Enhancing semantic search using ontologies: A hybrid information retrieval approach for industrial text

Syed Meesam Raza Naqvi[1,2], Mohammad Ghufran[2], Christophe Varnier [2], Jean-Marc Nicod[2] and Noureddine Zerhouni[1*]

[1]Fives CortX, Lyon, France
[2]SUPMICROTECH, CNRS, institut FEMTO-ST, Besançon, F-25000, France

Abstract

Despite the increased focus on data in Industry 4.0, textual data has received little attention in the production and engineering management literature. Data sources such as maintenance records and machine documentation usually are not used to help maintenance decision-making. Available studies mainly focus on categorizing maintenance records or extracting meta-data, such as time of failure, maintenance cost, etc. One of the main reasons behind this underutilization is the complexity and unstructured nature of the industrial text. In this study, we propose a novel hybrid information retrieval approach for industrial text using multi-modal learning. Maintenance operators can use the proposed system to query maintenance records and find similar solutions to a given problem. The proposed system utilizes heterogeneous (multi-modal) data, a combination of maintenance records, and machine ontology to enhance semantic search results. We used the state-of-the-art Large Language Models (LLMs); BERT (Bidirectional Encoder Representations from Transformers) for textual similarity. For similarity among ontology labels, we used a modified version of Wu-Palmer's similarity. A hybrid weighted similarity is proposed, incorporating text and ontology similarities to enhance semantic search results. The proposed approach is validated using an open-source dataset of real maintenance records from excavators collected over ten years from different mining sites. A retrieval comparison using only text and multi-modal data is performed to estimate the proposed system's effectiveness. Quantitative and qualitative analysis of results indicates a performance improvement of 8% using the proposed hybrid similarity approach compared to only text-based retrieval. To the best of our knowledge, this is the first study to combine LLMs and machine ontology for semantic search in maintenance records.

Keywords: *Industry 4.0, Industrial information integration, Machine documentation, Multimodal learning, Semantic search, Large Language Models (LLMs)*

https://doi.org/10.1016/j.jii.2025.100835

Introduction

Industry 4.0 generates large volumes of data due to increased digitization, automation, and data exchange in manufacturing processes [1]. The collected data is multi-modal, consisting of multiple modalities or types of information, such as sensor data, industrial text, images, audio, video, and machine ontologies. If leveraged correctly, this enriched information can improve manufacturing and help diagnose complex industrial systems. Prognostics and Health Management (PHM) is a complete industry management cycle [2]. It is a crucial enabler behind the reduced maintenance cost, increased reliability, and availability of manufacturing systems. PHM mainly deals with developing techniques for predicting machine failure and machine component's Remaining Useful Life (RUL) to facilitate predictive maintenance. Data is the key to the data-driven prognostic techniques that can help make predictive decisions through data-powered intelligent systems. The ultimate goal of data-driven techniques is to train Machine Learning (ML) and Artificial Intelligence (AI) models on the collected data to make future predictions.

The collected data should be used to improve manufacturing and support digitization. However, efficient use of this data is still challenging to achieve. There are many reasons behind the underutilization of the collected data, including fragmentation of collected data, inconsistent quality, lack of expertise, absence of data-driven decision-making culture [3] and privacy and security concerns [4]. Strategic initiatives are required to address these challenges to unlock the full potential of the data. Harnessing this underutilized data can drive insights, optimize processes, and enable informed decision-making. Machine learning and AI techniques strive to develop intelligent machines with human-like perception, understanding, and response [5]. However, traditional machine learning algorithms in industrial solutions are usually uni-modal (e.g., focused on only one data modality, i.e., times series, images, or text). This uni-modal focus also contributes to the lack of uti-

*Corresponding author: smeesamnaqvi@gmail.com
Accepted: March 13, 2025, **Published:** March 22, 2025

lization of data collected in the industry. It is also contrary to human perception as humans perceive environments and solve problems by integrating and analyzing information from various data sources. A maintenance operator resolves maintenance problems through past knowledge, machine documentation, and visual-auditory inspection. Developing algorithms and models that can process multi-modal data and generate insights is crucial. Each modality provides a unique perspective and complementary information, and combining these modalities can considerably enhance the understanding of complex industrial systems.

Multi-modal Machine Learning (MMML) is an interdisciplinary field that deals with developing techniques and models for processing, analyzing, and extracting information from multi-modal input [6]. MMML strives to explore the unique characteristics of different modalities and correlations among them. One of the underutilized data sources in the industry is industrial text and machine documentation (e.g., machine ontology representing machine knowledge). This study explores the possibility of leveraging these underutilized data sources through MMML. The industrial text, specifically Maintenance Work Orders (MWOs), typically contains decades of experience and health indicators for various assets. They contain maintenance, repair, or operations details and are a vast source of human knowledge. However, due to the unique characteristics of the content and the environment in which it is produced, processing industrial text presents several challenges. Here are some typical challenges associated with the analysis of the industrial text:

- *Technical language:* Industrial text often contains specialized terminologies and industry-specific jargon that is hard to process using standard natural language processing pipelines. Effective interpretation of industrial text requires domain-specific knowledge and expertise.
- *Noisy and unstructured:* Industrial text is noisy and unstructured because it contains spelling mistakes, inconsistent formatting, incomplete sentences, irrelevant information, and typographical errors. Manual preprocessing and cleaning of such industrial text require domain expertise. This process can also be time-consuming and impractical due to large volumes of data.
- *Privacy and security:* Industrial text data may contain sensitive information, such as Personally Identifiable Information (PII), trade secrets, or proprietary knowledge. While processing and analyzing such data, ensuring data privacy and implementing robust security measures are crucial.
- *Contextual understanding:* Processing industrial text often requires a deep contextual understanding of the underlying system/equipment. For instance, understanding the specific equipment, various parts, and related maintenance processes may be necessary when deciphering a Maintenance Work Order (MWO) entry. This contextual information must be captured for accurate processing and analysis. In multinational industrial setups, industrial text may be available in different languages. In such cases, language-specific processing methods, such as language identification, translation, and cross-lingual information retrieval, can be used for efficient understanding of the context information.

Addressing these challenges often requires various Natural Language Processing (NLP) techniques, such as data preprocessing, feature engineering, and fine-tuning models using domain-specific data. The unstructured and unique nature of the industrial text makes data preprocessing incredibly challenging since regular NLP pipelines fail to process such data. Figure 1 shows an NLP pipeline to process regular text, which usually fails when applied to industrial text. This study proposes a multi-modal machine learning-based methodology for semantic search in industrial text for information retrieval. The proposed methodology uses a combination of text and machine ontology to search through industrial text. To learn the representations (embeddings) of the industrial text, we used Bidirectional Encoder Representations from Transformers (BERT) [7], a Large Language Model (LLM) based on Transformers architecture. A hybrid similarity approach is proposed combining industrial text and machine ontology to enhance retrieval performance.

The main objective of the proposed system is to serve as a way to retain and exploit human knowledge in MWOs to solve new maintenance problems. If an experienced employee retires or leaves the enterprise, their expertise and experience go with them. MWOs inherently capture this experience. Modern industries focus on retaining and utilizing the knowledge of past experienced employees, and the proposed system is a step toward this effort. Inexperienced maintenance operators can also use the proposed system to enhance their knowledge and work independently. A detailed explanation of the methodology is presented in Section . The rest of the paper is organized as follows: Section presents the literature review; Section explains the evaluation methodology and results of the proposed system. Finally, we conclude the paper and present the future perspectives in Section .

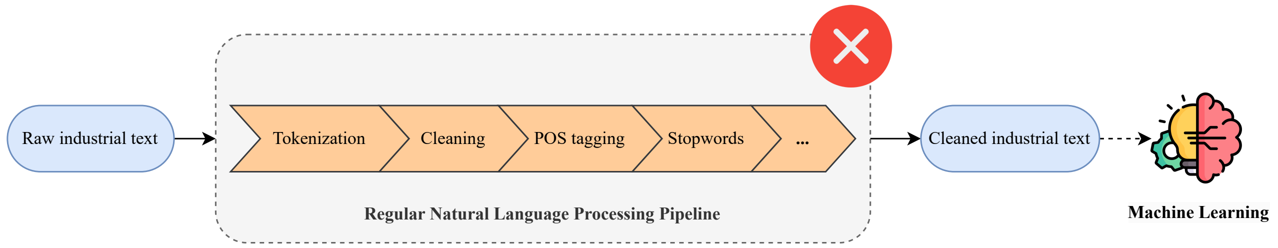

Figure 1: *Processing industrial text through regular NLP pipeline*

Literature Review

The underutilization of industrial data is a severe problem. With the increased focus on multi-modal machine learning, researchers are leveraging available data from various sources to develop intelligent industrial systems [6]. Machine learning algorithms have come a long way from models that can perform image and video analysis with near-human accuracy to models able to generate indistinguishable human-like text. Models such as Google's BARD and OpenAI's GPT-4 are multi-modal and can generate different responses depending on the underlying task [8].

