Labelling of Annotated Condition Monitoring Data Through Technical Language Processing

Karl Löwenmark[1], Cees Taal[2], Amit Vurgaft[2], Joakim Nivre[3], Marcus Liwicki[1] and Fredrik Sandin[1]

[1] *Embedded Intelligent Systems Laboratory (EISLAB), Luleå University of Technology,*
971 87 Luleå, Sweden, karl.lowenmark@ltu.se

[2] *SKF Research & Technology Development, Meidoornkade 14, 3992 AE Houten,*
P.O. Box 2350, 3430 DT Nieuwegein, The Netherlands

[3] *RISE Research Institutes of Sweden, Isafjordsgatan 22, 164 40 Kista, Sweden,*
P.O. Box 857, 501 15 Borås, Sweden

Abstract

We propose a novel approach, technical language labelling, to facilitate supervised intelligent fault diagnosis on unlabelled but annotated industry datasets using technical language processing. Condition monitoring (CM) is vital for high safety and resource efficiency in the green transition and digital transformation of the process industry. Computerised maintenance systems are required to facilitate CM scalability, and learning-based Intelligent Fault Diagnosis (IFD) methods are required to automate maintenance decisions and improve support for human analysts. A major challenge is the lack of labelled datasets from industry and the difficulty of transferring features from labelled lab datasets to unlabelled industry datasets. In this study, we investigate how the fault description annotations and maintenance work orders present in many CM datasets can be understood and used for IFD through Technical Language Processing, based on insights from recent advances in Natural Language Supervision joint pre-training of images and captions. We identify two distinct pipelines, one based on pre-training on large datasets, and one based on a human-centric approach and unsupervised clustering methods to transform annotations into labels, aided by insights from dimensionality reduction and visualisation techniques. Finally, we showcase one example of the small-data fault classification implementation on a CM industry dataset with a Sentence BERT model and conventional signal processing methods. Sets of features are used to overcome data imbalance and label misalignment, and we show that our model can separate sets of cable and sensor fault recordings from sets of bearing-related fault recordings with an F1-score of 92.6%. To our knowledge, this is the first system to create labels for CM data through pre-trained language models without requiring pre-defined taxonomies.

1. Introduction

In the digital and green transformation of the process industry to more sustainable production and operation, prognostics and health management of equipment is critical. Intelligent fault diagnosis (IFD) has been widely investigated to improve condition monitoring (CM) based maintenance (Manikandan & Duraivelu, 2021; T. Zhang et al., 2022). A major challenge in IFD implementation is that machines operate in different working conditions and processes, and industry data is almost exclusively unlabelled (Zhao et al., 2021). Furthermore, the data is imbalanced as faults are undesirable and critical fault development is largely prevented (T. Zhang et al., 2022). Therefore, labelled data is typically generated in lab environments where fault development can be induced, controlled, accelerated, and measured. However, there is a significant shift in features and noise levels between lab and industry environments, which has motivated research into transfer learning approaches (Lei et al., 2020; W. Li et al., 2022; T. Zhang et al., 2022), with the fundamental goal being to facilitate optimisation using unlabelled or weakly labelled industry data.

While strict labels are lacking, maintenance work order annotations are often present in condition monitoring datasets, with fault descriptions providing information similar to labels that sometimes include rich contextual and descriptive details, as well as uncertainties. Technical language processing (TLP) (Brundage et al., 2021) has been proposed to address the challenges present in low-resource technical language such as key technical terms not being present during pre-training of publicly available language models, being effectively out-of-vocabulary, while the data amount is insufficient to train a model specifically for technical language. Many approaches feature rule-based expert systems for language representation (Conte et al., 2021; Navinchandran et al., 2022), but more recently pre-trained language models have been integrated as well (Lowenmark et al., 2022; Cadavid et al., 2022). Can mapping annotations through TLP with the associated signals facilitate IFD models directly optimised on industry CM data?

Recent progress in machine learning combines visual and textual information through natural language supervision (NLS) (Radford et al., 2021; Kim et al., 2021; J. Li et al., 2022; Mu et al., 2022), and recent work in transferring NLS to technical language supervision (TLS) (Löwenmark et al., 2021) illustrates the potential of pre-training on hybrid datasets. However, obtaining large annotated industry datasets can be challenging due to intellectual property and privacy restrictions, and the scarce nature of annotations in comparison to signals. Therefore, we investigate a small-data solution for facilitating human-centric IFD based on unlabelled CM datasets with technical language annotations, and relate its strengths and weaknesses to technical and natural language supervision. We also visualise the properties of the signal and the text embedding spaces and show how they can offer insights about the data through joint embedding models.

The main contributions of this paper can be summarized as follows:

- We summarise work done on joint representations of images and captions (Section 2) and describe challenges and knowledge gaps in joint representations of CM signals and annotations or maintenance work orders (Section 3).
- We describe a human-centric framework to automate IFD on unlabelled but annotated CM data by unsupervised processing and clustering of annotations (Section 4). This method requires minimal human intervention compared to normal CM data analysis, and can be used to target specific fault classes through language-guided selection of clusters.
- We showcase how joint representations can be used to illustrate annotated CM datasets, implement a case-study method based on this framework by using sets of recording features to overcome weak supervision challenges, and present a cable and sensor fault detector using this method (Sections 5 and 6).

2. BACKGROUND

Early work in mapping language and signals focused mainly on a uni-directional pipeline, going for instance from image to caption through image captioning (Hossain et al., 2019), or from image + question to image and question-based answer through visual question answering (VQA) (Wu et al., 2017). In image captioning and VQA, datasets typically consist of images with associated text and tags written by human annotators, with captions for image captioning datasets and pairs of questions and answers for VQA (You et al., 2016). During training, images are encoded with a vision model, and decoded into text with a language model.

For image captioning, the decoding produces a caption that can be compared to human captions, and the error can then be backpropagated (Anderson et al., 2018). For VQA, the question is encoded alongside the image, and the output is compared to human-written question answers (Selvaraju et al., 2017). After the breakthroughs of GPT1 (Radford et al., 2018) and BERT (Devlin et al., 2019), vision language tasks benefited from improved language representations, with research moving towards computing joint image-text representations during training then decoding the joint representation to text during captioning or question answering testing (X. Li et al., 2020). This also facilitated self-supervision, such as masked language modeling as used in e.g. BERT, but conditioned on CNN image features (Sariyildiz et al., n.d.; Desai & Johnson, 2020), using either a pre-trained BERT model (Sariyildiz et al., n.d.) or by training the entire model from randomly initialised weights (Desai & Johnson, 2020).

However, training still relied on structured human captions and tags. Ideally, models would be trained in an unsupervised or self-supervised fashion from unstructured datasets without manually created captions, tags, or question-answer pairs. This would facilitate an immense upscaling of dataset size, and in turn model size, which per the general scaling laws of language models (Brown et al., 2020) should improve general understanding and representation of image-text properties. Furthermore, by correctly utilising the prompting capabilities of large language models and improved joint representations, unlabelled data could potentially even be used for supervision task, which will be described in the next section.

