

MaintNet: A Collaborative Open-Source Library for Predictive Maintenance Language Resources

Farhad Akhbardeh, Travis Desell, Marcos Zampieri

Rochester Institute of Technology, United States

{fa3019, tjdvse, mazgla}@rit.edu

Abstract

Maintenance record logbooks are an emerging text type in NLP. They typically consist of free text documents with many domain specific technical terms, abbreviations, as well as non-standard spelling and grammar, which poses difficulties to NLP pipelines trained on standard corpora. Analyzing and annotating such documents is of particular importance in the development of predictive maintenance systems, which aim to provide operational efficiencies, prevent accidents and save lives. In order to facilitate and encourage research in this area, we have developed MaintNet, a collaborative open-source library of technical and domain-specific language datasets. MaintNet provides novel logbook data from the aviation, automotive, and facilities domains along with tools to aid in their (pre-)processing and clustering. Furthermore, it provides a way to encourage discussion on and sharing of new datasets and tools for logbook data analysis.

1 Introduction

With the rapid development of information technologies, engineering systems are generating ever increasing amounts of data that are used by industries to improve products. Maintenance records are one of these types of data. They typically consist of event logbooks which are collected in many domains such as aviation, transportation, and healthcare (Tanguy et al., 2016; Altuncu et al., 2018). The analysis of maintenance records is particularly important in the development of predictive maintenance systems, which can be used to improve efficiencies as well as prevent accidents and reduce maintenance costs (Jarry et al., 2018).

Maintenance record datasets generally contain free text fields describing issues (or problems) and actions, as can be seen in the instances presented in Table 1.

ID	Issue/Problem	Date	Action
111552	R/H FWD UPPER BAFL SEAL NEEDS TO BE RESECURED	7/2/2012	INSTALLED POP RIVET TO RESECURE R/H FWD BAF SEEAL.
111563	CAP SCREW MISSING, L/H ENG #4 BAFL	7/3/2012	INSTALLED NEW SCREW. CHKD ENG
111574	CYL #1 BAFFLE CRACKED AT SCREW SUPPORT & FWD BAFL BELOWE #1	7/2/2012	FABRICATED PATCHES OF LIKE MATERIAL & RIVETED IAW CESSN
111585	#3 FWD PUSH ROD TUBE GSK LEAKING @ EGNINE	7/2/2012	REMOVED & REPLACED #3 FWD PUSH ROD TUBE SEALS. LEAK CHE

Table 1: Four examples from Maintnet’s aviation dataset.

Standard NLP tools, however, are typically trained on standard contemporary corpora (e.g. newspaper texts) and struggle when dealing with the domain specific terminology, abbreviations, and non-standard spelling which are abundant in maintenance records. To help encourage further study in this area, we present MaintNet¹, a collaborative, open-source library for technical language resources with a special focus on predictive maintenance data.

The four main contributions of this paper are the following:

¹Available at: <https://people.rit.edu/fa3019/MaintNet/>

1. The development of MaintNet, a user-friendly web-based platform that serves as a repository hosting a variety of resources related to predictive maintenance and technical logbook data.
2. The creation of several important language resources for technical language and predictive maintenance such as abbreviation lists, morphosyntactic information lists, and term banks for the aviation, automotive and facilities domains. All these resources as well as raw data from these domains are made freely available to the research community via MaintNet.
3. The development of several novel Python packages for (pre-)processing technical language including stop word removal, stemmers, lemmatizers, POS tagging, clustering methods, and more.
4. A collaborative environment in which the community can contribute with data and resources and interact with developers and other members of the community.

2 MaintNet Features

2.1 Language Resources

To the best of our knowledge, there are no freely available tools and libraries developed to process such data which makes MaintNet unique. MaintNet currently features datasets from the aviation, automotive, and facilities domains (see Table 2), and it will be expanded with the collaboration of the interested members of the NLP community working on similar topics.

Domain	Dataset	Inst.	Tokens	Source
Automotive	Car	617	4,443	Connecticut Open Data
Aviation	Maintenance	6,169	76,866	University of North Dakota Aviation Program
	Accident	5,268	162,533	Open Data by Socrata
	Safety	25,558	345,979	Federal Aviation Administration
Facility Maintenance	Operations	87,276	2,469,003	Baltimore City Maryland Preventive Maintenance

Table 2: The number of instances and tokens in each dataset/domain.

Predictive maintenance datasets are hard to obtain due to the sensitive information they contain. Therefore, we work closely with the data providers to ensure that any confidential and sensitive information in the dataset remains anonymous. In addition to the datasets, MaintNet further provides the user with domain specific abbreviation dictionaries, morphosyntactic annotation, and term banks. The abbreviation dictionaries contains abbreviated validated by domain experts. The morphosyntactic annotation contains the part of speech (POS) tag, compound, lemma, and word stems. Finally, the domain term banks contain the collected list of terms that are used in each domain along with a sample of usage in the corpus.

2.2 Pre-processing and Tools

One of the bottlenecks of automatically processing logbooks for predictive maintenance is that most of these datasets are not annotated with the reason of maintenance or the category of the issue. We implemented several (pre-)processing steps to clean and extract as much information from logbooks as possible. The pipeline is shown in Figure 1. The process starts with text normalization, including lowercasing, stop word and punctuation removal, and treating special characters with NLTK’s (Bird et al., 2009) regular expression library, followed by stemming (Snowball Stemmer), lemmatization (WordNet (Miller, 1992)), and tokenization (NLTK tokenizer). With use of the collected morphosyntactic information, POS annotation is carried out with the NLTK POS tagger. Term frequency-inverse document frequency (TF-IDF) is obtained using the *gensim tfidf model* (Rehurek and Sojka, 2010). Our analysis of the logbooks found that many of the misspellings and abbreviations lead to incorrect or non-existent dictionary look ups. To overcome this issue, we explored various state-of-the-art spellcheckers including Enchant², Pyspellchecker³, Symspellpy⁴, and Autocorrect⁵.