Researchers are now trying to utilize rarely used data sources such as industrial text and machine knowledge to develop algorithms and generate insights. Jose et al. explored the possibility of using additional data sources besides sensor data, such as maintenance reports and production line cameras from the manufacturing process in the context of PHM and multi-modal machine learning [6]. Recently, researchers from General Electric (GE) Digital, the National Institute of Standards and Technology (NIST), and the University of Western Australia (UWA) formalized this research problem as a new subdomain of Artificial Intelligence (AI). The proposed subdomain, Technical Language Processing (TLP), deals with developing NLP pipelines for processing technical language or domain-specific text [9]. TLP recognizes that technical/industrial language differs from regular text due to its unstructured nature and contextual nuances. It addresses the challenges associated with analyzing and understanding text containing domain-specific knowledge. The main goal is to create tools, domain-specific models, and pipelines to recognize industrial text well and derive meaningful insights.

Other researchers are applying statistical word sense representation approaches to produce semantic documents. These documents provide consistent exchangeable information at the word and sentence levels to ensure writers and readers have the same understanding of different concepts. The proposed approach is validated on the Chinese e-business data but can be extended to enhance semantic understanding in other domains, such as industrial text. [10]. Hao et al. proposed a scheduling procedure for a printed circuit board production facility using industrial text mining. The proposed approach tries to avoid scheduling bottlenecks by solving larger problems by tackling the sub-problems of bottleneck candidates [11]. In another study, Chung et al. proposed a methodology for equipment fault classification and expected failure time using pre-shipment data, including equipment manuals and maintenance documents. Compared to traditional approaches, the proposed methodology is unique because it uses no operational data collected after the shipment during equipment operation. The study shows a good example of how industrial text and documentation can predict faults [12]. Romero et al. proposed a hybrid approach based on combining a Long Short-Term Memory Network (LSTM) and Process Capability Assessment Ontology (PCAO) for evaluating the quality of business processes. LSTM handles enterprise process textual data, and PCAO is based on rules to calculate various process attributes such as ratings, capability level, etc. Since the quality of the business processes directly impacts the quality of the products and services, the proposed hybrid framework can help improve overall productivity by improving individual processes [13]. Business processes such as maintenance and asset management are fundamental to the availability of physical assets ensuring minimum risk of downtime and maintenance cost. Polenghi et al. explored the development process for ontological information to support maintenance and asset management. Since ontologies provide knowledge to support interoperability at both technical and semantic levels, the study can be a guideline for a generic approach to ontology development [14]. Naqvi et al. proposed a methodology to classify French maintenance work orders using large language models such as BERT [15].

Stewart et al. proposed Echidna, an interactive interface to visualize historic maintenance short text in the form of the knowledge graph [16]. Echidna is based on MWO2KG (Maintenace Work Order to Knowledge Graph), a deep learning-based technique to transform MWOs into the knowledge graph. Another study proposed using maintenance work orders to discover critical Key Performance Indicators (KPIs) through natural language processing techniques [17]. Nandyala et al. evaluated different word representation techniques in the context of Technical Language Processing. They compared classical techniques with more advanced text representation techniques like BERT [18]. Akshay et al. evaluated different approaches for generating maintenance work order representations (embedding) to develop a recommendation system for industrial text [19]. In [20], the author highlighted the effects of the underutilization of maintenance text on the maintenance knowledge intelligence in the manufacturing process. The author proposed TextPlan, a compositional framework for understanding and quantifying industrial text at the syntax and semantic levels. Sharma and Kumar proposed a hybrid information retrieval system based on skip-gram embeddings and domain ontology for unstructured text [21].

Developed initially to process text and gaining state-of-the-art performance on natural language processing tasks, transformer architecture is now gaining popularity in other domains, such as vision and time series [22, 23, 24, 25]. Zhao et al. proposed a self-learning approach to extract ontology from semi-structural process planning documents automatically. The extracted data from the documents is presented as concepts in the ontology. The authors then used important concepts to generate a mining dataset. Association rule mining is used to uncover the relevant patterns from the mining dataset. The developed ontology and generated knowledge base serve as a semantic representation of heterogeneous data used to help improve the decision-making of manufacturing activities for CNC machines [26]. Bao et al. proposed a methodology to represent the assembly process as a network of geometric elements and topological relationships of the product. The developed graph is then converted to fixed-length graph vector embeddings using node2vec. These representations are then used to develop an algorithm for predicting the execution time of assembly work steps [27]. Our previous studies proposed methodologies to develop a maintenance decision support system for maintenance work orders using embeddings from state-of-the-art transformers models. These studies explored how transformers models can be adopted for industrial use cases [28, 29]. We also proposed an architecture integrating human knowledge-centered maintenance decision-support solutions in a digital twin-enabled modern manufacturing environment [30]. This study continues our work toward improving semantic search performance in MWOs using multi-modal machine learning techniques.

Proposed Methodology

Industrial text is complex and usually consists of raw maintenance records without labels. MWOs may also contain the site, machine, manufacturer, or repair cost information. This information usually does not help much in retrieving relevant information at the system/subsystem level. This study proposes an information retrieval approach for industrial text using hybrid data sources (a combination of text and machine ontology). An ontology is developed to improve information retrieval performance by introducing system/subsystem-level context during the search. This section discusses the development of the Technical Language Processing (TLP) pipeline for an information retrieval system to support maintenance operations. The following subsections describe various steps in developing the proposed system, including dataset description, formalization of ontology, model development, and hybrid information retrieval process.

Dataset

The dataset used for the study consists of 5486 maintenance work orders from 8 similarly sized excavators. Five of the eight excavators are 1400 HP units (Set A), and the rest are 1440 HP units (Set B). These MWOs are collected during mining operations at various mine sites across Australia over ten years (from 2002 onward) [31, 32]. The shared dataset has two versions, raw and cleaned. Table 1 lists the columns in the raw and additional columns in the cleaned version of maintenance records. The original columns are in the raw version, and the cleaned version results from another study [33]. Out of the original columns in the raw version, the first column lists the date of the maintenance operation. Asserts columns have the assert id. The original short text column contains the textual description of the maintenance problem. PM type describes the type of maintenance, PM01 is the code for corrective maintenance, and PM02 is the code for preventive maintenance, and so on. The cost column shows the cost of the maintenance operation in Australian dollars. Figure 2 shows the word cloud for the excavator maintenance records dataset used in this study, showing frequent terms and systems appearing in maintenance records. It is evident by the frequency of some terms that most of the problems in the dataset are relevant to engine, bucket, leakages, etc.

The cleaned version has some additional columns

Table 1: *Default fields and our added column in excavator maintenance records*

Version	Column name	Column description
	BscStartDate	Maintenance operation date
	Asset	Id of the asset under maintenance
Raw	OriginalShorttext	Description of the maintenance problem
	PMType	Type of maintenance operation
	Cost	Cost of maintenance operation
	RunningTime	Running time of the asset
	MajorSystem	Major system associated with maintenance operation
	Part	Part needing maintenance
	Action	Maintenance action performed
	Variant	Maintenance operation variant
Cleaned	FM	Fault mode
	Location	Location of the fault
	Comments	Comment associated with the maintenance operation
	FuncLocation	Functional location of the fault
	SuspSugg	Suspend normal operation
	Rule	Rule associated with extracted information
Our addition	Ontology label	Ontology label, associated with maintenance work order

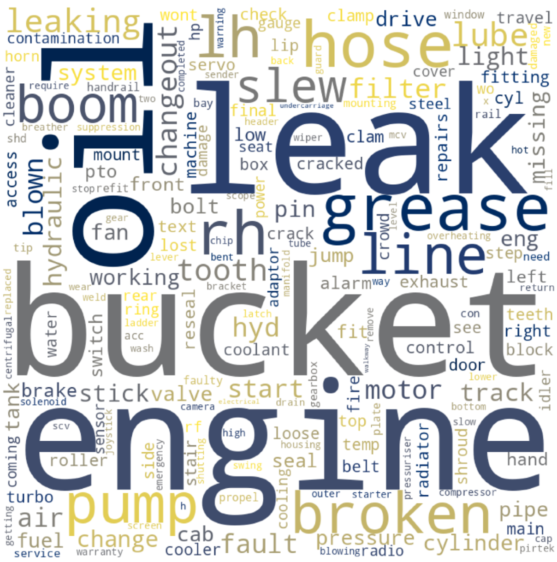

Figure 2: *Word cloud of excavator maintenance records*

added by using Data Extraction and Cleaning tool (DEST). DEST is a rule-based customizable MATLAB script that is used to identify the functional location, categorization, and critical level of the fault from raw work orders. The additional columns in the cleaned version include information such as the affected major system, part, the action performed, the location of the fault, etc. Although these columns provide additional context, there are lots of missing values. The detailed process to generate the cleaned version from the raw work orders is described in [33]. For this study, we added a new column (ontology label) to each record in the raw version using the information in the cleaned version. The ontology label column is added to standardize the assignment of relevant systems/subsystems for available work orders in the dataset and to solve missing information problems in the additional columns. Further explanation on ontology formalization and generation of ontology label column to the dataset is available in Section .