2.1. Natural Language Supervision

Natural language supervision (NLS) models require large datasets of unstructured captioned images from e.g. social media websites, resulting in millions of unique image-text pairs. With NLS, both image cap-

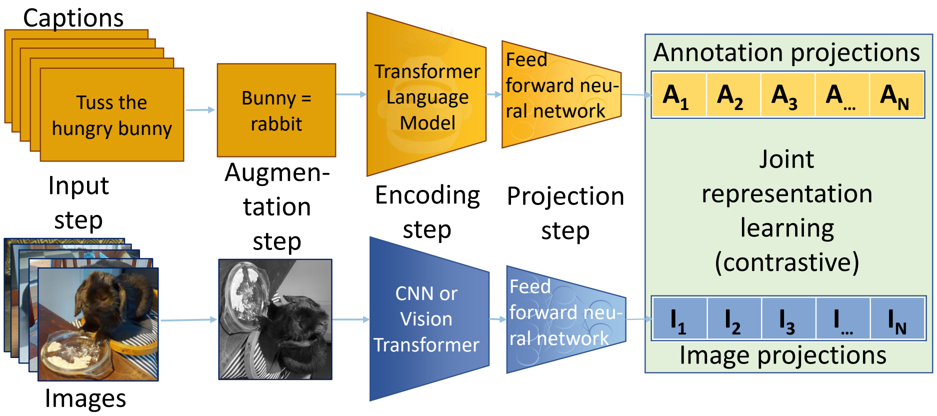
Captions
Tuss the hungry bunny
Input step
Bunny = rabbit
Augmentation step
Transformer Language Model
Encoding step
Feed forward neural network
Projection step
Annotation projections
A_1 A_2 A_3 $A_{...}$ A_N
Joint representation learning (contrastive)
I_1 I_2 I_3 $I_{...}$ I_N
Image projections
CNN or Vision Transformer
Feed forward neural network
Images

Figure 1. A general natural language supervision pipeline showcasing the five steps of natural language supervision. In the input step, unstructured pairs of images and captions are gathered from online sources, e.g. Wikipedia images with descriptions. In the augmentation step, multiple pairs based on the same image can be derived by altering either source, or certain images or captions can be removed based on conditions such as NSFW-filters. In the encoding step, features are computed through large encoders, sometimes pre-trained, which are then transformed to feature projections through feed-forward networks in the projection step. Finally, image and caption projections are mapped in the joint representation learning step, typically through contrastive learning.

tioning and visual question answering are possible, especially when fine-tuned for those tasks. The main difference compared to prior solutions to those tasks is the datasets used for pre-training of joint embedding spaces through noisy supervision (Jia et al., 2021), which no longer require reliable labels created by humans for supervision purposes. For example, ImageNet (Deng et al., 2009), one of the largest annotated datasets, consists of 1.35M images with 1 000 object categories, while LAION5B, one of the largest image-text-pair datasets available, consists of 5.85B samples. A large batch of images and captions are used in each training step, and the model is optimised to map correct pairs of images and captions to the same space in a high-dimensional projection space, for example by defining correct pairs as having a target dot product of 1 and all other pairs as having a target dot product of 0. Contrastive learning (Chen et al., 2020) is the most common technique used for this kind of pre-training, and was used in for instance the CLIP model (Radford et al., 2021), which popularised the term NLS and established its general framework.

To perform supervision tasks such as image classification, only general knowledge of what is represented in the images present in the test dataset is required. Classification is then achieved by choosing the image-caption pair with the highest dot product from a manually created query space representing the desired granularity of the classification, e.g. “a photo of an animal” vs “a photo of a vehicle”, or “a photo of a dog” vs “a photo of a cat”.

In total, we identify five major steps of an NLS model pipeline, as illustrated in Figure 1:

1. Input step – Gathering, selecting, and structuring input data.
2. Augmentation step – Overcoming challenges with data imbalance and label misalignment by modifying input images or text. This module can be added to the encoder and the joint representation steps as well.
3. Encoding step – Generating image and text features through large encoders, e.g. BERT or GPT3 and ResNet or the Vision Transformer (ViT). The encoders can be frozen or fine-tuned during optimisation of joint embedding spaces.
4. Projection step – Projecting high-dimensional features to a lower-dimensional space, typically through a feed-forward neural network.
5. Joint representation step – Comparing projections of text and image through, for instance, dot products of a large batch of captioned images.

The input step is one major improvement compared to previous vision language models, where data now can be gathered from any source of captioned images, without requiring human labelling efforts. This allows for considerably more training data and scaling to new data sources without requiring expensive labelling practices, which was shown by the ALIGN model to improve performance despite the added noise from poor samples (Jia et al., 2021).

The augmentation step, if used, is most commonly associated with the input data, where for instance text can be filtered based on hate speech detection and images based on a not-safe-for-work (NSFW) filter. For instance, LAION5B filtered its input samples with CLIP, featuring detection scores for watermarks, NSFW, and toxic content. However, data augmentation can also be introduced inside the encoding step, as in BLIP (J. Li et al., 2022), where synthetic captions are generated and noisy text-image pairs filtered through models fine-tuned on manually annotated datasets, after which the filtered dataset is used to pre-train a new model.

The encoding step is mainly based on established image and text encoder architectures, either with pre-trained weights or trained from scratch with the joint representation learning step. If weights are downloaded from a previous model, they can be either frozen or further trained during the optimisation of the model. CLIP uses two different image encoders, one based on ResNet-50 (K. He et al., 2016) and one based on the ViT (Dosovitskiy et al., 2020), and the GPT-2 text transformer from (Radford et al., 2019). Neither text nor image encoders are pre-trained in CLIP, though other models, such as Oscar (X. Li et al., 2020) and VinVL (P. Zhang et al., 2021), rely on pre-trained object detectors to extract region-of-interest features from images before learning joint representations.

Finally, the joint representation learning step typically consists of contrastive learning in a large batch of image-text pairs. Contrastive learning pushes positive samples closer in the projection space, while negative pairing projections are pushed away. This process can be assisted by conditioning positive and negative pairs on various properties, such as the covariance of encoding features used in CLOOB (Fürst et al., 2021). Alternatively, the contrastive step can be treated as a bidirectional loss from text to image and image to text, as was used in (Y. Zhang et al., 2020), which used annotated medical x-rays for contrastive pre-training. Another similar approach is to maximise the similarity between text and image segment tokens, as done in FILIP (Yao et al., 2021).

3. CHALLENGES AND KNOWLEDGE GAPS

While there are similarities between sets of images with captions, and CM signals with annotations and maintenance work orders, there are also important differences to consider and knowledge gaps to overcome. The transition from common natural language to technical language in the upper part of Figure 1, from images to CM signals in the blue part, and data imbalances and fault properties in the joint representation learning green part, all face challenges, namely:

1. Technical language encoding challenges
   (a) OOV: The language used in annotations contains technical terms and abbreviations that are semantically critical but out-of-vocabulary of conventional NLP language models.
   (b) Interoperability: Annotations and work orders have varying, non-standard formats, including unstructured free-text, semi-structured text in several fields, and tabular structured data. Varying vocabularies are used even in similar industries.
2. Signal encoding challenges
   (a) Feature complexity: Features contained in condition monitoring signals differ considerably from images. Most notably, some fault properties, such as severity, can depend on the evolution of features over multiple signals and operational data such as machine rotational speed.
   (b) Physics-based features: Additional analysis is often performed on signals, such as transforming from the time domain to the frequency domain, and analysing condition indicators based on characteristic frequencies, which is crucial for human analysis and annotation writing, compared to the straightforward learning-based encoding of images.
3. Joint Representation Learning challenges
   (a) Imbalance: Annotations associated with signals are considerably fewer than captions associated with images, and most signals in CM datasets are unannotated, resulting in imbalanced data (T. Zhang et al., 2022) and incomplete supervision (Zhou, 2017).
   (b) Time range: An annotation has one time stamp, while signals containing a fault has a time range. Therefore it is not trivial how the association between signals and annotations should be mapped.
   (c) Inexactness: Annotations are typically connected to an asset or sub-asset, which might contain multiple sensors or data types, with multiple recordings per day per sensor. Thus each annotation is connected to a set of features, without a clear definition of which recordings should be included in the set or which features indicate a fault. This is typically referred to as *inexact* supervision in weak supervision terminology (Zhou, 2017).

