Journal of Intelligent Manufacturing
https://doi.org/10.1007/s10845-024-02323-4

Technical language processing for Prognostics and Health Management: applying text similarity and topic modeling to maintenance work orders

Sarvesh Sundaram[1] · Abe Zeid[1]

Received: 22 June 2023 / Accepted: 5 January 2024

Abstract

Modern manufacturing paradigms have incorporated Prognostics and Health Management (PHM) to implement data-driven methods for fault detection, failure prediction, and assessment of system health. The maintenance operation has similarly benefitted from these advancements, and predictive maintenance is now being used across the industry. Despite these developments, most of the approaches in maintenance rely on numerical data from sensors and field devices for any sort of analysis. Text data from Maintenance Work Orders (MWOs) contain some of the most crucial information pertaining to the functioning of systems and components, but are still regarded as 'black holes', i.e., they store valuable data without being used in decision-making. The analysis of this data can help save time and costs in maintenance. While Natural Language Processing (NLP) methods have been very successful in understanding and examining text data from non-technical sources, progress in the analysis of technical text data has been limited. Non-technical text data are usually structured and consist of standardized vocabularies allowing the use of out-of-the-box language processing methods in their analysis. On the other hand, records from MWOs are often semi-structured or unstructured; and consist of complicated terminologies, technical jargon, and industry-specific abbreviations. Deploying traditional NLP to such data can result in an imprecise and flawed analysis which can be very costly. Owing to these challenges, we propose a Technical Language Processing (TLP) framework for PHM. To illustrate its capabilities, we use text data from MWOs of aircraft to address two scenarios. First, we predict corrective actions for new maintenance problems by comparing them with existing problems using syntactic and semantic textual similarity matching and evaluate the results with cosine similarity scores. In the second scenario, we identify and extract the most dominant topics and salient terms from the data using Latent Dirichlet Allocation (LDA). Using the results, we are able to successfully link maintenance problems to standardized maintenance codes used in the aviation industry.

Keywords Prognostics and Health Management · Maintenance · Natural language processing · Sentence similarity · Topic modeling · Smart manufacturing

Introduction

Maintenance has become an integral part of manufacturing and industrial operations. Almost every system and product that is in existence today requires some form of maintenance. The needs of a growing global population can be directly associated with the increase in consumption of goods and services, requiring the maintenance of more systems and equipment. Many industries are operating at their maximum level of capacity utilization, resulting in the reduced maintenance times for machinery and equipment (Krolikowski & Naggert, 2021). Studies have shown that maintenance can take up a large fraction of an organization's operational budget (Garg & Deshmukh, 2006), with manufacturers in the United States spending approximately $50 billion on maintenance costs (Thomas, 2018). It is becoming evident that maintenance should not be viewed purely as an isolated function, but instead as a competitive strategy (Mwanza & Mbohwa, 2015). This highlights the growing significance

✉ Sarvesh Sundaram
sundaram.s@northeastern.edu

[1] Department of Mechanical and Industrial Engineering, College of Engineering, Northeastern University, Boston, MA 02135, USA

Published online: 24 February 2024

of maintenance operations. Recent strains on global supply chains caused by the Covid-19 pandemic place an even greater emphasis on the importance of minimizing downtimes to make up for the time lost due to interruptions elsewhere.

Developments in Artificial Intelligence (AI) with Machine Learning (ML), Deep Learning (DL), and Natural Language Processing (NLP) are revolutionizing several industries and the way some tasks are performed. Self-driving cars, AI-based chatbots and real-time identification of objects on smart devices are just some of the real-world applications of these technologies. In the context of manufacturing and industrial operations, the area of Prognostics and Health Management (PHM) has incorporated AI-based methods to detect, predict and in some cases, diagnose failures. Maintenance strategies such as unplanned or reactive maintenance and planned or preventive maintenance are being supplemented by approaches such as predictive maintenance that use AI (Sundaram & Zeid, 2021). With the introduction of these state-of-the-art methods, maintenance approaches can reduce excess inventory, minimize system downtime, and improve overall performance. However, even with these developments, the modern maintenance operation faces several challenges such as (1) the lack of appropriate ontologies to integrate maintenance processes, (2) complexities involved in the implementation of specific maintenance strategies, (3) shortage of well-trained manpower for the maintenance of crucial systems, and (4) limitations in the analysis of multimodal and unstructured data from heterogeneous systems.

Industry 4.0 and Smart Manufacturing based architectures are promoting digitization of the industry and making manufacturing operations more interoperable (Zeid et al., 2019). These hierarchy-free models are enabling the planning and implementation of maintenance processes in a sustainable manner. Several reference ontology models have been proposed (Karray et al., 2012, 2019; Montero Jiménez et al., 2023) to address the challenge of integration of different maintenance processes. The second and third challenges that face modern maintenance are closely related. The complexities and difficulties associated with implementation of some maintenance approaches is exacerbated by the scarcity of trained technicians and engineers (Federal Energy Management Program, 2021). These concerns can be mitigated to some extent by technologies such as Augmented Reality (AR) and Remote Maintenance (RM) (Masoni et al., 2017; Mourtzis et al., 2017), and can enable faster training of the workforce and provide real-time assistance on the shopfloor. To analyze data from the shopfloor, data-driven PHM methods have successfully used sensor measurements to detect faults and predict the Remaining Useful Life (RUL) of systems. Sensor signals are often numerical or image-based, and AI models have been fine-tuned for such modes of data. While this addresses some of the requirements of the fourth challenge of modern maintenance, it does not tackle all modes of data.

One part of maintenance that contains troves of data but is yet to be exploited, is the technical text data stored in Maintenance Work Orders (MWOs). Text data from MWOs are sometimes considered as 'black holes', i.e., they are fed with so much data, but are seldom used to make data-driven decisions. Instead, they are often regarded as historical logs that are relied upon only when there is an absolute necessity. NLP can offer some hope with its ability to analyze text data and provide appropriate solutions where necessary. However, NLP's successes predominantly come from the analysis of text that are a part of non-technical sources. NLP can be used on different types of data using domain-adaptation or transfer-learning, but this assumes that to process data fromlow resource domains, there are high-resource domains that are somewhat similar consisting of annotated data (Ben-David et al., 2010). That is not the case when it comes to technical data, specifically data from MWOs. These documents are often unstructured or semi-structured, they adopt unique language dictionaries, and use colloquialisms and jargon that are domain specific. In order to process text data, the approach of Technical Language Processing (TLP) has been proposed (Brundage et al., 2021; Dima et al., 2021; Linhares & Dias, 2003). TLP uses a human-in-the-loop strategy to address the challenges posed by technical text data with the help of customized NLP methods.

The research presented in this work is aimed at highlighting how TLP can be transformative in PHM and improve the maintenance operation. First, we review the maintenance approaches and the role played by PHM. We then explain what non-technical and technical text data are, and how text from MWOs is unique. We demonstrate how traditional NLP fails on technical text, the why there is a need for TLP. We propose a TLP framework to implement in the PHM environment and suggest some potential areas of its use. To demonstrate the capabilities of the proposed framework, we identify two scenarios using MWOs of aircraft and apply customized text processing methods to it. We first use TLP to predict corrective actions for new maintenance problems using semantic and syntactic text similarity and evaluate it using the cosine similarity score; then we use Topic Modeling to extract the most dominant topics from the MWOs and use relevant keywords to link problems to standardized maintenance codes.

Maintenance strategies and the role of Prognostics and Health Management

There are several approaches to maintenance that have been adopted across the industry. Unplanned/reactive maintenance, also known as run-to-failure maintenance, is one of the traditional maintenance strategies in which the machine or component fails before any corrective measure is taken. This strategy can be expensive due to the high costs associated with repairing and replacing the machines or components, and additional losses associated with unplanned system shutdown (Basri et al., 2017). Another widely used approach is preventive maintenance in which the system is inspected at regularly scheduled intervals to identify any potential issues that might arise. In most cases, preventive maintenance is planned well in advance to be implemented at a set-time in the future and in order to minimize failure probability of a specific system/equipment (Kimura, 1997). Small and Medium-Sized Enterprises (SMEs) account for majority of the industry and Gross Domestic Product (GDP) (Offices of Industries & Economics, 2010), and rely largely on reactive or preventive maintenance approaches (Jin et al., 2016). Therefore, any improvements made to maintenance practices in SMEs will result in a tremendous amount of savings and potentially impact other areas of the industry as well.

The introduction of data-driven methods with ML and DL are bringing proactive maintenance approaches to the forefront (Sundaram & Zeid, 2021). Condition-monitoring of systems using Internet of Things (IoT) based sensors and field devices allow the measurement of system parameters in real-time that can be used to record large amount of data for analysis. This can be used along with historical data and domain knowledge to predict when failures would occur or how close they are to occurring. These methods have been used successfully in fault diagnosis of rotating machinery (Jiang et al., 2023), defect identification of stainless steel welds (Zhang et al., 2023), in monitoring and predicting the quality of solutions in electroplating (Granados et al., 2020), and in several other areas. While these methods provide excellent results in the real-world, they have limitations in terms of the types and formats of the data that can be used. Sensor signals and other measurements primarily take the form of numeric or image data. These methods are not equipped to handle text data, especially the technical text data from domain-specific maintenance operations.

Prognostics and diagnostics of systems and components can be enhanced if the analysis of technical text data can be incorporated into existing approaches. Consider the timeline for a typical maintenance operation shown in Fig. 1. There are many steps involved right from detecting a breakdown, creating an MWO, to fixing the problem, and closing the MWO. We can also observe how much time is spent in the entire process. In the current maintenance timeline, there are several opportunities for technical text processing that can help save time. In a reactive maintenance setting, once a breakdown is detected, corrective actions could be predicted based on the problem text. Similarly, in predictive maintenance, although time is saved by predicting when a failure would occur, time is still spent in diagnosing the type of failure and in repairing/replacing the necessary parts. In this case too, analyzing technical text can help speed up the process. There are a few more potential applications and use-cases for TLP in the context of the typical maintenance operations timeline that we will discuss.

Text data

Non-technical text

We interact with non-technical text data presented to us on a daily basis via different forms of media. On our mobile devices and computers, text data usually takes the form of digital newspaper articles, webpages, blogs, Short Message Service (SMS) text, e-books, etc. Almost all of these text data are non-technical and based on spoken languages or are derivatives of spoken languages. Since these languages are a part of everyday interaction, there is a lot of scope to use the data from it, which is exactly what NLP methods are built for. One of the main requirements for NLP models to succeed is the availability of large amounts of text data, also referred to as corpora, for model training purposes. Since non-technical data is available in abundance, it is not a problem for the NLP pipeline. Such non-technical text data have been assembled into several corpora and have been made publicly available. Corpora such as Penn Treebank Corpus (Marcus et al., 1993), British National Corpus (BNC Consortium, 2007) and WikiText-2 (Merity et al., 2017) are used for the generic training of NLP models, and each of them contain tens of thousands, if not millions of words. Some corpora are designed for specific tasks. Amazon Customer Reviews (Bhatt et al., 2015) and IMDb Movie Reviews (Maas et al., 2011) are commonly used to train models for Sentiment Analysis. The Reuters Corpus (RCV1) (Rose et al., 2002) is routinely used for text clustering. Similarly, there are corpora designed for Named-Entity Recognition (NER) and Parts-Of-Speech (POS) tagging tasks. Genre-specific corpora have also been compiled for formal text from academic articles, journals, and conferences; informal text from emails, blogs, and webpages; spoken language from language databases; etc. This brings to light how resource-rich some of the non-technical areas are, and how NLP models can be trained to achieve high levels of accuracy in their tasks.