Ontology formalization

An Ontology specifies concepts and relationships for an agent or a group of agents. It can be considered as a set of concept definitions [34]. The purpose of ontology is to show relations between concepts and categories. In the life sciences, ontologies have long been used to represent domain

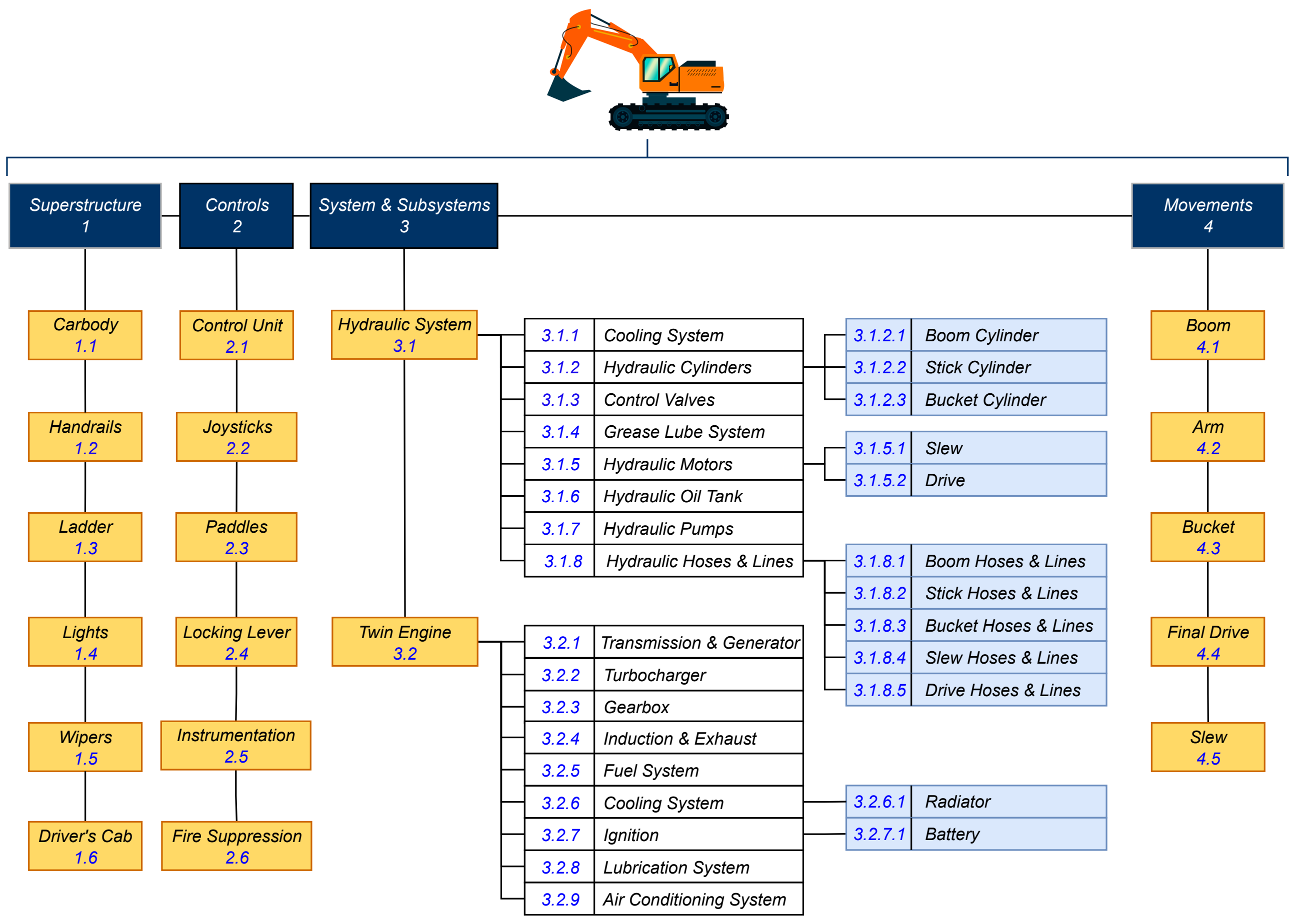

Figure 3: *Hydraulic excavator ontology*

knowledge formally. An ontological representation of domain knowledge can be used in the reasoning process and enhance domain understanding. Ontologies are employed practically in most major biological databases. Ontologies are now more frequently utilized as background or domain knowledge sources in similarity-based techniques and machine learning models. Ontology-machine learning integration techniques are still novel and under active development [35]. This study explores how ontology can enhance semantic search results in industrial text. To integrate domain knowledge of excavators in work orders, we developed an ontology for hydraulic excavators with the help of domain experts. Figure 3 presents the developed ontology for hydraulic excavators.

We tried to capture structural and functional domain knowledge related to excavators in the proposed ontology presented in Figure 3. The developed ontology represents concepts associated with hydraulic excavators. These concepts are divided into four main categories; first is the superstructure which contains concepts related to the structural aspects of hydraulic excavators, including car body, handrails, ladders, drivers cab, etc. The second category is controls, which include control units, joysticks, paddles, instrumentation, etc. The third category is systems and subsystems, which contain two central systems of hydraulic excavators, namely the hydraulic system and engine. These concepts are further subdivided into subsystems depending on the respective functionalities. The last category is Movements which lists different concepts that control various movements of the hydraulic excavators, such as the bucket, boom, arm, final drive, and slew movements. Each concept in the excavator ontology is assigned a hierarchical label based on the position. This hierarchical label gives a sense of depth and association depending on the positioning and relations to the main concept.

Integrating excavator ontology and work orders To integrate domain knowledge captured by the ontology into the maintenance work orders, we labeled each instance of work orders with the relevant hierarchical label from the excavator's ontology. We used the cleaned version of the dataset mentioned in Section to label the work orders in the raw dataset version with hierarchical labels. The work orders text is the same in the raw and cleaned dataset versions of the excavator MWOs. The reason behind using the

Table 2: *Sample excavator maintenance records*

Sr.	Maintenance Work Order (MWO)	Ontology label
1	REPLACE BUCKET TEETH 2 OFF REPLACE 2 BUCKET TEETH REPLACE 2 LOST TEETH BUCKET TEETH BROKEN X 2 Remove bucket teeth 3 & 4	4.3
2	Aircons not getting cold SHD24 Aircon not working SHD0024 airconditioner not getting cold	3.2.9
3	L/H/Side rear estop pull cord u/s	2.5
	Cracked mount l/h/s R/H pump box	3.1.7
	big oil leak - R.H.S. engine - R.H. bank	3.2.8

cleaned version to add the ontology label is the availability of additional system/subsystem-level information extracted through rule-based reasoning [33]. The cleaned version is only used to facilitate labeling. The reason for not adding columns from the cleaned version in the semantic search is lots of missing values. With the hierarchical ontology labels, we tried to squeeze the information available in the additional columns of the cleaned version into a single label.

Table 2 presents sample sets from excavator maintenance records with respective ontology labels. The samples in Table 2 show unconventional writing styles and complexities associated with the industrial text. Sample sets 1 and 2 contain cases related to the same problem but described differently by maintenance operators. Samples in set 1 are functional and associated with the bucket hence assigned bucket ontology label. Similarity samples in set 2 are associated with the cooling system and thus assigned the same label (3.2.9). Sample set 3 shows how sides of different systems/subsystems are described differently in the cases adding to the complexity of the industrial text. Cases in set 3 are associated with concepts and assigned ontology labels 2.5, 3.17, and 3.28, respectively.

Model development

To process text using computers, we need to represent text in a numerical representation. This section explains the steps to develop the machine learning model to process maintenance work orders. As previously stated in the introduction, to generate meaningful representations (embeddings) of maintenance work orders, we used one of the state-of-art NLP models Bidirectional Encoder Representations from Transformers (BERT). The following text explains the BERT model's data preparation and fine-tuning process for industrial text.

Data preprocssing The industrial text contains domain-specific knowledge and terminologies that are non-existent in the regular text. Therefore regular NLP pipelines are not suitable for processing industrial text. Other common problems associated with industrial text are structural irregularities, spelling mistakes, and the use of acronyms. These problems, paired with the different writing styles of different operators, add to the complexity of the industrial text and make it very hard to process [9]. The traditional approaches for processing industrial text mainly depend on application-specific custom pipelines. The objective of these pipelines is to normalize the industrial text before feeding it to the machine learning models. Developing these custom pipelines often requires domain knowledge and manual labor to cover all possible scenarios. These pipelines also require constant updates with time as new terms and scenarios are added to the target dataset. The proposed model (BERT) can be used to develop an automatic technical language processing pipeline that can process industrial text automatically without any preprocessing and normalization using traditional custom pipelines. To explore the capabilities of BERT on the domain-specific industrial text, we performed minimal preprocessing by only normalizing the case of characters to upper case.

Domain fine-tuning The preprocessed work orders can be used to fine-tune the BERT model on the domain-specific text. Although a pre-trained BERT model can convert the input text to embeddings (numerical feature vectors), studies show fine-tuning the model on domain-specific text considerably improves the performance and semantic understanding of the model for target text [28]. The process is also essential to understand complex and intricate patterns in the industrial text. Another reason behind fine-tuning is that BERT, a word embedding model, cannot generate meaningful sentence or paragraph-level embeddings. To adopt the BERT model to generate meaningful sentence or paragraph-level embeddings, sentence BERT[1] was proposed [36]. After fine-tuning, the BERT model can generate embeddings comparing similarities between input queries and past maintenance records. BERT is a famous text-processing model and has different variants. For this study, we used "bert-base-uncased" the model proposed in the original version of the BERT paper [7]. The fine-tuning process is further explained in the following text.