Based on these challenges we identify two viable paths for using annotations as supervision signals for associated CM signals: *technical language supervision* (TLS) and *technical language labelling* (TLL). Both paths rely on TLP integrated with a pre-trained language model to address the technical language

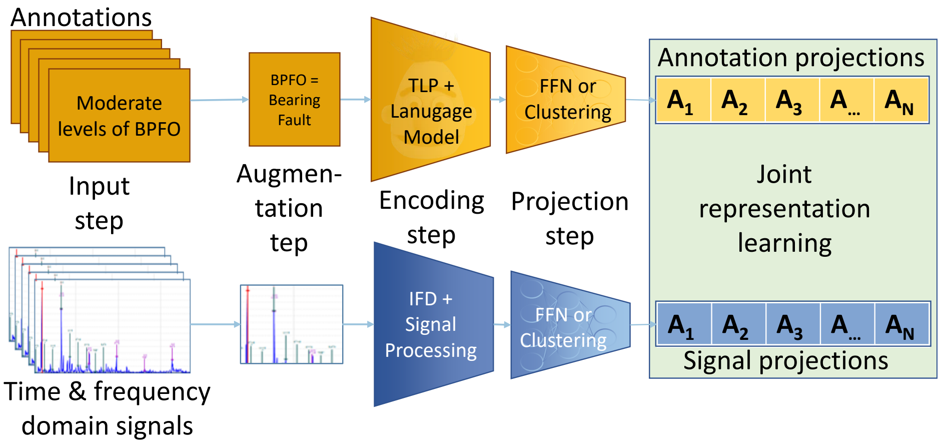

Figure 2. A technical language pipeline inspired by the NLS pipeline shown in Figure 1, with possibilities for both technical language supervision through joint representation learning (Löwenmark et al., 2021) and technical language labelling (this work). Image-caption pairs are now replaced by signal-annotation pairs. Augmentation is now based on human insight into industry specific properties of signals and annotations, and encoding models have significantly less data to train on, necessitating technical language processing and pre-trained language models in the language encoder, and signal processing, physics-based kinematics and IFD insights in the signal encoder. There is no significant change in the FFN projector, but clustering can also be employed when data is scarce but human insight is rich. Finally, the joint representation step can now be contrastive or mappings between clusters, based on the projection step.

encoding challenges. In TLS, signal encodings are learned through contrastive learning in the joint representation step, relying on scaling up data and model size for improved performance. TLL instead relies on signal processing methods for the signal encoding and unsupervised methods such as clustering, and integrating human knowledge for the joint representation step, to facilitate language-based learning without requiring large data. These paths go beyond present IFD state-of-the-art by integrating supervision signals in the form of unstructured text annotations, and are further detailed in the following sections.

4. THEORY

4.1. Technical Language Supervision

TLS was introduced in 2021 (Löwenmark et al., 2021), with an emphasis on IFD and a case-study to showcase the practical usefulness of the approach. Figure 2 showcases a pipeline both for TLS and TLL (introduced next section) based on the NLS pipeline shown in Figure 1. CM signals and technical language annotations are mapped to a joint embedding space in a similar approach as in natural language supervision, but with different types of input data, augmentation, and encoding steps.

In cases where annotations are scarce, it is both difficult to pre-train or fine-tune a language model to accurately represent technical language, and to find sufficiently large samples of data to train a TLS model. Therefore, it is desirable to make use of language model developments to represent knowledge stored in annotations as supervision signals in an unsupervised manner, without requiring large datasets. This is beneficial for the transfer and development of IFD models on industry data, and can also serve to develop pre-trained signal encoders for TLS, similarly to using a pre-trained vision transformer (Dosovitskiy et al., 2020) to represent images in natural language supervision.

4.2. Technical Language Labelling

TLL substitutes the data- and computationally-demanding learned projection in the joint representation step in Figure 2 with unsupervised methods, such as dimensionality reduction techniques and clustering algorithms, to project data from feature space to projection space. The joint representation learning is treated unidirectionally, mapping signal features to annotation clusters, annotation features to signal clusters, annotation/signal clusters to annotation/signal clusters, or joint cluster features, depending on which projections that are replaced with unsupervised methods. These clusters can be visualised and analysed through dimensionality reduction techniques such as PCA or t-SNE (Liu et al., 2018), which offer a tool for explorative data analysis of how TLP implementations affect language encodings. The process used in this study to predict the fault class based on signal features and annotation clusters can be described as follows:

1. Input step – Extract annotations and associate them with the related assets in the machinery. Choose which sensor recordings to include with regard to parameters such as the recording date compared to the annotation date. Associate

each annotation with all recordings in that time window, creating one annotation-signal pair per recording.

2. Augmentation step – Resample recordings to balance the fault classes by creating sets of features, as described below. Use visualisation insights to augment the class cluster distribution.
3. Encoding step – Embed the annotations using a language model and substitute technical terms with natural language descriptions (Lowenmark et al., 2022). Encode the signals using conventional signal processing techniques.
4. Projection step – Project the embeddings to clusters using unsupervised clustering techniques such as k-Means. Visualise the clusters using t-SNE. Cluster either the embedding space or the visualisation space.
5. Joint representation learning and classification step – Project the signal features to the clusters using a classifier such as a support vector machine. Evaluate classifier performance by predicting cluster labels based on unseen signal-annotation pairs.

The replacement of the feed-forward projection heads with an indirect mapping through clustering drastically reduces the complexity of the learning step, but also the potential for pre-training, as a significant amount of information is lost going from features to projection. Thus, pre-training or fine-tuning the encoders through joint representation learning becomes difficult, necessitating the use of a pre-trained language model augmented with TLP, and a signal encoder based either on signal processing methods, known physical properties such as characteristic frequencies, or a pre-trained IFD model. However, the same challenges that were listed regarding TLS are also present for language-based labelling, though with the possibility to apply a human-centric approach rather than data scaling to amend them.

4.3. Dealing with Imbalanced Data

CM industry datasets are typically highly imbalanced with regard to fault class distribution and healthy/unhealthy data distribution (Akhbardeh et al., 2021; Usuga-Cadavid et al., 2021; T. Zhang et al., 2022). Therefore, methods required to deal with this challenge are added after the feature extraction set.

Three common methods for dealing with imbalanced data are oversampling, undersampling, and synthetic sampling (H. He & Garcia, 2009). In oversampling, data and labels from the minority classes are randomly resampled to increase the number of samples until all classes are balanced, which results in a much larger dataset and prevents information loss, but also increases the risk for overfitting. In the case of undersampling, data is removed from the majority classes, resulting in a smaller dataset and loss of information, but with less risk for overfitting due to duplicate oversamples. Synthetic sampling attempts to generate new samples from minority classes to prevent both information loss and overfitting. A common algorithm is SMOTE (Chawla et al., 2002), which iteratively generates synthetic samples based on the five nearest neighbours of an existing sample. Synthetically upsampling time-series based signals is not trivial due to such signals typically being shift-variant. Therefore, algorithms such as SMOTE should (ideally) be applied only to shift-invariant signals, e.g. directly on low dimensional shift-invariant features computed from shift-variant signals, which also reduces the computation time.

