Engineering Symposium at Bánki
(ESB 2019)
http://bgk.uni-obuda.hu/esb/2019/

Artificial Intelligence in Maintenance: The Industrial Application of Natural Language Processing

[1]Szűcs, Balázs, [2]Dr. Ballagi, Áron

[1]AUDI HUNGARIA Zrt., 9027 Győr, Hungary, Balazs.Szucs@audi.hu
[2]Széchenyi István Egyetem, Department of Automation, 9026 Győr, Hungary, ballagi@sze.hu

Abstract

Text- and natural language processing improved a lot in the recent years. Applications like translators, chatbots and virtual assistants made the everyday life easier, but the industrial use of the technology and its potential in the maintenance planning remained unexploited. In this paper we present two possible ways how to utilize these algorithms to achieve their positive benefits. For humans it's often impossible to handle historical maintenance data and error logs due to their enormous amount or because the descriptions of errors are subjective. With the analysis of error messages, maintenance and shift logs, the correlations between failures and events can be detected, thus the effectiveness of the maintenance planning and the interventions can be increased.

Keywords: maintenance; industry 4.0; machine learning; artificial intelligence; natural language processing

1 Introduction

The artificial intelligence and the machine learning methods improved a lot in the recent years, more and more application modes are being introduced every year. In the era of social media the value of the data has increased, as a result, the pace of development has accelerated. New, more sophisticated methods have extended to other disciplines. The effect of these methods are rising in health sciences, healthcare and financial areas.

The industrial sector is no exception; researches are under way to exploit the potential of artificial intelligence methods. Lee et al. [1] presented a condition based predictive maintenance system to predict machine tooling faults. Liu et al. [2] showed a method to diagnose the fault of rotary machines. Pelham & Hockley [3] used NLP to analyse short maintenance logs. Stenström et al. [4] demonstrated a potential use case of NLP to analyse rail infrastructure maintenance logs. It can be stated, that several researches going on in the field of maintenance, artificial intelligence and natural language processing, however there is no paper found on the potential of NLP in the manufacturing industry.

In the following sections we will show two potential use case of utilizing natural language processing in the industry to improve machine maintenance efficiency.

2 Natural language processing

Natural language processing (NLP) is a subfield of linguistics and artificial intelligence. The aim of NLP is to convert the natural language from human interpretable form to computer interpretable form. The NLP methods grant the ability to computers to understand the natural language. NLP is used for machine translation, speech recognition, language and text generation,

question answering and for automated chatbots. Nowadays NLP is popular in medical science, healthcare, social media, advertising industry, human resources and in other services, but the potential in the industrial usage remains unexploited.

2.1 Pre-processing

To transform the digitalized text to computer interpretable, well-defined sequence of tasks have to be done. To parse the text corpus the first step is the pre-processing. During the pre-processing the text document can be segmented to smaller parts or sentences, the terms without relevant information like punctuations and stop words can be removed.

There are other pre-processing steps for information extraction, like lemmatization, stemming, part-of-speech (POS) tagging, etc. Lemmatization removes inflectional endings only and returns the base dictionary form of a word. Stemming reducing inflected or derived words to their root form. POS tagging determines the part-of-speech tags for each word.

2.2 Tokenization

After the pre-processing it's necessary to convert the human readable characters, words or sentences into numerical values. This can be done via tokenization, where each sentence, word or n-gram is associated with a numerical value and added to list or dictionary. An n-gram is a contiguous sequence of n items from a given text, it can be created from words, and word based characters and characters too. Some of the pre-processing steps, like POS tagging also include tokenization steps.

To convert the text to numbers, weighting or vectorization functions, like term frequency (TF), inverse document frequency (IDF) or the combination of previous methods, the term frequency-inverse document frequency (TF-IDF) frequently used [7].

Term frequency TF, also denoted as tf(t,d) counts the number of times each term *t* occurs in each document *d*. Typical weighting function shown in 1. Table. Log normalization and double normalization are used to prevent bias towards longer documents. The value of α is usually set to 0.5.

Table 1. Term frequency weighting schemes

Weighting scheme	***Weight***
Binary	0, 1
Raw count	$f_{t,d}$
Term frequency	$\frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}}$
Log normalization	$\log(1 + f_{t,d})$
Double normalizations	$\alpha + (1-\alpha)\frac{f_{t,d}}{max_{\{t' \in d\}} f_{t',d}}, 0 < \alpha < 1$

The inverse document frequency, IDF (Eq.5) is a measure of how much information the word provides, i.e., if it is common or rare across all documents.

$$idf(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|} \quad (1)$$

The **inverse document frequency, IDF** (Eq.2) is a measure of how much information the word provides, i.e., if it is common or rare across all documents.

$$idf(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|} \tag{2}$$

N: total number of documents in the corpus: $N = |D|$

$n_t = |\{d \in D : t \in d\}|$: Number of documents where the term *t* appears, often adjusted to $1 + |\{d \in D : t \in d\}|$ to avoid division-by-zero.