Although the first sentence BERT model was pro-

[1] `https://www.sbert.net/`

posed to be fine-tuned through supervised training using labeled data. This inspired development of unsupervised fine-tuning techniques to train sentence BERT models [37, 38, 39, 40]. Industrial text, especially maintenance text, is mostly unlabeled raw notes by maintenance operators. We also used the unsupervised fine-tuning technique Transformer-based Sequential Denoising Auto-Encoder (TSDAE) for this study [40]. TSDAE being the most efficient and having state-of-the-art performance among other fine-tuning techniques, is selected as the preferred method for this study [38, 39, 40]. This study mainly focuses on how hybrid data source (multi-modal data) text and ontology can enhance semantic search results. A comparison of different unsupervised fine-tuning approaches is beyond the scope of this study.

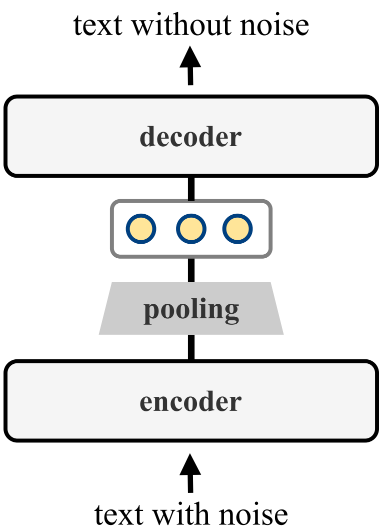

Figure 4: *TSDAE Architecture [40]*

Figure 4 shows the architecture of the TSDAE fine-tuning approach. The first step in TSDAE fine-tuning process is to create a noisy dataset from the original dataset (industrial text in our case). Different techniques can be used to create a noisy version of the original dataset, such as deleting or swapping certain words in the original text. In the original TSDAE paper, after experimenting with different noise techniques, authors found that deletion with a ratio of 0.6 produced the best results. The next step is to input these noisy training samples into an encoder model (BERT in our case). The encoder model converts the noisy input text to tokens and generates respective token embeddings. These token embeddings are then pooled into fixed-length sentence or paragraph-level embeddings using a pooling layer. The final step of the process is decoding noisy sentence embeddings to generate original text without noise. The weights of the encoder model are optimized based on decoder feedback during the fine-tuning process. After complete fine-tuning, we only use the encoder to generate sentence or paragraph-level embeddings. The TSDAE is a variation of the encoder-decoder transformer architecture which focuses key and value of cross attention on fixed-size sentence embedding instead of word embeddings.

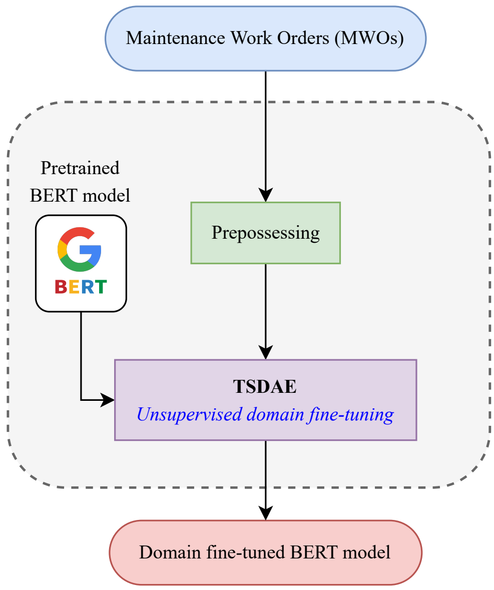

Figure 5: *Flow diagram of TSDAE based domain fine-tuning of pre-trained transformers model*

$$H^{(k)} = Attention\left(H^{(k-1)}, [s^T], [s^T]\right) \quad (1)$$

$$Attention\,(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d}}\right)V \quad (2)$$

Equation 1 and 2 show the formulas for the modified cross attention. In Equation 1, $H^{(k)} \in \mathbb{R}^{t \times d}$ represents the hidden states of decoder at k-th layer, within t decoding steps. d is the size of the fixed-length sentence embedding vector, and $[s^T] \in \mathbb{R}^{1 \times d}$ is a single row matrix of fixed-length sentence embedding vector. Q, K, and V in Equation 2 represent cross-attention for query, key, and value, respectively [40].

The inputs of the TSDAE algorithm are unlabeled domain-specific training data and a pre-trained transformer-based model. The output of the fine-tuning process is a domain fine-tuned sentence BERT model able to generate meaningful sentence or paragraph-level embeddings. Figure 5 shows the flow diagram of the TSDAE fine-tuning pipeline. The process starts with the maintenance work orders as input, followed by preprocessing (change to upper case). The preprocessed data and pre-trained BERT model are then input into the TSDAE fine-tuning algorithm, and the final output is the domain fine-tuned sentence BERT model. Because the weights of the encoder models are optimized based on the feedback from the decoder for fixed-size sentence embeddings, the resulting fine-tuned model generates meaningful sentence or paragraph-level embeddings instead of word embeddings by the original pre-trained input model. The length of the generated sentence embedding is the same as the input pre-trained BERT model (768 for the base model and 1024 for the large model).

Hybrid information retrieval system

After fine-tuning the BERT model on excavator maintenance work orders, we can now use this model for information retrieval. Figure 6 presents the information retrieval flow diagram using hybrid weighted similarity. The input to the retrieval process is a maintenance search query, and the output is top k similar maintenance work orders from the maintenance database. In case of a new maintenance problem, the operator inputs the problem description as a search query. The search query consists of input text describing the problem and the target concept label from ontology. For example, suppose the operator is searching for similar information to a problem in the engine's lubrication system. In that case, the operator will provide the problem description and respective ontology label (i.e., 3.2.8) from the ontology presented in Figure 3. The text from the query is preprocessed and converted to semantic embedding using a domain fine-tuned model. After processing the text, we extract the semantic embeddings of past maintenance records along the respective ontology labels assigned to each record during the ontology development and labeling process. Text embeddings are compared with text embeddings of the records in the maintenance database, and the query ontology label is compared with the respective ontology labels. Finally, a combined weighted hybrid similarity is calculated using text and ontology similarity output. The following sections describe the process of calculating various similarities in further detail.

Text similarity The resulting embeddings from the domain fine-tuned model are dense feature vectors. The standard similarity measure used to compare these embedding vectors is cosine similarity. Equation 3 shows the formula for cosine similarity between two non-zero vectors **A** and **B**.

$$Similarity_{cosine} = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \times \|\mathbf{B}\|} \tag{3}$$

Cosine similarity measures the angle between input vectors by calculating the dot product of these vectors divided by the product of their lengths. Cosine similarity is the appropriate measure for this comparison as it focuses on the separation between the vectors. Similar vector embeddings tend to cluster together in the embedding space. The output value of the cosine similarity ranges between -1 and 1. It is -1 for two opposite vectors, 0 for two orthogonal vectors, and 1 for proportional vectors. Since the generated vectors through the BERT model mainly belong to the positive space, cosine similarity values range between 0 and 1.

Ontology similarity Measuring the semantic similarity between concepts in an ontology is a well-researched topic. There are two primary approaches for calculating the topological similarity between ontological concepts. Edge-based techniques in which similarity between the compared concepts is based on information about edges and their types that lead to those concepts. Node-based techniques where similarity between concepts depends on the nodes in the ontology and their properties. Wu and Palmer's similarity is one of the straightforward and intuitive edge-based semantic similarity techniques [41]. Given an ontology Ω consisting of a set of nodes and a root node (R), Equation 4 presents the formula for measuring Wu and Palmer's similarity between two concepts ($C1$ and $C2$) in the ontology Ω.

$$Similarity_{wp}\ (C1,\ C2) = \frac{2 \cdot N}{N1 + N2} \tag{4}$$

To understand the formalization of Equation 4, consider an ontology extract from Ω presented in Figure 7 where $C1$ and $C2$ are the two concepts being compared. The conceptual similarity based on Equation 4 between $C1$ and $C2$ is $2N$ divided by $N1$ and $N2$, where N is the distance of the root node to a common ancestor (CA) between concepts ($C1$ and $C2$). $N1$ and $N2$ are distances of $C1$ (concept 1) and $C2$ (concept 2) from the root node. The output of Wu and Palmer's similarity ranges between 0 and 1. Wu and Palmer's similarity is extensively used in the literature due to its simplicity and performance. Various studies proposed variations of Wu and Palmer's similarity to increase performance and effectiveness [42, 43, 44, 45].

