Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 115 (2022) 184–189

www.elsevier.com/locate/procedia

10th CIRP Global Web Conference – Material Aspects of Manufacturing Processes

Applying Natural Language Processing in Manufacturing

Marvin Carl May *[a], Jan Neidhöfer[a], Tom Körner[a], Louis Schäfer[a], Gisela Lanza[a]

[a]*wbk Institute of Production Science, Karlsruhe Institute of Technology (KIT), Kaiserstr. 12, 76131 Karlsruhe, Germany*

* Corresponding author. Tel:+49-1523-950-2624; Fax:+49-721-60845005. *E-mail address:* marvin.may@kit.edu

Abstract

Despite great progress in the digitization of the industrial sector through Industry 4.0 and widely available data, data analysis is typically constrained to numerical data, not the synthesis of knowledge. Although valuable employee knowledge in manufacturing is often described textually, it is rarely formalized and effective application hindered. To close this gap, we introduce methods of Natural Language Processing (NLP) to leverage available text data in manufacturing. For this purpose, we develop a NLP pipeline to handle textual information from machine providers. We extend this with production specific information to reduce failure downtime. The resulting, formalized knowledge can furthermore be used as a basis for optimizing manufacturing processes.

Keywords: Natural Language Processing; Predictive Maintenance; Knowledge Extraction; Machine Learning; Production Planning

1. Introduction

Industry 4.0, widely regarded as a major development in manufacturing, is based on the ubiquitous acquisition, analysis and application of data and models in industrial settings. Fundamental is the ability to derive knowledge from *stupid* data. Until now, the scheme is effectively applied to tabular, numerical data, pictures or control sequences. In contrast to such numerical data, natural language in both written and spoken form is far more expressive and capable of describing and transferring knowledge [2]. However, an effective formalization and analysis of this knowledge is missing in manufacturing [18]. Instead of tapping into the vast knowledge that is expressed in natural language, humans are often manually providing numerical data to enable regular Industry 4.0 use-cases. However, standardized questionnaires, fault report templates or machine outputs for example are prone to a limited solution space. With a lot of challenges in manufacturing, there is a vast variety of different influencing factors that cannot be thought of in advance and therefore result in a huge amount of unstructured, natural text data. To that end, Natural Language Processing (NLP), the art of formalizing natural language to distill knowledge [14], can serve as a valuable method and holds immense potential. Thus, in order to effectively use the knowledge inherent to natural conversation and written texts in manufacturing - especially from old data where the structure cannot be changed - this paper proposes a novel NLP pipeline for the application in manufacturing. Based on the developed NLP pipeline a case study with real world industrial text data is performed and exemplary, suitable algorithms are presented. In particular, comments from machine operators that are digitally recorded and that describe failures and downtimes from manufacturing are analyzed for improving the accurateness of downtime estimations with domain knowledge through NLP. The approach can easily be adapted to processing hand-written comments by adding a preceding optical character recognition.

This article is structured as follows: Section 2 introduces the current state-of-the-art in Natural Language Processing in general, as well as an outline of the so far few applications in manufacturing. Based on the state-of-the-art and keeping the manufacturing setting in mind Section 3 introduces the novel, exemplified NLP pipeline approach. The approach is put to the test with a case study in Section 4. In Section 5 the findings are discussed and followed by a subsequent summary and outlook in Section 6.

2. Related Work

A brief state-of-the-art of NLP, as introduced in Section 2.1, and a general literature review of possible applications of NLP in manufacturing in Section 2.2 are presented.

10.1016/j.procir.2022.10.071

2.1. State-of-the-art Natural Language Processing

In manufacturing companies, text data is often available in an unorganized form. There is a relatively small body of literature that is directly concerned with using NLP methods in combination with Machine Learning (ML) for production related data [4]. Due to increasing data acquisition on the shopfloor with respect to textual data and the availability of increasingly powerful NLP algorithms, the application of NLP is promising in the production domain. In order to apply NLP in a production environment and ultimately tap into the unstructured but available human knowledge in form of texts and spoken language, a general understanding of current methods and algorithms is required.

