19.6
	ro-nl	18.5	10.3	13.4	16.8	16.6	16.7	16.5
	Average	19.1	10.8	14.9	17.7	17.7	17.8	17.5

Europarl

Disjoint Data

	Direction	Pivot ($X \rightarrow \text{en} \rightarrow Y$)	Baseline	+vardrop	Residual	+vardrop	Residual+Query	+vardrop
Supervised	da-en	-	38.3	38.5	38.1	38.2	37.7	37.3
	de-en	-	36.1	36.0	35.9	35.5	35.2	34.9
	es-en	-	43.0	43.2	43.0	42.9	42.5	42.1
	fi-en	-	32.6	32.8	32.4	31.9	31.9	31.0
	fr-en	-	39.4	39.5	39.1	39.0	38.8	37.5
	it-en	-	37.3	36.9	36.7	36.4	36.4	35.2
	nl-en	-	34.4	34.2	34.0	33.8	33.3	32.9
	pt-en	-	41.1	41.0	40.7	40.6	40.2	39.5
	en-da	-	36.5	36.7	36.2	36.1	35.8	35.7
	en-de	-	28.1	28.0	27.7	27.5	27.5	26.7
	en-es	-	42.5	42.6	42.2	42.2	41.7	41.6
	en-fi	-	22.5	22.2	22.2	21.6	21.3	20.8

Appendix A. Supplementary Information for Chapter 3

	Direction	Pivot ($X \rightarrow en \rightarrow Y$)	Baseline	+vardrop	Residual	+vardrop	Residual+Query	+vardrop
	en-fr	-	37.7	38.0	37.8	37.5	37.4	37.0
	en-it	-	32.7	32.7	32.4	32.2	32.0	31.8
	en-nl	-	29.7	29.9	29.6	29.4	29.5	29.1
	en-pt	-	38.3	38.4	38.0	38.0	37.9	37.6
	Average		35.6	35.7	35.4	35.2	34.9	34.4
Zero-shot	da-de	24.2	5.2	21.9	25.0	24.5	23.9	23.7
	da-es	33.1	15.8	32.4	32.8	32.5	32.3	31.6
	da-fi	18.1	5.0	16.6	18.1	17.1	17.2	16.6
	da-fr	30.6	9.9	28.1	29.4	29.3	28.9	28.4
	da-it	26.1	11.5	24.3	24.8	25.0	24.8	24.3
	da-nl	26.3	6.5	24.3	25.8	26.0	25.5	25.5
	da-pt	29.9	14.4	28.6	29.3	29.1	28.8	28.4
	de-da	29.2	9.5	26.9	29.2	29.1	27.5	28.3
	de-es	32.1	11.6	31.2	32.1	31.6	31.0	30.9
	de-fi	17.9	4.2	16.7	17.5	16.7	16.3	16.1
	de-fr	29.9	7.3	27.8	28.9	29.2	28.4	28.6
	de-it	25.6	8.8	23.4	24.6	24.5	23.7	23.6
	de-nl	25.9	5.5	23.7	26.1	26.1	25.2	25.4
	de-pt	29.2	10.8	28.0	28.9	28.4	28.2	27.9
	es-da	31.4	10.2	28.1	31.0	30.5	29.5	29.6
	es-de	24.8	4.3	21.4	24.3	23.8	23.4	23.1
	es-fi	19.4	4.5	17.3	18.6	17.5	17.7	17.0
	es-fr	34.8	9.1	33.4	34.8	34.9	34.2	34.1
	es-it	29.6	9.8	28.7	29.7	29.6	29.1	28.9
	es-nl	26.8	5.9	24.3	26.5	26.1	25.3	25.5
	es-pt	35.1	13.6	35.3	35.9	36.0	35.5	35.5
	fi-da	26.3	9.8	23.6	25.4	24.9	24.1	24.3
	fi-de	20.8	4.0	17.8	20.2	19.4	18.8	18.9
	fi-es	29.2	11.9	27.8	28.4	27.6	27.3	26.6
	fi-fr	27.2	7.6	24.3	25.4	25.4	24.7	24.3
	fi-it	23.2	8.4	20.9	21.8	21.4	20.9	20.1
	fi-nl	22.8	5.3	20.2	22.0	21.6	20.7	20.9
	fi-pt	26.5	10.8	25.0	25.3	24.8	24.0	23.9
	fr-da	29.4	8.6	25.6	28.6	28.3	26.8	26.4
	fr-de	23.6	3.4	19.3	23.1	22.7	21.6	21.3
	fr-es	35.9	11.8	35.5	36.4	36.2	35.1	34.7
	fr-fi	17.9	3.9	15.7	17.2	16.0	15.8	15.2
	fr-it	28.4	8.0	27.3	28.7	28.3	27.7	26.8
	fr-nl	26.3	5.0	22.8	25.9	25.4	24.9	24.7
	fr-pt	32.8	11.5	31.8	33.3	33.1	32.6	31.9
	it-da	27.6	8.3	24.3	26.8	26.7	25.2	25.4
	it-de	22.3	3.5	18.1	21.3	21.1	20.2	20.0
	it-es	33.7	10.9	33.9	34.4	34.4	33.5	33.0
	it-fi	17.0	3.4	13.4	15.9	13.8	14.6	13.0
	it-fr	31.4	6.9	30.2	31.6	31.4	30.8	30.7
	it-nl	24.9	5.0	21.6	24.0	23.7	23.4	23.3
	it-pt	30.8	10.6	30.4	31.3	31.1	30.9	30.7
	nl-da	27.7	10.1	25.5	27.7	27.8	26.6	27.0
	nl-de	23.2	4.4	20.0	23.5	23.4	22.5	22.7
	nl-es	30.7	12.6	30.0	30.8	30.6	30.1	29.8
	nl-fi	16.5	4.7	15.5	16.0	15.8	15.6	15.4
	nl-fr	28.8	8.3	27.1	28.1	28.3	27.6	27.8
	nl-it	24.5	9.2	22.5	23.9	23.8	23.1	23.3
	nl-pt	28.2	12.2	27.0	27.5	27.3	27.2	27.1
	pt-da	30.5	9.5	27.1	29.5	29.2	28.0	28.4

Direction	Pivot ($X \rightarrow en \rightarrow Y$)	Baseline	+vardrop	Residual	+vardrop	Residual+Query	+vardrop
pt-de	24.3	3.5	20.6	23.6	23.2	22.4	22.5
pt-es	37.7	14.3	37.7	38.3	38.2	37.5	37.4
pt-fi	18.4	4.1	16.8	17.9	17.1	17.2	16.6
pt-fr	34.1	7.7	32.6	33.9	34.1	33.4	33.3
pt-it	29.3	8.7	27.9	29.2	29.0	28.7	28.3
pt-nl	26.4	5.4	23.7	25.7	25.6	24.8	25.2
Average	27.1	8.2	25.1	26.7	26.4	25.8	25.6

Multiway Data

	Direction	Pivot ($X \rightarrow en \rightarrow Y$)	Baseline	Residual	Residual+Query
Supervised	da-en	-	36.3	36.1	35.0
	de-en	-	33.6	33.4	32.3
	es-en	-	40.5	40.1	39.1
	fi-en	-	30.3	30.1	29.2
	fr-en	-	37.0	36.8	35.2
	it-en	-	34.7	34.4	33.4
	nl-en	-	32.3	32.2	31.2
	pt-en	-	38.3	37.9	37.3
	en-da	-	35.7	35.7	34.8
	en-de	-	28.0	27.1	26.5
	en-es	-	42.0	41.4	41.2
	en-fi	-	21.3	20.8	20.5
	en-fr	-	37.5	37.2	36.7
	en-it	-	32.2	31.8	31.4
	en-nl	-	29.3	29.2	28.7
	en-pt	-	37.9	37.6	37.2
	Average		34.2	33.9	33.1
Zero-shot	da-de	23.3	10.0	23.9	22.8
	da-es	31.6	11.8	32.2	31.4
	da-fi	17.3	6.2	17.0	16.4
	da-fr	29.5	14.4	29.0	27.8
	da-it	25.1	14.3	24.8	24.2
	da-nl	25.2	11.2	25.4	24.1
	da-pt	28.6	12.9	28.9	28.3
	de-da	27.5	14.8	28.4	26.9
	de-es	30.5	10.1	31.4	30.7
	de-fi	16.2	5.2	16.2	15.1
	de-fr	28.7	11.6	28.5	27.2
	de-it	24.3	12.4	24.1	23.1
	de-nl	24.6	9.8	25.4	24.2
	de-pt	27.7	10.5	28.3	27.0
	es-da	30.3	16.4	30.0	28.9
	es-de	23.8	9.4	24.0	22.3
	es-fi	18.4	6.3	17.8	16.9
	es-fr	33.3	15.6	34.0	32.8
	es-it	28.4	17.3	29.4	28.5
	es-nl	25.9	10.7	26.2	24.7
	es-pt	33.2	16.1	35.4	34.6
	fi-da	24.9	13.4	24.6	24.0
	fi-de	19.6	7.9	19.4	18.3
	fi-es	27.8	10.3	27.7	27.2
	fi-fr	26.0	11.5	25.2	23.9

Appendix A. Supplementary Information for Chapter 3

	Direction	Pivot ($X \rightarrow en \rightarrow Y$)	Baseline	Residual	Residual+Query
Zero-shot (cont.)	fi-it	21.8	11.8	21.2	20.5
	fi-nl	21.9	9.0	21.4	20.4
	fi-pt	25.1	11.0	24.8	24.0
	fr-da	28.3	14.2	27.5	26.1
	fr-de	22.6	8.3	22.1	20.5
	fr-es	33.9	12.1	35.6	34.4
	fr-fi	17.0	5.3	16.4	15.3
	fr-it	26.9	15.3	28.1	26.9
	fr-nl	25.1	9.6	25.0	23.9
	fr-pt	31.0	12.8	32.5	31.3
	it-da	26.7	13.8	26.3	25.2
	it-de	21.5	7.7	21.0	19.4
	it-es	32.2	11.7	33.9	33.4
	it-fi	15.9	4.7	15.0	14.8
	it-fr	29.9	12.2	31.2	29.8
	it-nl	24.1	9.2	23.6	22.4
	it-pt	29.5	12.2	30.7	29.8
	nl-da	26.4	15.4	27.1	26.1
	nl-de	22.0	9.3	22.8	21.6
	nl-es	29.3	10.8	30.2	29.3
	nl-fi	15.8	5.6	15.5	14.7
	nl-fr	27.7	12.5	27.7	26.7
	nl-it	23.3	13.4	23.6	22.6
	nl-pt	26.7	12.1	27.1	26.2
	pt-da	29.3	14.3	28.8	27.7
	pt-de	23.4	8.3	23.0	21.6
	pt-es	35.8	14.2	38.0	37.0
	pt-fi	17.6	5.3	17.0	16.3
	pt-fr	32.6	13.8	33.4	32.1
	pt-it	27.9	16.0	28.6	28.1
	pt-nl	25.5	9.3	25.4	24.3
	Average	25.9	11.3	26.1	25.1

PMIndia

	Direction	Pivot ($X \rightarrow en \rightarrow Y$)	Baseline	+vardrop	Residual	+vardrop	Residual+Query
Supervised	te-en	-	31.2	30.2	30.7	30.2	29.7
	kn-en	-	31.4	31.1	31.3	31.1	30.3
	ml-en	-	28.9	27.9	28.2	27.9	27.9
	bn-en	-	25.6	25.4	25.0	25.4	25.0
	gu-en	-	35.3	34.7	34.7	34.7	33.5
	hi-en	-	37.8	37.1	37.0	37.1	35.8
	mr-en	-	29.0	28.7	28.8	28.7	28.1
	or-en	-	29.0	28.9	28.9	28.9	28.3
	pa-en	-	35.9	35.2	35.1	35.2	34.5
	en-te	-	16.3	16.7	16.1	16.1	15.7
	en-kn	-	33.4	33.4	32.6	32.6	32.2
	en-ml	-	20.4	20.8	20.3	20.3	20.1
	en-bn	-	22.6	22.5	21.9	21.9	21.8
	en-gu	-	39.3	39.6	38.9	38.9	38.0
	en-hi	-	32.5	32.6	31.5	31.5	30.8
	en-mr	-	25.7	25.7	25.4	25.4	24.8
	en-or	-	33.7	33.7	33.0	33.0	32.6

Appendix A. Supplementary Information for Chapter 3

	Direction	Pivot (X→en→Y)	Baseline	+vardrop	Residual	+vardrop	Residual+Query
	en-pa	-	38.8	38.9	38.0	38.0	37.1
	Average		30.4	30.2	29.9	29.8	29.2
Zero-shot	te-kn	24.9	0.6	3.2	2.5	16.3	1.3
	te-ml	15.8	1.0	3.8	3.2	10.8	1.5
	te-bn	18.0	0.7	2.7	2.0	12.4	2.0
	te-gu	28.9	1.2	2.4	2.3	19.3	0.9
	te-hi	21.6	0.4	0.5	1.0	12.8	1.9
	te-mr	19.5	1.2	3.8	2.6	12.6	1.7
	te-or	25.4	0.7	2.3	2.6	17.2	1.1
	te-pa	28.2	0.5	2.1	2.5	17.9	0.6
	kn-te	12.8	0.6	2.0	2.0	8.4	0.7
	kn-ml	16.0	0.9	3.6	3.1	10.9	1.8
	kn-bn	17.9	0.9	2.6	2.3	12.1	2.2
	kn-gu	29.3	1.1	2.6	2.4	19.9	0.8
	kn-hi	22.2	0.4	0.5	1.0	12.9	0.4
	kn-mr	19.7	1.2	3.6	2.9	12.8	1.7
	kn-or	25.9	0.6	2.4	2.6	17.4	1.4
	kn-pa	29.0	0.5	2.2	2.7	18.4	0.7
	ml-te	11.9	0.6	2.3	1.9	8.2	0.7
	ml-kn	24.0	0.5	3.3	2.8	16.0	1.2
	ml-bn	17.5	0.9	2.8	2.3	12.0	2.0
	ml-gu	27.3	1.1	2.4	2.3	17.9	0.9
	ml-hi	20.5	0.4	0.5	1.1	11.9	0.4
	ml-mr	18.9	1.3	3.9	3.0	12.1	1.8
	ml-or	24.8	0.6	2.3	2.5	16.3	1.3
	ml-pa	27.0	0.5	2.0	2.4	16.7	0.6
	bn-te	10.7	0.5	2.4	1.9	7.4	0.7
	bn-kn	21.6	0.7	2.9	2.8	14.2	1.3
	bn-ml	14.0	1.0	3.9	3.2	9.3	1.7
	bn-gu	25.3	1.1	2.7	2.1	17.8	1.0
	bn-hi	18.9	0.5	0.7	1.1	11.9	0.5
	bn-mr	17.4	1.6	4.2	3.0	11.7	1.8
	bn-or	23.8	0.9	3.0	3.1	16.7	1.3
	bn-pa	25.1	0.6	2.5	2.5	16.9	0.6
	gu-te	13.2	0.4	1.6	1.6	8.4	0.5
	gu-kn	26.6	0.5	2.3	2.7	16.6	1.2
	gu-ml	16.9	0.7	3.0	2.6	10.7	1.5
	gu-bn	19.2	0.7	1.8	1.9	12.5	1.8
	gu-hi	25.6	0.4	0.5	1.3	15.7	0.4
	gu-mr	21.4	1.4	3.2	2.8	14.5	1.7
	gu-or	28.1	0.6	2.0	2.9	18.1	1.3
	gu-pa	32.0	0.5	2.3	2.9	20.6	0.7
	hi-te	14.0	0.4	1.1	1.4	8.5	0.5
	hi-kn	28.0	0.5	1.3	1.9	15.6	1.0
	hi-ml	17.7	0.6	1.9	2.2	10.4	1.1
	hi-bn	20.0	0.5	1.2	1.4	12.9	1.2
	hi-gu	34.9	1.0	1.6	1.9	21.1	0.8
	hi-mr	22.6	1.0	2.6	2.4	14.5	1.6
	hi-or	30.6	0.5	1.4	2.0	18.1	1.0
	hi-pa	35.3	0.4	1.5	2.3	21.7	0.6
	mr-te	11.9	0.5	1.9	1.8	8.0	0.7
	mr-kn	24.1	0.6	2.8	2.5	15.7	1.4
	mr-ml	15.6	0.9	3.5	2.9	10.4	1.5
	mr-bn	17.4	0.9	2.4	2.2	12.1	2.1
	mr-gu	28.5	1.1	2.5	2.2	19.5	0.9