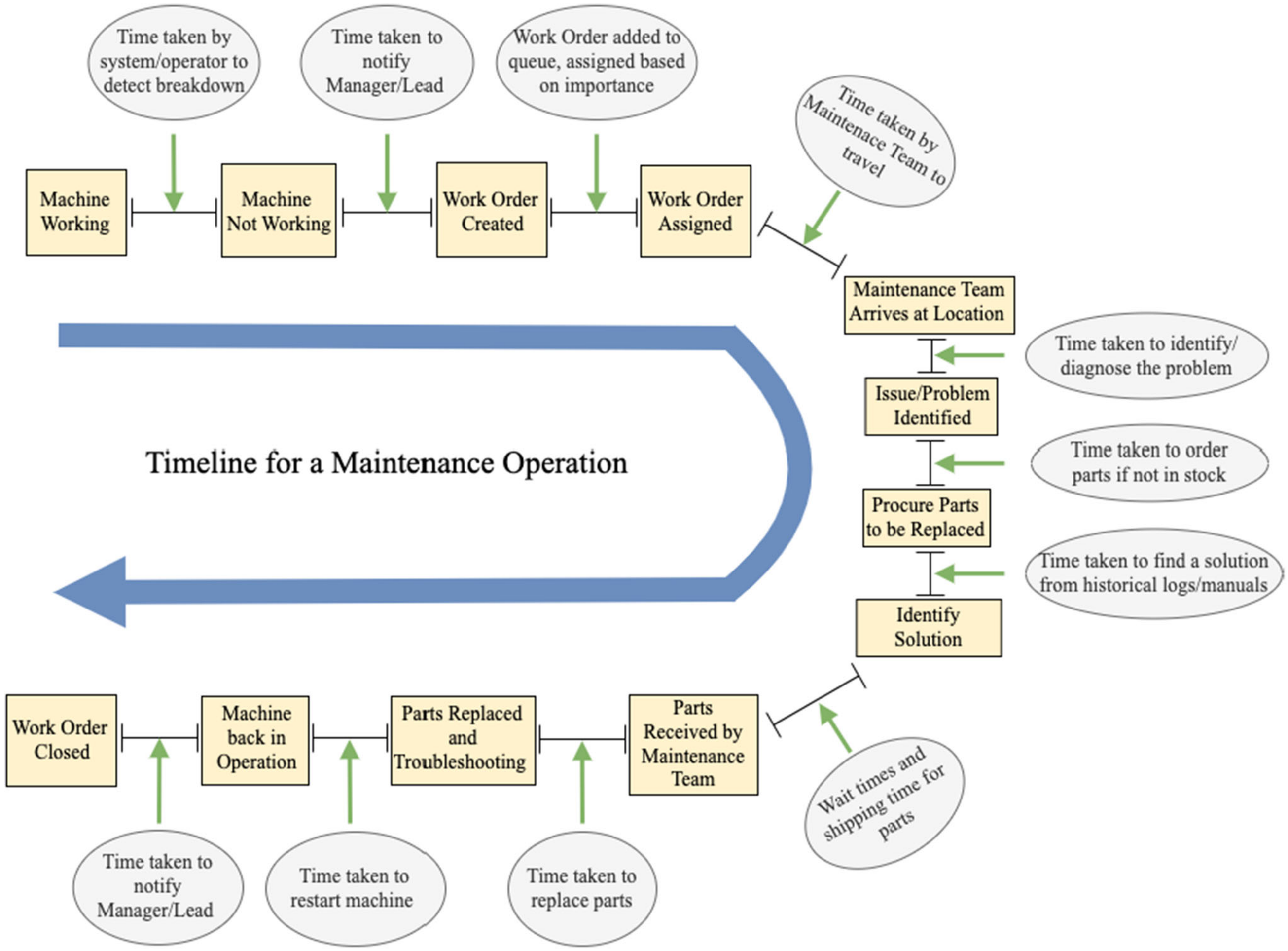
Time taken by system/operator to detect breakdown
Time taken to notify Manager/Lead
Work Order added to queue, assigned based on importance
Time taken by Maintenace Team to travel
Machine Working
Machine Not Working
Work Order Created
Work Order Assigned
Maintenance Team Arrives at Location
Time taken to identify/ diagnose the problem
Issue/Problem Identified
Time taken to order parts if not in stock
Procure Parts to be Replaced
Time taken to find a solution from historical logs/manuals
Identify Solution
Timeline for a Maintenance Operation
Work Order Closed
Machine back in Operation
Parts Replaced and Troubleshooting
Parts Received by Maintenance Team
Wait times and shipping time for parts
Time taken to notify Manager/Lead
Time taken to restart machine
Time taken to replace parts

Fig. 1 Timeline for a maintenance operation

Technical text

The term 'Technical text' is not a clearly defined one and has been used to described text data from various fields (Copeck et al., 1997). In the engineering and scientific community, the term 'technical' can take a few different meanings. It could signify the complex mathematical formulae and equations, or the sophisticated terminology used in a specific domain. It could also be considered as a subset of a language used by experts that are familiar with the knowledge about a specific task or process. In the field of law, the question of whether law and the associated legal text can be considered to be technical has been studied in detail (Schauer, 2015). Arguments have been made that law is to be interpreted as a technical language when it is intended to be understood and used by legal professionals (Fuller, 1958; Holmes, 1997; Horvath, 1954). Medical terminologies can be considered to be technical text too (Chung & Nation, 2004; Schironi, 2010). Similarly, there is technical text in chemistry (Meshalkin et al., 2015), in physics (Alexander & Kulikowich, 1994), and in molecular biology (Krallinger & Valencia, 2005; Wilbur & Yang, 1996). In manufacturing, technical data is observed in maintenance (Stenström et al., 2015), and also in product inspection logs (Sundaram & Zeid, 2023). It is clear that technical text exists in science, engineering, medicine, law, and several other domains. Although these types of text data are comfortably understood by domain experts, it is not as easily understood by text processing models. Figure 2 shows examples of technical text data from aerospace (Dangut et al., 2021), healthcare (Fleurence et al., 2014), chemistry (Tice et al., 2013), general engineering (Hodkiewicz et al., 2017), and nuclear energy operations (Olack, 2021) domains.

Text from maintenance work orders

MWOs are critical documents that record data from each step of a maintenance operation. These documents can provide an in-depth understanding of the system's performance over time, allowing the extraction of valuable knowledge. This type of technical data is unique due the fact that it

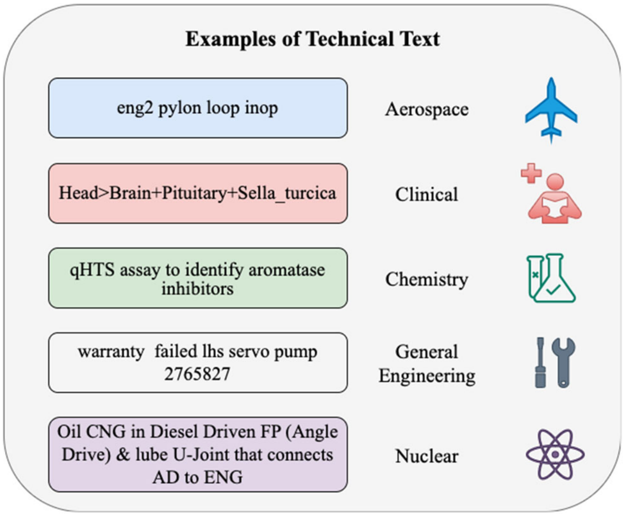
Examples of Technical Text
eng2 pylon loop inop
Aerospace
Head>Brain+Pituitary+Sella_turcica
Clinical
qHTS assay to identify aromatase inhibitors
Chemistry
warranty failed lhs servo pump 2765827
General Engineering
Oil CNG in Diesel Driven FP (Angle Drive) & lube U-Joint that connects AD to ENG
Nuclear

Fig. 2 Examples of technical text data from various disciplines

is generated by a human source as opposed to being readings from a machine or instrument (Bokinsky et al., 2013). The records essentially consist of thoughts about a specific problem expressed by technicians in a language they are comfortable using. Many industrial operations, particularly Small and Medium-sized Enterprises (SMEs), use hand-written formats for MWOs. The digitization of the shopfloor has made it possible to store data in a more efficient manner. Modern MWOs use Computerized Maintenance Management Systems (CMMS) to record data in a semi-structured format (Woods et al., 2020). The text used in MWOs differs significantly from other corpora, including a lot of technical ones, because engineers, operators, and technicians tend to use domain specific verbiage and jargon. MWOs often comprise of inconsistent language, incomplete entries, or sometimes no entries at all (O'Donoghue & Prendergast, 2004). In some cases, different shorthand notations are used to refer to the same component, which can lead to discrepancies (Rajpathak & Chougule, 2011). Text used in MWOs are also much shorter when compared to other text in the NLP corpora, even though they may contain the same number of records (Dima et al., 2021). MWOs typically consist of a unique identifier, the problem, action taken, and timestamp of when it was created and closed, etc. (Navinchandran et al., 2022). Table 1 shows a few attributes from MWOs for Airbus A320 aircraft (Witteman et al., 2021) as an example.

In Table 1, A/C depicts the aircraft number, Item conveys the maintenance task number, Description provides a description of the maintenance task, LAST EXEC DT shows when the last maintenance action was completed, and LIMIT EXEC DT shows the date before which the next inspection must be completed. We can observe the complex nature of MWOs, i.e., it is not easy to understand unless one is a domain expert.

Inability of natural language processing to handle technical text

NLP has been successful at adapting to a few technical disciplines. Technical text from medicine has been processed with good results (Chen et al., 2018; Zhou & Hripcsak, 2007). In supply chain, semantic text matching has been used in the management of transportation assets (Le & David Jeong, 2017). NLP has also adapted to technical text from finance in the form of chatbots (Khurana et al., 2023). With these successful adaptations, one may wonder why it cannot be used for text data from MWOs. We use insights from previous works (Brundage et al., 2021; Dima et al., 2021) to understand the drawbacks of NLP for technical text from MWOs.

Consider the second maintenance problem from Table 1 that reads *OUT FLAP L/E PANELS NO.S 2 AND 5 – INSTALL*. The maintenance jargon and short hand is not very easy to understand. Translated to spoken English, it means "out flaps on leading edge panels number 2 and 5 to be installed". To demonstrate the limitations of NLP on technical text, we apply a typical NLP pipeline (Alharbi et al., 2021; Mahmoudzadeh et al., 2020) as seen in Fig. 3 to preprocess the text from the MWO. In the first step of Tokenization, algorithms segment the text data into words and phrases, and these tokens become the input for the NLP task (Webster & Kit, 1992). The tokenizer converts the string of technical text into individual tokens, but the short hand of *L/E*, used to denote the leading edge is not tokenized correctly. It is split up into separate tokens for each character instead. We then use the individual tokens from tokenization as the input for the next step of stop word removal. Stop words are considered to be unimportant words that provide little information (Manning & Schutze, 1999). Removal of words such as *out*,

Table 1 Example of an MWO for Airbus A320 aircraft

A/C	Item	Description	LAST EXEC DT	LIMIT EXEC DT
Aircraft 2	572054-01-4	FUEL PUMPS 21QA AND 25QA - INSTALL	10/26/12	10/26/24
Aircraft 24	575303-01-1	OUT FLAP L/E PANELS NO.S 2 AND 5 - INSTALL	11/17/15	11/17/21