However, we have not identified work done in data augmentation based on joint properties of annotations and signals. Thus, there are potential gains for both IFD and TLS by implementing for instance bootstrapping of noisy data with artificial annotations similar to BLIP (J. Li et al., 2022), where an NLS model is augmented with synthetic captions. These captions are generated from uncaptioned web images with the image-grounded text decoder and filtered with the image-grounded text encoder, then used to expand the pre-training dataset for the NLS model. A similar approach for annotated industry data could thus be used to annotate unannnotated signals that share features with the annotated samples.

Alternatively, contrastive learning (Chen et al., 2020), or contrastive self-supervision (Grill et al., 2020), can be used purely on the signal level. Contrastive learning on images augments the data by using image properties such as that a picture of a dog, even if rotated, cropped, or colour shifted, etc., remains a dog, or that a picture of a another dog should share more latent features compared to a picture of e.g. a car. In these scenarios, contrastive models learn joint representations by passing two or more images with known properties, such as ”related” and ”unrelated”. By augmenting the data, the possible scope of the dataset is drastically increased, which typically leads to better downstream performance. Adopting the same approach for industry sensor data can thus alleviate the issue of annotation scarcity by increasing the number of ”samples” associated with one annotation, or by facilitating IFD pre-training of image representations prior to joint embedding learning in the NLS model.

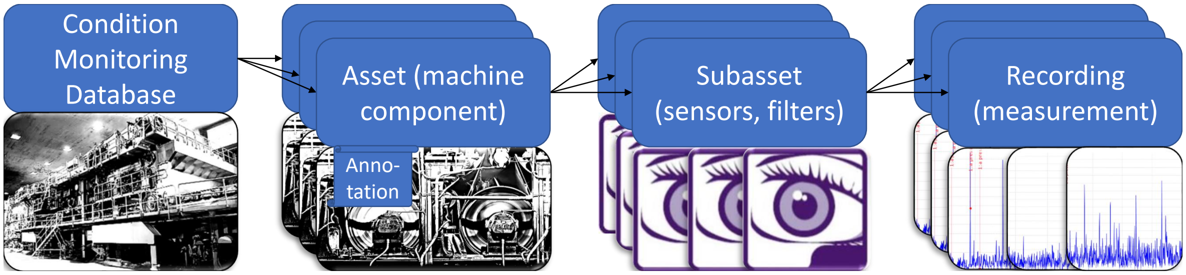
Condition Monitoring Database
Asset (machine component)
Anno-tation
Subasset (sensors, filters)
Recording (measurement)

Figure 3. Structure of a CM dataset.

5. METHOD

5.1. Case Study Dataset

To illustrate the properties of annotated CM datasets and showcase one example of language-based fault classification, we implement a case study model using data from two large Swedish paper mills with annotated condition monitoring signals from assets such as dryers, rollers and gearboxes etc. An overview of the structure of the CM dataset is shown in Figure 3.

The dataset consists of multiple assets (machine parts), where detected faults are annotated at the asset level. An asset consists of multiple subassets made up of different signal types or sensors. Each subasset in turn consists of multiple recordings – data such as vibration measurements and associated spectra – and metadata such as RPM, measured for several seconds and then stored multiple times per day, every day. Consequently, the annotation, at the asset level, is associated with multiple subassets, each associated with scores of recordings.

In our dataset, we have 2385 annotations over a span of five years, of which 319 are within the last 6 months and thus have associated signals still stored in the dataset. We choose a time span of ten days before and after the annotation time stamp, based on experiments to maximise data size while maintaining performance. Thus, a total of 38597 associated recordings are present in a span of ten days before and after the annotation date.

The faults described can be grouped as cable and sensor faults, bearing related faults such as ball-pass frequency outer race (BPFO), ball-pass frequency inner race (BPFI), mechanical looseness or imbalance (Randall & Antoni, 2011), or other miscellaneous faults or comments such as gearbox faults or comments on new sensor types. Many annotations detail that maintenance has been done, though the annotation date is not necessarily the same as the maintenance date. As data size grows, the number of unique annotations will also increase, necessitating a pretrained language model to ensure system scalability.

5.2. Explorative Data Analysis

We use a Swedish SentenceBERT language model (Rekathati, 2021) with technical language substitution (Lowenmark et al., 2022) as the TLP method to encode annotations from a Swedish process industry dataset, constituting the input step and the encoding step of the pipeline shown in Figure 2. The annotation embeddings were clustered using *k*-Means on the embeddings, and visualised using t-SNE projection, with the most common technical words of the embedding clusters serving as labels, constituting the projection step for the language section, shown in Figure 4. Vibration spectra associated with the annotations, defined as being within ten days of the annotation, were also visualised using t-SNE projection on the log of the spectra, and labelled based on which cluster their associated annotation belonged to.

Human knowledge was integrated by automatically grouping the clusters based on whether key-words indicating sensor and cable faults, or bearing-related faults, were present, shown in Figure 6, with the ungrouped figure shown in Figure 5. This illustrates one possible path through Figure 2, using clustering to go from features to low-dimensional representations, and clustered annotations as joint representations.

5.3. Fault Classification

Then, the annotation embedding was concatenated to each spectrum before projection, projected using t-SNE, then labelled and regrouped based on the annotation cluster to illustrate the joint properties, shown in Figure 8, with the ungrouped clusters shown in Figure 7.

We also implemented an SVM RBF classifier predicting annotation embedding clusters based on spectra and time-series features. The features were computed by common signal processing methods, namely average, max, standard deviation, kurtosis, skew, and peak-to-peak, of the vibration time series, spectra, and time-series transformed with a sliding t-scan window and through Mallat-Zhong wavelet transform (Mallat et al., 1992). While some methods make sense mainly in time or frequency space, we regard-

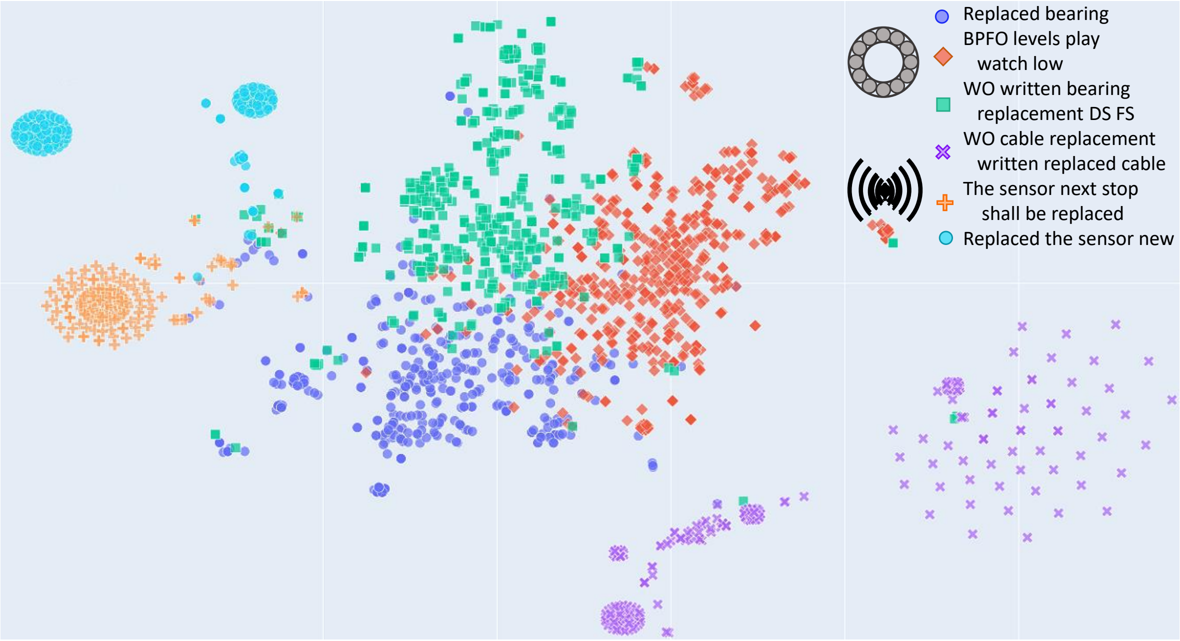

Figure 4. Two dimensional t-SNE transformation of annotation sentence embeddings, classified with *k*-Means and labelled with the most common technical words per cluster.

less saw improved performance by using all methods on all input signals.