A.5. Details on Implementation of Prior Approaches

Direction	Pivot ($X \rightarrow en \rightarrow Y$)	Baseline	+vardrop	Residual	+vardrop	Residual+Query
mr-hi	21.6	0.5	0.7	1.4	13.4	0.6
mr-or	25.3	0.7	2.2	2.6	17.0	1.4
mr-pa	27.6	0.6	2.2	2.8	17.0	0.7
or-te	11.4	0.6	2.0	1.8	8.0	0.6
or-kn	23.1	0.7	2.9	3.0	15.5	1.5
or-ml	14.9	0.9	3.5	3.2	10.3	1.6
or-bn	17.5	0.9	3.0	2.6	12.7	2.3
or-gu	28.2	1.3	3.0	2.7	19.6	1.0
or-hi	21.7	0.4	0.8	1.5	13.9	0.5
or-mr	18.7	1.5	4.2	3.1	12.3	1.7
or-pa	27.7	0.6	2.7	3.1	18.7	0.7
pa-te	12.8	0.5	1.6	1.8	8.5	0.5
pa-kn	26.4	0.6	2.3	2.6	16.2	1.1
pa-ml	16.6	0.8	3.2	2.7	10.6	1.5
pa-bn	18.9	0.8	2.0	1.9	12.6	1.9
pa-gu	32.4	1.4	2.6	2.6	22.0	1.0
pa-hi	26.8	0.5	0.6	1.5	17.2	0.4
pa-mr	21.3	1.4	3.4	3.0	13.8	1.7
pa-or	28.0	0.7	2.2	2.7	18.6	1.2
Average	22.1	0.8	2.4	2.3	14.3	1.1

A.5 Details on Implementation of Prior Approaches

For similarity regularization, we choose meanpooling and the Euclidean distance, since they provided better or more consistent performance than maxpooling and the cosine distance respectively in initial experiments. We use a weight of 0.1 on the similarity objective following [Pham et al. \(2019\)](#). For the adversarial language classification approach, we alternate the training between the translation and classification steps with a 1:1 ratio. The gradient reversal layer uses a factor of -1, therefore only reversing the gradient without changing their magnitude.

Appendix B

Supplementary Information for Chapter 4

B.1 Dataset Details

The training parallel data¹ are compiled by Wenzek et al. (2021) and include the following datasets: bible-uedin (Christodoulopoulos and Steedman, 2015), (Multi)CCAligned (El-Kishky et al., 2020), Gnome, ELRC, KDE4, GlobalVoices, OpenSubtitles, QED (Abdelali et al., 2014), MultiParaCrawl, TED2020, Tanzil, Tatoeba, Ubuntu, WikiMatrix (Schwenk et al., 2021a), wikimedia, and TICO-19 (Anastasopoulos et al., 2020).

B.2 Training, Inference, and Evaluation Details

Our implementation is based on FAIRSEQ (Ott et al., 2019). When training, one optimization step happens after 16384 target tokens. The learning rate is 0.0001 with the inverse square root schedule and 2500 warmup steps. As for regularization parameters, we use label smoothing of 0.1, dropout of 0.3, and attention dropout 0.1. The models are trained for 500K updates in total. An exception is the MULTI-BRIDGE experiment with more training data, where we trained for 800K updates in total. For inference, we use a beam size of 5.

The scale on the codebook loss and commitment loss (α_{codebook} and $\alpha_{\text{commitment}}$) are 1.0 and 1.001. We found the model sensitive to increasing $\alpha_{\text{commitment}}$, where higher values lead to index collapse². After exponentially decreasing it to approach 1.0, we settled at 1.001. For the probability of seeing the continuous encoder context, with a search among {0.1, 0.5, 0.7, 0.9}, we found 0.9 and 0.5 the best parameters for ID-BRIDGE and EN-BRIDGE respectively.

For BLEU evaluation, we use the sacreBLEU toolkit (Post, 2018) with signature `nrefs:1|case:mixed|eff:no|tok:flores101|smooth:exp|version:2.3.1`.

The tokenizer choice “tok:flores101” follows the official metric of the WMT shared task (Wenzek et al., 2021) where the datasets are sourced. BLEU using this tokenizer is also referred to as spBLEU (Goyal et al., 2022).

¹https://data.statmt.org/wmt21/multilingual-task/small_task2_filt_v2.tar.gz

²A potential reason is the encoder parameters are updated too aggressively by the commitment loss in these cases.

Appendix C

Supplementary Information for Chapter 5

C.1 Dataset Details

All our task-specific training data are retrieved from HuggingFace¹. The translation test sets are hosted by WMT². The alignment data are sourced from Tatoeba³ with its default version of v2021-07-22 at the time of writing. We filter out translations that are empty or include multiple sentences. The ablation English-German alignment data is from IWSLT 2017⁴ (Cettolo et al., 2017).

C.2 List of Languages

The languages involved in our downstream tasks are listed in Table C.1. The 35 languages in the initial analyses in §5.3 include all languages in slot filling and machine translation. They additionally include the following languages: am (Amharic), bn (Bengali), it (Italian), hu (Hungarian), hy (Armenian), id (Indonesian), kn (Kannada), ka (Georgian), mn (Mongolian), km (Khmer), ko (Korean), and lv (Latvian).

C.3 Training and Inference Details

Our implementation is based on HuggingFace Transformer (Wolf et al., 2020).

Training Hyperparameters

Fine-tuning is performed using LoRA (Hu et al., 2022) adapters with a rank of 8 for all attention components and linear projections (query, key, value, output, gate, up, down). We set LoRA’s α parameter to 16 and dropout to 0.1. The number of trainable parameters is 20,971,520 on Llama 3, and 20,185,088 on Qwen 2.5. We train at most 5 epochs on the task data. Training on all our tasks converged before reaching the max number of epochs. The learning rate is set to 5e-4 with inverse square root schedule and warmup

¹MASSIVE: <https://huggingface.co/datasets/AmazonScience/massive>
ALMA: <https://huggingface.co/datasets/haoranxu/ALMA-Human-Parallel>
UNER: https://huggingface.co/datasets/CohereForAI/aya_collection/viewer/templated_uner_llm

²<https://github.com/wmt-conference/wmt23-news-systems/tree/master/txt>

³<https://huggingface.co/datasets/Helsinki-NLP/tatoeba>

⁴<https://huggingface.co/datasets/IWSLT/iwslt2017>

Code	FLoRes Code	Full Name	Slot Filling	Machine Translation	JSON Generation
af	afr_Latn	Afrikaans	✓		
az	azj_Latn	North Azerbaijani	✓		
ar	arb_Arab	Modern Standard Arabic	✓		
cs	ces_Latn	Czech		✓	
cy	cym_Latn	Welsh	✓		
da	dan_Latn	Danish			✓
de	deu_Latn	German	✓	✓	
el	ell_Grek	Greek	✓		
en	eng_Latn	English	✓	✓	✓
es	spa_Latn	Spanish	✓		
fr	fra_Latn	French	✓		
he	heb_Hebr	Hebrew		✓	
hi	hin_Deva	Hindi	✓		
hr	hrv_Latn	Croatian			✓
is	isl_Latn	Icelandic	✓	✓	
ja	jpn_Jpan	Japanese	✓	✓	
jv	jav_Latn	Javanese	✓		
pt	por_Latn	Portuguese			✓
ru	rus_Cyrl	Russian	✓	✓	
sk	slk_Latn	Slovak			✓
sr	srp_Cyrl	Serbian			✓
sv	swe_Latn	Swedish			✓
sw	swh_Latn	Swahili	✓		
th	tha_Thai	Thai	✓		
tl	tgl_Latn	Tagalog	✓		
tr	tur_Latn	Turkish	✓		
uk	ukr_Cyrl	Ukrainian		✓	
ur	urd_Arab	Urdu	✓		
zh	zho_Hans	Chinese (Simplified)	✓	✓	✓

TABLE C.1: List of languages evaluated on different downstream tasks.

up ratio 0.03. We save checkpoints and evaluate every 200 optimization steps, and early stop if the development loss does not improve for 5 consecutive evaluations. For the temperature parameter τ in the contrastive loss, we searched among {0.1, 1.0, 1.5, 2.0} based on development loss on machine translation. For Llama we use 0.1, for Qwen we use 1.5.

Prompt Format

Slot Filling The system prompt is shortened from [He and Garner \(2023\)](#).⁵

- **System:** Given a command from the user, a voice assistant will extract entities essential for carry out the command. Your task is to extract the entities as words from the command if they fall under a predefined list of entity types.
- **User:** wake me up at five am this week
- **Assistant:** time: five am; date: this week
- **User (de):** wecke mich in dieser woche um fünf uhr auf
- **Assistant (de):** date: dieser woche; time: fünf uhr

⁵We stayed consistent to the original prompt text, including the typographical errors.

C.4. Results for Individual Languages

For **zero-shot slot filling** experiments, we need to specify more requirements in the system prompt with the template also following [He and Garner \(2023\)](#).

Machine Translation

- **System:** Translate the following sentences from English to German.
- **User:** Police arrest 15 after violent protest outside UK refugee hotel.
- **Assistant:** Polizei verhaftet 15 Menschen nach gewalttätigen Protesten vor einer Flüchtling-sunterkunft in Großbritannien

JSON Generation

- **User:** Please identify all the named entities mentioned in the input sentence provided below. Use only the categories: PER - person, ORG - organization, and LOC - location. Remember, nationalities are neither locations nor organizations, and organizations can represent other groups of people. Pay attention to the provided example. You should only output the results in JSON format, following a similar structure to the example result provided. Example sentence and results: Where in the world is Iguazu? "Results": [{ "TypeName": "LOC", "Text": "Iguazu", "Start": 22, "End": 28 }] Considering the input sentence below, what is the output result? Widely considered to be one of the most spectacular waterfalls in the world, the Iguazu Falls on the border of Argentina and Brazil, are a certainly must see attraction in the area.
- **Assistant:** "Results": [{ "TypeName": "LOC", "Text": "Iguazu Falls", "Start": 81, "End": 93 }, { "TypeName": "LOC", "Text": "Argentina", "Start": 111, "End": 120 }, { "TypeName": "LOC", "Text": "Brazil", "Start": 125, "End": 131 }]

Inference and Evaluation Details

We use greedy decoding in all experiments for easily reproducible results. For the model merging experiments, we searched among weights {0.5, 0.7, 0.9} for the task-specific LoRA modules on the MASSIVE development set and chose 0.9 for our experiments.

For BLEU evaluation, we use the sacreBLEU toolkit ([Post, 2018](#)). The evaluation signatures are `nrefs:1|case:mixed|eff:no|tok:13a|smooth:exp|version:2.4.2`, with the dedicated tokenizers `ja-mecab` and `zh` for Japanese and Chinese respectively.

Details for Retrieval

To evaluate cross-lingual retrieval performance, we adapt the implementation from LASER⁶ ([Schwenk et al., 2021a](#)) to process representations extracted offline.

C.4 Results for Individual Languages

The detailed results for [Table 5.3](#) and [Table 5.4](#) are presented in [Table C.2](#), [Table C.3](#) (slot filling) and [Table C.4](#) (machine translation).

⁶<https://github.com/facebookresearch/LASER/tree/main/tasks/xsim>

	Supervised					Transfer (aligned)				
	ar	en	es	ru	zh	cy	ja	jv	sw	tl
Llama 3 SFT	75.5	81.7	74.5	77.6	73.8	44.0	65.8	41.0	42.8	65.0
+ align	75.1	82.0	74.9	78.0	74.9	49.4	66.5	48.2	47.7	65.5
Qwen 2.5 SFT	74.7	81.1	74.0	77.5	74.1	27.0	67.3	32.9	23.5	57.4
+ align	74.9	82.5	74.8	78.0	75.1	36.5	68.3	39.6	30.4	57.8

TABLE C.2: Per-language F_1 results on slot filling: Supervised and Transfer (aligned).

	Transfer (other)									
	af	az	de	el	fr	hi	is	th	tr	ur
Llama 3 SFT	66.0	49.0	75.0	69.4	71.9	70.0	45.0	79.9	60.4	57.1
+ align	66.2	47.9	74.7	72.4	72.1	69.6	48.0	79.1	62.2	56.1
Qwen 2.5 SFT	58.9	45.9	74.6	63.3	70.8	60.0	34.4	79.9	59.9	46.5
+ align	63.1	42.5	74.6	63.3	70.9	61.3	35.8	80.2	58.1	47.2

TABLE C.3: Per-language F_1 results on slot filling: Transfer (other).

	Supervised X→En					Supervised En→X					Transfer X→En			Transfer En→X		
	cs	de	is	ru	zh	cs	de	is	ru	zh	he	ja	uk	he	ja	uk
BLEU																
Llama 3 SFT	37.8	43.0	28.3	32.0	22.5	25.9	35.5	10.6	25.2	38.9	39.3	17.5	38.7	14.5	14.2	17.7
+ align	38.4	43.1	29.1	32.4	23.0	24.7	34.7	10.9	24.4	38.1	39.8	18.8	38.4	16.0	15.6	19.5
Qwen 2.5 SFT	36.1	40.8	20.5	30.6	23.2	21.5	33.7	6.8	25.3	45.3	34.6	18.9	35.6	13.3	17.6	13.0
+ align	36.6	41.4	21.2	30.9	24.0	20.5	32.7	4.8	25.0	45.3	36.3	19.4	36.8	12.7	17.8	13.5
COMET																
Llama 3 SFT	85.2	84.9	81.0	82.4	79.7	84.3	81.8	68.7	83.3	84.2	83.6	79.8	85.1	75.7	83.5	79.7
+ align	85.5	84.9	81.1	82.4	79.8	83.8	81.6	69.0	83.3	84.0	83.6	80.1	85.2	77.1	84.2	80.8
Qwen 2.5 SFT	84.8	84.7	74.1	82.6	80.2	80.8	80.6	52.0	83.3	86.1	82.3	81.3	84.5	70.7	85.5	74.6
+ align	85.1	84.7	74.4	82.6	80.4	79.5	80.1	46.5	83.1	85.8	82.2	81.4	84.6	70.7	85.7	74.4

TABLE C.4: Per-language BLEU and COMET results on machine translation.

Appendix D

Supplementary Information for Chapter 7

D.1 Dataset Statistics

OPUS-100 Data for Transformer-Base

The data overview is in [Table D.1](#). For tokenization, we use the SentencePiece ([Kudo and Richardson, 2018](#)) model from NLLB-200 ([NLLB Team, 2024](#)). The model is trained to translate from and into English.