At the beginning of the NLP application, the textual documents have to be sorted, prepared and filtered [14]. Next, the texts are preprocessed in the data preparation step. Examples of preprocessing include the removal of unwanted characters, the lemmatization of words ,the removal of stop wordsor stemming. [14]. Lemmatiziation describes the grouping of inflicted words, such as "best" and "good" [20]. Stop words are words that occur so frequently in a document collection that it makes no sense to search for them or index them (e.g. the, that, is, ...).Stemming is described as a process of extracting a root word. After the data preparation, the vectorization takes place [25]. The vectorization is based on the embedding of words in vectors for the sake of representation (i.e. Word2Vec, fastText) or their frequency-based vectorization (e.g. One-Hot-Encoding, TF-IDF) [25]. This vectorization allows ML algorithms to process the text documents and to train the model. An evaluation metric can be determined and the algorithm or model can be adjusted accordingly.

A typical process for using ML on textual data is shown in Fig. 1.

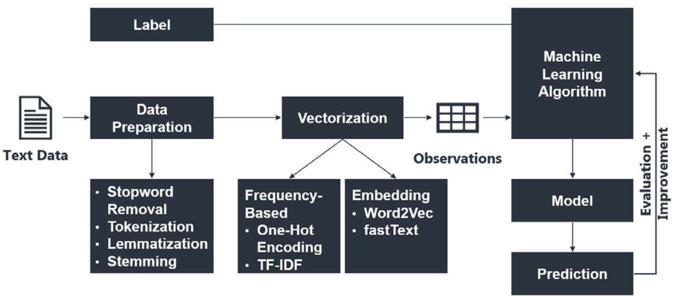

Fig. 1. Standard process for applying ML on textual data.

Both in industry and in companies in the Business-to-Customer (B2C) sector, data labeled by humans is very difficult to obtain. That is why pre-trained NLP models are used, especially in the B2C area. These models are pre-trained on large datasets such as accumulated Twitter data or Wikipedia documents [22]. An example of this is the bidirectional algorithm: BERT. The algorithm BERT (Bidirectional Encoder Representations from Transformers), developed in 2018, uses pre-training on text data from the web in order to then be able to adapt and apply the model to smaller tasks with the help of fine-tuning [8]. BERT is deep bidirectional meaning the algorithm uses the left and right context of words and, thus, attempts to distinguish words like ("seat bank or money bank"). Eventually BERT can generate a deeper understanding of the text than unidirectional models, such as: TF-IDF [8].With the TF-IDF score, it is possible to calculate the relevance between a word and a specific document without taking into account the position of a word in a sentence. For this purpose, the frequency with which a word occurs in a document is multiplied by the inverse document frequency of the word in a series of documents[6].

The development of the pre-trained algorithms is currently culminating in the GPT-3 algorithm [3]. GPT-3 is a trained deep neural network for NLP from OpenAI. It has been trained on 570 gigabytes of web text data and has over 175 billion parameters in total. Its structure is based on that of a transformer [24]. It trains by predicting a token based on a sequence of previous tokens. A token represents a word, an N-gram, or the like. In doing so, GPT-3 can also solve NLP tasks for which it was not originally trained [3].

These types of algorithms were not trained on text data within production, as each company uses a different terminology. This presents us with the challenge of a vectorization of text documents without prior training and knowledge according tot the sequential steps outlined in Table 1

Table 1. Overview - NLP and ML

Steps for NLP	Overview
1. Data preparation	Tokenization, stop-word removal, spelling correction, stemming, normalization
2. Vectorization	One-hot-encoding, TF-IDF, Bag of Words, Word2Vec, fastText, BERT
3. ML	Supervised (e.g. XGBoost, LightGBM, Random Forest), Unsupervised (e.g. k-Means, DB-SCAN, Expectation Maximization)

2.2. Literature Review

NLP is mainly applied in non-industrial sectors, since large collections of texts are already available. In the following, we describe some approaches of NLP outside the industrial sector in order to recognize the possibilities of already existing solutions for their transfer to industrial applications. One approach is to train a so-called language model with the goal of classifying a sentence whether it occurs as such in natural dialogs or not [7]. Net [1] develop a ML method in their research, which filters relevant features directly from the text to summarize a text automatically. A different application describes data-driven, automated, editorial creation of human-understandable as well as informative texts through ML algorithms. Here, the authors referred to the use case of automated reporting on the 2017 Finnish local elections [15].