We used a data augmentation method described below to create sets of features for each annotation cluster prediction, thus overcoming data imbalance and label misalignment issues. Finally, we tested this performance both on the clusters naturally formed through *k*-Means, and the joined clusters through the automated human-centric process, shown in Figures 9 and 10 respectively.

5.4. Overcoming Joint Representation Challenges

The challenges listed in Section 3 are overcome in three different parts of the TLL method. First, the technical language challenges 1a and 1b and overcome by using technical language substitution (Lowenmark et al., 2022) with a pre-trained language model, which can represent language with varying degrees of structure, including technical words covered by the substitution. Second, signal processing features from both the time and frequency domains are included to overcome the challenges posed in 2b. Challenge 2a can be addressed with sequential models such as recurrent neural networks or transformers, but to maintain the low complexity of the framework, we instead opt to address this challenge alongside the joint representation challenges, described below. Third, all three joint representation challenges are overcome by creating sets of samples for each annotation through random sampling of recordings associated with each annotation cluster.

In this process, one training sample for a particular cluster class is generated by concatenating features from $N_{samples}$ randomly chosen samples, where $N_{samples}$ determines the size of the new training sample. Thus, the input to the classifier becomes a set of features. and data can be balanced by sampling the same number of unique combinations of sets from all classes.

This results in three major benefits:

1. The likelihood of associating samples with appropriate features with the correct label increases.

2. Data imbalance challenges can be overcome without requiring over-, under-, or synthetic sampling.

3. For larger set sizes N_{sets} an extremely high number of unique sets can be generated, up to $\binom{N_{samples}}{N_{sets}}$, where $N_{samples}$ is the number of samples in the target class.

Another similar approach is to stack the N most recent recordings for each recording type per asset, but that does not allow for the scaling and balancing of dataset size that the method above does, and is much less likely to overcome the label misalignment, especially when two or more sensors are mapped to the same asset.

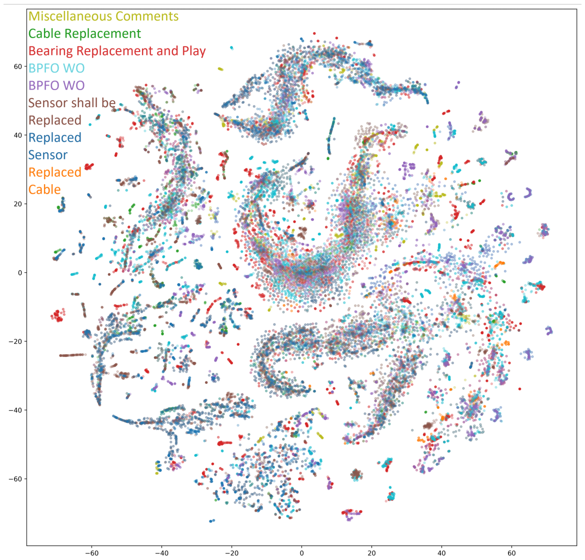

Figure 5. Two dimensional t-SNE transformation of CM spectra, classified with annotation clusters as seen in Figure 4 and labelled based on the most common technical words in the clusters. This showcases that without human insight there is limited consistency between annotation clusters and signal clusters.

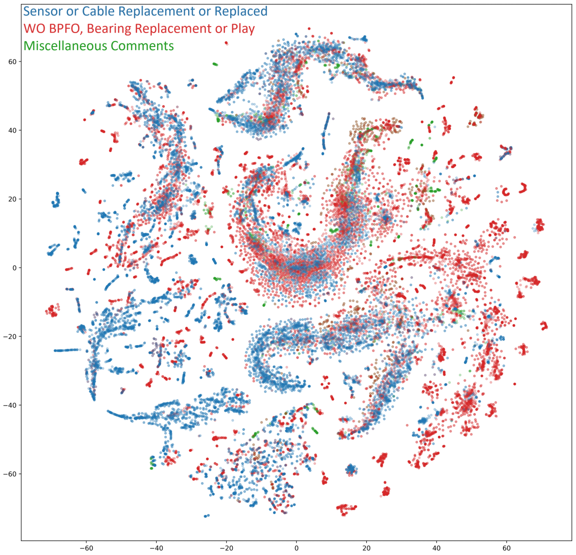

Figure 6. The same transformation as in Figure 5, but automatically relabelled per Table 1. Recordings now form visually distinct clusters going from cable and sensor faults to the left and bearing faults to the right, with clusters in the middle and upper sections consisting mostly of recordings associated with annotations detailing that MWOs have been completed.

6. Results and Discussion

6.1. Explorative Data Analysis

Figure 4 illustrates data exploration of the language space through a t-SNE projection of annotation embeddings, clustered with the *k*-Means algorithm. Each element in the Figure is one annotation embedding, and the "labels" of the clusters are the most common non-stopwords. The clusters show a clear separation between cable and sensor faults both from each other and from the large group of bearing-related annotations. Sensor faults are further split into annotations detailing the need for sensor replacement (orange crosses) and annotations describing that sensor replacement has been done (teal dots). However, cable replacements are not split with only six clusters, thus containing both 'WO [work order] written cable replacement' and 'cable replaced' annotations. The green, red and blue clusters mainly describe bearing faults, but there is not a distinct separation as with cable and sensor faults. The blue dots contain annotations mostly describing maintenance actions taken on bearings, while the green squares consist mostly of annotations describing that work orders have been written. However, the red rhomboids contain both 'BPFO' and 'play'(mechanical looseness)-related annotations, and both the red and green annotations also contain other bearing-related faults that were inseparable from each other. Thus, while the embeddings for various bearing faults are different, they are not sufficiently separable to facilitate ideal separation. Therefore, improved TLP methods or language models are required to further augment this step.

6.1.1. Joint Signal–Annotation Data Analysis

Figures 5, 6, 7, and 8 show further data exploration of properties between signals and annotations, and potential for joint mappings, with the goal to investigate if the data supports predicting the language clusters from the corresponding signals.

Figure 5 illustrates a standard data analysis approach, projecting log of spectra with t-SNE to a two-dimensional space. The labels are based on annotation clusters, as shown in Figure 4, but computed with eight embedding clusters to increase the resolution of the "labels". The figure shows a few large groupings of spectra, with multiple smaller groups distributed over the t-SNE projection space, likely maintaining only local similarities per t-SNE's objective.