Direction	# Sentences	# Tokens (en)	# Tokens (X)
en-es	1,000,000	15,482,094	16,422,413
en-de	1,000,000	17,952,717	20,142,507
en-fr	1,000,000	21,495,343	26,634,530
en-hi	534,319	8,723,899	10,913,496
en-it	1,000,000	14,435,382	15,524,589
en-ko	1,000,000	11,290,102	9,552,148
en-pt	1,000,000	13,879,742	14,410,909
en-ru	1,000,000	16,638,782	19,630,699

TABLE D.1: Overview of OPUS-100 data we used to train the Transformer-base.

Details on Domain Mismatch Data

For the grammatical gender control experiments with domain mismatch (§7.4.2), the training domain differs from the test sets in both style and length. An overview is shown in [Table D.2](#).

During training, an example tuple of (input, output, attribute label) is: ("the actor finished her work.", "La actriz terminó su trabajo.", 0: feminine) ("the actor finished his work.", "El actor terminó su trabajo.", 1: masculine). The training sentences are all artificial sentences following this simple subject-verb-object structure. This differs significantly from the test sets with public speaking texts.

D.2 Training and Inference Details

Our implementation is based on FAIRSEQ ([Ott et al., 2019](#)).

Split	Style	Avg. # output words per sent.
Train	artificial sentences	5.5
Test (supervised)	TED talks	25.4
Test (new target lang.)	TED talks	25.2
Test (new source & target lang.)	TED talks	26.2

TABLE D.2: Details on domain mismatch training setup.

Inference and Evaluation

For CoCoA-MT (Nadejde et al., 2022), many test inputs contain multiple sentences. When directly decoding, NLLB-200 (NLLB Team, 2024) suffered from severe under-translation, where the output translation only contains one sentence. We therefore split the input by sentence boundaries and decode sentence by sentence. When decoding, we use a beam size of 4. To evaluate BLEU and COMET scores, we concatenate the hypotheses and references from different attributes. It is also the case when reporting the multi-source results in Table 7.7. We use the sacreBLEU toolkit (Post, 2018) for BLEU evaluation. The evaluation signature is `bs:1000|rs:12345|c:mixed|e:no|tok:13a|s:exp|v:2.3.1`.

Details on Fine-Tuning

When finetuning NLLB-200, we use a batch size of 16K target tokens. For bilingual systems, we train for 30 updates. When training multilingually, we train for 60 updates. We use a learning rate of 0.0001 with an inverse squared root schedule and 20 warmup steps. Dropout is set to 0.1.

Details on Classifier Guidance

Attribute Classifier Training The classifier operates on meanpooled decoder hidden states and consists of two feedforward layers with ReLU activation in between. The first layer projects from the 1024 Transformer hidden dimension to 256, the second layer from 256 to C , the number of attribute classes. In our experiments, C is 2 for formality control (formal, informal) and 3 for gender control (feminine, masculine, neutral)¹.

We train the classifier on a frozen NLLB-200 600M model with an effective batch size of 32K target tokens. The learning rate is 0.002 with an inverse square root schedule and 20 warm-up steps. We use the Adam (Kingma and Ba, 2015) optimizer with betas of (0.9, 0.98). Dropout and label smoothing are set at 0.1. For formality control, we train the monolingual classifiers for 100 updates and multilingual for 250 updates. For the grammatical gender control, we train for 25 updates. Due to the small dataset and simplicity of the training data, training converges fast.

Hyperparameters For the classifier guidance hyperparameters, on the en→de training data of CoCoA-MT, we searched among step size [0.05, 0.1, 0.5], and number of iterations [3, 5]. We used 5 iterations and 0.1 step size for formality control, and 5 iterations and 0.05

¹As our test set only covers two genders, we only report scores on two genders.

D.3. Details on Human Evaluation

step size for grammatical gender control. We do not use KL regularization and postnorm fusion as in [Dathathri et al. \(2020\)](#), since they degraded performance in initial experiments.

Decoding Speed Decoding with our approach is slow due to the repeated gradient updates. For instance on formality control, decoding on the test sets of 600 sentences takes around 30 minutes.

D.3 Details on Human Evaluation

We randomly sampled 16 source English sentences containing second person pronouns from the CoCoA-MT test set, and collected 5 translations for each: from baseline NLLB-200, as well as from CG (multi) and FT (multi) for both formalities². A native speaker rated the 80 hypotheses.

During the evaluation, we learned that there are three levels of formality in Bengali, where: 1) the lowest formality level is only used between very close relations; 2) the next higher level is used between families or acquaintances; 3) the highest level is used between unfamiliar persons or those between higher social distances. We therefore asked the annotator to match each formality category to one integer point. That is, 1, 2, and 3 correspond to very informal, informal, and formal respectively. We also learned that the lowest formality level is only used between very close relations and therefore rare.

While scoring, the annotator was presented with the English source sentences and their Bengali translations together in random order, and asked to score translation quality on a 5-point scale (1 being the worst) and formality scores on a 3-point scale (1 being the least formal).

D.4 Comparison to Prior Works on Different Data Conditions

We compare our results to prior works that used *more relaxed* data conditions than ours for the zero-shot tasks. In [Table D.3](#), the first four systems are submissions to the unconstrained zero-shot track of the IWSLT 2023 formality control shared task ([Agarwal et al., 2023](#)). We compare to submissions in the unconstrained track, as our models would fall under this track due to the use of pre-trained models. The scores of other systems are from Table 48 of [Agarwal et al. \(2023\)](#). We grayed out our COMET scores, as it is unknown whether the organizers used the same underlying COMET model for evaluating these systems. Overall, our model’s performance is comparable to the leading systems.

²Due to resource constraints, we could not include the combination of finetuning and classifier guidance in the evaluation.

	Formality	BLEU	COMET	M-Acc
en→pt				
Ours	formal	40.3	85.3	100
	informal	43.9	86.0	83
Wu et al. (2023)	formal	45.4	77.4	100
	informal	49.1	78.5	100
Bahar et al. (2023)	formal	34.6	60.9	99
	informal	42.4	67.9	64
Lee et al. (2023)	formal	31.0	52.5	100
	informal	19.9	24.9	68
Vakharia et al. (2023)	formal	26.6	40.5	90
	informal	28.4	42.5	58
en→ru				
Ours	formal	33.2	84.4	100
	informal	33.0	84.4	93
Bahar et al. (2023)	formal	35.4	61.7	99
	informal	33.0	60.3	98
Wu et al. (2023)	formal	33.7	58.0	100
	informal	32.4	55.6	100
Lee et al. (2023)	formal	25.8	44.5	100
	informal	26.3	41.8	100
Vakharia et al. (2023)	formal	18.4	-17.1	99
	informal	14.9	-27.7	52
Vincent et al. (2023)	formal	unknown	unknown	100
	informal	unknown	unknown	99

TABLE D.3: Comparison to prior works with different data conditions.

Bibliography

- Ahmed Abdelali, Francisco Guzman, Hassan Sajjad, and Stephan Vogel. 2014. [The AMARA corpus: Building parallel language resources for the educational domain](#). In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1856–1862, Reykjavik, Iceland. European Language Resources Association (ELRA).
- David Ifeoluwa Adelani, A. Seza Doğruöz, André Coneglian, and Atul Kr. Ojha. 2024. [Comparing LLM prompting with cross-lingual transfer performance on indigenous and low-resource Brazilian languages](#). In *Proceedings of the 4th Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP 2024)*, pages 34–41, Mexico City, Mexico. Association for Computational Linguistics.
- Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2017. [Fine-grained analysis of sentence embeddings using auxiliary prediction tasks](#). In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24–26, 2017, Conference Track Proceedings*. OpenReview.net.
- Milind Agarwal, Sweta Agrawal, Antonios Anastasopoulos, Luisa Bentivogli, Ondřej Bojar, Claudia Borg, Marine Carpuat, Roldano Cattoni, Mauro Cettolo, Mingda Chen, William Chen, Khalid Choukri, Alexandra Chronopoulou, Anna Currey, Thierry Declerck, Qianqian Dong, Kevin Duh, Yannick Estève, Marcello Federico, and 43 others. 2023. [FINDINGS OF THE IWSLT 2023 EVALUATION CAMPAIGN](#). In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*, pages 1–61, Toronto, Canada (in-person and online). Association for Computational Linguistics.
- Roei Aharoni, Melvin Johnson, and Orhan Firat. 2019. [Massively multilingual neural machine translation](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3874–3884, Minneapolis, Minnesota. Association for Computational Linguistics.
- Wasi Ahmad, Zhisong Zhang, Xuezhe Ma, Eduard Hovy, Kai-Wei Chang, and Nanyun Peng. 2019. [On difficulties of cross-lingual transfer with order differences: A case study on dependency parsing](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2440–2452, Minneapolis, Minnesota. Association for Computational Linguistics.

- AI @ Meta, Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, and 543 others. 2024. [The llama 3 herd of models](#). *Preprint*, arXiv:2407.21783.
- Alham Fikri Aji, Nikolay Bogoychev, Kenneth Heafield, and Rico Sennrich. 2020. [In neural machine translation, what does transfer learning transfer?](#) In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7701–7710, Online. Association for Computational Linguistics.
- Farhad Akhbardeh, Arkady Arkhangorodsky, Magdalena Biesialska, Ondřej Bojar, Rajen Chatterjee, Vishrav Chaudhary, Marta R. Costa-jussa, Cristina España-Bonet, Angela Fan, Christian Federmann, Markus Freitag, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Leonie Harter, Kenneth Heafield, Christopher Homan, Matthias Huck, Kwabena Amponsah-Kaakyire, and 17 others. 2021. [Findings of the 2021 conference on machine translation \(WMT21\)](#). In *Proceedings of the Sixth Conference on Machine Translation*, pages 1–88, Online. Association for Computational Linguistics.
- Maruan Al-Shedivat and Ankur Parikh. 2019. [Consistency by agreement in zero-shot neural machine translation](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1184–1197, Minneapolis, Minnesota. Association for Computational Linguistics.
- Duarte Alves, Nuno Guerreiro, João Alves, José Pombal, Ricardo Rei, José de Souza, Pierre Colombo, and Andre Martins. 2023. [Steering large language models for machine translation with finetuning and in-context learning](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11127–11148, Singapore. Association for Computational Linguistics.
- Duarte Miguel Alves, José Pombal, Nuno M Guerreiro, Pedro Henrique Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G. C. de Souza, and Andre Martins. 2024. [Tower: An open multilingual large language model for translation-related tasks](#). In *First Conference on Language Modeling*.
- Antonios Anastasopoulos, Ondřej Bojar, Jacob Bremerman, Roldano Cattoni, Maha El-bayad, Marcello Federico, Xutai Ma, Satoshi Nakamura, Matteo Negri, Jan Niehues, Juan Pino, Elizabeth Salesky, Sebastian Stüker, Katsuhito Sudoh, Marco Turchi, Alexander Waibel, Changhan Wang, and Matthew Wiesner. 2021. [FINDINGS OF THE IWSLT 2021 EVALUATION CAMPAIGN](#). In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 1–29, Bangkok, Thailand (online). Association for Computational Linguistics.

Bibliography

- Antonios Anastasopoulos, Alessandro Cattelan, Zi-Yi Dou, Marcello Federico, Christian Federmann, Dmitriy Genzel, Francisco Guzmán, Junjie Hu, Macduff Hughes, Philipp Koehn, Rosie Lazar, Will Lewis, Graham Neubig, Mengmeng Niu, Alp Öktem, Eric Paquin, Grace Tang, and Sylwia Tur. 2020. [TICO-19: the translation initiative for COvid-19](#). In *Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020*, Online. Association for Computational Linguistics.
- Alan Ansell, Edoardo Ponti, Anna Korhonen, and Ivan Vulić. 2022. [Composable sparse fine-tuning for cross-lingual transfer](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1778–1796, Dublin, Ireland. Association for Computational Linguistics.
- Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Roei Aharoni, Melvin Johnson, and Wolfgang Macherey. 2019a. [The missing ingredient in zero-shot neural machine translation](#). *CoRR*, abs/1903.07091.
- Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George F. Foster, Colin Cherry, Wolfgang Macherey, Zhifeng Chen, and Yonghui Wu. 2019b. [Massively multilingual neural machine translation in the wild: Findings and challenges](#). *CoRR*, abs/1907.05019.
- Mikel Artetxe and Holger Schwenk. 2019a. [Margin-based parallel corpus mining with multilingual sentence embeddings](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3197–3203, Florence, Italy. Association for Computational Linguistics.
- Mikel Artetxe and Holger Schwenk. 2019b. [Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond](#). *Transactions of the Association for Computational Linguistics*, 7:597–610.
- Seth Ayccock and Rachel Bawden. 2024. [Topic-guided example selection for domain adaptation in LLM-based machine translation](#). In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 175–195, St. Julian’s, Malta. Association for Computational Linguistics.
- Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. [Layer normalization](#). *CoRR*, abs/1607.06450.
- Alexei Baevski, Steffen Schneider, and Michael Auli. 2020. [vq-wav2vec: Self-supervised learning of discrete speech representations](#). In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.
- Parnia Bahar, Patrick Wilken, Javier Iranzo-Sánchez, Mattia Di Gangi, Evgeny Matusov, and Zoltán Tüske. 2023. [Speech translation with style: AppTek’s submissions to the IWSLT subtitling and formality tracks in 2023](#). In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*, pages 251–260, Toronto, Canada (in-person and online). Association for Computational Linguistics.

- Ankur Bapna and Orhan Firat. 2019. [Simple, scalable adaptation for neural machine translation](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1538–1548, Hong Kong, China. Association for Computational Linguistics.
- Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. 2019. [Findings of the 2019 conference on machine translation \(WMT19\)](#). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 1–61, Florence, Italy. Association for Computational Linguistics.
- Christos Baziotis, Mikel Artetxe, James Cross, and Shruti Bhosale. 2022. [Multilingual machine translation with hyper-adapters](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1170–1185, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2017. [What do neural machine translation models learn about morphology?](#) In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 861–872, Vancouver, Canada. Association for Computational Linguistics.
- Yoshua Bengio, Aaron C. Courville, and Pascal Vincent. 2013a. [Representation learning: A review and new perspectives](#). *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(8):1798–1828.
- Yoshua Bengio, Nicholas Léonard, and Aaron C. Courville. 2013b. [Estimating or propagating gradients through stochastic neurons for conditional computation](#). *CoRR*, abs/1308.3432.
- Luisa Bentivogli, Arianna Bisazza, Mauro Cettolo, and Marcello Federico. 2016. [Neural versus phrase-based machine translation quality: a case study](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 257–267, Austin, Texas. Association for Computational Linguistics.
- Luisa Bentivogli, Beatrice Savoldi, Matteo Negri, Mattia A. Di Gangi, Roldano Cattoni, and Marco Turchi. 2020. [Gender in danger? evaluating speech translation technology on the MuST-SHE corpus](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6923–6933, Online. Association for Computational Linguistics.
- Alexandre Berard. 2021. [Continual learning in multilingual NMT via language-specific embeddings](#). In *Proceedings of the Sixth Conference on Machine Translation*, pages 542–565, Online. Association for Computational Linguistics.
- Dan Biderman, Jacob Portes, Jose Javier Gonzalez Ortiz, Mansheej Paul, Philip Greengard, Connor Jennings, Daniel King, Sam Havens, Vitaliy Chiley, Jonathan Frankle, Cody

Bibliography

- Blakeney, and John Patrick Cunningham. 2024. [LoRA learns less and forgets less](#). *Transactions on Machine Learning Research*. Featured Certification.
- Irina Bigoulaeva, Viktor Hangya, and Alexander Fraser. 2021. [Cross-lingual transfer learning for hate speech detection](#). In *Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion*, pages 15–25, Kyiv. Association for Computational Linguistics.
- Eleftheria Briakou, Colin Cherry, and George Foster. 2023. [Searching for needles in a haystack: On the role of incidental bilingualism in PaLM’s translation capability](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9432–9452, Toronto, Canada. Association for Computational Linguistics.
- Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. [Language models are few-shot learners](#). In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.
- Rich Caruana. 1997. [Multitask learning](#). *Mach. Learn.*, 28(1):41–75.
- Mauro Cettolo, Marcello Federico, Luisa Bentivogli, Jan Niehues, Sebastian Stüker, Katsuhito Sudoh, Koichiro Yoshino, and Christian Federmann. 2017. [Overview of the IWSLT 2017 evaluation campaign](#). In *Proceedings of the 14th International Conference on Spoken Language Translation*, pages 2–14, Tokyo, Japan. International Workshop on Spoken Language Translation.
- Changyou Chen, Jianyi Zhang, Yi Xu, Liqun Chen, Jiali Duan, Yiran Chen, Son Tran, Belinda Zeng, and Trishul Chilimbi. 2022. [Why do we need large batchsizes in contrastive learning? A gradient-bias perspective](#). In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.
- Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2021. [InfoXLM: An information-theoretic framework for cross-lingual language model pre-training](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3576–3588, Online. Association for Computational Linguistics.
- Ting-Rui Chiang, Yi-Pei Chen, Yi-Ting Yeh, and Graham Neubig. 2022. [Breaking down multilingual machine translation](#). In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2766–2780, Dublin, Ireland. Association for Computational Linguistics.