Text data is prevalent in manufacturing, but often not used for computational analysis due to its complexity [18]. As about 80 % of the available knowledge is available in text form, manufacturing in particular represents a use case for NLP methods [10]. Moreover, researching available knowledge and merging data accounts accounts for about 70 % of the the design phase of projects [16]. Therefore, it can be concluded that the application of supporting NLP methods may lead to increased efficiency and time savings in various scenarios.

In the field of manufacturing, more precisely in the maintenance of machines, there are several approaches using NLP and ML. Understanding maintenance logs of past events by combining NLP and ML with human understanding [23] and extracting hidden knowledge from short texts from machines for maintenance decisions [4]. Furthermore, an application aimed at retrieving possible solutions for new problems by searching for a similar problem stored in remote technical assistance reports for troubleshooting [2], and NLP to find the root cause of a failure in the Semiconductor Industry [9] have been documented. The NLP procedures can be based on retrievable, central knowledge databases often found in manufacturing environments. In the area of administration, NLP may be relevant for employee or applicant management and for identifying and evaluating potential suppliers [17]. An NLP system for problem-solving management with an integrated content-based recommender engine to enhance knowledge management on the shopfloor exists [21]. This can be extended towards the integration of digital shopfloor management [19]. Communication within the company and with the outside world is also a potential target for NLP applications. Chatbots are commonly used for these purposes.

However, aside from Chatbots and shopfloor management analyses performed on less specific terms and language [19], the application of NLP in manufacturing is not capable of end to end prediction based on textual knowledge but rather focused on very well described, unique tasks. Hence, overcoming the infancy of NLP in manufacturing requires an easily transferable and applicable NLP pipeline for end to end prediction or classification based on available textual data.

3. Own Approach

Applications of NLP in manufacturing environments come in many facets as described before. Fig. 2 shows some of the possible applications of NLP in manufacturing environments that have been described in literature and that can envisioned with current NLP Algorithms. However, pre-trained models such as BERT cannot be used because the production vocabulary is different from everyday language. Using a large amount of data for training the models is therefore important. Thus, the structured approach, from shopfloor based textual data gathering to the resulting prediction or classification is missing.

Using production text data for NLP requires preprocessing before the application of Machine Learning algorithms as the data is unstructured and uncleaned. The structure is visualized in Fig. 2. Available manufacturing execution system (MES) and or textual data needs to be identified and made available on a large scale (step 0). This includes the selection of suitable data and texts that are meaningful and provide significant insights for the later algorithms to make feasible predictions, classifications or decisions. Subsequently, the data should be cleaned (step 1). This includes the removal of punctuation marks, leading spaces, special characters, and code fragments. Converting all terms to lower case is another helpful procedure to improve analytics due to adequate text preparation (step 2). In the production context in particular, spell checking and an autocorrection are a useful addition as comments are prone to contain spelling mistakes. After the following tokenization (step 3: chopping sentences in shorter tokens, i.e. consisting of one to three words) stop words should be removed from the corpus as these regularly occurring words contribute little or no information (step 4). The next step is the lemmatization of the text, whereby the individual terms are converted into their basic form (step 5). Subsequently, the text data is processed by a vectorizer (step 6). Common methods such as Bag-of-Words, TF-IDF or Word2Vec can be used for this. Only then are various algorithms applicable (step 7) to perform tasks ranging from maintenance decision selection and root cause analyses to downtime predictions.

In order to generate knowledge using NLP for a specific task in manufacturing, a varying sequence of steps is necessary. The necessary procedure depends on the objective, available data and technology. In this section we describe a new approach of predicting machine downtime. To the best of our knowledge, this application that has not been covered yet by research despite providing a promising outlook. Note that for in-depth comparison of individual algorithms or techniques per step we refer the reader to standard NLP literature [14] and comparisons [20].

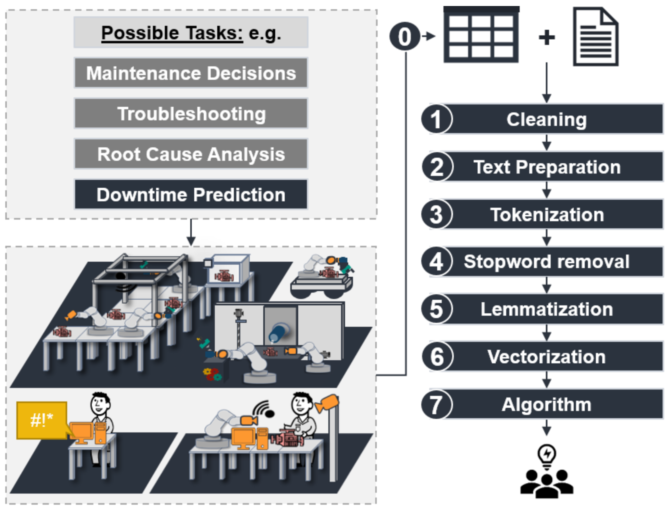

Fig. 2. Use of NLP in manufacturing environments.