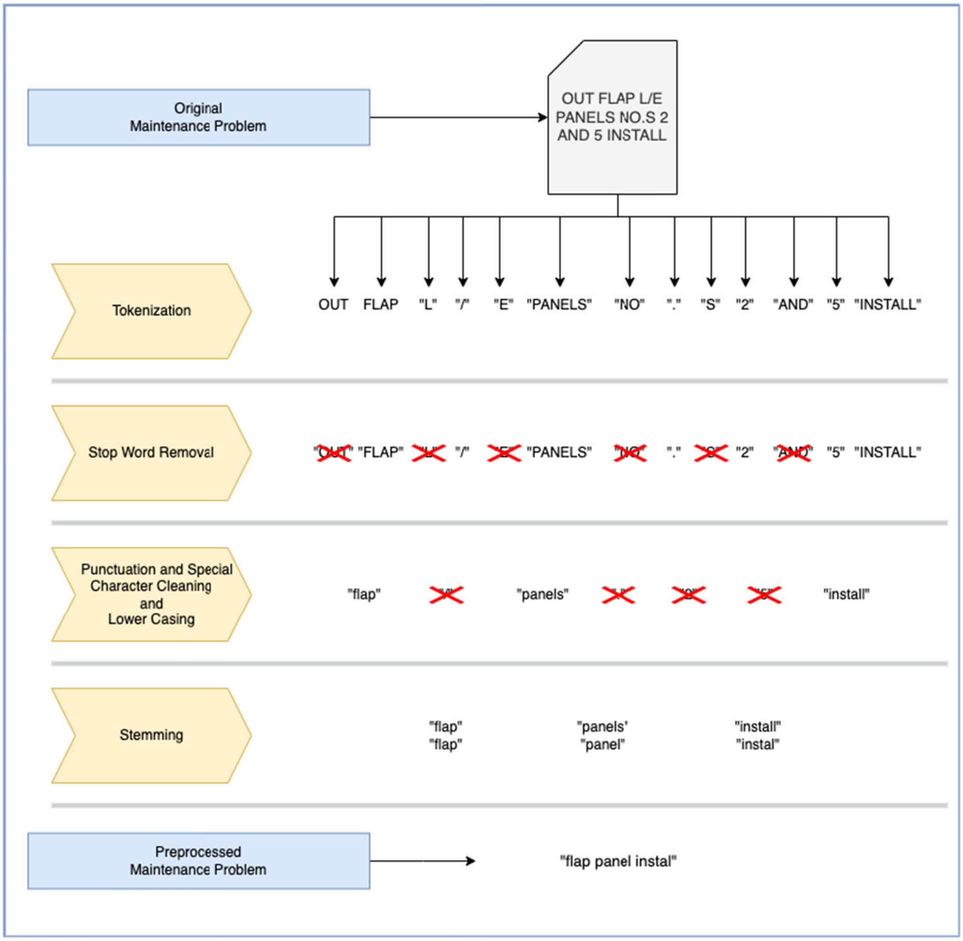
Original
Maintenance Problem
OUT FLAP L/E PANELS NO.S 2 AND 5 INSTALL
Tokenization
OUT FLAP "L" "/" "E" "PANELS" "NO" "." "S" "2" "AND" "5" "INSTALL"
Stop Word Removal
~~"OUT"~~ "FLAP" ~~"L"~~ "/" ~~"E"~~ "PANELS" ~~"NO"~~ "." ~~"S"~~ "2" ~~"AND"~~ "5" "INSTALL"
Punctuation and Special Character Cleaning and Lower Casing
"flap" ~~"/"~~ "panels" ~~"."~~ ~~"2"~~ ~~"5"~~ "install"
Stemming
"flap"
"flap"
"panels'
"panel"
"install"
"instal"
Preprocessed
Maintenance Problem
"flap panel instal"

Fig. 3 Traditional NLP's flaws when used on technical text from MWOs

no, *and*, and the letters *l*, *e*, and *s* may be useful in a non-technical scenario but are very important in the context of MWOs. In this case, the letters *l* and *e* state that the part to be installed is along the leading edge panel. Stop word removal also removes the word *no*, by assuming that it is used to convey negation, when in reality it is short hand for "number" in the context of this MWO. The next step of text cleaning performs some basic operations such as punctuation and special character removal, and lower casing of the text string. This step removes the punctuation mark used for forward slash ("/"), and also removes the numbers *2* and *5* that are used to identify the panels. The following step of stemming is used to find the stem words or the root words. The word *panels* is reduced to its stem, "panel" and *install* is reduced to "instal". Technical jargon and abbreviations would not be stemmed to their separate root words due to their complex nature. This concludes the preprocessing stage and results in the preprocessed text reading "flap panel instal". Although this phrase consists of some of the keywords from the original technical text, it has completely lost its meaning. The location of the leading edge has been lost during the preprocessing, and number identifying the panels has also been discarded by the NLP pipeline. If this processed text string were to be used as input to a prediction model, the results could misrepresent the severity or seriousness of the actual maintenance problem, or in some case, even lead to hazardous consequences.

The complexities involved in processing technical text from MWOs are quite clear, and traditional NLP is not quite suited for this task. Domain adaptation from other technical domains is also not possible because the maintenance data is so unique. Transfer learning cannot be used either due to the absence of similarly structured technical data on which models can be trained. It is evident that domain-knowledge needs to be incorporated into the text processing pipeline considering the peculiarities of technical text used in MWOs. To demonstrate how this can be made possible, we propose a framework to implement TLP to process technical text from MWOs that uses expert knowledge.

A technical language processing framework for Prognostics and Health Management

One of the major reasons why NLP is unsuccessful on MWOs is that the complexity of technical text makes it extremely difficult for a computerized model to understand the context in which certain words or terminology are used. To overcome this obstacle, expert knowledge must be incorporated during various stages of processing for decision-making. We propose TLP framework for PHM that incorporates domain knowledge at each step of processing as shown in Fig. 4.

The first step in the TLP framework involves obtaining the MWOs that have been generated. These work orders can consist of a varying number of fields such as an identifier, the opening and closing times of the work order, the maintenance problem, the corrective action taken, the name and an identifier for the technician, the part replaced, the costs involved, etc. Since there is no universal format for MWOs, it is essential to understand the technical text that is presented in it. To decipher the short hand, technical jargon, abbreviations, and other colloquialisms used, expert knowledge needs to be utilized. Historical data and use-cases are used as references to analyze raw data that is presented (Brundage et al., 2021). Dictionaries with technical terminology, abbreviations, morphosyntactic information, stop words and other relevant data are collected by talking with technicians, operators, and maintenance engineers. This data, in addition to the data from sources such as maintenance manuals, CMMS, and other metadata are collectively referred to as "fortuitous" data (Dima et al., 2021; Plank, 2016).

Once there is sufficient understanding of the text, appropriate preprocessing steps can be selected. Methods such as Tokenization, stop word removal and lemmatization can be used in a such a way that it preserves the meaning of the original text. Unnecessary stop words are screened, technical jargon is substituted with understandable verbiage, and any punctuations that do not alter the meaning of the data are removed. The preprocessed text is then examined to ensure that the quality of the data has not been lost. If it is deemed that the preprocessing is not satisfactory, appropriate steps are revised.

The preprocessed technical text can now be used for a variety of tasks. There are several areas of PHM and maintenance that can benefit from TLP. With the help of domain experts and analysts, the areas where TLP will be effective can be identified, and appropriate computational resources can be procured. TLP can be used to predict corrective actions for new maintenance problems by comparing them with existing ones from historical MWOs. In the same vein, it can also be used to match the technical text to identify what parts might need to be replaced and refer the technicians to the necessary manuals/documentation. If there is some form of annotating to be performed on MWOs, a TLP model can assist performing that task too. Another potential area of where TLP can optimize the maintenance process is by incorporating it directly into a predictive maintenance framework. Such prescriptive maintenance systems that are able to handle multimodal data have been proposed (Ansari et al., 2019, 2020) but it remains to be seen how they can be incorporated across the industry. Nevertheless, it shows that there are many areas where TLP can be used with PHM to improve maintenance.

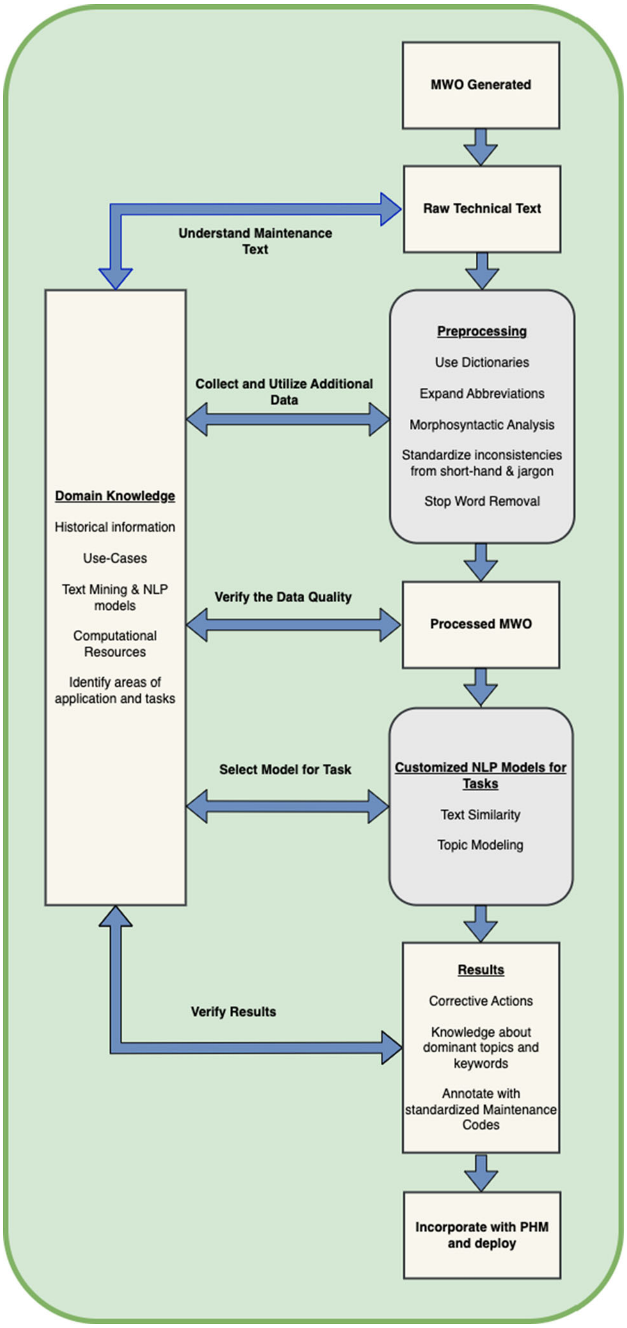

Fig. 4 A framework for technical language processing

Once it is established where TLP would be effective in the maintenance timeline, text processing methods compatible with the tasks can be identified. Our understanding is that text similarity methods with semantic and syntactic similarity can be very useful for a variety of tasks—matching new maintenance problems to existing problems actions to predict corrective actions, using keywords from processes to identify replacement strategies for tools/components, and several other tasks. Another suitable method for data from MWOs is Topic Modeling. It identifies the most dominant topics and keywords from maintenance problems, helping to quantify what systems/components utilizes the most resources. Topic Modeling also assists in linking MWOs from dominant topics to standardized industry codes, making annotation of the data easier for future tasks. The results from all of the methods applied to the preprocessed MWOs are evaluated by experts in the field to verify its validity.

When the performance of the text processing methods is deemed to be satisfactory, i.e., a sufficient number of corrective actions have been successfully predicted for new maintenance problems, or new problems have been effectively linked to appropriate topics, the next steps are considered. These steps involve deploying it as a part of CMMS, or as a standalone tool for PHM. As noted previously, it can also be incorporated as a part of the predictive maintenance paradigm. The goal of the TLP framework to enable quick and efficient decision making in the maintenance operation by assisting engineers, technicians, operators, and maintainers. There are many possible use-cases for such a framework across several industries. To demonstrate the potential applications of the proposed framework, we identify two scenarios in which it can be used to improve the maintenance operation of aircraft.

Applying technical language processing to aircraft maintenance work orders

There are a limited number of publicly available datasets with technical text. We use the Aviation Maintenance dataset that is originally from University of North Dakota's Aviation Program, made available by researchers (Akhbardeh et al., 2020a) on the open-source web-interface called Maintnet.[1] The data consists of three attributes out of which only two are of interest to us—the Problem which contains text about the maintenance problem/issue, and Action or which contains text that describes the action taken to address the problem. A set of supplemental data files that contain abbreviations dictionaries, morphosyntactic information, and domain term banks are also provided. Table 2 shows four randomly selected records from the MWO.

Table 2 Randomly selected records from aviation maintenance data available at Maintnet (see Fn. 1)

Ident	Problem	Action
100111	L/H engine #4 AFT baffle cracked in multiple places	Installed repaired baffling
100532	Scroll to PC flitter tube has excessive fretting under float	Ordered line
103139	Accidentally oversped engine before pulling throttle back, E	Evaluated data on FDM card & found engine overspeed to
104077	CYL #2 & 4 rocker cover gaskets are leaking	Removed & replaced rocker cover gaskets

The creators of Maintnet have specified that the original data consisted of text that represented the maintenance problem, the action, the ATA chapter codes (maintenance codes as prescribed by the Air Transport Association), open and close dates for the MWOs, and a work order identifier (Akhbardeh et al., 2020b). However, due to the presence of sensitive information and privacy concerns, the data was de-identified and confidential information was removed from the publicly available version. The absence of annotations makes it challenging to apply any form of text processing methods especially since there are only two attributes. However, this opens up a window of opportunity for unsupervised methods to be used.