However, it is difficult to distinguish clear patterns between annotation and signal representations from this level alone. Therefore, since some clusters have similar or even identical cluster labels, human-centric knowledge can be integrated to group labels based on whether they describe similar fault properties, in particular cable/sensor-related faults and bearing-related faults. This is illustrated in Figure 6 based on the reclustering shown in Table 1, where some patterns start to emerge; cable and sensor faults tend to the west and south-west of the Figure, while bearing-related faults occupy the east. The clusters in the

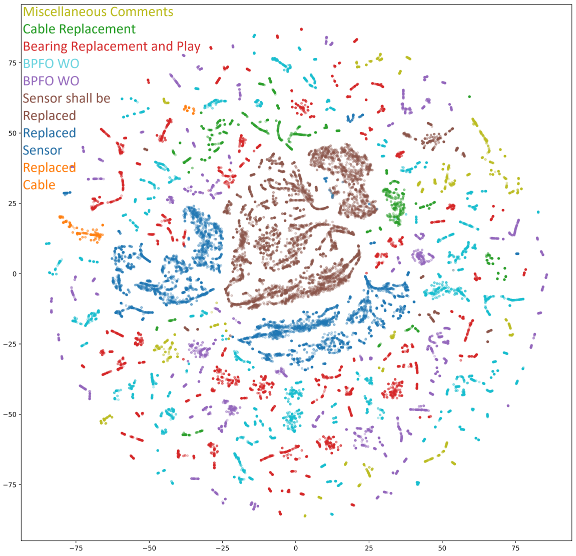

Figure 7. Two dimensional t-SNE transformation of CM spectra concatenated with annotation embeddings, classified with annotation clusters as seen in Figure 4 and labelled based on the content of the most common technical words.

Semantic meaning of annotation	Human-centric relabelling
Miscellaneous Comments	‘Miscellaneous Comments’
Cable Replacement	‘Sensor’, ‘Cable Replacement’, or ‘Replaced’
Bearing Replacement and Play	‘WO BPFO’, ‘Bearing Replacement’, or ‘Play’
BPFO WO	‘WO BPFO’, ‘Bearing Replacement’, or ‘Play’
BPFO WO	‘WO BPFO’, ‘Bearing Replacement’, or ‘Play’
Sensor shall be Replaced	‘Sensor’, ‘Cable Replacement’, or ‘Replaced’
Replaced Sensor	‘Sensor’, ‘ ‘Cable Replacement’, or ‘Replaced’
Replaced Cable	‘Sensor’, ‘Cable Replacement’, or ‘Replaced’

Table 1. The most common technical words, computed based on a word frequency per cluster threshold then reformulated to more clearly show the faults described in each cluster, and human-centric-based label.

center, north and north-west are all mainly made up of annotation indicating that maintenance has been performed, in particular the large centermost cluster, which explains why there are signals of both groups of faults.

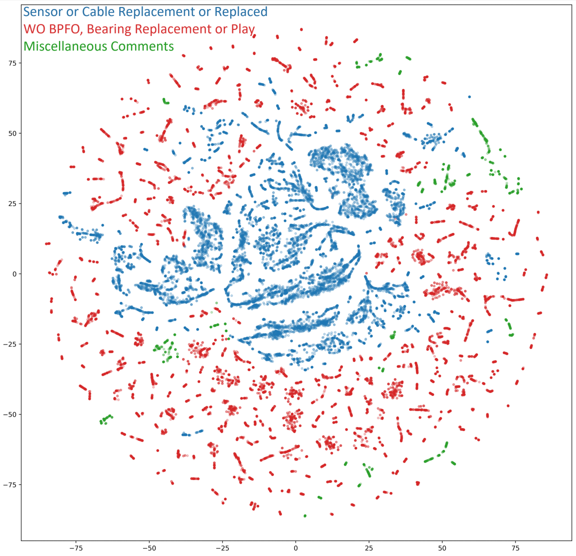

Figure 8. The same transformation as in Figure 7, but automatically relabelled per Table 1. Sensor and cable faults now show a clear pattern based in the center of the projection, with bearing faults, miscellaneous comments and a few sensor cable faults occupying the periphery.

The data is still difficult to fully interpret however, which is why we also explored joining signals with annotation embeddings before t-SNE projection, shown in Figure 7. There is now no overlap between the “labels” with well-separated local clusters, though the colours are distributed with no clear coherent global patterns. Since the annotation embeddings used for the “labels” are now also part of the input, it is expected that local separability would improve, though the clear local separability certainly points to a correlation between the annotation and signal properties.

Finally, the human-centric process is repeated for the joint visualisation, shown in Figure 8. A surprisingly clear pattern now emerges, where sensor and cable faults now occupy the center of the joint representation representations, with various bearing-related faults occupying the periphery. In particular, sensor/cable faults occupy the same space in the center of the figure despite being globally separated in the embedding visualisation of Figure 4. These results indicate that human-centric augmentation of data exploration and automated embedding clusters results in more distinguishable patterns for data interpretation, which supports the merits of building a classifier based on signal-embedding pairs.

The data exploration is meant to illustrate data properties, but could in itself be used for classification by distilling a network to mimic the projection step, concatenating embeddings from common fault description annotations to unannotated spectra, and using for instance k-nearest neighbours to evaluate which

Cluster nr	Most common technical words	Relabelling nr
1	WO cable replacement written replaced cable	1
2	WO written bearing replacement DS FS	2
3	BPFO levels play watch low	2
4	The sensor next stop shall be_replaced	1
5	Replaced the_sensor new	1
6	Replaced bearing	2

Table 2. Annotations cluster number; the most common technical words, computed based on a word frequency per cluster threshold; and new label assigned based on meaning of the most common technical words.

cluster the new spectra should belong to. However, this would require extracting and adding a wide array of unannotated points, and likely also adaptation to industry-specific needs. Therefore, we do not investigate that possibility further in this study, but we note that adding unannotated points is a logical next step.

Clustering can be done in either the embedding space, the visualisation space, or in any intermediate dimension. Using dimensionality reduction techniques first reduces the complexity of the clustering and makes the results easier to interpret visually, but using the embedding space directly will arguably lead to more accurate assessment of how the language model represents the annotations. However, we saw little difference in performance using either method, and a manual inspection showed slightly more coherent results by clustering in higher dimensions such as the embedding space.

6.2. Classifier

The confusion matrix shown in Figure 9 shows the performance of the classifier described in 5.3, trained on sets with

$$N_{sets} = 50, N_{samples} = 100000$$

created from signals extracted from five to one days before the annotation date, and tested on data from the day leading up to the annotation. This split had superior results compared to using data from up to ten before the annotation date, maintaining a causal[1] 80/20 train/test split, as well as compared to including data from after the annotation date. The labels are based on the clusters from Figure 4.

The challenge of features overlapping between classes is clearly visible in the confusion matrix, as “WO cable replacement” (1), “sensor shall be replaced” (4), and “replaced the sensor” (5), all see an overlap in predictions, which also is true for the “WO written bearing replacement” (2), “BPFO” (3), and “replaced bearing” (6) clusters. These misclassifications do not indicate poor performance, as the features indicating BFPO might also warrant a bearing replacement, and an annotation describing that a bearing has just been replaced likely associates with signals with severe bearing fault features, e.g. from BPFO, leading up to that replacement. Thus, when joining cluster labels through human-centric reclustering, shown in Table 2, the performance is significantly improved, and in particular the number of false alarms for cable and sensor fault sets decreases drastically.

Forming sets of features increases the level of abstraction of the supervision task, but is necessary to overcome the challenge of inconsistent features in the signal space. Without the sets, the model is unable to perform well as far too many data points are feature-less, and thus many annotation-signal pairs have different annotation labels associated with the same feature-less feature-space.

However, some misclassifications remain even in the joined confusion matrix, which likely arises due to the errors between cable replacement/replaced and replaced bearing in Figure 9. These could be due to some annotations being written days after the replacement date, resulting in healthy features being included in both sets. Thus, some testing samples will contain mostly healthy data, which is difficult to separate between classes that should be based on unhealthy data. To overcome this with a large-data approach, a transformer- or RNN-based model which attends to a wide span of signals to predict class belonging is likely the best solution. Obtaining sufficient amounts of annotated data would require widespread industry collaboration with annotated datasets from multiple companies. An alternate small-data approach is to have a dynamic scope for data extraction based on the fault description, either learned from language model representations or by a TLP-based expert system based on human knowledge of typical dataset properties.