Table 2. Inverse document frequency weighting schemes

Weighting scheme	***Weight ($n_t = \|\{d \in D : t \in d\}\|$)***
Unary	1
Inverse document frequency	$log \frac{N}{n_t} = -log \frac{n_t}{N}$
Inverse document frequency smooth	$log \left(\frac{N}{1 + n_t}\right)$
Inverse document frequency max	$\log \left(\frac{max_{\{t' \in d\}} n_{t'}}{1 + n_t}\right)$
Probabilistic inverse document frequency	$\log \left(\frac{N - n_t}{n_t}\right)$

A high weight in **TF-IDF** is reached by a high term frequency (in the given document) and a low document frequency of the term in the whole collection of documents; the weights hence tend to filter out common terms.

$$tfidf(t, d, D) = tf(t, d) \cdot idf(t, D) \tag{3}$$

2.3 Toolbox

We used Python to design, build and evaluate machine learning (ML) algorithms and models. Python is an interpreted, high-level, general-purpose programming language, which is very popular in data science, ML and AI. To build our models we used the libraries scikit-learn [6], NLTK [7], spaCy [8]. These libraries are open-source and highly supported by the data science and AI community.

2.4 Summary

The above mentioned techniques also called *word embedding* which is a collective name to methods where words or phrases from the vocabulary are mapped to vectors of real numbers. There are other groups of complex models, like the Word2Vec [9].

At the end of the *NLP pipeline* the *corpora* is ready to feed into the machine learning algorithms and deep neural networks for classification or other task. In the following sections we are showing how to use these methods for maintenance log analysis and entity based error message analysis, which we use to improve maintenance effectivity.

3 NLP analysis of maintenance log

Most manufacturing companies records the maintenance activities, part replacements and machine refurbishments. The goal of this activity is to create a statistics from machine downtimes, analyse maintenance times and intervals, error causes, maintenance cost, improve maintenance process and to track quality relevant interventions. These logs are usually man-made, which consists differences in wording of fault descriptions for similar or the same faults. Even with few numbers of production machines, the unique wordings and the increasing amount of records over time makes harder to detect patterns and correlations between failures. To aid this problem, NLP and unsupervised learning comes in handy.

The log consist 343 record for 79 machines. To cluster the records and to detect similarities we used unsupervised NLP clustering. The fault descriptions are labelled (fault nature and fault class), the dataset would be a good starting point to train classification models, but the number of samples are too few, the model would not generalize well.

To cluster the samples first we pre-processed the text corpora. We removed the punctuations with the method of regular expression matching, then the stop words like *"repaired", "replacement", "after replacement", etc.* These terms are additional information to the reader (for e.g.: *"Workpiece positioning pin break. After replacement OK."),* they mean the repairing is done, but have no additional information about the fault. As the part of the pre-processing steps, we converted the text to lowercase. This step is usually recommended, it can reduce the number of tokens, because during the tokenization the same words or n-grams which begins with capitals, or contains capital letters, treated as one term.

For tokenization we used n-grams: we have split each word to unigrams, bigrams and trigrams, than we count their term frequency (TF) in each document. Longer n-grams, like trigrams or four-grams recommended when we dealing with languages which contains *digraphs*.

The tokenization and the token counting creates a *vocabulary,* a *sparse matrix* that contains the n-gram terms count frequency. In the columns are the n-grams, in the rows. This numerical representation of the documents made it available to feed them into machine learning algorithms.

To cluster the error logs, we used the *Density-Based Spatial Clustering of Applications with Noise (DBSCAN*) [10] algorithm. DBSCAN groups together points that are close to each other based on a distance measurement. To measure the distance between the sparse vectors we used cosine similarity. Cosine similarity is used to measure the angle between two normalized vectors, thus to calculate the similarity of the vectors. [11]

As the result of the clustering, from the 343 maintenance log, 169 records were grouped together to separate clusters. That means 49.27 % of the faults are recurring. The other 174 record are unique entry.

By analysing the repetitive faults, the maintenance strategies tasks can be improved, or the serious fault can be avoided. The above example shows that, that the simplest NLP analysis can boost the maintenance efficiency. Further goals is to extend the analysis to longer term records.

4 NLP Aided Maintenance

Besides manually recorded maintenance logs there are automatically recorded fault message logs. A moderately complex production machine can have more than thousands of error messages. These messages are generally addressed to the operators. But there are complex faults, which cannot be solved by the operator alone, despite the error messages known. As well as, there are cases when the expertise personnel cannot solve the problem as soon as possible. To help the solution of errors, NLP can be used.