Given how the technical text from the aviation MWOs is unstructured, we devise two scenarios to demonstrate the capabilities of TLP. First, we use the problem and action attributes as historical data and apply syntactic and semantic text similarity models to predict corrective actions for new maintenance problems. In the second scenario, we apply Topic Modeling using Latent Dirichlet Allocation (LDA) to identify the most important topics and salient terms and link them to the standardized ATA maintenance codes.

Preprocessing maintenance work orders

The preprocessing steps we consider are broadly the same for both the scenarios, barring a few minor differences that will be highlighted. As we have seen in Table 2 the technical text from the MWOs of aircraft consist of abbreviations, short-hand, and some aviation maintenance specific jargon. To ensure that we capture all the important information from the data, we need to follow some preprocessing steps. Traditional NLP preprocessing steps such as Tokenization and English language stop word removal can significantly alter the meaning of the text from the MWOs. For example, technical terms such as *reswaged* and *resafetied* may not be tokenized correctly. Models trained on non-technical data

[1] https://people.rit.edu/fa3019/MaintNet/.

will not be able to tokenize the technical slang. Words such as *On* and *Off* provide very important context in the MWO. The removal of stop words based on the English language might result in the elimination of important technical terms that could potentially alter the meaning of the maintenance problem and subsequently influence any maintenance decisions. Therefore, a custom stop word list would be more appropriate for text from MWOs.

The first step in preprocessing is converting abbreviations, short-hand, and colloquialisms into their full versions or expanded forms respectively. This will allow easier understanding of these phrases and will be more presentable to an audience that is not exclusively comprised of domain experts. We use the list of abbreviations provided with the aviation dataset. But upon further examination of the data, we realize there are many more abbreviations that needed to be expanded. Terms such as *r/h* and *l/h* meaning “right hand” and “left hand” respectively have been used extensively in the MWOs. Other short hand included *i/b* for “inboard”, *o/b* for “outboard”, *a/c* for “aircraft”, *cht* for “cylinder head temperature”, *egt* for “exhaust gas temperature”, *c/w* for “complied with”, *fdm* for “flight data monitoring”, and many more. We tried to be as exhaustive as possible in the task of replacing abbreviations with the full words and phrases. In the next step, we focus on eliminating stop words. We know that stop words can either be a nuisance or of great value in technical text. So, we need to ensure that the stop words we choose to remove are appropriate in the context of the MWOs. We identified a list of stop words suitable for TLP (Sarica & Luo, 2021). However, words in this list were selected from various technical texts that were not domain specific. For the MWOs in consideration, this is not a complete list that is representative of the text. We customize the list of stop words so that they do not drastically change the meaning of the text. There are some words in the list such as *along*, *good*, *many*, *straight*, *forward*, and *upon* which are important in providing context to the maintenance problem and actions, so they are retained.

Scenario 1: predicting maintenance actions using text similarity

The value of saving time by improving maintenance processes is often ignored or overlooked (Mobley, 2002). In situations where delays in repairs could be very expensive, improving maintenance times becomes a priority. In the aviation industry, devising corrective actions for new maintenance problems is a time sensitive process in a highly stressful environment (Latorella & Prabhu, 2000). Any errors made in such situations could be catastrophic. To assist in the decision making process in such a highly challenging environment, TLP could be put to use.

In this scenario, we use the text data from the MWOs to predict corrective actions for new maintenance problems. We use the keywords present in the new maintenance problem and compare them with ones from historical problems in the MWOs. Two approaches are considered: syntactic text similarity and semantic text similarity. There are research works that apply these methods to the electric power industry operations (Wan et al., 2021), biomedical texts (Phan et al., 2019), medical records (Warnekar & Carter, 2003), railway safety (Qurashi et al., 2020), production line failures (Tekgöz et al., 2023), and also to aviation maintenance (Naqvi et al., 2022). Our aim is to place a greater emphasis on how these methods can be applied to technical data and how effective they are in the context of improving the maintenance operation by assisting in the decision making process.

Syntactic textual similarity

When a new maintenance problem arises, it is first preprocessed to convert all the technical abbreviations to their expanded forms. Each word in the new maintenance problem is recorded in order to allow syntactic and semantic matching. To determine syntactically similar matches of the new maintenance problem, it is compared with all of the historical problems that have already been recorded in the MWOs. One might consider only exclusively selecting the historical problems that contain any of the keywords present in the new maintenance problem, but there is a potential issue with that approach. A stop word or a word that is extensively used across all maintenance problems might be present in the text. Figure 5 shows the raw text from new maintenance problem, the text after preprocessing, the extracted keywords, and how problems in the historical MWOs are matched with the keywords. In this instance, we see how the word *had* appears in many maintenance problems. If we use this approach to determine how close the existing maintenance problem is to the new problem, we could end up getting many records that are irrelevant due to the presence of an unrelated keyword, a stop word that might have slipped through, or even a technical word that is not pertinent to the meaning of the text. To address this, we use the Term Frequency-Inverse Document Frequency (TF-IDF) method to find the most common terms. The TF-IDF implementation in Python involves transforming all the text from the maintenance problem attribute into a feature matrix. First, the frequencies of all words in the MWOs are recorded. TF-IDF then incorporates a concept called inverse document frequency (Sparck Jones, 1972). This measures the amount of information provided by a word by looking at how frequently it is present in the corpus.

We use the Scikit-learn library’s (Pedregosa et al., 2011) TF-IDF implementation in Python. TF-IDF is defined as the product of $tf(t, D)$ (TF or term-frequency) and $idf(t, D)$

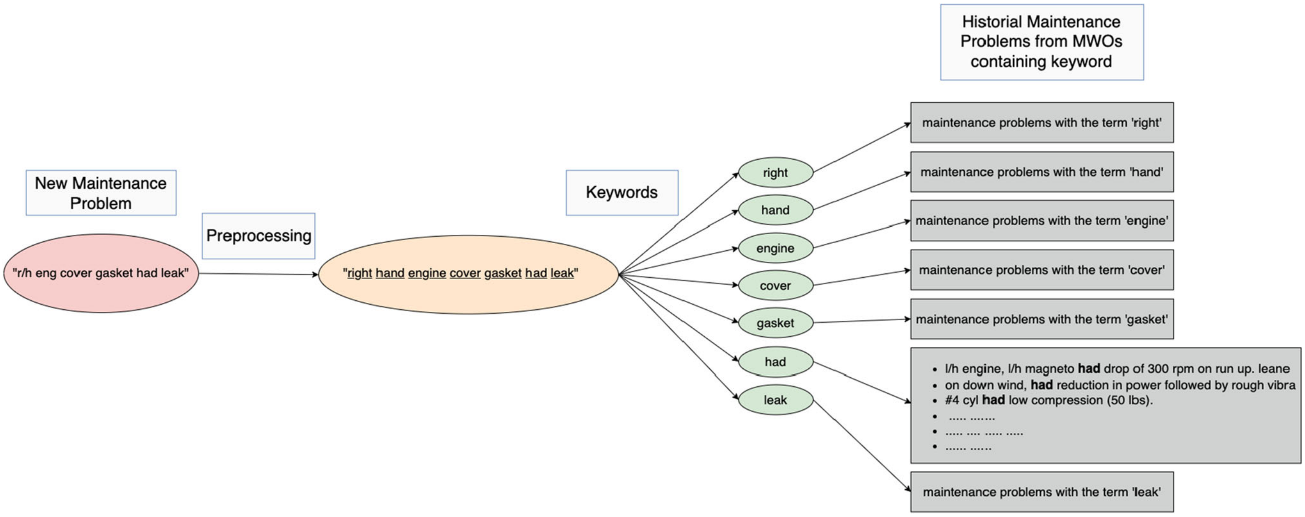
Historial Maintenance Problems from MWOs containing keyword
New Maintenance Problem
Preprocessing
Keywords
"r/h eng cover gasket had leak"
"right hand engine cover gasket had leak"
right
hand
engine
cover
gasket
had
leak
maintenance problems with the term 'right'
maintenance problems with the term 'hand'
maintenance problems with the term 'engine'
maintenance problems with the term 'cover'
maintenance problems with the term 'gasket'
l/h engine, l/h magneto **had** drop of 300 rpm on run up. leane
on down wind, **had** reduction in power followed by rough vibra
#4 cyl **had** low compression (50 lbs).
....
....
....
maintenance problems with the term 'leak'

Fig. 5 Keyword based matching

(IDF or inverse document frequency) where TF is the number of times a word or term appears in the total number of documents, and IDF computes how uncommon or rare a term is across all documents. The IDF can be mathematically expressed as shown in (1):

$$idf(t, D) = \log\left(\frac{(N+1)}{df(t)+1}\right) + 1 \tag{1}$$

In TF-IDF, t is each term or word, N is the number of documents (D), and $df(t)$ is the number of documents that contain the words t. This method uses smoothing, i.e., the constant of "1" is added to both the numerator and denominator to prevent divisions by zero. First the TF is calculated by considering each term from the string, which in our case is each word from the maintenance problem. Then, the TF-IDF is calculated by obtaining the product of TF and IDF [from (1)]. This converts the string from the maintenance problem into a vector. Similarly, TF-IDF is applied to all the problems from the MWOs, converting the entire document into a vectorized representation. For text similarity calculations, new maintenance problems are also converted into a vector. To identify how similar the vectorized versions of the new problem are to the existing problems, we use the cosine similarity metric, which is the angle between the two vectors using an inner product. Given two vectors A and B, the cosine similarity between them can be calculated as shown in (2):

$$cosinesimilarity(A, B) = \frac{A.B}{\|A\|\|B\|} \tag{2}$$

where $A.B$ is the dot product between vectors A and$B$; and $\|A\|$ and $\|B\|$ are the L2 norm of vectors A and B respectively. In our case, A can be considered to be an existing maintenance problem from the dataset, and B can be the new maintenance problem. To calculate their dot product which is the numerator of (2), the vector B would be transposed so that it can be multiplied with vector$A$. To calculate the denominator of (2), the L2 norm for each vector, A and$B$, is computed as the square root of the sum of squared vector values, and the results are multiplied. To put it into context, the vectorized representations of the maintenance problem will consist of numerical values representing each word. These values are squared and summed, and the square root of the total would result in the L2 norm. The product of L2 norms of the existing maintenance problem (A) and new maintenance problem (B) would provide us with the denominator for (2). Using these computed values, we can obtain the cosine similarity score between an existing maintenance problem and a new maintenance problem.

Semantic textual similarity

Implementing semantic text similarity is a little more complex as compared to syntactic text similarity. Methods such as Bag of Words (BoW) and TF-IDF cannot effectively capture the similarity between different words that convey the same meaning or idea (Chandrasekaran & Mago, 2021). It is even more difficult to detect different words that could convey the same concept when using technical vocabulary. According to research, semantic similarity can be classified into corpus-based similarity and knowledge-based similarity (Gomaa & Fahmy, 2013). Due to the lack of publicly available corpora for technical text data, specifically maintenance related data, we consider a pre-trained sentence transformer model. Transformer models are basically encoder—decoder models. The task of the encoder is to take an input of raw text and map it to a numerical sequence. The decoder then uses the output of the encoder along with other contextual information to generate a meaningful output (Vaswani et al., 2017). One of the state-of-the-art models is the Bidirectional Encoder Representations from Transforms or BERT (Devlin et al., 2019). For tasks such as semantic textual similarity, BERT has been modified with Siamese networks and is called Sentence-BERT or SBERT (Reimers & Gurevych, 2019). The sentence transformers library from Huggingface (Wolf et al., 2020) provides many different methods that we can use for our task of semantic textual similarity. Considering this is an unsupervised learning task, we choose to use the pre-trained model 'all-mpnet-base-v2'[2] model which is trained on more than 1 billion records for our task. When we feed the model with a new maintenance problem, the sentence transformer will consider all the historical maintenance problems and match them based on semantic similarity. We then use the cosine similarity metric to identify how semantically similar the new problem is to existing problems.