- Nadezhda Chirkova and Vassilina Nikoulina. 2024. [Key ingredients for effective zero-shot cross-lingual knowledge transfer in generative tasks](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7222–7238, Mexico City, Mexico. Association for Computational Linguistics.
- Minseok Choi, Kyunghyun Min, and Jaegul Choo. 2024. [Cross-lingual unlearning of selective knowledge in multilingual language models](#). In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10732–10747, Miami, Florida, USA. Association for Computational Linguistics.
- Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, and 48 others. 2023. [Palm: Scaling language modeling with pathways](#). *J. Mach. Learn. Res.*, 24:240:1–240:113.
- Christos Christodoulopoulos and Mark Steedman. 2015. [A massively parallel corpus: the bible in 100 languages](#). *Lang. Resour. Evaluation*, 49(2):375–395.
- Alexandra Chronopoulou, Matthew Peters, Alexander Fraser, and Jesse Dodge. 2023a. [AdapterSoup: Weight averaging to improve generalization of pretrained language models](#). In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2054–2063, Dubrovnik, Croatia. Association for Computational Linguistics.
- Alexandra Chronopoulou, Jonas Pfeiffer, Joshua Maynez, Xinyi Wang, Sebastian Ruder, and Priyanka Agrawal. 2024. [Language and task arithmetic with parameter-efficient layers for zero-shot summarization](#). In *Proceedings of the Fourth Workshop on Multilingual Representation Learning (MRL 2024)*, pages 114–126, Miami, Florida, USA. Association for Computational Linguistics.
- Alexandra Chronopoulou, Dario Stojanovski, and Alexander Fraser. 2023b. [Language-family adapters for low-resource multilingual neural machine translation](#). In *Proceedings of the Sixth Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2023)*, pages 59–72, Dubrovnik, Croatia. Association for Computational Linguistics.
- Chenhui Chu, Raj Dabre, and Sadao Kurohashi. 2017. [An empirical comparison of domain adaptation methods for neural machine translation](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 385–391, Vancouver, Canada. Association for Computational Linguistics.
- Common Crawl. 2025. Common crawl monthly archives language statistics. <https://commoncrawl.github.io/cc-crawl-statistics/plots/languages>. Accessed: 2025-09-17.

Bibliography

- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. [Unsupervised cross-lingual representation learning at scale](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.
- Alexis Conneau and Guillaume Lample. 2019. [Cross-lingual language model pretraining](#). In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 7057–7067.
- Asa Cooper Stickland, Alexandre Berard, and Vassilina Nikoulina. 2021. [Multilingual domain adaptation for NMT: Decoupling language and domain information with adapters](#). In *Proceedings of the Sixth Conference on Machine Translation*, pages 578–598, Online. Association for Computational Linguistics.
- Anna Currey and Kenneth Heafield. 2019. [Zero-resource neural machine translation with monolingual pivot data](#). In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 99–107, Hong Kong. Association for Computational Linguistics.
- Raj Dabre and Sadao Kurohashi. 2017. [MMCR4NLP: multilingual multiway corpora repository for natural language processing](#). CoRR, abs/1710.01025.
- Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. [Transformer-XL: Attentive language models beyond a fixed-length context](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, Florence, Italy. Association for Computational Linguistics.
- Praveen Dakwale and Christof Monz. 2017. [Fine-tuning for neural machine translation with limited degradation across in- and out-of-domain data](#). In *Proceedings of Machine Translation Summit XVI: Research Track*, pages 156–169, Nagoya Japan.
- Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. [Plug and play language models: A simple approach to controlled text generation](#). In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.
- Yue Deng, Wenxuan Zhang, Sinno Jialin Pan, and Lidong Bing. 2024. [Multilingual jailbreak challenges in large language models](#). In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

- Prafulla Dhariwal and Alexander Quinn Nichol. 2021. [Diffusion models beat gans on image synthesis](#). In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 8780–8794.
- Daxiang Dong, Hua Wu, Wei He, Dianhai Yu, and Haifeng Wang. 2015. [Multi-task learning for multiple language translation](#). In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1723–1732, Beijing, China. Association for Computational Linguistics.
- Paul-Ambroise Duquenne, Hongyu Gong, Ning Dong, Jingfei Du, Ann Lee, Vedanuj Goswami, Changhan Wang, Juan Pino, Benoît Sagot, and Holger Schwenk. 2023a. [SpeechMatrix: A large-scale mined corpus of multilingual speech-to-speech translations](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16251–16269, Toronto, Canada. Association for Computational Linguistics.
- Paul-Ambroise Duquenne, Hongyu Gong, Benoît Sagot, and Holger Schwenk. 2022. [T-modules: Translation modules for zero-shot cross-modal machine translation](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5794–5806, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Paul-Ambroise Duquenne, Holger Schwenk, and Benoît Sagot. 2023b. [SONAR: sentence-level multimodal and language-agnostic representations](#). *CoRR*, abs/2308.11466.
- David Eigen, Marc’Aurelio Ranzato, and Ilya Sutskever. 2014. [Learning factored representations in a deep mixture of experts](#). In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Workshop Track Proceedings*.
- Ahmed El-Kishky, Vishrav Chaudhary, Francisco Guzmán, and Philipp Koehn. 2020. [CCAligned: A massive collection of cross-lingual web-document pairs](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5960–5969, Online. Association for Computational Linguistics.
- Muhammad ElNokrashy, Amr Hendy, Mohamed Maher, Mohamed Afify, and Hany Hassan Awadalla. 2022. [Language tokens: A frustratingly simple approach improves zero-shot performance of multilingual translation](#). In *Proceedings of the 15th biennial conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 70–82, Orlando, USA. Association for Machine Translation in the Americas.
- Taisei Enomoto, Hwichan Kim, Zhousi Chen, and Mamoru Komachi. 2025. [A fair comparison without translationese: English vs. target-language instructions for multilingual LLMs](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the*

Bibliography

- Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 649–670, Albuquerque, New Mexico. Association for Computational Linguistics.
- Johannes Eschbach-Dymanus, Frank Essenberger, Bianka Buschbeck, and Miriam Exel. 2024. [Exploring the effectiveness of LLM domain adaptation for business IT machine translation](#). In *Proceedings of the 25th Annual Conference of the European Association for Machine Translation (Volume 1)*, pages 610–622, Sheffield, UK. European Association for Machine Translation (EAMT).
- Carlos Escolano, Marta R. Costa-jussà, José A. R. Fonollosa, and Mikel Artetxe. 2020. [Training multilingual machine translation by alternately freezing language-specific encoders-decoders](#). *CoRR*, abs/2006.01594.
- Carlos Escolano, Marta R. Costa-Jussà, and José A. R. Fonollosa. 2021. [From bilingual to multilingual neural-based machine translation by incremental training](#). *Journal of the Association for Information Science and Technology*, 72(2):190–203.
- Ethnologue. 2025. [Languages of the world](#). Accessed: 2025-09-13.
- Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Michael Auli, and Armand Joulin. 2021. [Beyond english-centric multilingual machine translation](#). *J. Mach. Learn. Res.*, 22:107:1–107:48.
- Weston Feely, Eva Hasler, and Adrià de Gispert. 2019. [Controlling Japanese honorifics in English-to-Japanese neural machine translation](#). In *Proceedings of the 6th Workshop on Asian Translation*, pages 45–53, Hong Kong, China. Association for Computational Linguistics.
- Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. [Language-agnostic BERT sentence embedding](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.
- Patrick Fernandes, Daniel Deutsch, Mara Finkelstein, Parker Riley, André Martins, Graham Neubig, Ankush Garg, Jonathan Clark, Markus Freitag, and Orhan Firat. 2023. [The devil is in the errors: Leveraging large language models for fine-grained machine translation evaluation](#). In *Proceedings of the Eighth Conference on Machine Translation*, pages 1066–1083, Singapore. Association for Computational Linguistics.
- Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. [Multi-way, multilingual neural machine translation with a shared attention mechanism](#). In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 866–875, San Diego, California. Association for Computational Linguistics.

- Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, Swetha Ranganath, Laurie Crist, Misha Britan, Wouter Leeuwis, Gokhan Tur, and Prem Natarajan. 2023. [MASSIVE: A 1M-example multilingual natural language understanding dataset with 51 typologically-diverse languages](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4277–4302, Toronto, Canada. Association for Computational Linguistics.
- Markus Freitag and Yaser Al-Onaizan. 2016. [Fast domain adaptation for neural machine translation](#). *CoRR*, abs/1612.06897.
- Markus Freitag and Orhan Firat. 2020. [Complete multilingual neural machine translation](#). In *Proceedings of the Fifth Conference on Machine Translation*, pages 550–560, Online. Association for Computational Linguistics.
- Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F. T. Martins. 2022. [Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust](#). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 46–68, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Kunihiko Fukushima. 1975. [Cognitron: A self-organizing multilayered neural network](#). *Biol. Cybern.*, 20(3–4):121–136.
- Yarin Gal and Zoubin Ghahramani. 2016. [A theoretically grounded application of dropout in recurrent neural networks](#). In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 1019–1027.
- Yaroslav Ganin and Victor S. Lempitsky. 2015. [Unsupervised domain adaptation by backpropagation](#). In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1180–1189. JMLR.org.
- Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. [SimCSE: Simple contrastive learning of sentence embeddings](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Xavier Garcia, Yamini Bansal, Colin Cherry, George F. Foster, Maxim Krikun, Melvin Johnson, and Orhan Firat. 2023. [The unreasonable effectiveness of few-shot learning for machine translation](#). In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 10867–10878. PMLR.

Bibliography

- Xavier Garcia, Noah Constant, Ankur Parikh, and Orhan Firat. 2021. [Towards continual learning for multilingual machine translation via vocabulary substitution](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1184–1192, Online. Association for Computational Linguistics.
- Neeraj Gaur, Brian Farris, Parisa Haghani, Isabel Leal, Pedro J. Moreno, Manasa Prasad, Bhuvana Ramabhadran, and Yun Zhu. 2021. [Mixture of informed experts for multilingual speech recognition](#). In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2021, Toronto, ON, Canada, June 6-11, 2021*, pages 6234–6238. IEEE.
- Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. 2023. [Imagebind one embedding space to bind them all](#). In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 15180–15190. IEEE.
- Hongyu Gong, Vishrav Chaudhary, Yuqing Tang, and Francisco Guzmán. 2021. [LAWDR: language-agnostic weighted document representations from pre-trained models](#). *CoRR*, abs/2106.03379.
- Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*, chapter 15. Representation Learning. MIT Press.
- Edward Gow-Smith and Danae Sánchez Villegas. 2023. [Sheffield’s submission to the AmericasNLP shared task on machine translation into indigenous languages](#). In *Proceedings of the Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP)*, pages 192–199, Toronto, Canada. Association for Computational Linguistics.
- Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc’Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. [The Flores-101 evaluation benchmark for low-resource and multilingual machine translation](#). *Transactions of the Association for Computational Linguistics*, 10:522–538.
- Milan Gritta and Ignacio Iacobacci. 2021. [XeroAlign: Zero-shot cross-lingual transformer alignment](#). In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 371–381, Online. Association for Computational Linguistics.
- Jiatao Gu, Yong Wang, Kyunghyun Cho, and Victor O.K. Li. 2019. [Improved zero-shot neural machine translation via ignoring spurious correlations](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1258–1268, Florence, Italy. Association for Computational Linguistics.
- Shuhao Gu and Yang Feng. 2020. [Investigating catastrophic forgetting during continual training for neural machine translation](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4315–4326, Barcelona, Spain (Online). International Committee on Computational Linguistics.

- Tianrui Gu, Kaie Chen, Siqi Ouyang, and Lei Li. 2023. [PlayGround low resource machine translation system for the 2023 AmericasNLP shared task](#). In *Proceedings of the Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP)*, pages 173–176, Toronto, Canada. Association for Computational Linguistics.
- Nuno M. Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André F. T. Martins. 2024. [xcomet: Transparent machine translation evaluation through fine-grained error detection](#). *Transactions of the Association for Computational Linguistics*, 12:979–995.
- Thanh-Le Ha, Jan Niehues, and Alex Waibel. 2016. [Toward multilingual neural machine translation with universal encoder and decoder](#). In *Proceedings of the 13th International Conference on Spoken Language Translation*, Seattle, Washington D.C. International Workshop on Spoken Language Translation.
- Thanh-Le Ha, Jan Niehues, and Alexander Waibel. 2017. [Effective strategies in zero-shot neural machine translation](#). In *Proceedings of the 14th International Conference on Spoken Language Translation*, pages 105–112, Tokyo, Japan. International Workshop on Spoken Language Translation.
- Barry Haddow and Faheem Kirefu. 2020. [Pmindia - A collection of parallel corpora of languages of india](#). *CoRR*, abs/2001.09907.
- Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. [Dimensionality reduction by learning an invariant mapping](#). In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006)*, 17-22 June 2006, New York, NY, USA, pages 1735–1742. IEEE Computer Society.
- Katharina Hämmerl, Jindřich Libovický, and Alexander Fraser. 2024. [Understanding cross-lingual Alignment—A survey](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10922–10943, Bangkok, Thailand. Association for Computational Linguistics.
- Viktor Hangya and Alexander Fraser. 2019. [Unsupervised parallel sentence extraction with parallel segment detection helps machine translation](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1224–1234, Florence, Italy. Association for Computational Linguistics.
- Dan He, Minh-Quang Pham, Thanh-Le Ha, and Marco Turchi. 2023. [Gradient-based gradual pruning for language-specific multilingual neural machine translation](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 654–670, Singapore. Association for Computational Linguistics.
- Kaiming He, Ross B. Girshick, and Piotr Dollár. 2019. [Rethinking imagenet pre-training](#). In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 4917–4926. IEEE.