Wu and Palmer's similarity is based on edge counting and calculates the similarity between the two concepts compared to the distance from their nearest common subsuming parent. The more general the subsuming, the lower the similarity between the concepts and vice versa. This adds a limitation for the ontologies where we have uniform distances (i.e., all the semantic connections between concepts have the same weight). For instance, measuring the similarity between the concepts "Boom Cylinder" and "Hydraulic Pumps" exceeds the similarity between concepts "System & Subsystems" and "Boom Cylinder". Given a concept, Wu and Palmer's similarity

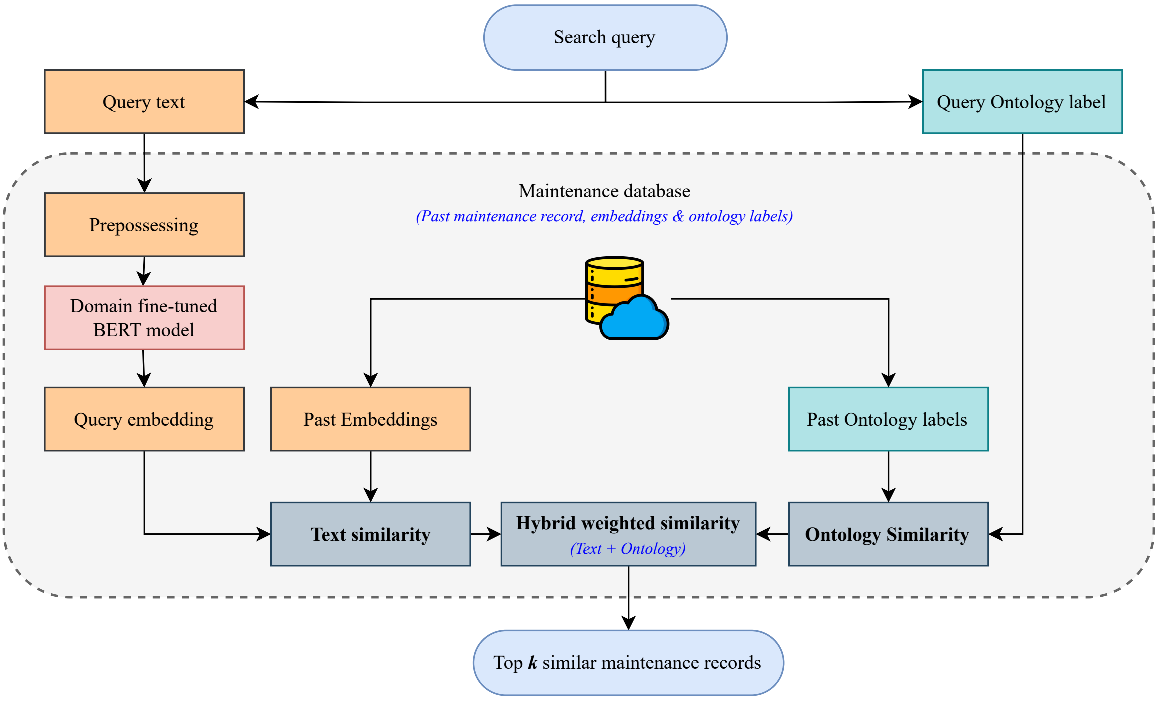

Figure 6: *Flow diagram of the proposed hybrid information retrieval system*

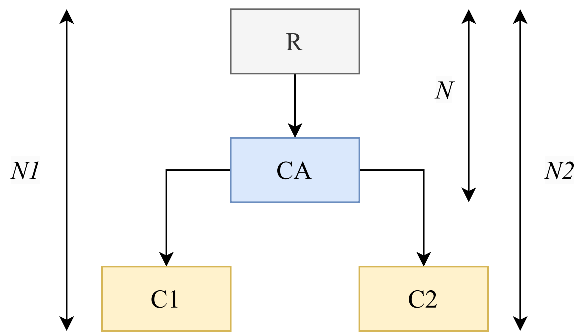

Figure 7: *Sample ontology extract from ontology* Ω

gives a higher value of similarity between that concept and concepts in its vicinity compared to the concepts in the same hierarchy. Let "System & Subsystems", "Boom Cylinder", and "Hydraulic Pumps" are concepts $C1$, $C2$, and $C3$, respectively; we can verify the above example by performing the following calculations:

$$Similarity_{wp}(C1, C2) = 2 * 1/(1 + 4) = 0.4$$

$$Similarity_{wp}(C2, C3) = 2 * 2/(4 + 3) = 0.57$$

To overcome this limitation, we use a modified version of Wu and Palmar's similarity, the tbk similarity [42]. This similarity measure is inspired by the advantages of Wu and Palmar's similarity and uses the same formula with the addition of a penalty factor to overcome its limitation. Equation 5 represents the formulation of the tbk similarity. The new term introduced to the parent Equation 4 is $PF(C1, C2)$, which is the penalty factor.

$$Similarity_{tbk}\ (C1,\ C2) = \frac{2 \cdot N}{N1 + N2} \times PF(C1, C2) \tag{5}$$

$$PF(C1, C2) = (1 - \lambda).(Min(N1, N2) - N) + \lambda(|N1 - N2| + 1)^{-1}$$

The coefficient λ in the penalty factor equation is a boolean to indicate if the compared concepts belong to the same hierarchy or neighborhood. Its value is 0 if the compared concepts are from the same hierarchy and 1 if they belong to the same neighborhood. Compared concepts belong to the same hierarchy if connected by a continuous path, whereas compared concepts are considered in the same neighborhood if they belong to the same system/subsystem but are not connected directly by a continuous path. $Min(N1, N2)$ is the minimum of the distance among $N1$ and $N2$ for concepts $C1$ and $C2$. This approach only penalizes if the compared concepts are in the same neighborhood to compensate for Wu and Palmar's similarity limitation. Detailed explanations of this similarity measure can be found in the original paper [42].

Hybrid weighted similarity We propose a hybrid weighted similarity measure to retrieve relevant excavator maintenance records based on the combined semantic similarity of text and ontology presented in Figure 6. The idea is to develop a similarity measure for hybrid data sources to support multi-modal similarity analysis. Equation 6 shows the proposed hybrid similarity formula. In Equation 6, $similarity_{text}$ is cosine similarity described in Section that is the similarity between the embedding of the query text and the embeddings of the past maintenance records from domain fine-tuned model. The value $similarity_{ontology}$ represents the similarity between query ontology labels and ontology labels of the respective past maintenance records.

$$Similarity_{hybrid} = (Similarity_{text} * w_{text}) + (Similarity_{ontology} * w_{ontology}) \quad (6)$$

The weights assigned to textual and ontological semantic similarities are the w_{text} and $w_{ontology}$ where $w_{text} + w_{ontology} = 1$. The proposed weighted similarity measure tries to leverage insights from two different sources. It is also flexible and adaptable because of the inclusion of weights to fine-tune the contribution from different data sources. The proposed approach is easily adaptable and can be extended to additional future data sources to enable multi-modal similarity calculation.

Results and discussion

We performed various analyses to assess the performance of the developed approach. This section presents the results of the proposed approach, including the evaluation process and discussion. To better understand the performance and practicality of the proposed approach, we present two different versions of the results in this section (precision-based and pattern analysis-based). Precision-based results provide the quantitative perspective, whereas pattern analysis-based results present the qualitative aspect by showing the quality of similar cases obtained through the proposed approach.

Evaluation

Section describes the development process of a hybrid information retrieval system using multi-modal data. Since the target industrial text contains raw maintenance records, no ground truth is provided with the dataset. We proposed precision-based and pattern-based analysis to measure the performance of the hybrid information retrieval system. To quantify the performance, we developed a set of queries associated with the different systems/subsystems in the excavator. We tested the system on a total of 103 queries related to different maintenance issues of excavators to see if the retrieval system generates relevant results. Tested queries are based on careful analysis of different excavator maintenance checklists to identify various fault associations. Table 3 presents common issues associated with various systems/subsystems of hydraulic excavators on which we based our queries.

These queries are then used as sample problems to test the performance of the proposed system. We extracted top k results for each query using similarity analysis. Analysis shows that the maintenance database has at least five similar instances for each target query in the query set, so the value of k is set to 5. After developing the test queries and identifying the value of k, we generated similar maintenance records against all queries. We performed this retrieval step separately with text-based and hybrid weighted similarity (text + ontology) to compare the performance gain due to the proposed approach. Maintenance specialists then label the results of these two retrieval strategies for the corresponding input queries. This step is crucial to verify the relevance of extracted similar work orders and calculate the performance of the two strategies. If the proposed work order is similar or relevant to the input query, they labeled it True Positive (TP); if not, they labeled it False Positive (FP). Finally, we determined the precision for each query based on the number of true positives and false positives among the top k similar work orders. Equation 7 shows the formula for precision calculation, which represents the proportion of correct suggestions by the proposed system.

$$Precision = \frac{True\ Positive}{True\ Positive\ +\ False\ Positive} \quad (7)$$

To test the performance gain due to the proposed multi-modal technique, we compared the precision results only using text similarity with hybrid weighted similarity (text and ontology). Table 4 presents precision-based results on the target system level. We also inspected the results labeled by maintenance specialists to identify interesting patterns. Table 5 presents the pattern-based analysis of the results where we compared the predicted similar cases obtained using only text with the predicted similar cases obtained using the hybrid approach (text + ontology). A detailed analysis of precision and pattern-based results is presented in Section .