Results of textual similarity methods

To demonstrate how syntactic and semantic textual similarity can be used, we generate several new instances of maintenance problems. The first new problem is almost identical to a consistently appearing existing maintenance problem in the MWOs. The second maintenance problem is also identical, but only semantically. The third maintenance problem shows how close semantic and syntactic matches can be obtained if similar words appear in both pieces of text. Table 3 shows the new maintenance problems; their preprocessed versions after stop words removal and abbreviation expansion; the matching syntactically similar and semantically similar problems with their matching corrective action predictions and cosine similarity scores respectively. For each new maintenance problem, we pick the top three matching problems and their corresponding corrective actions. We can see that in the first instance, the syntactic and semantic matches both show existing problems that are very similar to the new problem, and this logical similarity is further justified by their cosine similarity scores. In the second instance, the new maintenance problem consists of the word *disintegrating*, which does not appear anywhere in the existing problems from the MWOs. However, using the sentence transformer model, semantically similar textual matches are found with the word *worn*. In the same instance, the syntactically similar matches do not appear to be a good fit. For the third new maintenance problem, the words *not clean* are semantically matched with the word *dirty* to find a corrective action. For the syntactic matches of the third problem, existing maintenance problems where engine cleaning is needed are identified with their respective corrective actions.

It is important to note here that the new maintenance problems considered have not been seen by the text similarity models. These new or unseen problems are not present in the MWO data, they are generated by us to demonstrate how text similarity would be useful in a real-world scenario. It is also necessary to mention that several existing maintenance problems in the MWO data are repeated. However, their corresponding corrective actions are different even though the maintenance problem is exactly the same. For example, consider the existing maintenance problem from the aircraft MWOs, *#1 cyl has low compression*. This same problem has several possible corrective actions such as: *ran a/c, ops and leak good*, *serviced with 8 quarts of mineral oil*, *removed rocket cover and staked valve with no effect*, etc. This means that there is no true corrective action for any maintenance problem and that corrective actions can vary based on several factors. These factors can be related to the operator, the maintenance task, time spent inspecting the problem, tools available to inspect the problem, and any other additional context. Table 4 in Appendix shows additional instances of new maintenance problems with syntactically and semantically similar existing problems along with the corresponding corrective actions. Note the problem *low compression observed on cyl #4*. It has been semantically matched to the existing problem *low compression on cylinder 3 (55)* twice, but they both suggest different corrective actions. Due to the absence of a true corrective action for any given problem, we consider the prediction of corrective actions as an unsupervised learning problem, and use text similarity methods in our attempt to solve it.

The corrective actions shown in Tables 3 and 4 seem appropriate for almost all the cases of new maintenance problems. However, due to the limited context available about the maintenance problem and the lack of a ground-truth in terms of corrective action, any result from the text similarity methods needs to be verified using expert-knowledge. The text similarity based corrective action prediction is aimed toward making the job of the operator or technician easier by helping save time in the maintenance task. In cases where the maintenance problem is more complex, this method would help provide the technicians with several options to take corrective actions. Regardless of the type of maintenance problem and predictive method, the aviation industry is highly regulated, and necessitates the use of domain knowledge before making maintenance decisions.

Scenario 2: topic modeling

The original MWOs consisted of a few more attributes such as the Air Transport Association of America chapter codes, commonly referred to as ATA chapter/code. ATA codes are a standardized numbering system used by pilots, engineers, and maintenance technicians across the industry. Over the

[2] https://huggingface.co/sentence-transformers/all-mpnet-base-v2.

Table 3 New maintenance problems matched with syntactically and semantically similar existing problems to predict corrective actions with cosine similarity scores

New maintenance problem	Preprocessed new maintenance problem	Syntactically similar problem using TF-IDF	Corrective action using syntactic match	Cosine similarity	Semantically similar problem using a sentence transformer	Corrective action using semantic match	Cosine similarity
l/h and r/h rocker cover gasket leak	Left hand and right hand rocker cover gasket leak	Right hand engine 2 and 3 rocker box cover gasket leaking	Replaced gaskets	0.93	2 Rocker cover gasket leaking, right hand eng	Removed and replaced rocker cover gasket	0.93
		1, 3 and 4 rocker cover gasket leaking on left hand engine	Leak check good	0.92	2 and 3 rocker cover gaskets leaking, left hand eng	Removed and replaced rocker cover gaskets	0.93
		Right hand 2 and 4 rocker cover gaskets leaking	Removed and replaced with new gaskets	0.92	Left hand 4 rocker cover gasket leaks	Removed and replaced gasket	0.92
Baffle plugs disintegrating	Baffle plugs disintegrating	Both baffle plugs missing too many tabs	Installed baffle plugs	0.66	Baffle plugs worn	Replaced both baffle plugs	0.81
		Left hand and right hand baffle plugs missing tabs	Replaced left hand and right hand baffle plugs	0.66	Baffle plugs worn out	Removed and replaced plugs	0.80
		Baffle plugs have many missing fingers	Installed 2 new baffle plugs	0.65	Engine baffle plugs bad	Installed new baffle plugs	0.80
Eng is not clean	Engine is not clean	Both lower engine cowls need cleaning	Cleaned both cowls	0.81	Engine is dirty	Washed engine	0.82
		Engine oil cooler can be cleaned externally	Cleaned engine oil cooler	0.78	Engine air filter dirty	Removed and replaced engine air filter. Next due 7/31/17	0.75
		Oil leak noted on front engine section	No leak noted on run up	0.77	Engine requires overhaul	Unmasked and cleaned parts, installed new cylinders on engine	0.65

years, the ATA codes have been modified to what is now the Joint Aircraft System/Component (JASC) codes, colloquially referred to as JASC/ATA codes (FAA Flight Standards Service, 2008). Understandably, these codes are a very critical part of the record-keeping process of different airlines, private aircraft operators, and manufacturers. However, it is not an easy task to link the maintenance problems to these codes (U.S. Department of Transportation & Federal Aviation Administration, 2017), and this is an industry-wide issue due to the complexities of maintenance problems. In a lot of cases, MWOs are manually linked to the ATA codes which can be a time-consuming process. Since many of the maintenance issues reported in the MWOs are similar, we can apply TLP to process the data and predict the matching ATA codes. A technique known as Topic Modeling helps in identifying and extracting keywords to determine the most dominant topics prevalent in the data. Topic Modeling has been applied in several domains such as aviation safety (Rose et al., 2022), software maintenance tasks (Sun et al., 2015), and railway fault diagnosis (Wu, 2018). We use Topic Modeling to extract the most dominant keywords and identify suitable topics that can be linked with the standardized ATA codes.

Latent Dirichlet allocation (LDA)

The algorithm we choose for Topic Modeling is Latent Dirichlet Allocation (LDA), which is an unsupervised probabilistic model introduced for machine learning (Blei et al., 2003). It characterizes documents as a mixture of topics where each topic is represented as a distribution over words (Jelodar et al., 2019). A few assumptions made for LDA are: (1) each document is represented as a probabilistic distribution over topics, (2) the topic distributions in all the documents have a common Dirichlet prior, (3) each topic is also represented as a probabilistic distribution over words, and (4) word distributions in all the topics have a common Dirichlet prior as well. To understand the math behind LDA, let's define a few terms and equations.

Consider a corpus D with documents M having a vocabulary of size N. The documents d consists of N_d words with $d \in \{1, \ldots, M\}$. The generative process for each document W in corpus D assumes:

- Choose a multinomial distribution ϕ_t for a topic t with $t \in \{1, \ldots, T\}$ from a Dirichlet distribution with a parameter β.
- Choose another multinomial distribution θ_d for document d with $d \in \{1, \ldots, M\}$ from a Dirichlet distribution with parameter α.
- For a word w_n with $n \in \{1, \ldots, N_d\}$ in a document d,
  (a) Select a topic z_n from θ_d,
  (b) Select a word w_n from ϕ_{zn}.

For the above process, the observed variables are the words, the latent variables are ϕ and θ, and the hyperparameters are β and α. The probability of a corpus is shown in (3):

$$p(D|\alpha, \beta) = \prod_{d=1}^{M} \int p(\theta_d|\alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn}, \beta) \right) d\theta_d \tag{3}$$

To make this task as close to what it would be like in the industry, we only consider the maintenance problem and do not use the corrective action data. The same pre-processing steps outlined in section "Preprocessing maintenance work orders" are applied to the MWOs with some minor changes. First, some more stop words are removed since words like *right hand*, *need*, *have*, *and*, etc. have little importance when we are trying to identify a topic to match with ATA codes. Next, we perform word tokenization on the data, which means that we convert words into numeric representations. Since we have preprocessed the data pretty thoroughly, word tokenization should be effective. LDA then identifies the most dominant topics in the MWO along with the associated keywords that are considered to be salient terms. To determine how all the words are associated to each topic is determined by making two important considerations. First, we consider a ranking measure called *Lift* (Taddy, 2012), which is the ratio of a word's probability within a topic to its marginal probability across the entire corpus. Then, we consider a parameter called *Relevance* (Sievert & Shirley, 2014), a method for ranking words within topics explained in the following manner.

ϕ_{kw} denotes the probability of a word $w \in \{1, \ldots, V\}$ for a topic $k \in \{1, \ldots, K\}$ where V is the number of words in the vocabulary and p_w is the marginal probability of a word w in the corpus. The value of ϕ is estimated using LDA and p_w is determined from the empirical distribution of the data.

Relevance of word w to topic k given weight parameter λ, where $0 \leq \lambda \leq 1$, is shown in (4):

$$r(w, k|\lambda) = \lambda \log(\phi_{kw}) + (1 - \lambda) \log\left(\frac{\phi_{kw}}{p_w}\right) \tag{4}$$

where λ is the weight assigned to the probability of the word w under topic k relative to its *Lift*.

Results of topic modeling with LDA

For LDA model, we experimented with different learning rates and a different number of topics to obtain the most effective representation of the data in the MWOs. We find that the learning rate of 0.7 using the batch learning is most suited for our task and the model is run for 20 iterations. We identify that the optimal number of topics is between two and three, so we use three topics to characterize the data. We create a visualization in Python using the LDAvis method (Sievert & Shirley, 2014) to generate a global perspective of the topics (see Fig. 6). The areas of the circles are representative of the relative prevalence of each topic in the MWO corpus. The inter-topic distances used in this model are computed using the Jensen–Shannon divergence (Fuglede & Topsoe, 2004) and scaled with principal components as the axes using Principal Component Analysis (Wold et al., 1987). We observe large, non-overlapping topics conveying that the topics are distinct. Here, Topics 1, 2, and 3 contain 39.9%, 33%, and 27.1% of all the tokens in the corpus respectively.