Bibliography

- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. [Deep residual learning for image recognition](#). In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society.
- Mutian He and Philip N. Garner. 2023. [Can chatgpt detect intent? evaluating large language models for spoken language understanding](#). In *24th Annual Conference of the International Speech Communication Association, Interspeech 2023, Dublin, Ireland, August 20-24, 2023*, pages 1109–1113. ISCA.
- Kevin Heffernan, Onur Çelebi, and Holger Schwenk. 2022. [Bitext mining using distilled sentence representations for low-resource languages](#). In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2101–2112, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Dan Hendrycks, Kimin Lee, and Mantas Mazeika. 2019. [Using pre-training can improve model robustness and uncertainty](#). In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 2712–2721. PMLR.
- Geoffrey Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, and Brian Kingsbury. 2012. [Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups](#). *IEEE Signal Processing Magazine*, 29(6):82–97.
- Sepp Hochreiter. 1998. [The vanishing gradient problem during learning recurrent neural nets and problem solutions](#). *Int. J. Uncertain. Fuzziness Knowl. Based Syst.*, 6(2):107–116.
- Jimin Hong, Gibbeum Lee, and Jaewoong Cho. 2024. [Accelerating multilingual language model for excessively tokenized languages](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11095–11111, Bangkok, Thailand. Association for Computational Linguistics.
- Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Larous-silhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. [Parameter-efficient transfer learning for NLP](#). In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. [Lora: Low-rank adaptation of large language models](#). In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.
- Junjie Hu, Melvin Johnson, Orhan Firat, Aditya Siddhant, and Graham Neubig. 2021. [Explicit alignment objectives for multilingual bidirectional encoders](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational*

- Linguistics: Human Language Technologies*, pages 3633–3643, Online. Association for Computational Linguistics.
- Ke Hu, Bo Li, Tara Sainath, Yu Zhang, and Françoise Beaufays. 2023. [Mixture-of-Expert Conformer for Streaming Multilingual ASR](#). In *Proc. INTERSPEECH 2023*, pages 3327–3331.
- Tianxiang Hu, Pei Zhang, Baosong Yang, Jun Xie, Derek F. Wong, and Rui Wang. 2024. [Large language model for multi-domain translation: Benchmarking and domain CoT fine-tuning](#). In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 5726–5746, Miami, Florida, USA. Association for Computational Linguistics.
- Dieuwke Hupkes, Mario Giulianelli, Verna Dankers, Mikel Artetxe, Yanai Elazar, Tiago Pimentel, Christos E. Christodoulopoulos, Karim Lasri, Naomi Saphra, Arabella Sinclair, Dennis Ulmer, Florian Schottmann, Khuyagbaatar Batsuren, Kaiser Sun, Koustuv Sinha, Leila Khalatbari, Maria Ryskina, Rita Frieske, Ryan Cotterell, and Zhijing Jin. 2023. [A taxonomy and review of generalization research in NLP](#). *Nat. Mac. Intell.*, 5(10):1161–1174.
- Gabriel Ilharco, Marco Túlio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. [Editing models with task arithmetic](#). In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.
- Vivek Iyer, Bhavitvya Malik, Wenhao Zhu, Pavel Stepachev, Pinzhen Chen, Barry Haddow, and Alexandra Birch. 2024. [Exploring very low-resource translation with LLMs: The University of Edinburgh’s submission to AmericasNLP 2024 translation task](#). In *Proceedings of the 4th Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP 2024)*, pages 209–220, Mexico City, Mexico. Association for Computational Linguistics.
- Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. [Unsupervised dense information retrieval with contrastive learning](#). *Trans. Mach. Learn. Res.*, 2022.
- João Maria Janeiro, Benjamin Piwowarski, Patrick Gallinari, and Loic Barrault. 2025. [MEXMA: Token-level objectives improve sentence representations](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 23960–23995, Vienna, Austria. Association for Computational Linguistics.
- Baijun Ji, Zhirui Zhang, Xiangyu Duan, Min Zhang, Boxing Chen, and Weihua Luo. 2020. [Cross-lingual pre-training based transfer for zero-shot neural machine translation](#). In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 115–122. AAAI Press.

Bibliography

- Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. [Google’s multilingual neural machine translation system: Enabling zero-shot translation](#). *Transactions of the Association for Computational Linguistics*, 5:339–351.
- Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. [The state and fate of linguistic diversity and inclusion in the NLP world](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.
- Lukasz Kaiser, Samy Bengio, Aurko Roy, Ashish Vaswani, Niki Parmar, Jakob Uszkoreit, and Noam Shazeer. 2018. [Fast decoding in sequence models using discrete latent variables](#). In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2390–2399. PMLR.
- Damjan Kalajdzievski. 2024. [Scaling laws for forgetting when fine-tuning large language models](#). *CoRR*, abs/2401.05605.
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. [Scaling laws for neural language models](#). *CoRR*, abs/2001.08361.
- Amir Hossein Kargaran, Ali Modarressi, Nafiseh Nikeghbal, Jana Diesner, François Yvon, and Hinrich Schuetze. 2025. [MEXA: Multilingual evaluation of English-centric LLMs via cross-lingual alignment](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 27001–27023, Vienna, Austria. Association for Computational Linguistics.
- Nitish Shirish Keskar, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. 2019. [CTRL: A conditional transformer language model for controllable generation](#). *CoRR*, abs/1909.05858.
- Zae Myung Kim, Laurent Besacier, Vassilina Nikoulina, and Didier Schwab. 2021. [Do multilingual neural machine translation models contain language pair specific attention heads?](#) In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2832–2841, Online. Association for Computational Linguistics.
- Diederik P. Kingma and Jimmy Ba. 2015. [Adam: A method for stochastic optimization](#). In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.
- James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. 2017. [Overcoming catastrophic forgetting in neural networks](#). *Proceedings of the National Academy of Sciences*, 114(13):3521–3526.

- Rebecca Knowles and Samuel Larkin. 2021. [NRC-CNRC systems for Upper Sorbian-German and Lower Sorbian-German machine translation 2021](#). In *Proceedings of the Sixth Conference on Machine Translation*, pages 999–1008, Online. Association for Computational Linguistics.
- Catherine Kobus, Josep Maria Crego, and Jean Senellart. 2017. [Domain control for neural machine translation](#). In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017, Varna, Bulgaria, September 2 - 8, 2017*, pages 372–378. INCOMA Ltd.
- Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena Karpinska, Philipp Koehn, Benjamin Marie, Christof Monz, Kenton Murray, Masaaki Nagata, Martin Popel, Maja Popović, and 3 others. 2024a. [Findings of the WMT24 general machine translation shared task: The LLM era is here but MT is not solved yet](#). In *Proceedings of the Ninth Conference on Machine Translation*, pages 1–46, Miami, Florida, USA. Association for Computational Linguistics.
- Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Philipp Koehn, Benjamin Marie, Christof Monz, Makoto Morishita, Kenton Murray, Makoto Nagata, Toshiaki Nakazawa, Martin Popel, and 2 others. 2023. [Findings of the 2023 conference on machine translation \(WMT23\): LLMs are here but not quite there yet](#). In *Proceedings of the Eighth Conference on Machine Translation*, pages 1–42, Singapore. Association for Computational Linguistics.
- Tom Kocmi, Vilém Zouhar, Christian Federmann, and Matt Post. 2024b. [Navigating the metrics maze: Reconciling score magnitudes and accuracies](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1999–2014, Bangkok, Thailand. Association for Computational Linguistics.
- Philipp Koehn. 2005. [Europarl: A parallel corpus for statistical machine translation](#). In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand.
- Yeskendir Koishekenov, Alexandre Berard, and Vassilina Nikoulina. 2023. [Memory-efficient NLLB-200: Language-specific expert pruning of a massively multilingual machine translation model](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3567–3585, Toronto, Canada. Association for Computational Linguistics.
- Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey E. Hinton. 2019. [Similarity of neural network representations revisited](#). In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 3519–3529. PMLR.

Bibliography

- Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq Joty, Richard Socher, and Nazneen Fatema Rajani. 2021. [GeDi: Generative discriminator guided sequence generation](#). In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4929–4952, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Ortiz Suarez, and 33 others. 2022. [Quality at a glance: An audit of web-crawled multilingual datasets](#). *Transactions of the Association for Computational Linguistics*, 10:50–72.
- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. [Imagenet classification with deep convolutional neural networks](#). In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, pages 1106–1114.
- Taku Kudo and John Richardson. 2018. [SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.
- Sneha Kudugunta, Ankur Bapna, Isaac Caswell, and Orhan Firat. 2019. [Investigating multilingual NMT representations at scale](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1565–1575, Hong Kong, China. Association for Computational Linguistics.
- Yoohwan Kwon and Soo-Whan Chung. 2023. [Mole : Mixture of language experts for multi-lingual automatic speech recognition](#). In *IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2023, Rhodes Island, Greece, June 4-10, 2023*, pages 1–5. IEEE.
- Sonia Laguna, Ricards Marcinkevics, Moritz Vandenhirz, and Julia E. Vogt. 2024. [Beyond concept bottleneck models: How to make black boxes intervenable?](#) In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.
- Wen Lai, Viktor Hangya, and Alexander Fraser. 2023. [Extending multilingual machine translation through imitation learning](#). *CoRR*, abs/2311.08538.
- Wen Lai, Viktor Hangya, and Alexander Fraser. 2024a. [Style-specific neurons for steering LLMs in text style transfer](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 13427–13443, Miami, Florida, USA. Association for Computational Linguistics.

- Wen Lai, Mohsen Mesgar, and Alexander Fraser. 2024b. [LLMs beyond English: Scaling the multilingual capability of LLMs with cross-lingual feedback](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 8186–8213, Bangkok, Thailand. Association for Computational Linguistics.
- Surafel Melaku Lakew, Mattia Di Gangi, and Marcello Federico. 2019a. [Controlling the output length of neural machine translation](#). In *Proceedings of the 16th International Conference on Spoken Language Translation*, Hong Kong. Association for Computational Linguistics.
- Surafel Melaku Lakew, Marcello Federico, Matteo Negri, and Marco Turchi. 2019b. [Multilingual neural machine translation for zero-resource languages](#). *CoRR*, abs/1909.07342.
- Guillaume Lample, Alexis Conneau, Marc’Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. [Word translation without parallel data](#). In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.
- David Landsman, Jerry Zikun Chen, and Hussain Zaidi. 2022. [BeamR: Beam reweighing with attribute discriminators for controllable text generation](#). In *Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022*, pages 422–437, Online only. Association for Computational Linguistics.
- Angeliki Lazaridou, Adhiguna Kuncoro, Elena Gribovskaya, Devang Agrawal, Adam Liska, Tayfun Terzi, Mai Gimenez, Cyprien de Masson d’Autume, Tomás Kociský, Sebastian Ruder, Dani Yogatama, Kris Cao, Susannah Young, and Phil Blunsom. 2021. [Mind the gap: Assessing temporal generalization in neural language models](#). In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 29348–29363.
- Hang Le, Juan Pino, Chaghan Wang, Jiatao Gu, Didier Schwab, and Laurent Besacier. 2021. [Lightweight adapter tuning for multilingual speech translation](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 817–824, Online. Association for Computational Linguistics.
- Hyunji Lee, Danni Liu, Supriti Sinhamahapatra, and Jan Niehues. 2025. [How do multimodal foundation models encode text and speech? an analysis of cross-lingual and cross-modal representations](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 600–610, Albuquerque, New Mexico. Association for Computational Linguistics.
- Seungjun Lee, Hyeonseok Moon, Chanjun Park, and Heuseok Lim. 2023. [Improving formality-sensitive machine translation using data-centric approaches and prompt engineering](#). In *Proceedings of the 20th International Conference on Spoken Language Translation*

Bibliography

- (*IWSLT 2023*), pages 420–432, Toronto, Canada (in-person and online). Association for Computational Linguistics.
- Dmitry Lepikhin, Hyoungho Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. 2021. [Gshard: Scaling giant models with conditional computation and automatic sharding](#). In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.
- Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. [The power of scale for parameter-efficient prompt tuning](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Daoyang Li, Haiyan Zhao, Qingcheng Zeng, and Mengnan Du. 2025a. [Exploring multilingual probing in large language models: A cross-language analysis](#). *Preprint*, arXiv:2409.14459.
- Kenneth Li, Oam Patel, Fernanda B. Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023a. [Inference-time intervention: Eliciting truthful answers from a language model](#). In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.
- Tianjian Li and Kenton Murray. 2023. [Why does zero-shot cross-lingual generation fail? an explanation and a solution](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12461–12476, Toronto, Canada. Association for Computational Linguistics.
- Xiang Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B. Hashimoto. 2022. [Diffusion-lm improves controllable text generation](#). In *NeurIPS*.
- Yichen Li, Zhiting Fan, Ruizhe Chen, Xiaotang Gai, Luqi Gong, Yan Zhang, and Zuozhu Liu. 2025b. [FairSteer: Inference time debiasing for LLMs with dynamic activation steering](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 11293–11312, Vienna, Austria. Association for Computational Linguistics.
- Ziheng Li, Shaohan Huang, Zihan Zhang, Zhi-Hong Deng, Qiang Lou, Haizhen Huang, Jian Jiao, Furu Wei, Weiwei Deng, and Qi Zhang. 2023b. [Dual-alignment pre-training for cross-lingual sentence embedding](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3466–3478, Toronto, Canada. Association for Computational Linguistics.
- Jindřich Libovický and Alexander Fraser. 2021a. [Findings of the WMT 2021 shared tasks in unsupervised MT and very low resource supervised MT](#). In *Proceedings of the Sixth Conference on Machine Translation*, pages 726–732, Online. Association for Computational Linguistics.

- Jindřich Libovický and Alexander Fraser. 2021b. [The LMU Munich systems for the WMT21 unsupervised and very low-resource translation task](#). In *Proceedings of the Sixth Conference on Machine Translation*, pages 989–994, Online. Association for Computational Linguistics.
- Jindřich Libovický, Rudolf Rosa, and Alexander Fraser. 2020. [On the language neutrality of pre-trained multilingual representations](#). In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1663–1674, Online. Association for Computational Linguistics.
- Zehui Lin, Liwei Wu, Mingxuan Wang, and Lei Li. 2021. [Learning language specific sub-network for multilingual machine translation](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 293–305, Online. Association for Computational Linguistics.
- Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. 2021a. [DExperts: Decoding-time controlled text generation with experts and anti-experts](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6691–6706, Online. Association for Computational Linguistics.
- Chaoqun Liu, Wenxuan Zhang, Yiran Zhao, Anh Tuan Luu, and Lidong Bing. 2025. [Is translation all you need? a study on solving multilingual tasks with large language models](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 9594–9614, Albuquerque, New Mexico. Association for Computational Linguistics.
- Danni Liu and Jan Niehues. 2021. [Maastricht university’s multilingual speech translation system for IWSLT 2021](#). In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 138–143, Bangkok, Thailand (online). Association for Computational Linguistics.
- Danni Liu and Jan Niehues. 2022. [Learning an artificial language for knowledge-sharing in multilingual translation](#). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 188–202, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Danni Liu and Jan Niehues. 2024a. [How transferable are attribute controllers on pretrained multilingual translation models?](#) In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 334–348, St. Julian’s, Malta. Association for Computational Linguistics.