In the following, we further describe the plausible use case of forecasting machine downtime by analyzing text data in combination with machine information. The goal of the approach is to develop a pipeline that is able to forecast machine downtime of production lines or single machines. Typically, in modern manufacturing sites, production lines and machines are connected to a MES that tracks events and production data. The data is automatically generated by the system during production events and saved in tabular form in databases. The data typically contains information about the affected equipment, the type of error, meta information of the production such as timestamps and event duration. In many cases, machine operators or production staff can manually add text comments to provide additional information. Operators that work at a specific line or machine supervise production and processes and have enhanced knowledge about certain, potentially recurring events. Therefore, the added comments contain important information. On the other hand, due to missing standards, the comments may vary greatly in form and content. In addition, since production companies often have established their own jargon and fixed abbreviations, the production text data deviates from the everyday language. Likewise, grammatical and spelling are of secondary importance to the mere transmission of information and are often neglected by operators in the hectic production environment. We assume that the added text data contains relevant information about certain events and that the information provided by machine operators correlates to some extend with the resulting downtime. It can therefore be assumed that the additional information in the form of text comments can significantly contribute to the prediction of production downtime. Based on this prerequisites the proposed NLP pipeline is filled with concrete algorithms and real world data in the case study in Section 4.

4. Case Study

In the following, we describe a case study applying NLP for downtime prediction in manufacturing environments. We develop a pipeline to predict the downtime of machine stops based on text comments from machine operators in combination with alarm and events (A&E) data from production lines. We use the data to classify faults in terms of their severity with regard to its downtime. This information can be used by the department management to distribute production orders to other production lines, schedule maintenance or cleaning actions, or assign staff to different tasks. We examine combined data from multiple manufacturing departments and production lines of a production plant.

The data used in this case study comes from a production facility of a consumer goods manufacturer (step 0). The products manufactured in the production facility belong to the same business unit. The individual departments are responsible for different production steps in the production of the product. The individual production lines within the departments also differ from each other, e.g. in the use of certain machines or their arrangement. However, the production lines of each department produce identical or similar products. The production data is collected by the Line Events Data System (LEDS) of an overarching MES and stored in a central database. In addition to meta-information on production, all events on production lines are recorded. Events describe, for example, setup processes, machine stops, maintenance and project-based downtimes. Events are divided into planned and unplanned events and classified using a boolean identifier.

The data set includes 98,434 observations of unplanned stops from 27 different production lines. Each observation accounts for a specific unplanned machine stop. As soon as a stop occurs, the production line, equipment and component causing it, the defined error code and time are recorded. The duration of the stop in minutes is calculated from the difference to the following event and inserted subsequently. Since the error is not necessarily of technical nature, but for example, results from incorrect settings, product characteristics or wear, the production line operators add an individual comment to the respective entry. This aims at describing the cause of the error or the reason for the production downtime as concisely as possible. The machine-generated data is partly inconsistent, e.g. certain pieces of equipment are listed under different names. Likewise, some names of production lines, machines or components have changed over time, but have not been corrected subsequently. The data is, therefore, cleaned (step 1) before the actual analysis and also freed from missing entries. All events that do not describe an unplanned downtime are removed. To predict the severity of the machine stop, we use only the machine operator comments, the location of the error, and the cause of the machine stop in our analysis.

The comment data also contains various errors. In addition to spelling errors, there are code fragments, superfluous punctuation marks, and leading spaces. The quality of the data requires text preparation (step 2) before use. First, all textual data is stripped of punctuation, as it provides no information content in our case. Then, all textual entries are converted to lower case and tokenized word by word (step 3). Stop words are removed next (step 4). Stop words include not only generally frequently occurring terms but also, for example, artifacts from code fragments, which are specified manually after a frequency analysis. In the real dataset, there are 32,635 events in the light error category, 41,880 events in the medium error category, and 23,919 events in the severe error category. We can, thus, rule out a serious imbalance. Given the nature of the described problems lemmatization (step 5) provides little additional support and is, thus, simplified. Note, that the degree and type of lemmatization is highly dependent on the actual use case and language used.