To understand what *relevance* and *lift* mean in the context of the MWOs, we can look at Fig. 7. The visualization shows

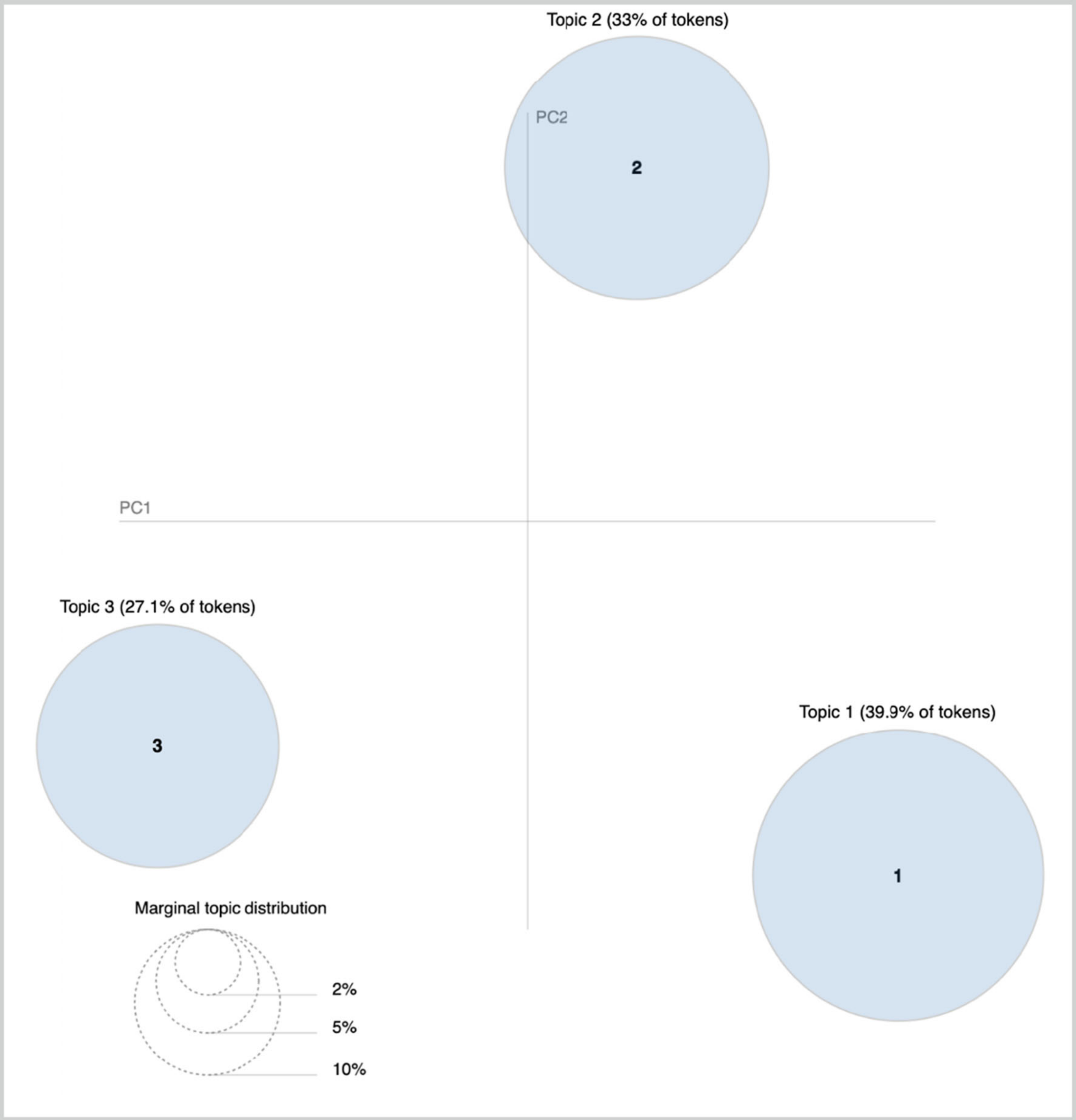

Fig. 6 Inter-topic distance map via multidimensional scaling

six bar graphs, with two graphs for each topic. For each topic, we evaluate the words associated with it by setting the value of the weight parameter as $\lambda = 1$ and $\lambda = 0$. The blue bar for any word represents the frequency of that word in the overall model and the red bar represents the frequency of that within a specific topic. The slider on the top of each graph controls the *relevance* metric. A word's association with a topic can be considered to be high if the frequency of its occurrence in that topic is high. This is achieved by setting $\lambda = 1$. In this case, words are sorted by the frequency of their occurrence in the topic, represented by the length of the red bars. A word can also be considered to be highly associated with a topic if its *lift* is high, i.e., how much the frequency of a word in a topic stands out above its overall frequency in the model. In other words, the ratio between the red and blue bars. For Topic 1, almost the same words are shown in both graphs, making it clear that those words are representative of the model. For Topic 2, we observe that the most frequently occurring terms are *baffle*, *engine*, *oil*, *cylinder*, *seal*, etc. We also note that with $\lambda = 0$, for words such as *oil*, *cracked*, *seal*, *aft*, *forward*, *screw*, *side*, etc., stand out compared to their overall frequency in the dataset of MWOs. This gives us some more information about the topic. In the case of Topic 3, the most relevant terms by the frequency of their occurrence are *engine*, *cylinder*, *baffle*, *intakes*, etc. The words that stand out the most are *intakes*, *compression*, *low*, *rpm*, *plug*, etc.

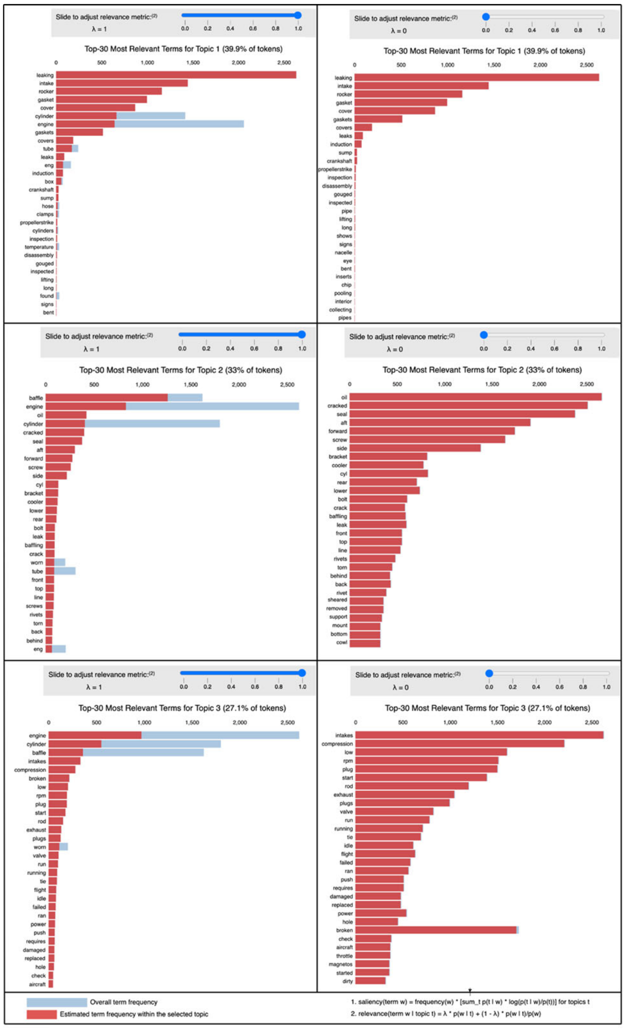

Fig. 7 Most relevant terms in Topics 1, 2, and 3 for different values of the relevance metric

We use this information learned from Topic Modeling to identify what JASC/ATA codes (FAA Flight Standards Service, 2008) these topics can be linked to. The JASC code of 7160 is related to all maintenance done with regards to the Engine Air Intake System. Topic 1 seems to have the keywords that fit into this maintenance category. The terms *leaking* and *intake* are dominant, and records from the MWOs that belong to Topic 1 could fall be annotated with code 7160. For Topic 2, the dominant terms are *baffle*, *engine*, and *oil*. The JASC code of 8550 pertains to Reciprocating Engine Oil System, so Topic 2 could be related to this code. Codes 7261—Turbine Engine Oil System, and 7900—Engine Oil System could also be appropriate annotations for the maintenance problems from this Topic 2. For the Topic 3, the terms *intakes*, *compression*, *seal*, *engine*, and *baffle* are dominant. The JASC Code of 8530 pertaining to Reciprocating Engine Cylinder Section seems to fit the description given the keywords of Topic 3. With the limited information available in the MWOs, we can see why linking standardized maintenance codes to maintenance issues can be a tedious and difficult task. Even after preprocessing with TLP, we cannot provide a 100% linking of maintenance problems to the codes because we do not have any context of the issue other than the short phrases in the MWOs. Yet, our method with Topic Modeling is capable of identifying dominant topics and getting approximate matches to the maintenance codes with the closest descriptions.

Research contribution and discussion

NLP has been very effective at analyzing text data from various domains. It has also been adapted to technical text from medicine, supply chain, and finance. Technical text data from MWOs though, is quite distinctive. It is generated by a human source, so the language used by operators, technicians, etc. is not easy to understand due to the highly specific colloquialisms, technical jargon, short hand, abbreviations and the domain-specific terminology. Off-the-shelf NLP methods perform poorly when provided with such data. TLP can be used to process highly complex technical text by incorporating domain knowledge. Given this research opportunity, we highlight how TLP can be used as a disruptive strategy to advance PHM and maintenance by:

(1) Proposing a TLP framework for PHM that uses expert-knowledge in a human-in-the-loop format.
(2) Applying the framework to MWOs from aircraft to:
    (a) Predict corrective actions for new maintenance problems by using syntactic and semantic text similarity methods,
    (b) Identify dominant topics and keywords from maintenance problems, matching them with standardized maintenance codes, and annotating all the records using those codes.

Our research shows that TLP can be effective in tackling the complexities and heterogeneity of the technical text data. For the aircraft MWOs, we found that the data only has two columns that convey meaningful information. We also discovered that it takes an extensive study of the domain knowledge to identify what the common acronyms, abbreviations, short hand, and technical jargon used in the industry are. Stop word removal on this data was an extremely challenging task. It took several iterations to come up with the best list of stop words for our tasks. Upon examining the data, we observed that most of the maintenance problems in the MWOs are related to the power plant or very closely related to the engine and its associated components. The MWOs could have been for a smaller aircraft or a training aircraft, in which case the most important concerns would have been highlighted. For a much larger dataset, we would have probably encountered work orders from other areas of aircraft maintenance. However, the proposed framework can be extended to larger data from different domains, as long as it integrates appropriate domain-knowledge.

Conclusion and future work

In this work, we demonstrate how TLP can be applied to the PHM paradigm. We review the current maintenance strategies and the different types of text data. The notable distinctiveness of technical text data from MWOs is emphasized by examining some sample maintenance records. While NLP is successful in processing non-technical data and has been adapted successfully to certain technical text, we demonstrate how it underperforms on technical text from MWOs. The MWOs considered are highly unstructured, with a large number of technical abbreviations, short hand and maintenance related colloquialisms. To overcome these challenges, we propose a TLP framework for PHM that and provide an outline of all the steps involved. The framework uses knowledge from domain experts in a human-in-the-loop format. We also identify the potential areas of application for TLP using the framework. To demonstrate its practical applications, we apply it to technical text data from MWOs of aircraft and identify two relevant scenarios of application.

We first use the existing data from the MWOs to help in predicting corrective actions for new maintenance problems. We identify existing maintenance problems using both syntactic and semantic textual similarity techniques and predict the top three corrective actions for new maintenance problems based on cosine similarity scores. We use text similarity methods because the data lacks any form of annotation to allow classification models to be used, and maintenance problems do not necessarily have a true corrective action. The corrective actions can vary depending on a variety of factors such as the technician or operator, the inspection performed to assess the problem, the time spent inspecting it, the severity of the problem, environmental factors, etc. Due to the limited context available to us from the data, we choose an unsupervised learning approach to demonstrate how corrective actions can be predicted using only the text from a new maintenance problem. To calculate syntactic and semantic similarity, we use TF-IDF and a BERT transformer respectively. Our predictions show that the recommended corrective actions are appropriate for the new problems presented, and our results are reaffirmed with cosine similarity scores. The results also highlight instances where a corrective action predicted using syntactic match might be more suitable to the maintenance problem than one obtained using a semantic match, and vice-versa. This confirms that there is a need for both syntactic and semantic textual similarity methods when assessing complex maintenance data. In the second scenario, we apply Topic Modeling to the technical text from MWOs. To replicate what this scenario might be like in the real-world, we decide to only use the data from existing maintenance problems, and do not use the data from corrective actions in modeling the topics. We use LDA to extract the most dominant topics and salient terms from the problems. We identify three dominant topics and use metrics such as *lift* and *relevance* to select the most appropriate keywords for each topic. We then use these keywords and the description of the JASC/ATA codes to match the maintenance problems from each of the topics to an appropriate code. This helps us to identify what JASC/ATA code a new maintenance problems corresponds to and can be annotated with. We find that the data considered consists of maintenance problems that are predominantly related to the aircraft's engine and its nearby systems. Our results are consistent with the text descriptions provided with the standardized codes.