Bibliography

- Danni Liu and Jan Niehues. 2024b. [Recent highlights in multilingual and multimodal speech translation](#). In *Proceedings of the 21st International Conference on Spoken Language Translation (IWSLT 2024)*, pages 235–253, Bangkok, Thailand (in-person and online). Association for Computational Linguistics.
- Danni Liu and Jan Niehues. 2025a. Conditions for catastrophic forgetting in multilingual translation. In *Proceedings of the Fifth Workshop on Multilingual Representation Learning (MRL 2025)*, Suzhou, China. Association for Computational Linguistics.
- Danni Liu and Jan Niehues. 2025b. [Middle-layer representation alignment for cross-lingual transfer in fine-tuned LLMs](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15979–15996, Vienna, Austria. Association for Computational Linguistics.
- Danni Liu, Jan Niehues, James Cross, Francisco Guzmán, and Xian Li. 2021b. [Improving zero-shot translation by disentangling positional information](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1259–1273, Online. Association for Computational Linguistics.
- Junpeng Liu, Kaiyu Huang, Hao Yu, Jiuyi Li, Jinsong Su, and Degen Huang. 2023. [Continual learning for multilingual neural machine translation via dual importance-based model division](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12011–12027, Singapore. Association for Computational Linguistics.
- Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. [Multilingual denoising pre-training for neural machine translation](#). *Transactions of the Association for Computational Linguistics*, 8:726–742.
- Yichao Lu, Phillip Keung, Faisal Ladhak, Vikas Bhardwaj, Shaonan Zhang, and Jason Sun. 2018. [A neural interlingua for multilingual machine translation](#). In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 84–92, Brussels, Belgium. Association for Computational Linguistics.
- Yizhou Lu, Mingkun Huang, Xinghua Qu, Pengfei Wei, and Zejun Ma. 2022. [Language adaptive cross-lingual speech representation learning with sparse sharing sub-networks](#). In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022, Virtual and Singapore, 23-27 May 2022*, pages 6882–6886. IEEE.
- Marco Lui and Timothy Baldwin. 2011. [Cross-domain feature selection for language identification](#). In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 553–561, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

- Samin Mahdizadeh Sani, Pouya Sadeghi, Thuy-Trang Vu, Yadollah Yaghoobzadeh, and Gholamreza Haffari. 2025. [Extending LLMs to new languages: A case study of llama and Persian adaptation](#). In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 8868–8884, Abu Dhabi, UAE. Association for Computational Linguistics.
- Zhuoyuan Mao and Yen Yu. 2024. [Tuning LLMs with contrastive alignment instructions for machine translation in unseen, low-resource languages](#). In *Proceedings of the Seventh Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2024)*, pages 1–25, Bangkok, Thailand. Association for Computational Linguistics.
- Kelly Marchisio, Jialiang Guo, Cheng-I Lai, and Philipp Koehn. 2019. [Controlling the reading level of machine translation output](#). In *Proceedings of Machine Translation Summit XVII: Research Track*, pages 193–203, Dublin, Ireland. European Association for Machine Translation.
- Kelly Marchisio, Wei-Yin Ko, Alexandre Berard, Théo Dehaze, and Sebastian Ruder. 2024. [Understanding and mitigating language confusion in LLMs](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6653–6677, Miami, Florida, USA. Association for Computational Linguistics.
- Michael Matena and Colin Raffel. 2022. [Merging models with fisher-weighted averaging](#). In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.
- Stephen Mayhew, Terra Blevins, Shuheng Liu, Marek Suppa, Hila Gonen, Joseph Marvin Imperial, Börje Karlsson, Peiqin Lin, Nikola Ljubešić, Lester James Miranda, Barbara Plank, Arij Riabi, and Yuval Pinter. 2024. [Universal NER: A gold-standard multilingual named entity recognition benchmark](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4322–4337, Mexico City, Mexico. Association for Computational Linguistics.
- Michael McCloskey and Neal J. Cohen. 1989. [Catastrophic interference in connectionist networks: The sequential learning problem](#). volume 24 of *Psychology of Learning and Motivation*, pages 109–165. Academic Press.
- Yan Meng and Christof Monz. 2024. [Disentangling the roles of target-side transfer and regularization in multilingual machine translation](#). In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1828–1840, St. Julian’s, Malta. Association for Computational Linguistics.
- Paul Michel and Graham Neubig. 2018. [Extreme adaptation for personalized neural machine translation](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 312–318, Melbourne, Australia. Association for Computational Linguistics.

Bibliography

- Tomás Mikolov, Quoc V. Le, and Ilya Sutskever. 2013. [Exploiting similarities among languages for machine translation](#). *CoRR*, abs/1309.4168.
- Itai Mondshine, Tzuf Paz-Argaman, and Reut Tsarfaty. 2025. [Beyond English: The impact of prompt translation strategies across languages and tasks in multilingual LLMs](#). In *Proceedings of the Eighth Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2025)*, pages 81–104, Albuquerque, New Mexico, U.S.A. Association for Computational Linguistics.
- Aaron Mueller, Garrett Nicolai, Arya D. McCarthy, Dylan Lewis, Winston Wu, and David Yarowsky. 2020. [An analysis of massively multilingual neural machine translation for low-resource languages](#). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3710–3718, Marseille, France. European Language Resources Association.
- Benjamin Muller, Yanai Elazar, Benoît Sagot, and Djamé Seddah. 2021. [First align, then predict: Understanding the cross-lingual ability of multilingual BERT](#). In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2214–2231, Online. Association for Computational Linguistics.
- Carlos Mullov, Quan Pham, and Alexander Waibel. 2024. [Decoupled vocabulary learning enables zero-shot translation from unseen languages](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6693–6709, Bangkok, Thailand. Association for Computational Linguistics.
- Maria Nadejde, Anna Currey, Benjamin Hsu, Xing Niu, Marcello Federico, and Georgiana Dinu. 2022. [CoCoA-MT: A dataset and benchmark for contrastive controlled MT with application to formality](#). In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 616–632, Seattle, United States. Association for Computational Linguistics.
- Jan Niehues. 2020. [Machine translation with unsupervised length-constraints](#). In *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 21–35, Virtual. Association for Machine Translation in the Americas.
- Xing Niu, Sudha Rao, and Marine Carpuat. 2018. [Multi-task neural models for translating between styles within and across languages](#). In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1008–1021, Santa Fe, New Mexico, USA. Association for Computational Linguistics.
- NLLB Team. 2024. [Scaling neural machine translation to 200 languages](#). *Nat.*, 630(8018):841–846.
- Mohammad Norouzi, Tomás Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg Corrado, and Jeffrey Dean. 2014. [Zero-shot learning by convex](#)

- combination of semantic embeddings. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.
- Franz Josef Och. 2005. [Statistical machine translation: Foundations and recent advances](#). In *Proceedings of Machine Translation Summit X: Tutorial notes*, Phuket, Thailand.
- Shu Okabe, Katharina Hämmerl, and Alexander Fraser. 2025. [Improving parallel sentence mining for low-resource and endangered languages](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 196–205, Vienna, Austria. Association for Computational Linguistics.
- Shereen Oraby, Lena Reed, Shubhangi Tandon, Sharath T.S., Stephanie Lukin, and Marilyn Walker. 2018. [Controlling personality-based stylistic variation with neural natural language generators](#). In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 180–190, Melbourne, Australia. Association for Computational Linguistics.
- Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. [fairseq: A fast, extensible toolkit for sequence modeling](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. [Training language models to follow instructions with human feedback](#). In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.
- Siqi Ouyang, Rong Ye, and Lei Li. 2023. [WACO: Word-aligned contrastive learning for speech translation](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3891–3907, Toronto, Canada. Association for Computational Linguistics.
- Sinno Jialin Pan and Qiang Yang. 2010. [A survey on transfer learning](#). *IEEE Trans. Knowl. Data Eng.*, 22(10):1345–1359.
- Xiao Pan, Mingxuan Wang, Liwei Wu, and Lei Li. 2021. [Contrastive learning for many-to-many multilingual neural machine translation](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 244–258, Online. Association for Computational Linguistics.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. [Bleu: a method for automatic evaluation of machine translation](#). In *Proceedings of the 40th Annual Meeting of*

Bibliography

- the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.
- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. [Deep contextualized word representations](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.
- Aleksandar Petrov, Emanuele La Malfa, Philip H. S. Torr, and Adel Bibi. 2023. [Language model tokenizers introduce unfairness between languages](#). In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.
- Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. [MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673, Online. Association for Computational Linguistics.
- Ngoc-Quan Pham, Tuan Nam Nguyen, Thanh-Le Ha, Sebastian Stüker, Alexander Waibel, and Dan He. 2021a. [Multilingual speech translation KIT @ IWSLT2021](#). In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 154–159, Bangkok, Thailand (online). Association for Computational Linguistics.
- Ngoc-Quan Pham, Tuan-Nam Nguyen, Sebastian Stüker, and Alex Waibel. 2021b. [Efficient weight factorization for multilingual speech recognition](#). In *Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August - 3 September 2021*, pages 2421–2425. ISCA.
- Ngoc-Quan Pham, Jan Niehues, Thanh-Le Ha, and Alexander Waibel. 2019. [Improving zero-shot translation with language-independent constraints](#). In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 13–23, Florence, Italy. Association for Computational Linguistics.
- Jerin Philip, Alexandre Berard, Matthias Gallé, and Laurent Besacier. 2020. [Monolingual adapters for zero-shot neural machine translation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4465–4470, Online. Association for Computational Linguistics.
- Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. [How multilingual is multilingual BERT?](#) In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.
- Telmo Pires, Robin Schmidt, Yi-Hsiu Liao, and Stephan Peitz. 2023. [Learning language-specific layers for multilingual machine translation](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14767–14783, Toronto, Canada. Association for Computational Linguistics.

- Emmanouil Antonios Platanios, Mrinmaya Sachan, Graham Neubig, and Tom Mitchell. 2018. [Contextual parameter generation for universal neural machine translation](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 425–435, Brussels, Belgium. Association for Computational Linguistics.
- Maja Popović. 2015. [chrF: character n-gram F-score for automatic MT evaluation](#). In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.
- Matt Post. 2018. [A call for clarity in reporting BLEU scores](#). In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.
- Taido Purason and Andre Tättar. 2022. [Multilingual neural machine translation with the right amount of sharing](#). In *Proceedings of the 23rd Annual Conference of the European Association for Machine Translation*, pages 91–100, Ghent, Belgium. European Association for Machine Translation.
- Kunxun Qi, Hai Wan, Jianfeng Du, and Haolan Chen. 2022. [Enhancing cross-lingual natural language inference by prompt-learning from cross-lingual templates](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1910–1923, Dublin, Ireland. Association for Computational Linguistics.
- Zhi Qu, Chenchen Ding, and Taro Watanabe. 2024. [Languages transferred within the encoder: On representation transfer in zero-shot multilingual translation](#). *CoRR*, abs/2406.08092.
- Qwen Team, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiayi Yang, Jingren Zhou, Junyang Lin, and 24 others. 2025. [Qwen2.5 technical report](#). *Preprint*, arXiv:2412.15115.
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. [Learning transferable visual models from natural language supervision](#). In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.
- Alec Radford and Karthik Narasimhan. 2018. [Improving language understanding by generative pre-training](#).
- Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. [Language models are unsupervised multitask learners](#).
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. [Direct preference optimization: Your language model is secretly a](#)

Bibliography

- [reward model](#). In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.
- Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. 2017. [SVCCA: singular vector canonical correlation analysis for deep learning dynamics and interpretability](#). In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6076–6085.
- Vinay Venkatesh Ramasesh, Aitor Lewkowycz, and Ethan Dyer. 2022. [Effect of scale on catastrophic forgetting in neural networks](#). In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.
- Vikas Raunak, Roman Grundkiewicz, and Marcin Junczys-Dowmunt. 2024. [On instruction-finetuning neural machine translation models](#). In *Proceedings of the Ninth Conference on Machine Translation*, pages 1155–1166, Miami, Florida, USA. Association for Computational Linguistics.
- Anton Razzhigaev, Matvey Mikhalechuk, Elizaveta Goncharova, Ivan Oseledets, Denis Dimitrov, and Andrey Kuznetsov. 2024. [The shape of learning: Anisotropy and intrinsic dimensions in transformer-based models](#). In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 868–874, St. Julian’s, Malta. Association for Computational Linguistics.
- Sylvestre-Alvise Rebuffi, Hakan Bilen, and Andrea Vedaldi. 2017. [Learning multiple visual domains with residual adapters](#). In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 506–516.
- Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022. [COMET-22: Unbabel-IST 2022 submission for the metrics shared task](#). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 578–585, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. [COMET: A neural framework for MT evaluation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.
- Aquia Richburg and Marine Carpuat. 2024. [How multilingual are large language models fine-tuned for translation?](#) In *First Conference on Language Modeling*.
- Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. 2024. [Steering llama 2 via contrastive activation addition](#). In *Proceedings of the*

- 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 15504–15522, Bangkok, Thailand. Association for Computational Linguistics.
- Annette Rios, Mathias Müller, and Rico Sennrich. 2020. [Subword segmentation and a single bridge language affect zero-shot neural machine translation](#). In *Proceedings of the Fifth Conference on Machine Translation*, pages 528–537, Online. Association for Computational Linguistics.
- Elijah Rippeth, Sweta Agrawal, and Marine Carpuat. 2022. [Controlling translation formality using pre-trained multilingual language models](#). In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, pages 327–340, Dublin, Ireland (in-person and online). Association for Computational Linguistics.
- Anthony V. Robins. 1995. [Catastrophic forgetting, rehearsal and pseudorehearsal](#). *Connect. Sci.*, 7(2):123–146.
- Nathaniel Robinson, Perez Ogayo, David R. Mortensen, and Graham Neubig. 2023. [Chat-GPT MT: Competitive for high- \(but not low-\) resource languages](#). In *Proceedings of the Eighth Conference on Machine Translation*, pages 392–418, Singapore. Association for Computational Linguistics.
- Bernardino Romera-Paredes and Philip H. S. Torr. 2015. [An embarrassingly simple approach to zero-shot learning](#). In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 2152–2161. JMLR.org.
- Guy D. Rosin and Kira Radinsky. 2022. [Temporal attention for language models](#). In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1498–1508, Seattle, United States. Association for Computational Linguistics.
- Sebastian Ruder, Matthew E. Peters, Swabha Swayamdipta, and Thomas Wolf. 2019a. [Transfer learning in natural language processing](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*, pages 15–18, Minneapolis, Minnesota. Association for Computational Linguistics.
- Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2019b. [A survey of cross-lingual word embedding models](#). *J. Artif. Int. Res.*, 65(1):569–630.
- David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. 1986. [Learning representations by back-propagating errors](#). *Nature*, 323:533–536.
- Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. 2021. [How good is your tokenizer? on the monolingual performance of multilingual language models](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3118–3135, Online. Association for Computational Linguistics.

Bibliography

- Ashutosh Saboo and Timo Baumann. 2019. [Integration of dubbing constraints into machine translation](#). In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 94–101, Florence, Italy. Association for Computational Linguistics.
- Gabriele Sarti, Phu Mon Htut, Xing Niu, Benjamin Hsu, Anna Currey, Georgiana Dinu, and Maria Nadejde. 2023. [RAMP: Retrieval and attribute-marking enhanced prompting for attribute-controlled translation](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1476–1490, Toronto, Canada. Association for Computational Linguistics.
- Danielle Saunders and Steve DeNeefe. 2024. [Domain adapted machine translation: What does catastrophic forgetting forget and why?](#) In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12660–12671, Miami, Florida, USA. Association for Computational Linguistics.
- Danielle Saunders, Rosie Sallis, and Bill Byrne. 2020. [Neural machine translation doesn’t translate gender coreference right unless you make it](#). In *Proceedings of the Second Workshop on Gender Bias in Natural Language Processing*, pages 35–43, Barcelona, Spain (Online). Association for Computational Linguistics.
- Andrea Schioppa, David Vilar, Artem Sokolov, and Katja Filippova. 2021. [Controlling machine translation for multiple attributes with additive interventions](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6676–6696, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Jürgen Schmidhuber. 1990. Towards compositional learning with dynamic neural networks. Technical Report FKI-129-90, Institut für Informatik, Technische Universität München.
- Holger Schwenk. 2018. [Filtering and mining parallel data in a joint multilingual space](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 228–234, Melbourne, Australia. Association for Computational Linguistics.
- Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2021a. [WikiMatrix: Mining 135M parallel sentences in 1620 language pairs from Wikipedia](#). In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1351–1361, Online. Association for Computational Linguistics.
- Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, Armand Joulin, and Angela Fan. 2021b. [CCMatrix: Mining billions of high-quality parallel sentences on the web](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6490–6500, Online. Association for Computational Linguistics.

- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. [Controlling politeness in neural machine translation via side constraints](#). In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 35–40, San Diego, California. Association for Computational Linguistics.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. [Neural machine translation of rare words with subword units](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.
- Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. [Self-attention with relative position representations](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 464–468, New Orleans, Louisiana. Association for Computational Linguistics.
- Noam Shazeer. 2020. [GLU variants improve transformer](#). *CoRR*, abs/2002.05202.
- Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarczyk, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. 2017. [Outrageously large neural networks: The sparsely-gated mixture-of-experts layer](#). In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.
- Junhong Shen, Neil A. Tenenholz, James Brian Hall, David Alvarez-Melis, and Nicolò Fusi. 2024. [Tag-llm: Repurposing general-purpose llms for specialized domains](#). In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.
- Shivalika Singh, Freddie Vargus, Daniel D’souza, Börje Karlsson, Abinaya Mahendiran, Wei-Yin Ko, Herumb Shandilya, Jay Patel, Deividas Mataciunas, Laura O’Mahony, Mike Zhang, Ramith Hettiarachchi, Joseph Wilson, Marina Machado, Luisa Moura, Dominik Krzemiński, Hakimeh Fadaei, Irem Ergun, Ifeoma Okoh, and 14 others. 2024. [Aya dataset: An open-access collection for multilingual instruction tuning](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11521–11567, Bangkok, Thailand. Association for Computational Linguistics.
- Richard Socher, Milind Ganjoo, Christopher D. Manning, and Andrew Y. Ng. 2013. [Zero-shot learning through cross-modal transfer](#). In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 935–943.
- Vladimir Solovyev, Danni Liu, and Jan Niehues. 2024. [Language-independent representations improve zero-shot summarization](#). In *Proceedings of the 2024 Conference of the*

Bibliography

- North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 772–782, Mexico City, Mexico. Association for Computational Linguistics.
- Jianlin Su, Murtadha H. M. Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. [Roformer: Enhanced transformer with rotary position embedding](#). *Neurocomputing*, 568:127063.
- Zhaochen Su, Zecheng Tang, Xinyan Guan, Lijun Wu, Min Zhang, and Juntao Li. 2022. [Improving temporal generalization of pre-trained language models with lexical semantic change](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6380–6393, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. [How to fine-tune BERT for text classification?](#) In *Chinese Computational Linguistics - 18th China National Conference, CCL 2019, Kunming, China, October 18-20, 2019, Proceedings*, volume 11856 of *Lecture Notes in Computer Science*, pages 194–206. Springer.
- Simeng Sun, Maha Elbayad, Anna Sun, and James Cross. 2023. [Efficiently upgrading multilingual machine translation models to support more languages](#). In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1513–1527, Dubrovnik, Croatia. Association for Computational Linguistics.
- Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. [Sequence to sequence learning with neural networks](#). In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.
- Sho Takase and Naoaki Okazaki. 2019. [Positional encoding to control output sequence length](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3999–4004, Minneapolis, Minnesota. Association for Computational Linguistics.
- Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2021. [Multilingual translation from denoising pre-training](#). In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3450–3466, Online. Association for Computational Linguistics.
- Brian Thompson, Jeremy Gwinnup, Huda Khayrallah, Kevin Duh, and Philipp Koehn. 2019. [Overcoming catastrophic forgetting during domain adaptation of neural machine translation](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2062–2068, Minneapolis, Minnesota. Association for Computational Linguistics.

- Brian Thompson, Huda Khayrallah, Antonios Anastasopoulos, Arya D. McCarthy, Kevin Duh, Rebecca Marvin, Paul McNamee, Jeremy Gwinnup, Tim Anderson, and Philipp Koehn. 2018. [Freezing subnetworks to analyze domain adaptation in neural machine translation](#). In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 124–132, Brussels, Belgium. Association for Computational Linguistics.
- Jörg Tiedemann. 2012. [Parallel data, tools and interfaces in OPUS](#). In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).
- Jörg Tiedemann. 2020. [The tatoeba translation challenge – realistic data sets for low resource and multilingual MT](#). In *Proceedings of the Fifth Conference on Machine Translation*, pages 1174–1182, Online. Association for Computational Linguistics.
- Andros Tjandra, Sakriani Sakti, and Satoshi Nakamura. 2020. [Transformer VQ-VAE for unsupervised unit discovery and speech synthesis: Zerospeech 2020 challenge](#). In *21st Annual Conference of the International Speech Communication Association, Interspeech 2020, Virtual Event, Shanghai, China, October 25-29, 2020*, pages 4851–4855. ISCA.
- Antonio Torralba, Phillip Isola, and William T. Freeman. 2024. [Foundations of Computer Vision](#), chapter 30. Representation Learning. Adaptive Computation and Machine Learning series. MIT Press.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, and 49 others. 2023. [Llama 2: Open foundation and fine-tuned chat models](#). CoRR, abs/2307.09288.
- Priyesh Vakharia, Shree Vignesh S, and Pranjali Basmatkar. 2023. [Low-resource formality controlled NMT using pre-trained LM](#). In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*, pages 321–329, Toronto, Canada (in-person and online). Association for Computational Linguistics.
- Aäron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. 2017. [Neural discrete representation learning](#). In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6306–6315.
- Eva Vanmassenhove, Christian Hardmeier, and Andy Way. 2018. [Getting gender right in neural machine translation](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3003–3008, Brussels, Belgium. Association for Computational Linguistics.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In *Advances*

Bibliography

- in *Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.
- Raúl Vázquez, Alessandro Raganato, Jörg Tiedemann, and Mathias Creutz. 2019. [Multi-lingual NMT with a language-independent attention bridge](#). In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 33–39, Florence, Italy. Association for Computational Linguistics.
- Sebastian Vincent, Robert Flynn, and Carolina Scarton. 2023. [MTCue: Learning zero-shot control of extra-textual attributes by leveraging unstructured context in neural machine translation](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8210–8226, Toronto, Canada. Association for Computational Linguistics.
- Thuy-trang Vu, Shahram Khadivi, Xuanli He, Dinh Phung, and Gholamreza Haffari. 2022a. [Can domains be transferred across languages in multi-domain multilingual neural machine translation?](#) In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 381–396, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Tu Vu, Aditya Barua, Brian Lester, Daniel Cer, Mohit Iyyer, and Noah Constant. 2022b. [Overcoming catastrophic forgetting in zero-shot cross-lingual generation](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9279–9300, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Chen Wang, Yuchen Liu, Boxing Chen, Jiajun Zhang, Wei Luo, Zhongqiang Huang, and Chengqing Zong. 2022. [Discrete cross-modal alignment enables zero-shot speech translation](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5291–5302, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Dong Wang and Thomas Fang Zheng. 2015. [Transfer learning for speech and language processing](#). In *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA 2015, Hong Kong, December 16-19, 2015*, pages 1225–1237. IEEE.
- Hetong Wang, Pasquale Minervini, and Edoardo Ponti. 2024. [Probing the emergence of cross-lingual alignment during LLM training](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12159–12173, Bangkok, Thailand. Association for Computational Linguistics.
- Weixuan Wang, Minghao Wu, Barry Haddow, and Alexandra Birch. 2025. [Bridging the language gaps in large language models with inference-time cross-lingual intervention](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5418–5433, Vienna, Austria. Association for Computational Linguistics.

- Wenxuan Wang, Guodong Ma, Yuke Li, and Binbin Du. 2023a. [Language-Routing Mixture of Experts for Multilingual and Code-Switching Speech Recognition](#). In *Proc. INTERSPEECH 2023*, pages 1389–1393.
- Yifan Wang, Zewei Sun, Shanbo Cheng, Weiguo Zheng, and Mingxuan Wang. 2023b. [Controlling styles in neural machine translation with activation prompt](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 2606–2620, Toronto, Canada. Association for Computational Linguistics.
- Aman Kassahun Wassie, Mahdi Molaei, and Yasmin Moslem. 2024. [Domain-specific translation with open-source large language models: Resource-oriented analysis](#). *CoRR*, abs/2412.05862.
- Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022a. [Finetuned language models are zero-shot learners](#). In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.
- Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022b. [Emergent abilities of large language models](#). *Trans. Mach. Learn. Res.*, 2022.
- Guillaume Wenzek, Vishrav Chaudhary, Angela Fan, Sahir Gomez, Naman Goyal, Somya Jain, Douwe Kiela, Tristan Thrush, and Francisco Guzmán. 2021. [Findings of the WMT 2021 shared task on large-scale multilingual machine translation](#). In *Proceedings of the Sixth Conference on Machine Translation*, pages 89–99, Online. Association for Computational Linguistics.
- Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. [CCNet: Extracting high quality monolingual datasets from web crawl data](#). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.
- Genta Winata, Lingjue Xie, Karthik Radhakrishnan, Shijie Wu, Xisen Jin, Pengxiang Cheng, Mayank Kulkarni, and Daniel Preotiuc-Pietro. 2023. [Overcoming catastrophic forgetting in massively multilingual continual learning](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 768–777, Toronto, Canada. Association for Computational Linguistics.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. [Transformers: State-of-the-art natural](#)

Bibliography

- language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.
- Di Wu, Yibin Lei, Andrew Yates, and Christof Monz. 2024a. [Representational isomorphism and alignment of multilingual large language models](#). In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 14074–14085, Miami, Florida, USA. Association for Computational Linguistics.
- Junhong Wu, Yuchen Liu, and Chengqing Zong. 2024b. [F-MALLOC: Feed-forward memory allocation for continual learning in neural machine translation](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7180–7192, Mexico City, Mexico. Association for Computational Linguistics.
- Liwei Wu, Shanbo Cheng, Mingxuan Wang, and Lei Li. 2021. [Language tags matter for zero-shot neural machine translation](#). In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3001–3007, Online. Association for Computational Linguistics.
- Shijie Wu and Mark Dredze. 2019. [Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.
- Zhanglin Wu, Zongyao Li, Daimeng Wei, Hengchao Shang, Jiaxin Guo, Xiaoyu Chen, Zhiqiang Rao, Zhengzhe Yu, Jinlong Yang, Shaojun Li, Yuhao Xie, Bin Wei, Jiawei Zheng, Ming Zhu, Lizhi Lei, Hao Yang, and Yanfei Jiang. 2023. [Improving neural machine translation formality control with domain adaptation and reranking-based transductive learning](#). In *Proceedings of the 20th International Conference on Spoken Language Translation (IWSLT 2023)*, pages 180–186, Toronto, Canada (in-person and online). Association for Computational Linguistics.
- Zhaofeng Wu, Xinyan Velocity Yu, Dani Yogatama, Jiasen Lu, and Yoon Kim. 2025. [The semantic hub hypothesis: Language models share semantic representations across languages and modalities](#). In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.
- Yongqin Xian, Bernt Schiele, and Zeynep Akata. 2017. [Zero-shot learning - the good, the bad and the ugly](#). In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 3077–3086. IEEE Computer Society.
- Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2024. [A paradigm shift in machine translation: Boosting translation performance of large language models](#).

- In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.
- Haoran Xu, Weiting Tan, Shuyue Li, Yunmo Chen, Benjamin Van Durme, Philipp Koehn, and Kenton Murray. 2023. [Condensing multilingual knowledge with lightweight language-specific modules](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1575–1587, Singapore. Association for Computational Linguistics.
- Weijia Xu, Batool Haider, and Saab Mansour. 2020. [End-to-end slot alignment and recognition for cross-lingual NLU](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5052–5063, Online. Association for Computational Linguistics.
- Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. [mT5: A massively multilingual pre-trained text-to-text transformer](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.
- Kevin Yang and Dan Klein. 2021. [FUDGE: Controlled text generation with future discriminators](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3511–3535, Online. Association for Computational Linguistics.
- Mu Yang, Andros Tjandra, Chunxi Liu, David Zhang, Duc Le, and Ozlem Kalinli. 2023. [Learning ASR pathways: A sparse multilingual ASR model](#). In *IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2023, Rhodes Island, Greece, June 4-10, 2023*, pages 1–5. IEEE.
- Shan Yang, Heng Lu, Shiyin Kang, Liumeng Xue, Jinba Xiao, Dan Su, Lei Xie, and Dong Yu. 2020. [On the localness modeling for the self-attention based end-to-end speech synthesis](#). *Neural Networks*, 125:121–130.
- Yilin Yang, Akiko Eriguchi, Alexandre Muzio, Prasad Tadepalli, Stefan Lee, and Hany Hassan. 2021. [Improving multilingual translation by representation and gradient regularization](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7266–7279, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Rong Ye, Mingxuan Wang, and Lei Li. 2022. [Cross-modal contrastive learning for speech translation](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5099–5113, Seattle, United States. Association for Computational Linguistics.
- Zheng Xin Yong, Hailey Schoelkopf, Niklas Muennighoff, Alham Fikri Aji, David Ifeoluwa Adelani, Khalid Almubarak, M Saiful Bari, Lintang Sutawika, Jungo Kasai, Ahmed

Bibliography

- Baruwa, Genta Winata, Stella Biderman, Edward Raff, Dragomir Radev, and Vassilina Nikoulina. 2023. [BLOOM+1: Adding language support to BLOOM for zero-shot prompting](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11682–11703, Toronto, Canada. Association for Computational Linguistics.
- Katherine Yu, Haoran Li, and Barlas Oguz. 2018. [Multilingual seq2seq training with similarity loss for cross-lingual document classification](#). In *Proceedings of the Third Workshop on Representation Learning for NLP*, pages 175–179, Melbourne, Australia. Association for Computational Linguistics.
- Fei Yuan, Shuai Yuan, Zhiyong Wu, and Lei Li. 2024. [How vocabulary sharing facilitates multilingualism in LLaMA?](#) In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12111–12130, Bangkok, Thailand. Association for Computational Linguistics.
- Changtong Zan, Liang Ding, Li Shen, Yibing Zhen, Weifeng Liu, and Dacheng Tao. 2024. [Building accurate translation-tailored llms with language aware instruction tuning](#). *CoRR*, abs/2403.14399.
- Biao Zhang, Ankur Bapna, Rico Sennrich, and Orhan Firat. 2021. [Share or not? learning to schedule language-specific capacity for multilingual translation](#). In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.
- Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. 2024. [When scaling meets LLM finetuning: The effect of data, model and finetuning method](#). In *The Twelfth International Conference on Learning Representations*.
- Biao Zhang and Rico Sennrich. 2019. [Root mean square layer normalization](#). In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 12360–12371.
- Biao Zhang and Rico Sennrich. 2021. [Edinburgh’s end-to-end multilingual speech translation system for IWSLT 2021](#). In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 160–168, Bangkok, Thailand (online). Association for Computational Linguistics.
- Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020a. [Improving massively multilingual neural machine translation and zero-shot translation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1628–1639, Online. Association for Computational Linguistics.
- Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. 2017a. [Understanding deep learning requires rethinking generalization](#). In *5th International*

- Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.
- Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017b. [Adversarial training for unsupervised bilingual lexicon induction](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1959–1970, Vancouver, Canada. Association for Computational Linguistics.
- Shucong Zhang, Erfan Loweimi, Peter Bell, and Steve Renals. 2020b. [When can self-attention be replaced by feed forward layers?](#) *CoRR*, abs/2005.13895.
- Chengzhi Zhong, Fei Cheng, Qianying Liu, Junfeng Jiang, Zhen Wan, Chenhui Chu, Yugo Murawaki, and Sadao Kurohashi. 2024. [Beyond english-centric llms: What language do multilingual language models think in?](#) *Preprint*, arXiv:2408.10811.
- Changfeng Zhu, Heng Yu, Shanbo Cheng, and Weihua Luo. 2020a. [Language-aware interlingua for multilingual neural machine translation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1650–1655, Online. Association for Computational Linguistics.
- Lichao Zhu, Maria Zimina, Maud B  nard, Behnoosh Namdar, Nicolas Ballier, Guillaume Wisniewski, and Jean-Baptiste Yun  s. 2023a. [Investigating techniques for a deeper understanding of neural machine translation \(NMT\) systems through data filtering and fine-tuning strategies](#). In *Proceedings of the Eighth Conference on Machine Translation*, pages 275–281, Singapore. Association for Computational Linguistics.
- Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Lingpeng Kong, Jiajun Chen, Lei Li, and Shujian Huang. 2023b. [Multilingual machine translation with large language models: Empirical results and analysis](#). *CoRR*, abs/2304.04675.
- Yun Zhu, Parisa Haghani, Anshuman Tripathi, Bhuvana Ramabhadran, Brian Farris, Hainan Xu, Han Lu, Hasim Sak, Isabel Leal, Neeraj Gaur, Pedro J. Moreno, and Qian Zhang. 2020b. [Multilingual Speech Recognition with Self-Attention Structured Parameterization](#). In *Proc. Interspeech 2020*, pages 4741–4745.
- Vil  m Zouhar, Pinzhen Chen, Tsz Kin Lam, Nikita Moghe, and Barry Haddow. 2024a. [Pitfalls and outlooks in using COMET](#). In *Proceedings of the Ninth Conference on Machine Translation*, pages 1272–1288, Miami, Florida, USA. Association for Computational Linguistics.
- Vil  m Zouhar, Shuoyang Ding, Anna Currey, Tatyana Badeka, Jenyuan Wang, and Brian Thompson. 2024b. [Fine-tuned machine translation metrics struggle in unseen domains](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 488–500, Bangkok, Thailand. Association for Computational Linguistics.

Bibliography

Maike Züfle and Jan Niehues. 2025. [Contrastive learning for task-independent SpeechLLM-pretraining](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 8469–8490, Vienna, Austria. Association for Computational Linguistics.

Acknowledgments

Jan, thank you for giving me plenty of freedom in research, and for being a supervisor I can be unfiltered with. Prof. Fraser, Prof. Henkel, Prof. Meyerhenke, thank you for agreeing to take up the reviewer or examiner role of this thesis.

To colleagues from the AI4LT lab: Lukas, Maïke, Margit, Maximilian, Sai, Silke, Supriti, Tu Anh, Yuka, Zhaolin, as well as colleagues from our neighboring lab: Binh, Carlos, Christian, Dogucan, Enes, Fabian, Ilya, Irem, Laura, Leonard, Nam, Quan, Seyma, Stefan, Yining, thank you for being such a nice and quirky team, making me look forward to every trip to Karlsruhe. To my ex-colleagues in Maastricht, thanks for the good time and I think of you with a smile.

To my friends and family who helped improve this thesis or the upcoming defense: Bulat Khaertdinov, Esam Ghaleb, Florian Wimmenauer, Jerry Spanakis, Mado Ntekouli, Ričards Marcinkevičs, Shu Okabe, Yuting Zhao, thank you for pouring your own time into making this work better.

To our past and current project collaborators: Felix Stahlberg, James Cross, Paco Guzmán, Xian Li, thank you for the feedback and discussions. To my internship mentors: Juan Pino and Changhan Wang, as well as members of the Universal Speech Translation Team at Meta, thank you for the learning experience. I have learned a tremendous amount from all of you.

To all the students I worked with, thank you, I have learned a lot from you too.

I should also thank Deutsche Bahn for training me to handle the uncontrollable.

Florian, in my master's thesis acknowledgement, I promised to work on my "ill-defined boundaries of day and night". I seem to have managed. This is probably another achievement over the years leading up to this thesis.

Publications

In this Thesis

This thesis is based on the following publications:

Danni Liu and Jan Niehues, “Conditions for Catastrophic Forgetting in Multilingual Translation”, in *Fifth Workshop on Multilingual Representation Learning, MRL 2025*.

Danni Liu and Jan Niehues, “Middle-Layer Representation Alignment for Cross-Lingual Transfer in Fine-Tuned LLMs”, in *63rd Annual Meeting of the Association for Computational Linguistics, ACL 2025*.

Danni Liu and Jan Niehues, “Recent Highlights in Multilingual and Multimodal Speech Translation”, in *21st International Conference on Spoken Language Translation, IWSLT 2024*.

Danni Liu and Jan Niehues, “How Transferable are Attribute Controllers on Pretrained Multilingual Translation Models?”, in *18th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2024*.

Danni Liu and Jan Niehues, “Learning an Artificial Language for Knowledge-Sharing in Multilingual Translation”, in *Seventh Conference on Machine Translation, WMT 2022*.

Danni Liu, Jan Niehues, James Cross, Francisco Guzmán, and Xian Li, “Improving Zero-Shot Translation by Disentangling Positional Information”, in *59th Annual Meeting of the Association for Computational Linguistics, ACL 2021*.

Other Publications

The following works were published during the period of the doctoral research, but are not included in this thesis. Publications marked with * are based on supervised student research projects.

Sara Papi, Maike Züfle, Marco Gaido, Beatrice Savoldi, **Danni Liu**, Ioannis Douros, Luisa Bentivogli, and Jan Niehues, “MCIF: Multimodal Crosslingual Instruction-Following Benchmark from Scientific Talks”, *Under Review*.

Hyunji Lee, **Danni Liu**, Supriti Sinhamahapatra, and Jan Niehues, “How do Multimodal Foundation Models Encode Text and Speech? An Analysis of Cross-Lingual and Cross-Modal Representations”, in *the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics, NAACL 2025*.*

Siqi Li, **Danni Liu**, and Jan Niehues, “Optimizing Rare Word Accuracy in Direct Speech Translation with a Retrieval-and-Demonstration Approach”, in *the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024*.*

Tu Anh Dinh, Carlos Mullov, Leonard Bärmann, Zhaolin Li, **Danni Liu**, Simon Reiß, Jueun Lee, Nathan Lerzer, Jianfeng Gao, Fabian Peller-Konrad, Alexander Waibel, Tamim Asfour, Michael Beigl, Rainer Stiefelhagen, Carsten Dachsbacher, Klemens Böhm, Jan Niehues, “SciEx: Benchmarking Large Language Models on Scientific Exams with Human Expert Grading and Automatic Grading”, in *the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024*.

Lukas Hilgert, **Danni Liu**, and Jan Niehues, “Evaluating and Training Long-Context Large Language Models for Question Answering on Scientific Papers”, in *1st Workshop on Customizable NLP: Progress and Challenges in Customizing NLP for a Domain, Application, Group, or Individual, CustomNLP4U Workshop 2024*.*

Vladimir Solovyev, **Danni Liu**, and Jan Niehues, “Language-Independent Representations Improve Zero-Shot Summarization”, in *the 2024 Conference of the North American Chapter of the Association for Computational Linguistics, NAACL 2024*.*

Yunus Demirag, **Danni Liu**, and Jan Niehues, “Benchmarking Diffusion Models for Machine Translation”, in *18th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop, EACL 2024 SRW*.*

Christian Huber, Tu Anh Dinh, Carlos Mullov, Ngoc-Quan Pham, Thai Binh Nguyen, Fabian Retkowsky, Stefan Constantin, Enes Ugan, **Danni Liu**, Zhaolin Li, Sai Koneru, Jan Niehues, and Alexander Waibel, “End-to-End Evaluation for Low-Latency Simultaneous Speech Translation”, in *the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, EMNLP 2023 Demo*.

Danni Liu, Changan Wang, Hongyu Gong, Xutai Ma, Yun Tang, and Juan Pino, “From Start to Finish: Latency Reduction Strategies for Incremental Speech Synthesis in Simultaneous Speech-to-Speech Translation”, in *23rd Annual Conference of the International Speech Communication Association, Interspeech 2022*.

Tu Anh Dinh, **Danni Liu**, and Jan Niehues, “Tackling Data Scarcity in Speech Translation using Zero-shot Multilingual Machine Translation Techniques”, in *2022 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022*.*

Sai Koneru, **Danni Liu**, and Jan Niehues, “Unsupervised Machine Translation On Dravidian Languages”, in *First Workshop on Speech and Language Technologies for Dravidian Languages, DravidianLangTech Workshop 2021*.*

Danni Liu, Gerasimos Spanakis, and Jan Niehues. “Low-Latency Sequence-to-Sequence Speech Recognition and Translation by Partial Hypothesis Selection”, in *21st Annual Conference of the International Speech Communication Association, Interspeech 2020*.

Danni Liu, Jan Niehues, and Gerasimos Spanakis. “Adapting End-to-End Speech Recognition for Readable Subtitles”, in *17th International Conference on Spoken Language Translation, IWSLT 2020*.

Shared Task System Descriptions

The following system description papers resulted from participation in shared tasks during the period of the doctoral research.

Bibliography

Zhaolin Li, Yining Liu, **Danni Liu**, Tuan Nam Nguyen, Enes Yavuz Ugan, Tu Anh Dinh, Carlos Mullov, Alexander Waibel, and Jan Niehues, “KIT’s Low-resource Speech Translation Systems for IWSLT2025: System Enhancement with Synthetic Data and Model Regularization”, in *22nd International Conference on Spoken Language Translation, IWSLT 2025*. **(Winning system in North Levantine Arabic, Tunisian Arabic, and Maltese to English translation)**

Sai Koneru, Thai-Binh Nguyen, Ngoc-Quan Pham, **Danni Liu**, Zhaolin Li, Alexander Waibel, and Jan Niehues, “Blending LLMs into Cascaded Speech Translation: KIT’s Offline Speech Translation System for IWSLT 2024”, in *21st International Conference on Spoken Language Translation, IWSLT 2024*.

Zhaolin Li, Enes Yavuz Ugan, **Danni Liu**, Carlos Mullov, Tu Anh Dinh, Sai Koneru, Alexander Waibel, and Jan Niehues, “The KIT Speech Translation Systems for IWSLT 2024 Dialectal and Low-resource Track”, in *21st International Conference on Spoken Language Translation, IWSLT 2024*. **(Winning system in Maltese to English translation)**

Danni Liu, Thai Binh Nguyen, Sai Koneru, Enes Yavuz Ugan, Ngoc-Quan Pham, Tuan Nam Nguyen, Tu Anh Dinh, Carlos Mullov, Alexander Waibel, and Jan Niehues, “KIT’s Multilingual Speech Translation System for IWSLT 2023”, in *20th International Conference on Spoken Language Translation, IWSLT 2023*. **(Winning system in constrained multilingual track)**

Peter Polák, **Danni Liu**, Ngoc-Quan Pham, Jan Niehues, Alexander Waibel, and Ondřej Bojar, “Towards Efficient Simultaneous Speech Translation: CUNI-KIT System for Simultaneous Track at IWSLT 2023”, in *20th International Conference on Spoken Language Translation, IWSLT 2023*.

Abteen Ebrahimi, Manuel Mager, Adam Wiemerslage, Pavel Denisov, Arturo Oncevay, **Danni Liu**, Sai Koneru, Enes Yavuz Ugan, Zhaolin Li, Jan Niehues, Monica Romero, Ivan G Torre, Tanel Alumäe, Jiaming Kong, Sergey Polezhaev, Yury Belousov, Wei-Rui Chen, Peter Sullivan, Ife Adebara, Bashar Talafha, Alcides Alcoba Inciarte, Muhammad Abdul-Mageed, Luis Chiruzzo, Rolando Coto-Solano, Hilaria Cruz, Sofía Flores-Solórzano, Aldo Andrés Álvarez López, Ivan Meza-Ruiz, John E Ortega, Alexis Palmer, Rodolfo Joel Zevallos Salazar, Kristine Stenzel, Thang Vu, and Katharina Kann, “Findings of the Second AmericasNLP Competition on Speech-to-Text Translation”, in *Proceedings of the NeurIPS 2022 Competitions Track*. **(Winning system in speech translation and machine translation tracks)**

Ngoc-Quan Pham, Tuan Nam Nguyen, Thai-Binh Nguyen, **Danni Liu**, Carlos Mullov, Jan Niehues, and Alexander Waibel, “Effective combination of pretrained models - KIT@IWSLT 2022”, in *19th International Conference on Spoken Language Translation, IWSLT 2022*.

Peter Polák, Ngoc-Quan Pham, Tuan Nam Nguyen, **Danni Liu**, Carlos Mullov, Jan Niehues, Ondřej Bojar, and Alexander Waibel, “CUNI-KIT System for Simultaneous Speech Translation Task at IWSLT 2022”, in *19th International Conference on Spoken Language Translation, IWSLT 2022*. **(Winning system in simultaneous translation track)**

Danni Liu and Jan Niehues, “Maastricht University’s Large-Scale Multilingual Machine Translation System for WMT 2021”, in *Sixth Conference on Machine Translation, WMT 2021*.

Danni Liu and Jan Niehues, “Maastricht University’s Multilingual Speech Translation System for IWSLT 2021”, in *18th International Conference on Spoken Language Translation, IWSLT 2021*.

Curriculum Vitae

Education

- 02/2020 – 11/2025 **Ph.D. Computer Science**
Karlsruhe Institute of Technology, Germany (07/2022 – 11/2025)
Maastricht University, Netherlands (02/2020 – 06/2022)
Advisor: Jan Niehues
- 02/2018 – 01/2020 **M.Sc. Artificial Intelligence** (cum laude)
Maastricht University, Netherlands
- 09/2014 – 08/2017 **B.Sc. Data Science and Knowledge Engineering** (cum laude)
Maastricht University, Netherlands
- 09/2011 – 08/2014 **B.Sc. International Business Administration** (cum laude)
Erasmus University Rotterdam, Netherlands

Professional Experience

- 07/2021 – 01/2022 **Facebook / Meta AI**, Research Intern.
- 03/2019 – 08/2019 **Nuance Communications (now Microsoft)**, Research Intern.
- 08/2015 – 01/2019 **Open Line**, Trainee → Business Intelligence Consultant.

Teaching and Mentoring

- 2020 – 2025 **Supervision of Theses & Internships**
15 finished bachelor and master theses, leading to 7 publications.
- 2022 – 2025 **Teaching Assistant**, Karlsruhe Institute of Technology, Germany
Courses: Natural Language Processing, multiple seminars
- 2019 – 2021 **Teaching Assistant**, Maastricht University, Netherlands
Courses: Advanced Natural Language Processing, Probability and Statistics
- 2012 - 2013 **Teaching Assistant**, Erasmus University Rotterdam, Netherlands
Courses: Skills 1: IT, mentoring course for first-year undergraduates