6.3. Future Research

The most important future research area for language-based supervision on industry data is to integrate unannotated data. Unannotated, presumably healthy, data makes up a significant majority of all data in CM datasets and in day-to-day work in process industries. Evaluating unannotated data is even more

[1] Split data so that training data always is older than test data. For example, if the chosen time span is signals from five days before the annotation date up to and including the annotation date, then split the data so that the last day before the annotation is the testing data. If data from five days prior to five days after is used, split the data so that signals from four and five days after the annotation are used for testing.

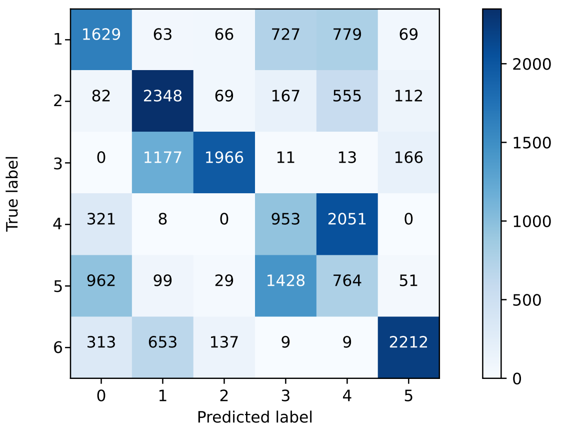

Figure 9. Multiclass cluster label prediction based on sets of recordings, with an F1-score of 50.7%.

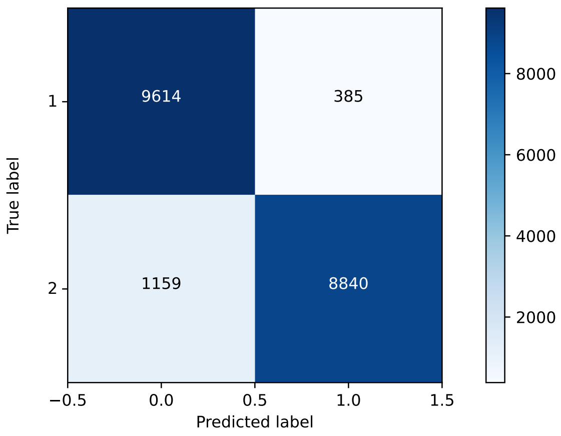

Figure 10. Sorted cluster label prediction based on sets of recordings with and F1-score of 92.6%.

difficult than noisy annotated data, but is critical to ensure that models developed can be integrated in industry pipelines without producing time-consuming false positive alarms or feedback for analysts. Analysts can also be integrated to provide real ground truth labels for a small dataset, which would allow for more accurate model evaluations.

In the vision-language pre-training domain, further research can be done in noisy supervision, which has been shown to converge despite noise in both text and image input data (Jia et al., 2021). However, no methods tailored specifically for industry challenges have yet been developed, which would greatly benefit any IFD scheme based on language data. For example, rule-based expert systems from TLP research could be further expanded to include the time-horison of common faults, so that a “BPFO detected low levels keep watch” annotation indicates that recordings from before and after the annotation date can be used, while “Sensor replaced OK” indicates that only recordings prior to the annotation date contain fault features. This can then be further expanded through active learning by evaluating feature levels with regard to the expected time horison from a model trained on a noisy datset, to essentially bootstrap the dataset similarly to BLIP (J. Li et al., 2022).

Augmentation based on text or image properties could potentially be introduced in the joint representation learning step to maximise agreement between not only image-text pairs, but also between samples of similar meaning, similar to data augmentation of labelled images in contrastive learning (Chen et al., 2020). Contrastive learning works well given large batch sizes, large datasets, and large models trained for a long time. However, some researchers argue that the contrastive learning space has extreme variance outside of the boundaries of encountered samples, and that other joint representation learning methods based on energy functions might be less computationally demanding and also generalise better (Sobal et al., 2022).

Many objectives in TLP research are related to event extraction (Tong et al., 2020; Lai et al., 2020; Wang et al., 2021; Lou et al., 2021), which detects text anomalies and thus helps with text structuring by automatically tagging what happened and when it happened. Event extraction has also been investigated with datasets from aviation industry (Akhbardeh et al., 2020; Madeira et al., 2021), thus associated with the more production-oriented data that has been used in TLP, wherein solutions for data imbalance have also been suggested by oversampling worse performing data distributions (Akhbardeh et al., 2021).

7. CONCLUSION

We introduced a human-centric method to automate IFD on unlabelled but annotated CM data by unsupervised processing and clustering of annotations. We illustrated how sensor and annotation representations correlate in a CM dataset, and show a test case where signal features were mapped to annotation embedding clusters. By adding human insight, we achieved an F1-score of 92.6% when detecting sets of cable and sensor fault vs bearing fault features.

Surveying the literature on TLP, we show that there is potential for new research in the merging of CM signals with associated annotations and MWOs for the implementation of IFD models on industry datasets. Thus, we describe the general framework of NLS, the mapping between images and natural language captions, and describe methods and challenges to translate findings in this field to the technical domain. In particular, TLS, based on joint pre-training on large datasets, and TLL, are identified as viable paths to facilitate optimisation of machine learning algorithms on unlabelled but annotated CM datasets.

Acknowledgement

This work is supported by the Strategic innovation program Process industrial IT and Automation (PiIA), a joint investment of Vinnova, Formas and the Swedish Energy Agency, reference number 2019-02533. The analysis of the results was done with help from Håkan Sirkka, a condition monitoring analyst with experience of the industry data used. We thank the members of the project reference group including Per-Erik Larsson, Kjell Lundberg, Håkan Sirkka and Peter Wikström, for valuable inputs. Karl Löwenmark thanks Tom Breuer for helpful discussions and a coding session on clustering and visualisation.

References

Akhbardeh, F., Alm, C. O., Zampieri, M., & Desell, T. (2021, August). Handling Extreme Class Imbalance in Technical Logbook Datasets. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (pp. 4034–4045). Online: Association for Computational Linguistics.

Akhbardeh, F., Desell, T., & Zampieri, M. (2020, December). MaintNet: A Collaborative Open-Source Library for Predictive Maintenance Language Resources. In *Proceedings of the 28th International Conference on Computational Linguistics: System Demonstrations* (pp. 7–11). Barcelona, Spain (Online): International Committee on Computational Linguistics (ICCL).

Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., & Zhang, L. (2018). Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (p. 6077-6086).

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., … Amodei, D. (2020). *Language Models are Few-Shot Learners.*

Brundage, M. P., Sexton, T., Hodkiewicz, M., Dima, A., & Lukens, S. (2021). Technical language processing: Unlocking maintenance knowledge. *Manufacturing Letters*, *27*, 42-46.

Cadavid, J. P. U., Lamouri, S., Grabot, B., & Fortin, A. (2022). Using deep learning to value free-form text data for predictive maintenance. *International Journal of Production Research*, *60*, 4548 - 4575.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002, jun). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*, 321–357.

Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020). A simple framework for contrastive learning of visual representations [Conference paper]. In (Vol. PartF168147-3, p. 1575 – 1585). (Cited by: 1448)

Conte, A., Bolland, C., Phan, L., Brundage, M., & Sexton, T. (2021). The Impact of Data Quality on Maintenance Work Order Analysis: A Case Study in HVAC Work Durations..

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition* (p. 248-255).

Desai, K., & Johnson, J. (2020). *VirTex: Learning Visual Representations from Textual Annotations.* arXiv.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.*

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., … Houlsby, N. (2020). *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale.* arXiv.