We suggest that future works in the area of TLP for PHM be focused on incorporating technical text processing in conjunction with predictive maintenance techniques to form a prescriptive approach. We also propose collaboration among organizations and industries to standardize maintenance terminology and jargon, which would enable the construction of large corpora for maintenance text. TLP can also be extended to other areas of manufacturing with the abundance of text data. Quality inspection logs generate a lot of text data on which TLP can be used to help determine whether a product needs to be reworked/scrapped. The results of our research reemphasize that TLP is here to stay and our outlook for its widespread incorporation in the industry is optimistic.

Funding Open access funding provided by Northeastern University Library.

Data availability The data use for analysis in this research can be found at: https://people.rit.edu/fa3019/MaintNet/.

Declarations

Competing interests Authors declare no competing interests to declare.

Appendix

See Table 4.

Table 4 Results of text similarity based corrective action prediction showing additional instances of new maintenance problems

New maintenance problem	Preprocessed new maintenance problem	Syntactically similar problem using TF-IDF	Corrective action using syntactic match	Cosine similarity	Semantically similar problem using a sentence transformer	Corrective action using semantic match	Cosine similarity
Low compression observed on cyl #4	Low compression observed on cylinder 4	Cylinder 2 compression low. air noted and observed out of exhaust	Performed compression on 2 cylinder after run and noted 70/80	0.94	Cylinder 4 has low compression	Rechecked compressions after post run, ops ck good	0.90
		Right hand engine 4 cylinder low on compression (30 lbs)	Added 8 qts of mineral oi. ran engine, no leaks. operational safety section check	0.87	Low compression on cylinder 3 (55)	Operations safety section and leak check good	0.88
		Low compression on cylinder 1	Found to have 57/80 psi compressor and burnt exhaust valve	0.86	Low compression on cylinder 3 (55)	Removed cylinder, found to be burnt exhaust valved, installed	0.88
Assistance required for start up	Assistance required for start up	Start assist required	Started aircraft using normal procedures	0.76	Aircraft will not start	Brought a/c in hangar to warm up. a/c started	0.50
		Pilot requires start assist	Started aircraft	0.73	Pilot reports start assist	Started aircraft	0.39
		Crew requested start assist	Started engine for crew	0.66	Pilot requests start assist	Started aircraft	0.33
Metal dust in oil	Metal dust in oil	Small amount of metal found in right hand engine oil filter	Aw lycoming si 1492d, oil change and filter insp require	0.82	Engine to be removed for metal found in oil	Installed engine on airframe. Began routing fuel lines	0.66
		Engine to be removed for metal found in oil	Washed fuselage and baffles. Installed scroll and cleaned	0.79	Small amount of metal found in right hand engine oil filter	Aw lycoming si 1492d, oil change and filter insp require	0.65
		Engine to be removed for metal found in oil	Removed engine from airframe	0.79	Fuzz in oil cooler	Removed fuzz	0.58

References

Akhbardeh, F., Desell, T., & Zampieri, M. (2020a). MaintNet: A collaborative open-source library for predictive maintenance language resources. In M. Ptaszynski & B. Ziolko (Eds.), *Proceedings of the 28th international conference on computational linguistics: System demonstrations* (pp. 7–11). International Committee on Computational Linguistics (ICCL). https://doi.org/10.18653/v1/2020.coling-demos.2

Akhbardeh, F., Desell, T., & Zampieri, M. (2020b). NLP tools for predictive maintenance records in MaintNet. In D. Wong & D. Kiela (Eds.), *Proceedings of the 1st conference of the Asia-Pacific chapter of the association for computational linguistics and the 10th international joint conference on natural language processing: System demonstrations* (pp. 26–32). Association for Computational Linguistics. https://aclanthology.org/2020.aacl-demo.5

Alexander, P. A., & Kulikowich, J. M. (1994). Learning from physics text: A synthesis of recent research. *Journal of Research in Science Teaching, 31*(9), 895–911. https://doi.org/10.1002/tea.3660310906

Alharbi, M., Roach, M., Cheesman, T., & Laramee, R. S. (2021). VNLP: Visible natural language processing. *Information Visualization, 20*(4), 245–262.

Ansari, F., Glawar, R., & Nemeth, T. (2019). PriMa: A prescriptive maintenance model for cyber-physical production systems. *International Journal of Computer Integrated Manufacturing, 32*(4–5), 482–503. https://doi.org/10.1080/0951192X.2019.1571236

Ansari, F., Glawar, R., & Sihn, W. (2020). Prescriptive maintenance of CPPS by integrating multimodal data with dynamic Bayesian networks. In J. Beyerer, A. Maier, & O. Niggemann (Eds.), *Machine learning for cyber physical systems* (pp. 1–8). Springer. https://doi.org/10.1007/978-3-662-59084-3_1

Basri, E. I., Razak, I. H. A., Ab-Samat, H., & Kamaruddin, S. (2017). Preventive maintenance (PM) planning: A review. *Journal of Quality in Maintenance Engineering, 23*(2), 114–143. https://doi.org/10.1108/JQME-04-2016-0014

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., & Vaughan, J. W. (2010). A theory of learning from different domains. *Machine Learning, 79*, 151–175. https://doi.org/10.1007/s10994-009-5152-4

Bhatt, A., Patel, A., Chheda, H., & Gawande, K. (2015). Amazon review classification and sentiment analysis. *International Journal of Computer Science and Information Technologies, 6*(6), 5107–5110.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research, 3*, 993–1022. https://doi.org/10.5555/944919.944937

BNC Consortium. (2007). British national corpus. *Oxford Text Archive Core Collection*. http://www.natcorp.ox.ac.uk/

Bokinsky, H., McKenzie, A., Bayoumi, A., McCaslin, R., Patterson, A., Matthews, M., Schmidley, J., & Eisner, L. (2013). Application of natural language processing techniques to marine V-22 maintenance data for populating a CBM-oriented database. In *AHS airworthiness, CBM, and HUMS specialists' meeting* (pp. 463–472). https://www.proceedings.com/19340.html

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

Chandrasekaran, D., & Mago, V. (2021). Evolution of semantic similarity—A survey. *ACM Computing Surveys (CSUR), 54*(2), 1–37. https://doi.org/10.1145/3440755

Chen, X., Xie, H., Wang, F. L., Liu, Z., Xu, J., & Hao, T. (2018). A bibliometric analysis of natural language processing in medical research. *BMC Medical Informatics and Decision Making, 18*(1), 1–14. https://doi.org/10.1186/s12911-018-0594-x

Chung, T. M., & Nation, P. (2004). Identifying technical vocabulary. *System, 32*(2), 251–263. https://doi.org/10.1016/j.system.2003.11.008

Copeck, T., Barker, K., Delisle, S., Szpakowicz, S., & Delannoy, J.-F. (1997). What is technical text? *Language Sciences, 19*(4), 391–423. https://doi.org/10.1016/S0388-0001(97)00003-X

Dangut, M. D., Skaf, Z., & Jennions, I. K. (2021). An integrated machine learning model for aircraft components rare failure prognostics with log-based dataset. *ISA Transactions, 113*, 127–139.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, & T. Solorio (Eds.), *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)* (pp. 4171–4186). Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1423

Dima, A., Lukens, S., Hodkiewicz, M., Sexton, T., & Brundage, M. P. (2021). Adapting natural language processing for technical text. *Applied AI Letters, 2*(3), e33. https://doi.org/10.1002/ail2.33

FAA Flight Standards Service. (2008). *Federal aviation administration joint aircraft system/component code table and definitions* (AFS-620). Mike Monroney Aeronautical Center. https://av-info.faa.gov/sdrx/documents/JASC_Code.pdf

Federal Energy Management Program. (2021). *OMETA: An integrated approach to operations, maintenance, engineering, training, and administration*. https://www1.eere.energy.gov/femp/pdfs/OM_3.pdf

Fleurence, R. L., Curtis, L. H., Califf, R. M., Platt, R., Selby, J. V., & Brown, J. S. (2014). Launching PCORnet, a national patient-centered clinical research network. *Journal of the American Medical Informatics Association, 21*(4), 578–582.

Fuglede, B., & Topsoe, F. (2004). Jensen-Shannon divergence and Hilbert space embedding. In *International symposium on information theory, 2004. ISIT 2004. Proceedings*. https://doi.org/10.1109/ISIT.2004.1365067

Fuller, L. L. (1958). Positivism and fidelity to law—A reply to Professor Hart. *Harvard Law Review, 71*, 630. https://doi.org/10.2307/1338226

Garg, A., & Deshmukh, S. G. (2006). Maintenance management: Literature review and directions. *Journal of Quality in Maintenance Engineering, 12*(3), 205–238. https://doi.org/10.1108/13552510610685075

Gomaa, W. H., & Fahmy, A. A. (2013). A survey of text similarity approaches. *International Journal of Computer Applications, 68*(13), 13–18. https://doi.org/10.5120/11638-7118

Granados, G. E., Lacroix, L., & Medjaher, K. (2020). Condition monitoring and prediction of solution quality during a copper electroplating process. *Journal of Intelligent Manufacturing, 31*(2), 285–300. https://doi.org/10.1007/s10845-018-1445-4

Hodkiewicz, M. R., Batsioudis, Z., Radomiljac, T., & Ho, M. T. (2017). Why autonomous assets are good for reliability—The impact of 'operator-related component' failures on heavy mobile equipment reliability. In *Annual conference of the PHM society, 2017*, *9*(1). https://doi.org/10.36001/phmconf.2017.v9i1.2449

Holmes, O. W. (1997). The path of the law. *Harvard Law Review, 110*(5), 991–1009. https://doi.org/10.2307/1342108

Horvath, B. (1954). Jurisprudence, men and ideas of the law. *The American Journal of Comparative Law, 3*(3), 448–451. https://doi.org/10.2307/837969

Jelodar, H., Wang, Y., Yuan, C., Feng, X., Jiang, X., Li, Y., & Zhao, L. (2019). Latent Dirichlet allocation (LDA) and topic modeling:

Models, applications, a survey. *Multimedia Tools and Applications, 78*, 15169–15211. https://doi.org/10.1007/s11042-018-6894-4

Jiang, C., Chen, H., Xu, Q., & Wang, X. (2023). Few-shot fault diagnosis of rotating machinery with two-branch prototypical networks. *Journal of Intelligent Manufacturing, 34*(4), 1667–1681. https://doi.org/10.1007/s10845-021-01904-x

Jin, X., Weiss, B. A., Siegel, D., & Lee, J. (2016). Present status and future growth of advanced maintenance technology and strategy in US manufacturing. *International Journal of Prognostics and Health Management*. https://doi.org/10.36001/ijphm.2016.v7i3.2409

Karray, M. H., Ameri, F., Hodkiewicz, M., & Louge, T. (2019). ROMAIN: Towards a BFO compliant reference ontology for industrial maintenance. *Applied Ontology, 14*(2), 155–177. https://doi.org/10.3233/AO-190208

Karray, M. H., Chebel-Morello, B., & Zerhouni, N. (2012). A formal ontology for industrial maintenance. *Applied Ontology, 7*(3), 269–310.

Khurana, D., Koli, A., Khatter, K., & Singh, S. (2023). Natural language processing: State of the art, current trends and challenges. *Multimedia Tools and Applications, 82*(3), 3713–3744. https://doi.org/10.1007/s11042-022-13428-4

Kimura, Y. (1997). Maintenance tribology: Its significance and activity in Japan. *Wear, 207*(1–2), 63–66. https://doi.org/10.1016/S0043-1648(96)07472-8

Krallinger, M., & Valencia, A. (2005). Text-mining and information-retrieval services for molecular biology. *Genome Biology, 6*(7), 1–8. https://doi.org/10.1186/gb-2005-6-7-224

Krolikowski, P. M., & Naggert, K. (2021). Semiconductor shortages and vehicle production and prices. *Federal Reserve Bank of Cleveland, Economic Commentary*, *2021–17*. https://doi.org/10.26509/frbc-ec-202117

Latorella, K. A., & Prabhu, P. V. (2000). A review of human error in aviation maintenance and inspection. *International Journal of Industrial Ergonomics, 26*(2), 133–161. https://doi.org/10.1016/S0169-8141(99)00063-3

Le, T., & David Jeong, H. (2017). NLP-based approach to semantic classification of heterogeneous transportation asset data terminology. *Journal of Computing in Civil Engineering, 31*(6), 04017057. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000701

Linhares, J. C., & Dias, A. (2003). A linguistic approach proposal for mechanical design using natural language processing. In N. J. Mamede, I. Trancoso, J. Baptista, & M. das Graças Volpe Nunes (Eds.), *Computational processing of the Portuguese language* (pp. 171–174). Springer Berlin Heidelberg. https://doi.org/10.1007/3-540-45011-4_25

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies—Volume 1* (pp. 142–150). https://doi.org/10.5555/2002472.2002491

Mahmoudzadeh, A., Elgart, Z., Arezoumand, S., Hansen, T., & Das, S. (2020). Designing transit agency job descriptions for optimal roles: An analytical text-mining approach. In *International conference on transportation and development 2020* (pp. 356–368). https://doi.org/10.1061/9780784483169.030

Manning, C., & Schutze, H. (1999). *Foundations of statistical natural language processing*. MIT Press. https://mitpress.mit.edu/9780262133609/

Marcus, M. P., Santorini, B., & Marcinkiewicz, M. A. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics, 19*(2), 313–330.