Following the preprocessing, we convert the text data into a vector format (step 6). As methods we use in this case TF-IDF and Bag-of-Words. Bag-of-Words (BoW) is a method for selecting features by creating bags for each instance type. In NLP a Bow is described as a vector of number of word occurrences. Afterwards, the data is divided into X and y where X is a matrix of text data, and y a vector of the corresponding downtime class. Furthermore, we divide the data set into a training set and a test set with the distribution 80% and 20%, respectively. We use Gradient Boosting Decision Tree (GBDT) algorithms XGBoost [5] and LightGBM [12] as classification models for our analysis as Fig. 3 shows. GBDT algorithms iterate over trees and fit them to the gradient of the loss function with respect to the output of the previous tree. The benefits of XGBoost and LightGBM are their scalability and speed that are achieved through out-of-core computation and their ability to handle sparse data. Data sparsity can occur because of missing values or as a result of feature-encoding and might cause prolonged learning sequences in conventional algorithms [5, 12] We define both models as multi-class classifiers with softmax as their objective function. Afterwards, we use grid search in conjunction with k-fold cross-validation to tune the hyperparameters and determine the best hypothesis. The model with the best accuracy is then used for evaluation. We also consider the weighted F1-score as well as the one-vs-one ROC-AUC (ROC-AUC-OVO). In addition, we apply a zero rate classifier to the data set to separate the performance of our models from the baseline classification (step 7).

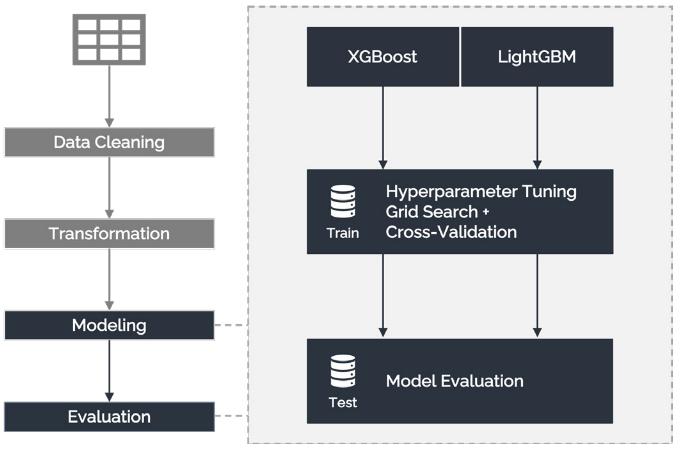

Fig. 3. Pipeline approach and modeling for downime prediction.

The results of the evaluation are summarized in Table 2. Using Bag-of-Words for vectorization results in an accuracy of 57.21 % and 55.48 %, a weighted F1-score of 57.09 % and 55.13 %, and a ROC AUC of 74.11 % and 73.77 % for XGBoost and LightGBM, respectively. The use of TF-IDF vectorization resulted in an accuracy of 58.87 % (62.06 %), a weighted F1-score of 58.26 % (62.10 %), and a ROC AUC of 77.52 % (79.97 %). Word2Vec results in an accuracy of 43.47 % (46.15 %), a weighted F1-score of 28.91 % (45.31 %), and a ROC AUC of 60.92 % (62.85 %). Baseline classification results in 34 %.

The results show that TF-IDF delivers the overall better performance compared to Bag-of-Words or Word2Vec. This indicates performance superiority for more complex and accurate vectorization algorithms. The need for careful vectorization and sophisticated algorithms is, thus, reinforced. When using Bag-of-Words, XGBoost and LightGBM show very similar performance. When using TF-IDF, the LightGBM model provides the better results. Compared to the zero rate classification, both models document a significant increase in performance.

Table 2. Evaluation results for both algorithms and vectorization approaches.