Fürst, A., Rumetshofer, E., Lehner, J., Tran, V., Tang, F., Ramsauer, H., … Hochreiter, S. (2021). *CLOOB: Modern Hopfield Networks with InfoLOOB Outperform CLIP.* arXiv.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., … Valko, M. (2020). *Bootstrap your own latent: A new approach to self-supervised Learning.* arXiv.

He, H., & Garcia, E. A. (2009). Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering*, *21*(9), 1263-1284.

He, K., Zhang, X., Ren, S., & Sun, J. (2016, June). Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

Hossain, M. Z., Sohel, F., Shiratuddin, M. F., & Laga, H. (2019, feb). A Comprehensive Survey of Deep Learning for Image Captioning. *ACM Comput. Surv.*, *51*(6).

Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., … Duerig, T. (2021). Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision.

Kim, W., Son, B., & Kim, I. (2021). ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision. In *ICML.*

Lai, V. D., Nguyen, T. N., & Nguyen, T. H. (2020, November). Event Detection: Gate Diversity and Syntactic Importance Scores for Graph Convolution Neural Networks. In *Proceedings of the 2020*

Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 5405–5411). Online: Association for Computational Linguistics.

Lei, Y., Yang, B., Jiang, X., Jia, F., Li, N., & Nandi, A. K. (2020). Applications of machine learning to machine fault diagnosis: A review and roadmap. *Mechanical Systems and Signal Processing*, *138*, 106587.

Li, J., Li, D., Xiong, C., & Hoi, S. (2022). Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning* (pp. 12888–12900).

Li, W., Huang, R., Li, J., Liao, Y., Chen, Z., He, G., ... Gryllias, K. (2022). A perspective survey on deep transfer learning for fault diagnosis in industrial scenarios: Theories, applications and challenges. *Mechanical Systems and Signal Processing*, *167*, 108487.

Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., ... Gao, J. (2020, August). Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks. In *ECCV*.

Liu, S., Bremer, P.-T., Thiagarajan, J. J., Srikumar, V., Wang, B., Livnat, Y., & Pascucci, V. (2018). Visual Exploration of Semantic Relationships in Neural Word Embeddings. *IEEE Transactions on Visualization and Computer Graphics*, *24*(1), 553-562.

Lou, D., Liao, Z., Deng, S., Zhang, N., & Chen, H. (2021, August). MLBiNet: A Cross-Sentence Collective Event Detection Network. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)"* (pp. 4829–4839). Online: Association for Computational Linguistics.

Lowenmark, K., Taal, C., Nivre, J., Liwicki, M., & Sandin, F. (2022). Processing of Condition Monitoring Annotations with BERT and Technical Language Substitution: A Case Study. In *PHM Society European Conference* (Vol. 7, pp. 306–314).

Löwenmark, K., Taal, C., Schnabel, S., Liwicki, M., & Sandin, F. (2021). Technical Language Supervision for Intelligent Fault Diagnosis in Process Industry. *arXiv e-prints*, arXiv:2112.07356.

Madeira, T., Melicio, R., Valerio, D., & Santos, L. (2021). Machine Learning and Natural Language Processing for Prediction of Human Factors in Aviation Incident Reports. *Aerospace*, *8*(2).

Mallat, S., Zhong, S., et al. (1992). Characterization of signals from multiscale edges. *IEEE Transactions on pattern analysis and machine intelligence*, *14*(7), 710–732.

Manikandan, S., & Duraivelu, K. (2021). Fault diagnosis of various rotating equipment using machine learning approaches – a review. *Proceedings of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering*, *235*(2), 629-642.

Mu, N., Kirillov, A., Wagner, D., & Xie, S. (2022). SLIP: Self-supervision Meets Language-Image Pre-training. In S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, & T. Hassner (Eds.), *Computer Vision – ECCV 2022* (pp. 529–544). Cham: Springer Nature Switzerland.

Navinchandran, M., Sharp, M., Brundage, M. P., & Sexton, T. (2022). Discovering critical KPI factors from natural language in maintenance work orders. *Journal of Intelligent Manufacturing*, *33*, 1859-1877.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., ... Sutskever, I. (2021). Learning Transferable Visual Models From Natural Language Supervision.

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. (2018). Improving language understanding by generative pre-training. OpenAI.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, *1*(8), 9.

Randall, R. B., & Antoni, J. (2011). Rolling element bearing diagnostics—A tutorial. *Mechanical Systems and Signal Processing*, *25*(2), 485-520.

Rekathati, F. (2021). *The KBLab Blog: Introducing a Swedish Sentence Transformer.*

Sariyildiz, M. B., Perez, J., & Larlus, D. (n.d.). Learning Visual Representations with Caption Annotations. In A. Vedaldi, H. Bischof, T. Brox, & J.-M. Frahm (Eds.), *Computer vision – eccv 2020.*

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *2017 IEEE International Conference on Computer Vision (ICCV)* (p. 618-626).

Sobal, V., S, J., Jalagam, S., Carion, N., Cho, K., & LeCun, Y. (2022). *Joint Embedding Predictive Architectures Focus on Slow Features.* arXiv.

Tong, M., Xu, B., Wang, S., Cao, Y., Hou, L., Li, J., & Xie, J. (2020, July). Improving Event Detection via Open-domain Trigger Knowledge. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 5887–5897). Online: Association for Computational Linguistics.

Usuga-Cadavid, J. P., Grabot, B., Lamouri, S., & Fortin, A. (2021). Artificial Data Generation with Language Models for Imbalanced Classification in Maintenance. *Service Oriented, Holonic and Multi-Agent Manufacturing Systems for Industry of the Future*.

Wang, Z., Wang, X., Han, X., Lin, Y., Hou, L., Liu, Z., ... Zhou, J. (2021, August). CLEVE: Contrastive Pre-training for Event Extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (pp. 6283–6297). Online: Association for Computational Linguistics.

Wu, Q., Teney, D., Wang, P., Shen, C., Dick, A., & van den Hengel, A. (2017). Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding*, *163*, 21-40. (Language in Vision)

Yao, L., Huang, R., Hou, L., Lu, G., Niu, M., Xu, H., ... Xu, C. (2021). FILIP: Fine-grained Interactive Language-Image Pre-Training. *ArXiv*, *abs/2111.07783*.

You, Q., Jin, H., Wang, Z., Fang, C., & Luo, J. (2016). Image Captioning with Semantic Attention. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (p. 4651-4659).

Zhang, P., Li, X., Hu, X., Yang, J., Zhang, L., Wang, L., ... Gao, J. (2021). VinVL: Revisiting Visual Representations in Vision-Language Models. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 5575-5584.

Zhang, T., Chen, J., Li, F., Zhang, K., Lv, H., He, S., & Xu, E. (2022). Intelligent fault diagnosis of machines with small & imbalanced data: A state-of-the-art review and possible extensions. *ISA Transactions*, *119*, 152-171.

Zhang, Y., Jiang, H., Miura, Y., Manning, C. D., & Langlotz, C. P. (2020). *Contrastive Learning of Medical Visual Representations from Paired Images and Text.* arXiv.

Zhao, Z., Zhang, Q., Yu, X., Sun, C., Wang, S., Yan, R., & Chen, X. (2021). Applications of Unsupervised Deep Transfer Learning to Intelligent Fault Diagnosis: A Survey and Comparative Study. *IEEE Transactions on Instrumentation and Measurement*, *70*, 1-28.

Zhou, Z.-H. (2017, 08). A brief introduction to weakly supervised learning. *National Science Review*, *5*(1), 44-53.