Masoni, R., Ferrise, F., Bordegoni, M., Gattullo, M., Uva, A. E., Fiorentino, M., Carrabba, E., & Di Donato, M. (2017). Supporting remote maintenance in industry 4.0 through augmented reality. *Procedia Manufacturing, 11*, 1296–1302. https://doi.org/10.1016/j.promfg.2017.07.257

Merity, S., Xiong, C., Bradbury, J., & Socher, R. (2017). Pointer sentinel mixture models. In *5th international conference on learning representations*. https://openreview.net/pdf?id=Byj72udxe

Meshalkin, V. P., Panina, E. A., & Bykov, R. S. (2015). Principles of developing an interactive system for the semantic processing of scientific and technical texts on chemical technology of reagents and ultrapure substances. *Theoretical Foundations of Chemical Engineering, 49*, 422–426. https://doi.org/10.1134/S0040579515040314

Mobley, R. K. (2002). *An introduction to predictive maintenance*. Elsevier. https://doi.org/10.1016/B978-0-7506-7531-4.X5000-3

Montero Jiménez, J. J., Vingerhoeds, R., Grabot, B., & Schwartz, S. (2023). An ontology model for maintenance strategy selection and assessment. *Journal of Intelligent Manufacturing, 34*(3), 1369–1387. https://doi.org/10.1007/s10845-021-01855-3

Mourtzis, D., Zogopoulos, V., & Vlachou, E. (2017). Augmented reality application to support remote maintenance as a service in the robotics industry. *Procedia CIRP, 63*, 46–51. https://doi.org/10.1016/j.procir.2017.03.154

Mwanza, B. G., & Mbohwa, C. (2015). Design of a total productive maintenance model for effective implementation: Case study of a chemical manufacturing company. *Procedia Manufacturing, 4*, 461–470. https://doi.org/10.1016/j.promfg.2015.11.063

Naqvi, S. M. R., Ghufran, M., Meraghni, S., Varnier, C., Nicod, J.-M., & Zerhouni, N. (2022). CBR-based decision support system for maintenance text using NLP for an aviation case study. In *2022 Prognostics and Health Management conference (PHM-2022 London)* (pp. 344–349). https://doi.org/10.1109/PHM2022-London52454.2022.00067

Navinchandran, M., Sharp, M. E., Brundage, M. P., & Sexton, T. B. (2022). Discovering critical KPI factors from natural language in maintenance work orders. *Journal of Intelligent Manufacturing, 33*(6), 1859–1877. https://doi.org/10.1007/s10845-021-01772-5

O'Donoghue, C. D., & Prendergast, J. G. (2004). Implementation and benefits of introducing a computerised maintenance management system into a textile manufacturing company. *Journal of Materials Processing Technology, 153–154*, 226–232. https://doi.org/10.1016/j.jmatprotec.2004.04.022

Offices of Industries and Economics. (2010). *Small and medium-sized enterprises: Characteristics and performance* (Investigation No. 332-510). United States International Trade Commission. https://www.usitc.gov/publications/332/pub4189.pdf

Olack, D. (2021, August). *Application of data analytics to mine nuclear plant maintenance data*. Data Science and Artificial Intelligence Regulatory Applications Workshops, Charlotte, NC. https://www.nrc.gov/docs/ML2127/ML21277A144.pdf

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research, 12*, 2825–2830. https://doi.org/10.5555/1953048.2078195

Phan, M. C., Sun, A., & Tay, Y. (2019). Robust representation learning of biomedical names. In A. Korhonen, D. Traum, & L. Màrquez (Eds.), *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 3275–3285). Association for Computational Linguistics. https://doi.org/10.18653/v1/P19-1317

Plank, B. (2016). What to do about non-standard (or non-canonical) language in NLP. *arXiv Preprint*. https://doi.org/10.48550/arXiv.1608.07836

Qurashi, A. W., Holmes, V., & Johnson, A. P. (2020). Document processing: Methods for semantic text similarity analysis. In *2020 international conference on INnovations in Intelligent SysTems* *and Applications (INISTA)* (pp. 1–6). https://doi.org/10.1109/INISTA49547.2020.9194665

Rajpathak, D., & Chougule, R. (2011). A generic ontology development framework for data integration and decision support in a distributed environment. *International Journal of Computer Integrated Manufacturing, 24*(2), 154–170. https://doi.org/10.1080/0951192X.2010.531291

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In K. Inui, J. Jiang, V. Ng, & X. Wan (Eds.), *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)* (pp. 3982–3992). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1410

Rose, R. L., Puranik, T. G., Mavris, D. N., & Rao, A. H. (2022). Application of structural topic modeling to aviation safety data. *Reliability Engineering & System Safety, 224*, 108522. https://doi.org/10.1016/j.ress.2022.108522

Rose, T., Stevenson, M., & Whitehead, M. (2002). The Reuters corpus volume 1—From yesterday's news to tomorrow's language resources. In M. González Rodríguez & C. P. Suarez Araujo (Eds.), *Proceedings of the third international conference on language resources and evaluation (LREC'02)*. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2002/pdf/80.pdf

Sarica, S., & Luo, J. (2021). Stopwords in technical language processing. *PLoS ONE, 16*(8), e0254937. https://doi.org/10.1371/journal.pone.0254937

Schauer, F. (2015). Is law a technical language? *San Diego Law Review, Forthcoming*, *52*, 501. https://ssrn.com/abstract=2689788

Schironi, F. (2010). Technical languages: Science and medicine. In *A companion to the ancient Greek language* (pp. 338–353). Wiley. https://doi.org/10.1002/9781444317398.ch23

Sievert, C., & Shirley, K. (2014). LDAvis: A method for visualizing and interpreting topics. In J. Chuang, S. Green, M. Hearst, J. Heer, & P. Koehn (Eds.), *Proceedings of the workshop on interactive language learning, visualization, and interfaces* (pp. 63–70). Association for Computational Linguistics. https://doi.org/10.3115/v1/W14-3110

Sparck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation, 28*(1), 11–21. https://doi.org/10.1108/eb026526

Stenström, C., Al-Jumaili, M., & Parida, A. (2015). Natural language processing of maintenance records data. *International Journal of COMADEM, 18*(2), 33–37.

Sun, X., Li, B., Li, Y., & Chen, Y. (2015). What information in software historical repositories do we need to support software maintenance tasks? An approach based on topic model. In R. Lee (Ed.), *Computer and information science* (pp. 27–37). Springer International Publishing. https://doi.org/10.1007/978-3-319-10509-3_3

Sundaram, S., & Zeid, A. (2021). Smart Prognostics and Health Management (SPHM) in smart manufacturing: An interoperable framework. *Sensors, 21*(18), Article 18. https://doi.org/10.3390/s21185994

Sundaram, S., & Zeid, A. (2023). Artificial intelligence-based smart quality inspection for manufacturing. *Micromachines, 14*(3), 570. https://doi.org/10.3390/mi14030570

Taddy, M. (2012). On estimation and selection for topic models. In N. D. Lawrence & M. Girolami (Eds.), *Proceedings of the fifteenth international conference on artificial intelligence and statistics* (Vol. 22, pp. 1184–1193). PMLR. https://proceedings.mlr.press/v22/taddy12.html

Tekgöz, H., Omurca, S. İ, Koç, K. Y., Topçu, U., & Çelik, O. (2023). Semantic similarity comparison between production line failures for predictive maintenance. *Advances in Artificial Intelligence Research, 3*(1), 1–11. https://doi.org/10.54569/aair.1142568

Thomas, D. S. (2018). *The costs and benefits of advanced maintenance in manufacturing* (pp. 1–45). National Institute of Standards and Technology, NIST AMS 100-18. https://doi.org/10.6028/NIST.AMS.100-18

Tice, R. R., Austin, C. P., Kavlock, R. J., & Bucher, J. R. (2013). Improving the human hazard characterization of chemicals: A Tox21 update. *Environmental Health Perspectives, 121*(7), 756–765.

U.S. Department of Transportation & Federal Aviation Administration. (2017). *Survey of service difficulty reports for in-service damage in transport category aircraft* (Tech Report DOT/FAA/TC-TN17/70). https://rosap.ntl.bts.gov/view/dot/57873

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Proceedings of the 31st international conference on neural information processing systems* (pp. 6000–6010).

Wan, W., Liu, Y., Han, X., & Wang, H. (2021). Evaluation model of power operation and maintenance based on text emotion analysis. *Mathematical Problems in Engineering, 2021*, 1–8. https://doi.org/10.1155/2021/2824689

Warnekar, P., & Carter, J. (2003). HIV terms coverage by a commercial nomenclature. In *AMIA annual symposium proceedings. AMIA symposium*, *2003* (p. 1046). https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1480038/

Webster, J. J., & Kit, C. (1992). Tokenization as the initial phase in NLP. In *Proceedings of the 14th conference on computational linguistics—Volume 4* (pp. 1106–1110). https://doi.org/10.3115/992424.992434

Wilbur, W. J., & Yang, Y. (1996). An analysis of statistical term strength and its use in the indexing and retrieval of molecular biology texts. *Computers in Biology and Medicine, 26*(3), 209–222. https://doi.org/10.1016/0010-4825(95)00055-0

Witteman, M., Deng, Q., & Santos, B. F. (2021). A bin packing approach to solve the aircraft maintenance task allocation problem. *European Journal of Operational Research, 294*(1), 365–376.

Wold, S., Esbensen, K., & Geladi, P. (1987). Principal component analysis. *Chemometrics and Intelligent Laboratory Systems, 2*(1–3), 37–52. https://doi.org/10.1016/0169-7439(87)80084-9

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., … Rush, A. (2020). Transformers: State-of-the-art natural language processing. In Q. Liu, & D. Schlangen (Eds.), *Proceedings of the 2020 conference on empirical methods in natural language processing: System demonstrations* (pp. 38–45). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-demos.6

Woods, C., Hodkiewicz, M., & French, T. (2020). Requirements for adaptive user interfaces for industrial maintenance procedures: A discussion of context, requirements and research opportunities. In *Proceedings of the 31st Australian conference on human-computer-interaction* (pp. 322–326). https://doi.org/10.1145/3369457.3369487

Wu, S. (2018). Short text mining for fault diagnosis of railway system based on multi-granularity topic model. In *2018 8th international conference on logistics, informatics and service sciences (LISS)* (pp. 1–6). https://doi.org/10.1109/LISS.2018.8593228

Zeid, A., Sundaram, S., Moghaddam, M., Kamarthi, S., & Marion, T. (2019). Interoperability in smart manufacturing: Research challenges. *Machines, 7*(2), Article 2. https://doi.org/10.3390/machines7020021

Zhang, R., Zhao, N., Fu, L., Bai, X., & Cai, J. (2023). Recognizing defects in stainless steel welds based on multi-domain feature expression and self-optimization. *Journal of Intelligent Manufacturing, 34*(3), 1293–1309. https://doi.org/10.1007/s10845-021-01849-1

Zhou, L., & Hripcsak, G. (2007). Temporal reasoning with medical data—A review with emphasis on medical natural language processing. *Journal of Biomedical Informatics, 40*(2), 183–202. https://doi.org/10.1016/j.jbi.2006.12.009

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.