Vectorization	Metric	XGBoost	LightGBM
Bag-of-Words	Accuracy	57.21 %	55.48 %
	F1 Weighted	57.09 %	55.13 %
	ROC AUC OVO	74.11 %	73.77 %
TF-IDF	Accuracy	58.87 %	62.06 %
	F1 Weighted	58.26 %	62.10 %
	ROC AUC OVO	77.52 %	79.97 %
Word2Vec	Accuracy	43.57 %	46.15 %
	F1 Weighted	28.91 %	45.31 %
	ROC AUC OVO	60.92 %	62.85 %
Baseline Classifier	Accuracy	34 %	

5. Discussion

In our case study, we use gradient boosting ML algorithms to predict the severity of machine downtime. Based on these, it can be seen that the downtime comments written by shopfloor workers do offer explanatory power with regard to the variation in downtime. However, the forecasts lack high accuracy for direct use in operations. The nature and the information content of the data play a major role in the accuracy of the prediction. With just under 100,000 observations, the data set is small, in particular for a NLP problem. This is further exacerbated by the division into training and test data sets. Also missing are some features that could have a significant impact on downtime, such as information about the product or team. Some randomness in downtimes is also conceivable. In the preparation of the data, autocorrection as well as the filtering of outliers could have a positive impact on the result. For autocorrection, a production or company-specific lexicon should be used, since the vocabulary is not completely covered with common implementations. Selection bias plays a vital role in modeling and evaluation, which is advocated by the large variation in results. Thus, the deployment model should always be trained on the entire data set. Further experiments are advised. Here, larger data sets as well as filtering of the data could be tried and a more comprehensive preparation could be performed. The introduction of a standardized comment format could have a positive effect on the result in the long term. Thus, industry readers are advised to keep standardized comment formats, the introduction of specific autocorrection lexicons and early data gathering on their bucketlist for enabling large scale NLP in manufacturing.

6. Summary & Outlook

In a nutshell, the need for applying NLP in manufacturing is evident. With increasing awareness, availability of natural language based knowledge on the shopfloor and easily applicable NLP pipelines this need can be addressed. Thus, this paper proposes a simplified NLP pipeline for the application in manufacturing and shows the effective application in a case study where machine downtimes are successfully predicted based on textual descriptions. Downtime prediction being a first step in the application of supervised learning through NLP with good results, follow up studies can shed light on the benefits in a long term application. Beyond supervised approaches presented here, a thorough analysis of NLP use cases with unsupervised and reinforcement learning is of high interest. For instance the explainability of intelligent, reinforcement learning based production control [13] shall be improved. Moreover, transformers should be analyzed and potentially trained with very large datasets from manufacturing and shopfloor. As the application in manufacturing and on the shopfloor differs greatly from standard NLP tasks, a comparison of individual techniques on all levels can be beneficial for future NLP users. In the future, the approach can easily be extended in order to be able to process hand-written comments by adding a preceding optical character recognition algorithms. Also, a comprehensive NLP application study in manufacturing can reveal further promising use cases and may lead to the integration of several NLP use cases, e.g. downtime prediction, maintenance decisions and digital shopfloor management [11, 19]. By leveraging more on this available knowledge, the true potential of NLP in manufacturing can be unleashed.

Acknowledgements

This research work was undertaken in the context of DIGIMAN4.0 project ("DIGItal MANufacturing Technologies for Zero-defect Industry 4.0 Production", http://www.digiman4-0.mek.dtu.dk/). DIGIMAN4.0 is a European Training Network supported by Horizon 2020, the EU Framework Programme for Research and Innovation (Project ID: 814225).

References

[1] , .

[2] Alfeo, A.L., Cimino, M.G., Vaglini, G., 2021. Technological troubleshooting based on sentence embedding with deep transformers. Journal of Intelligent Manufacturing 32, 1699–1710.

[3] Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D., 2020. Language models are few-shot learners. doi:10.48550/ARXIV.2005.14165.

[4] Brundage, M.P., Sexton, T., Hodkiewicz, M., Dima, A., Lukens, S., 2021. Technical language processing: Unlocking maintenance knowledge. Manufacturing Letters 27, 42–46. doi:https://doi.org/10.1016/j.mfglet.2020.11.001.

[5] Chen, T., Guestrin, C., 2016. Xgboost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining .

[6] Chen, W., Yuan, X., Zhang, S., Wu, J., Zhang, Y., Wang, Y., 2020. Ferryman at SemEval-2020 task 3: Bert with TFIDF-weighting for predicting the effect of context in word similarity, in: Proceedings of the Fourteenth Workshop on Semantic Evaluation, International Committee for Computational Linguistics, Barcelona (online). pp. 281–285. doi:10.18653/v1/2020.semeval-1.35.