Discussion

This section discusses results presented in Table 4 and 5 from both quantitative and qualitative per-

Table 3: *Common issues associated with various system/subsystems of hydraulic excavators*

Sr.	System/Subsystem	Common issues
1	Bucket	Crack, wear, damage, leakage, lubrication
2	Boom	Wear, damage, leaks, lubrication
3	Slew/swing	Cracks, leaks, slipping brakes, overheating
4	Engine	Excessive smoke, overheating, leaks
5	Radiator	Damage, leaks, blockage
6	Ignition system	Starting issues, battery issues
7	Air conditioning	Damage, Not cooling, Not warming
8	Hydraulics	Leaks, crack, cooling fan issues, blown rings
9	Pumps, gearboxes & grease lines	Damage, cracks, low pressure, need overhaul
10	Driver's cabin	Joystick, seat adjustment, gauges, alarms, instrumentation
11	Car body, superstructure	Broken rails and ladders, cracks, leaks

Table 4: *Comparison of information retrieval performance based on precision analysis*

Sr.	System/Subsystem	Precision		Performance gain
		Only Text	*Text + Ontology*	(%)
1	Bucket	0.92	**1.00**	8
2	Boom	0.91	**0.98**	7
3	Slew/swing	0.90	**1.00**	10
4	Engine	0.90	**1.00**	10
5	Radiator	0.90	**1.00**	10
6	Ignition system	1.00	1.00	0
7	Air conditioning	0.92	0.92	0
8	Hydraulics	0.93	**0.98**	5
9	Pumps, gearboxes & grease lines	0.92	**0.96**	4
10	Driver's cabin	0.88	**1.00**	12
11	Car body, superstructure	0.77	**0.94**	17
	Overall average	0.90	**0.98**	8

Table 5: *Comparison of information retrieval performance based on pattern analysis*

Sr.	Predicted Similar Case Only text		Predicted Similar Case Text + Ontology	
1	***Query:*** *Oil leak bucket*		***Ontology label:*** *3.1.8.3*	
	OIL LEAK AT BACK OF BUCKET	✓	Oil leak on bucket cylinder hose	✓
	OIL LEAK BUCKET CLAM	✓	GREASE LEAK AT BUCKET	✓
	OIL LEAK	×	Grease Leak front of bucket	✓
	OIL LEAK	×	REPAIR GREASE LEAK AT BUCKET.	✓
	Oil leak on bucket cylinder hose	✓	BLOWN HYDRAULIC OIL HOSE ON BUCKET	✓
	Precision	0.6	*Precision*	1.0
2	***Query:*** *Bucket teeth 2 off*		***Ontology label:*** *4.3*	
	REPLACE BUCKET TEETH 2 OFF	✓	REPLACE BUCKET TEETH 2 OFF	✓
	REPLACE 2 BUCKET TEETH	✓	REPLACE 2 BUCKET TEETH	✓
	BUCKET TEETH BROKEN X 2	✓	BUCKET TEETH BROKEN X 2	✓
	REPLACE 2 BUCKET TEETH AND KEEPERS	✓	REPLACE 2 BUCKET TEETH AND KEEPERS	✓
	Remove bucket teeth 3 & 4	✓	Remove bucket teeth 3 & 4	✓
	Precision	1.0	*Precision*	1.0
3	***Query:*** *Engine hose blown*		***Ontology label:*** *3.2*	
	Blown engine oil hose	✓	blown head gasket r/h engine	✓
	LH ENGINE TURBO HOSE BLOWN	✓	Blown engine oil hose	✓
	BLOWN HOSE ON RH ENGINE	✓	LH ENGINE TURBO HOSE BLOWN	✓
	BLOWN HYDRAULIC HOSE	×	HOSE ON AFTERCOOLER BLOWN OFF	✓
	BLOWN TURBO HOSE L/H ENGINE	✓	BLOWN HOSE ON RH ENGINE	✓
	Precision	0.8	*Precision*	1.0
4	***Query:*** *Engine blowing smoke*		***Ontology label:*** *3.2.4*	
	R/H engine blowing smoke	✓	R/H engine blowing smoke	✓
	L.H ENGINE BLOWING SMOKE	✓	L.H ENGINE BLOWING SMOKE	✓
	BLOWING EXCESSIVE SMOKE	×	L/H ENGINE BLOWING EXCESSIVE WH SMOKE.	✓
	L/H ENGINE BLOWING EXCESSIVE WH SMOKE.	✓	R/H ENGINE BLOWING WHITE SMOKE .replaced	✓
	R/H ENGINE BLOWING WHITE SMOKE .replaced	✓	RH ENG BLOWING EXCESSIVE SMOKE	✓
	Precision	0.8	*Precision*	1.0
5	***Query:*** *Engine oil leak*		***Ontology label:*** *3.2.8*	
	Engine oil leak	✓	Engine oil leak	✓
	OIL LEAK UNDER ENGINE	✓	OIL LEAK UNDER ENGINE	✓
	OIL LEAK	×	OIL LEAK ON LH ENGINE	✓
	OIL LEAK	×	Leak under engine	✓
	OIL LEAK ON LH ENGINE	✓	leaking engine oil hose	✓
	Precision	0.6	*Precision*	1.0
6	***Query:*** *Slew circuit / motor overheating*		***Ontology label:*** *3.1.1*	
	Slew circuit overheating	✓	Slew circuit overheating	✓
	No 4 slew overheating	✓	No 4 slew overheating	✓
	SLEW MOTORS HIGH TEMP	✓	SLEW MOTORS HIGH TEMP	✓
	Right slew motor leaking oil	×	REPORTED HIGH SLEW TEMP	✓
	bypassing oil into slew motor	×	REPAIR SLEW TEMP WARNING FAULT	✓
	Precision	0.6	*Precision*	1.0
7	***Query:*** *Broken clamp boom*		***Ontology label:*** *3.1.8.1*	
	broken bolts stauff clamp on boom	✓	broken bolts stauff clamp on boom	✓
	Broken bolt stauff clamp top of boom	✓	Broken bolt stauff clamp top of boom	✓
	REPAIR BROKEN STAUFF CLAMPS ON BOOM	✓	REPAIR BROKEN STAUFF CLAMPS ON BOOM	✓
	loose clamp on pipe back of boom	✓	loose clamp on pipe back of boom	✓
	clamp had broken bolts	×	STAUFF CLAMPS ON BOOM	✓
	Precision	0.8	*Precision*	1.0
8	***Query:*** *Change boom cylinder*		***Ontology label:*** *3.1.2.1*	
	CHANGE BOOM CYLINDER RHS	✓	CHANGE BOOM CYLINDER RHS	✓
	CHANGE OUT LH BOOM CYLINDER	✓	CHANGE OUT LH BOOM CYLINDER	✓
	Change LH Boom Cylinder (Terex)	✓	Change LH Boom Cylinder (Terex)	✓
	Changeout CYLINDER BOOM - LEFT	✓	Changeout CYLINDER BOOM - LEFT	✓
	Change LH bucket roll cylinder	×	CHANGEOUT RIGHT HAND BOOM CYLINDER	✓
	Precision	0.8	*Precision*	1.0

Continued on next page

Sr.	**Predicted Similar Case** $_{\text{Only text}}$		**Predicted Similar Case** $_{\text{Text + Ontology}}$	
9	***Query:*** *Aircons not cold*		***Ontology label:*** *3.2.9*	
	airconditioner not getting cold	✓	Aircons not getting cold SHD24	✓
	Aircons not getting cold SHD24	✓	air con. not getting cold.	✓
	air con. not getting cold.	✓	airconditioner not getting cold	✓
	Air conditioners not working	✓	Air conditioners not working	✓
	Air con blowing hot air	✓	HEATER NOT GETTING WARM	×
	Precision	1.0	*Precision*	0.8
10	***Query:*** *Replace / repair seat*		***Ontology label:*** *1.6*	
	Repair seat	✓	Repair seat	✓
	Repair seat slide	✓	Repair seat slide	✓
	Repair Slide On Seat	✓	Repair Slide On Seat	✓
	repair / replace broken handrail	×	Seat adjuster U/S	✓
	repair/replace broken hand rail	×	Seat backrest can not adjust	✓
	Precision	0.6	*Precision*	1.0
11	***Query:*** *Repair Gauge*		***Ontology label:*** *2.5*	
	Repair Fuel monitoring system	✓	REPAIR R/H FUEL GAUGE FAULT	✓
	CHECK AND REPAIR GREASE SYSTEM	×	Murphy switch gauge needs replacing	✓
	CHECK /REPAIR FUEL LEVEL WARNING	×	Replace engine temp gauges	✓
	REPAIR CAMERAS	×	Reposition LH Engine murphy gauge	✓
	REPAIR R/H FUEL GAUGE FAULT	✓	RHS Fuel gauge intermittent	✓
	Precision	0.4	*Precision*	1.0
12	***Query:*** *Eng warning sign coming*		***Ontology label:*** *3.2*	
	ENG WARNING LIGHT ON	✓	ENG WARNING LIGHT ON	✓
	LH ENG WARNING LIGHT IS ON	✓	LH ENG WARNING LIGHT IS ON	✓
	slew warning light coming	×	R/H ENG WARNING LIGHT ON	✓
	R/H ENG WARNING LIGHT ON	✓	engine warning light coming on	✓
	engine warning light coming on	✓	R/H ENGINE WARNING LIGHT ACTIVE	✓
	Precision	0.8	*Precision*	1.0
13	***Query:*** *Pump drive low pressure*		***Ontology label:*** *3.1.7*	
	LH PUMP DRIVE LOW PRESSURE FAULT	✓	RH PUMP GEAR PRESSURE LOW.	✓
	RH PUMP GEAR PRESSURE LOW.	✓	RH PUMP PRESSURE LOW	✓
	RH PUMP PRESSURE LOW	✓	LH PUMP DRIVE LOW PRESSURE FAULT	✓
	RH PUMP DRIVE BOX PRESSURE LOW FAULT	✓	RH PUMP DRIVE BOX PRESSURE LOW FAULT	✓
	L/H PUMP GEARBOX LOW PRESSURE.	✓	RH PUMP GEARBOX PRESSURE LOW @R6	✓
	Precision	1.0	*Precision*	1.0
14	***Query:*** *Pump pressure low*		***Ontology label:*** *3.1.7*	
	RH PUMP PRESSURE LOW	✓	RH PUMP PRESSURE LOW	✓
	RH PUMP GEAR PRESSURE LOW.	✓	RH PUMP GEAR PRESSURE LOW.	✓
	System low on pressure	×	PUMP GEAR PRESSURE LOW ALARM ACTIVATED	✓
	R/H PUMP BOX LUBE PRESSURE LOW	✓	RH PUMP GEARBOX PRESSURE LOW @R6	✓
	PUMP GEAR L/H PRESSURE TOO LOW	✓	R/H PUMP BOX LUBE PRESSURE LOW	✓
	Precision	0.8	*Precision*	1.0
15	***Query:*** *Oil leak pump*		***Ontology label:*** *3.1.7*	
	OIl leak at grease pump	✓	Oil leak at pump gearbox	✓
	OIL LEAK	×	Oil leak from PTO	✓
	OIL LEAK	×	RH PRE LUBE OIL PUMP LEAKING	✓
	Oil leak at pump gearbox	✓	OIl leak at grease pump	✓
	HYDRAULIC OIL LEAK	×	OIL LEAK	×
	Precision	0.4	*Precision*	0.8