In some cases, the operator does not know where the exact fault location is, or who the right, expertise person to notify is. These problem can be solved by the named entity recognition (NER). The named entity recognition identifies the entities in a document, text corpora or in a fault message. These entities can be persons, locations or even machine parts or components. Knowing the source of fault, the fault location can be visualised on the machine blueprint, the component documentation can be display automatically and the right person can be notified automatically.

To train our named entity recognition, we analysed 4533 fault messages. The messages belongs to industrial washing machines, turning, grinding and milling machines. To speed up the analysis, we used in the previous section presented method.

After the pre-processing (punctuation and stop word removal) steps, we split the error messages to unique 2891 words, then we clustered them. Since the error messages contains partially repetitive parts, that means the component names are repetitive, the TF-IDF vectorization was ineffective, because the TF-IDF weights filter out the common terms. We used TF vectorization. To cluster the entities we also used DBSCAN here.

After clustering 2192 were clustered successfully, after revision we used this clusters to label our data. 699 term had to be categorized and labelled by hand. We identified 70 categories, each for on entity. The performance of the model shown in Table 3.

Table 3. The models Precision, Recall and F-Score

Precision	***Recall***	***F-Score***
94.67	92.99	93.82

Due to the automatic recognition of machine components, the right maintenance personnel can be notified automatically, thus the maintenance effectivity can be increased.

5 Conclusion

The NLP analysis of maintenance logs and error messages provides promising result. As seen above, the presented methods are well applicable to speed up and improve the maintenance processes and the maintenance strategies. The formerly mentioned techniques can delivers fast results, especially when large scale, long-term maintenance logs are available.

The terminology of maintenance and industrial processes differs from the everyday language, but with appropriate pre-processing, the identification of the relevant stop words and the proper weighting functions, the processing of maintenance logs and fault messages are just as simple as the processing of average texts.

The machine learning methods, also the natural language processing have great potential in the industrial sector. The application of these algorithms can improve the quality of products, manufacturing and control processes [12], and the maintenance processes and strategies, thereby the Overall Equipment Effectiveness (OEE) of the production facilities. Further research on this topic is recommended.

6 Further goals

Besides improving the presented models and methods, further goals are to visualise the faulty machine components on the machine blueprints and layouts, to load the corresponding documentations and drawings of the components automatically, and based on the fault messages to display similar faults and their potential solutions from maintenance logs with using machine

learning algorithms. To extend these models behaviour with the analysis of the process signals and diagnostic measurements, to achieve prescriptive maintenance is also the aim of the further work.

7 References

[1] Lee, W. J. et al. (2019). Predictive Maintenance of Machine Tool Systems Using Artificial Intelligence Techniques Applied to Machine Condition Data. Procedia CIRP. 80. 506-511. doi:10.1016/j.procir.2018.12.019.

[2] Liu, R., Yang, B., Zio, E. & Chen, X. (2018). Artificial intelligence for fault diagnosis of rotating machinery: A review. Mechanical Systems and Signal Processing. 108. 33-47. doi:10.1016/j.ymssp.2018.02.016.

[3] Pelham, J.G., & Hockley, C. (2017). Analysis of Short form Maintenance Records for NFF Using NLP, Phrase Matching, and Bayesian Learning.

[4] Stenström, C., Aljumaili, M. & Aditya, P. (2015). Natural language processing of maintenance records data. International Journal of COMADEM. 18. 33-37.

[5] TF Manning, C.D., Raghavan, P. & Schutze, H. (2008). "Scoring, term weighting, and the vector space model" (PDF). Introduction to Information Retrieval. p. 100. doi:10.1017/CBO9780511809071.007. ISBN 978-0-511-80907-1.

[6] Pedregosa et al., (2011). Scikit-learn: Machine Learning in Python, JMLR 12, pp. 2825-2830

[7] Bird, S., Loper, E. & Klein, E. (2009). Natural Language Processing with Python. O'Reilly Media Inc.

[8] Honnibal, M. & Montani, I. (2018). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

[9] Goldberg, Y. & Levy, O. (2014). "word2vec Explained: Deriving Mikolov et al.'s Negative-Sampling Word-Embedding Method". arXiv:1402.3722 [cs.CL].

[10] Ester et al. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96). AAAI Press. pp. 226–231. CiteSeerX 10.1.1.121.9220. ISBN 1-57735-004-9.

[11] Singhal, A. (2001). Modern Information Retrieval: A Brief Overview. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 24, 35--43.

[12] Szűcs, B. & Ballagi, Á. (2019). Reducing Pseudo-error Rate of Industrial Machine Vision Systems with Machine Learning Methods, Acta Technica Jaurinensis. doi: 10.14513/actatechjaur.v12.n4.511.