[7] Collobert, R., Weston, J., 2008. A unified architecture for natural language processing, in: Proceedings of the 25th international conference on Machine learning - ICML '08, ACM Press. doi:10.1145/1390156.1390177.

[8] Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. doi:10.48550/ARXIV.1810.04805.

[9] Ezukwoke, K., Toubakh, H., Hoayek, A., Batton-Hubert, M., Boucher, X., Gounet, P., 2021. Intelligent fault analysis decision flow in semiconductor industry 4.0 using natural language processing with deep clustering, in: 2021 IEEE 17th International Conference on Automation Science and Engineering (CASE), IEEE. pp. 429–436.

[10] He, L., Dong, B., Jiang, P., 2021. A heuristic grafting strategy for manufacturing knowledge graph extending and completion based on nature language processing: Knowtree. IEEE Access 9, 90847–90862. doi:10.1109/ACCESS.2021.3092019.

[11] Kandler, M., May, M.C., Kurtz, J., Kuhnle, A., Lanza, G., 2021. Development of a human-centered implementation strategy for industry 4.0 exemplified by digital shopfloor management, in: Towards Sustainable Customization: Bridging Smart Products and Manufacturing Systems. Springer, pp. 738–745.

[12] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.Y., 2017. Lightgbm: A highly efficient gradient boosting decision tree, in: NIPS.

[13] Kuhnle, A., May, M.C., Schaefer, L., Lanza, G., 2021. Explainable reinforcement learning in production control of job shop manufacturing system. International Journal of Production Research , 1–23.

[14] Kulkarni, A., Shivananda, A., 2019. Natural Language Processing Recipes. Apress. doi:10.1007/978-1-4842-4267-4.

[15] Leppänen, L., Munezero, M., Granroth-Wilding, M., Toivonen, H., 2017. Data-driven news generation for automated journalism, in: Proceedings of the 10th International Conference on Natural Language Generation, Association for Computational Linguistics. doi:10.18653/v1/w17-3528.

[16] Li, X., Zhang, S., Huang, R., Huang, B., Xu, C., Kuang, B., 2018. Structured modeling of heterogeneous cam model based on process knowledge graph. The International Journal of Advanced Manufacturing Technology 96, 4173–4193.

[17] Mittal, V., Mehta, P., Relan, D., Gabrani, G., 2020. Methodology for resume parsing and job domain prediction. Journal of Statistics and Management Systems 23, 1265–1274.

[18] Moon, J., Park, G., Yang, M., Jeong, J., 2022. Design and verification of process discovery based on nlp approach and visualization for manufacturing industry. Sustainability 14, 1103.

[19] Müller, M., Alexandi, E., Metternich, J., 2021. Digital shop floor management enhanced by natural language processing. Procedia CIRP 96, 21–26.

[20] Müller, M., Longard, L., Metternich, J., 2022. Comparison of preprocessing approaches for text data in digital shop floor management systems. Procedia CIRP 107, 179–184.

[21] Müller, M., Terziev, G., Metternich, J., Landmann, N., 2020. Knowledge management on the shop floor through recommender engines. Procedia Manufacturing 52, 344–349. doi:10.1016/j.promfg.2020.11.057.

[22] Qiu, X., Sun, T., Xu, Y., Shao, Y., Dai, N., Huang, X., 2020. Pre-trained models for natural language processing: A survey. Science China Technological Sciences 63, 1872–1897. doi:10.1007/s11431-020-1647-3.

[23] Sexton, T., Brundage, M.P., Hoffman, M., Morris, K.C., 2017. Hybrid datafication of maintenance logs from ai-assisted human tags, in: 2017 IEEE International Conference on Big Data (Big Data), pp. 1769–1777. doi:10.1109/BigData.2017.8258120.

[24] Shao, T., Guo, Y., Chen, H., Hao, Z., 2019. Transformer-based neural network for answer selection in question answering. IEEE Access 7, 26146–26156. doi:10.1109/access.2019.2900753.

[25] Singh, A.K., Shashi, M., 2019. Vectorization of text documents for identifying unifiable news articles. Int. J. Adv. Comput. Sci. Appl 10.