spectives as previously stated in Section . The developed queries belong to different categories and functional localization of the hydraulic excavators. Table 4 presents the average precision-based results at the system/subsystem level calculated for 103 test queries. We also calculated the overall precision for each approach. It can be observed in the results that if we only use text-based similarity to retrieve similar maintenance records, we achieve an overall precision of 0.9. This performance demonstrates that large language models can identify intricate patterns in complex industrial text. We further observe that using the proposed hybrid similarity approach, we improve 8% with a precision of 0.98 by combining text and ontology similarity. Results indicate that, in some cases, the results of the text-based approach are equal or comparable to the hybrid approach. Still, in all instances, the hybrid approach performs equally to the text-based approach or improves the results. For example, for queries related to the car body and superstructure, we achieved maximum performance improvement of 17% using the proposed hybrid approach. There was no gain in the case of ignition and air conditioning systems. The results are coherent with the fact that the two subsystems are independent of the others and do not have shared vocabulary or concepts that would require discrimination with the help of an ontology. The results show that hybrid similarity approaches using multi-modal data sources can improve the information retrieval performance of complex unstructured industrial text. After precision-based analysis, we inspected the predicted similar cases to identify intricate patterns behind the performance improvement. Table 5 presents a pattern analysis of results generated by the text-based and hybrid approaches. We carefully selected 15 instances from 103 search queries for detailed performance analysis and showed interesting patterns. We color-coded the results from both approaches for better visibility and separation. We also highlighted the interesting patterns for quick identification across different presented cases. Here we discuss these cases in detail for a better understanding.

The first case is related to a leakage problem at the bucket. Since leakages are expected at the lines and hoses connecting the hydraulic system to the bucket cylinder, we selected an ontology label for "Stick Hoses & Lines (3.1.8.3)" as our query ontology label. Pattern-based analysis of the predicted results shows some relevant cases in text-only retrieval. But, we also have some very general cases, such as "OIL LEAK", which can not be associated with the leakage at the bucket level. On the other hand, predicted similar cases using the hybrid approach can all be associated with bucket-related leakage problems. The second example is especially interesting in performance and the level of intricate patterns that state-of-the-art semantic similarity approaches can achieve. The text part of the query mentions that two bucket teeth are missing. As the problem is associated with the structural integrity of the bucket, we assigned the ontology label "Bucket (4.3)" to the search query. The resulting similar cases are quite interesting, presenting the complex maintenance records and the capability of the proposed approach. In this case, we have the same predicted results for both approaches (text-based and hybrid). These results include patterns such as "BROKEN $\times$ 2", "Replace 2", and "teeth 3 & 4 (two teeth)" against "teeth 2 off".

Similarly, in the engine-related queries, we see similar trends. For example, if a hose connected to the engine is blown or damaged, we used the ontology label "Twin Engine (3.2)" with the search query since it's a general problem at the engine level. If we compare the predicted similar cases, we find similar trends with some unrelated generic predictions from the text-based approach, while the hybrid approach proposes more relevant cases. We see the same pattern in other engine-related queries (4 and 5) with specific ontology labels related to induction & exhaust, and lubrication systems. In case 6, the query is related to high slew circuit temperature. Since the slew circuit is operated through the hydraulic system, this problem concerns the hydraulic cooling system, so we assigned the ontology label "Cooling System (3.1.1)" to the query. Results indicate that the proposed hybrid approach could predict 5 out of 5 relevant cases compared to only the text-based approach having few irrelevant cases. Instances 7 and 8 are related to the excavator boom that helps to move the excavator attachments. The first instance is related to the loose clamp problem; although it seems to be a small issue, it is critical due to the work environment and the machine's proper functioning. Loose clamps can cause hanging hoses and lines, and damaging other critical systems. Since the problem is associated with boom hoses and lines, we assigned the ontology label "Boom Hoses & Lines (3.1.8.1)" to the search query. The second case related to boom is regarding boom cylinder, so the assigned ontology label to the query is "Boom Cylinder (3.1.2.1)". Compared approaches performed similarly in both cases, while the hybrid approach performed slightly better.

Out of the total of 103 search query instances, there is only one instance (case 9) where the proposed hybrid approach has lower precision with 4 out of 5 true positive cases compared to 5 out of 5 using only a text-based approach. This particular query is about the fault in the air conditioning system with the ontology label "Air Conditioning System (3.2.9)". The query is related to the air conditioner not cooling, but the hybrid approach predicted a case about the heater, not heating which has the opposite meaning. Cases 10 and 11 are related to the driver's cabin and instrumentation; thus, relevant search queries are assigned ontology labels "Driver's Cab (1.6)" and "Instrumentation (2.5)" respectively. Case 10 concerns the driver's seat repair, which is critical for the operator's comfort. It can be observed in the presented results that the hybrid approach predicted 5 out of 5 true positive results (precision 1.0) compared to 3 out of 5 (precision 0.6) in the case of the text-based approach. Case 11 concerns the repair of a monitoring gauge; although it seems like a simple query, we only have 2 out of 5 true positive results (precision 0.4) compared to 5 out of 5 (precision 1.0) with the hybrid approach.

Case 12 is another case related to the engine concerning a warning light being on. Since the fault is not specified, the assigned query ontology label is "Twin Engine (3.2)", covering all engine-related issues. For this case, both approaches have results with similar patterns, where the text-based approach has a precision of 0.8 (4/5 true positive cases) compared to the hybrid approach with a precision of 1.0 (5/5 true positive cases). It is important to note that both approaches successfully translated "ENG" from search query text to "Engine" in the predicted results. For the problem related to low pump drive pressure (case 13), both approaches have similar results with slightly different indexing and precision of 1.0. The concerned ontology label for pump-related faults is "Hydraulic Pumps (3.1.7)". For case 14, which is also related to low pump pressure, we have slightly better precision with the hybrid approach than the text-based approach. The last case in Table 5 is related to the leakage at the pump where the text-based approach performs poorly with a precision of 0.4, half the precision compared to the hybrid approach (0.8). It is important to note that maintenance specialists predicted the case with the acronym PTO (power takeoff) since it is performed by the pump, which is why the case is relevant.

The proposed approach was initially developed for French maintenance reports from various milling machines at Fives CortX to provide maintenance decision support. For the French maintenance records, we observed a performance improvement of 22% where the text-based solution had an average precision of 0.66, and the hybrid approach had a precision of 0.88. The developed application is currently in production and used by maintenance operators to facilitate maintenance decision-making. Due to data privacy issues, only results on open-source excavator maintenance records from actual mining sites are presented in this study. In conclusion, precision-based and pattern-based analysis shows that the proposed hybrid approach for multi-modal data performs considerably better than the text-based approach using only one data source.

Conclusion and future work

We proposed a hybrid information retrieval methodology for complex industrial text in this study. The method enhances search performance using multi-modal data (text and machine ontology). A hybrid weighted similarity is proposed to incorporate insights from different data sources. A quantitative and qualitative analysis of the results for different queries shows that incorporating data from different sources results in more relevant cases in the returned results. Different queries based on the problems of the various excavator subsystems show a maximum performance improvement of 17% and an overall performance improvement of 8% for all systems combined. This study is a step forward in developing semantic search systems for underutilized sources, such as maintenance records using multi-modal data sources. Although research on maintenance decision support is ongoing, there is a need to apply cutting-edge techniques, such as Technical Language Processing (TLP) and multi-modal learning. Several research avenues can be taken into consideration as an extension of this study: (i) Using ontologies with extended node properties, such as type, function, cost, etc., (ii) Research on using generative Large Language models (LLMs) to facilitate semantic search in industrial text using local document analysis, and (iii) Development of interactive chat style, simple and efficient Human-machine Interface (HMI) for semantic search in industrial text.

Acknowledgments

This work has been supported by the EIPHI Graduate School (contract "ANR-17-EURE-0002").

References

[1] Fangyu Li et al. "Towards big data driven construction industry". In: *Journal of Industrial Information Integration* (2023), p. 100483.

[2] Adalberto Polenghi et al. "Ontology-augmented Prognostics and Health Management for shopfloor-synchronised joint maintenance and production management decisions". In: *Journal of Industrial Information Integration* 27 (2022), p. 100286.

[3] Fadi Shrouf, Joaquin Ordieres, and Giovanni Miragliotta. "Smart factories in Industry 4.0: A review of the concept and of energy management approached in production based on the Internet of Things paradigm". In: *2014 IEEE international conference on industrial engineering and engineering management*. IEEE. 2014, pp. 697–701.

[4] Alessio Botta et al. "Integration of cloud computing and internet of things: a survey". In: *Future generation computer systems* 56 (2016), pp. 684–700.

[5] Bhaskar Mondal. "Artificial intelligence: state of the art". In: *Recent Trends and Advances in Artificial Intelligence and Internet of Things* (2020), pp. 389–425.

[6] Sagar Jose, Khanh T P Nguyen, and Kamal Medjaher. "Multimodal Machine Learning in Prognostics and Health Management of Manufacturing Systems". In: *Artificial Intelligence for Smart Manufacturing: Methods, Applications, and Challenges*. Springer, 2023, pp. 167–197.

[7] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proceedings of NAACL-HLT*. Association for Computational Linguistics, 2019, pp. 4171–4186.

[8] Md Saidur Rahaman et al. "From ChatGPT-3 to GPT-4: a significant advancement in ai-driven NLP tools". In: *Journal of Engineering and Emerging Technologies* 2.1 (2023), pp. 1–11.

[9] Michael P Brundage et al. "Technical language processing: Unlocking maintenance knowledge". In: *Manufacturing Letters* 27 (2021), pp. 42–46.

[10] Guangyi Xiao et al. "Semantic input method of Chinese word senses for semantic document exchange in e-business". In: *Journal of Industrial Information Integration* 3 (2016), pp. 31–36.

[11] Po-Chien Hao and Bertrand MT Lin. "Text mining approach for bottleneck detection and analysis in printed circuit board manufacturing". In: *Computers & Industrial Engineering* 154 (2021), p. 107121.

[12] Euisuk Chung, Kyoungchan Park, and Pilsung Kang. "Fault classification and timing prediction based on shipment inspection data and maintenance reports for semiconductor manufacturing equipment". In: *Computers & Industrial Engineering* 176 (2023), p. 108972.

[13] Marcelo Romero et al. "A hybrid deep learning and ontology-driven approach to perform business process capability assessment". In: *Journal of Industrial Information Integration* 30 (2022), p. 100409.

[14] Adalberto Polenghi et al. "Knowledge reuse for ontology modelling in Maintenance and Industrial Asset Management". In: *Journal of Industrial Information Integration* 27 (2022), p. 100298.

[15] Syed Meesam Raza Naqvi et al. "Leveraging free-form text in maintenance logs through BERT transfer learning". In: *International conference on deep learning, artificial intelligence and robotics*. Springer. 2021, pp. 63–75.

[16] Michael Stewart et al. "MWO2KG and Echidna: Constructing and exploring knowledge graphs from maintenance data". In: *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* (2022), p. 1748006X221131128.

[17] Madhusudanan Navinchandran et al. "Discovering critical KPI factors from natural language in maintenance work orders". In: *Journal of Intelligent Manufacturing* (2021), pp. 1–19.

[18] Ajay Varma Nandyala et al. "Evaluating word representations in a technical language processing pipeline". In: *PHM Society European Conference*. Vol. 6. 2021, pp. 17–17.

[19] Akshay Peshave et al. "Evaluating Vector Representations of Short Text Data for Automating Recommendations of Maintenance Cases". In: *Annual Conference of the PHM Society*. Vol. 14. 2022, pp. 1–13.

[20] Fazel Ansari. "Cost-based text understanding to improve maintenance knowledge intelligence in manufacturing enterprises". In: *Computers & Industrial Engineering* 141 (2020), p. 106319.

[21] Anil Sharma and Suresh Kumar. "Machine learning and ontology-based novel semantic document indexing for information retrieval". In: *Computers & Industrial Engineering* 176 (2023), p. 108940.

[22] Qinglong Zhang and Yu-Bin Yang. "Rest: An efficient transformer for visual recognition". In: *Advances in neural information processing systems* 34 (2021), pp. 15475–15485.

[23] Alexey Dosovitskiy et al. "An image is worth 16x16 words: Transformers for image recognition at scale". In: *arXiv preprint arXiv:2010.11929* (2020).

[24] Yong Liu et al. "Non-stationary transformers: Exploring the stationarity in time series forecasting". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 9881–9893.

[25] George Zerveas et al. "A transformer-based framework for multivariate time series representation learning". In: *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*. 2021, pp. 2114–2124.

[26] Yuanyuan Zhao et al. "An ontology self-learning approach for CNC machine capability information integration and representation in cloud manufacturing". In: *Journal of Industrial Information Integration* 25 (2022), p. 100300.

[27] Qiangwei Bao et al. "A node2vec-based graph embedding approach for unified assembly process information modeling and workstep execution time prediction". In: *Computers & Industrial Engineering* 163 (2022), p. 107864.

[28] Syed Meesam Raza Naqvi et al. "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)*. IEEE. 2022, pp. 344–349.

[29] Syed Meesam Raza Naqvi et al. "Generating Semantic Matches Between Maintenance Work Orders for Diagnostic Decision Support". In: *Annual Conference of the PHM Society*. Vol. 14. 2022, pp. 1–9.

[30] Syed Meesam Raza Naqvi et al. "Human knowledge centered maintenance decision support in digital twin environment". In: *Journal of Manufacturing Systems* 65 (2022), pp. 528–537.

[31] *Excavator maintenance work orders*. `https://prognosticsdl.systemhealthlab.com/`. Accessed: 2023-06-01. 2021.

[32] Melinda R Hodkiewicz et al. "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*. Vol. 9. 2017, pp. 1–7.

[33] Melinda Hodkiewicz and Mark Tien-Wei Ho. "Cleaning historical maintenance work order data for reliability analysis". In: *Journal of Quality in Maintenance Engineering* 22.2 (2016), pp. 146–163.

[34] Thomas R Gruber. "A translation approach to portable ontology specifications". In: *Knowledge acquisition* 5.2 (1993), pp. 199–220.

[35] Maxat Kulmanov et al. "Semantic similarity and machine learning with ontologies". In: *Briefings in bioinformatics* 22.4 (2021), bbaa199.

[36] Nils Reimers and Iryna Gurevych. "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 2019, pp. 3982–3992.

[37] Fredrik Carlsson et al. "Semantic re-tuning with contrastive tension". In: *International Conference on Learning Representations*. 2020, pp. 1–21.

[38] Tianyu Gao, Xingcheng Yao, and Danqi Chen. "SimCSE: Simple Contrastive Learning of Sentence Embeddings". In: *Conference on Empirical Methods in Natural Language Processing, EMNLP*. Association for Computational Linguistics (ACL). 2021, pp. 6894–6910.

[39] Sverker Janson et al. "Semantic re-tuning with contrastive tension". In: *International Conference on Learning Representations, 2021*. 2021, pp. 1–21.

[40] Kexin Wang, Nils Reimers, and Iryna Gurevych. "TSDAE: Using Transformer-based Sequential Denoising Auto-Encoder for Unsupervised Sentence Embedding Learning". In: *Findings of the Association for Computational Linguistics: EMNLP*. 2021, pp. 671–688.

[41] Zhibiao Wu and Martha Palmer. "Verbs semantics and lexical selection". In: *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*. 1994, pp. 133–138.

[42] Thabet Slimani, B Ben Yaghlane, and Khaled Mellouli. "A new similarity measure based on edge counting". In: *International Journal of Computer and Information Engineering* 2.11 (2008), pp. 3851–3855.

[43] K MS, DKC Shet, and DU Acharya. "A New Similarity Measure For Taxonomy Based On Edge Counting". In: *nternational Journal of Web & Semantic Technology (IJWesT)* 3 (2012), p. 4.

[44] Abdeslem Dennai and Sidi Mohammed Benslimane. "A new measure of the calculation of semantic distance between ontology concepts". In: *International Journal of Information Technology and Computer Science (IJITCS)* 7.7 (2015), p. 48.

[45] Djamel Guessoum, Moeiz Miraoui, and Chakib Tadj. "A modification of wu and palmer semantic similarity measure". In: *The Tenth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*. 2016, pp. 42–46.