²<https://www.abisource.com/projects/enchant/>

³<https://github.com/barrust/pyspellchecker>

⁴<https://github.com/wolfgarbe/SymSpell>

⁵<https://github.com/fsondej/autocorrect>

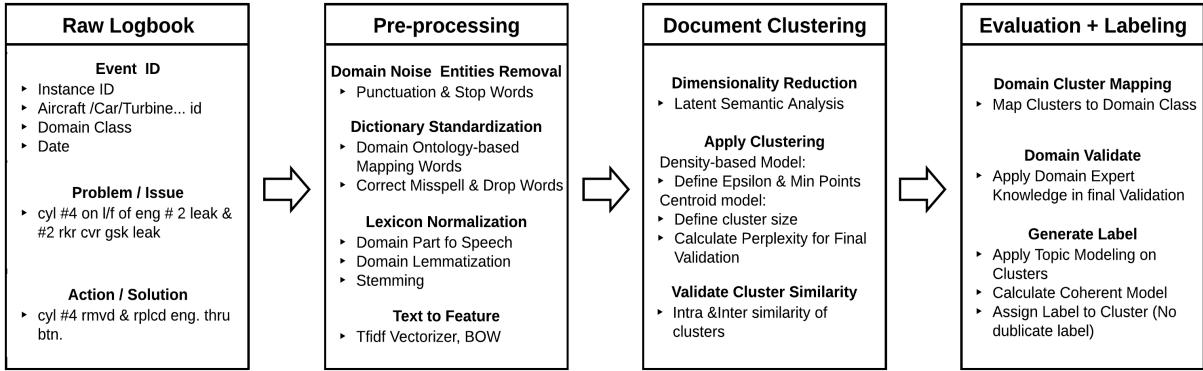


Figure 1: A pipeline of pre-processing and information extraction of maintenance dataset in MaintNet.

Given the inaccuracy of existing techniques, we developed methods of correcting syntactic errors, typos, and abbreviated words using a Levenshtein distance-based algorithm (Aggarwal and Zhai, 2012). This method uses a dictionary of domain specific words and maps the various possible misspelled words into the correct format by selecting the most similar word in the dictionary. The Levenshtein algorithm was chosen over other distance metrics (*e.g.*, Euclidian, Cosine) as it allows us to control the minimum number of string edits. The results of our method compared to other spellchecking techniques in a sub set of the aviation dataset is presented in Table 3.

Total Number of Documents	200
Total Unique Tokens	106
Total Non-standard	97
Success Rate (%)	
Enchant	59
PySpellchecker	12
Autocorrect	45
Levenshtein	97

Table 3: Results of the spelling correction and abbreviation expansion methods.

WordNet was used to lemmatize the document, however it requires defining a POS tagger parameter which we want to lemmatize (the wordNet default is “noun”). As the maintenance instances typically consist of verb, noun, adverb and adjective words that define a problem, action and occurrence, by using “verb” as the POS parameter, there is an issue of mapping important noun words such as “left” (*e.g.* left engine) to “leave” or “ground” to “grind”. To resolve this issue, as we discussed in 2.1, we created an exception list using developed morphosyntactic information for the WordNet lemmatizer to ignore mapping words which could be multiple parts of speech. We convert the terms and words into a numerical representation using libraries such as *tfidfvectorizer* (ElSahar et al., 2017) resulting in a large matrix of document terms (DT).

MaintNet also features implementations of popular clustering algorithms applied to logbook data that are made freely available to the research community. The motivation behind this is that most of this data is not annotated, which requires a domain expert to group instances into categories. Clustering techniques were used to help in this process. We use truncated singular value decomposition (SVD) (ElSahar et al., 2017) known as latent semantic analysis (LSA), to perform a linear dimensionality reduction. We chose truncated SVD (LSA) over principal component analysis (PCA)(ElSahar et al., 2017) in our system, due to the fact LSA can directly be applied to our *tfidf* DT matrix and it focuses on document and term relationships where PCA focuses on a term covariance matrix (eigendecomposition of the correlation). We experimented with different 4 clustering techniques: k-means (Jain, 2010), Density-Based Spatial Clustering of Applications with Noise (DBSCAN) (Ester et al., 1996), Latent Dirichlet Analysis (LDA) (Vorontsov et al., 2015), and hierarchical clustering (Aggarwal and Zhai, 2012). For comparison of the results, the silhouette and inertia (Fraley and Raftery, 1998) metrics were used to determine the number of clusters for k-means (both provided similar results), and perplexity (Fraley and Raftery, 1998) and

coherence (Vorontsov et al., 2015) scores were used for LDA. DBSCAN and hierarchical clustering do not require a predetermined number of clusters.

For evaluation, we used a standard measurement of cluster cohesion including high intra-cluster similarity and low inter-cluster similarity. We chose 3 different similarity algorithms including Levenshtein, Jaro, and cosine (Fraley and Raftery, 1998) to calculate intra- and inter-cluster similarity. The cosine similarity metric is commonly used and is independent of the length of document, while Jaro is more flexible by providing a rating of matching strings. We collected human annotated instances by a domain expert to serve as our gold standard, and these are provided on MaintNet to encourage research into improving unsupervised clustering of maintenance logbooks.

2.3 Community Participation

Finally, MaintNet provides various webpages for users to communicate with each other and the project developers; as well as upload data to share with the community (see Figure 2). We hope this will help further facilitate discussion and research in this under explored area.

The screenshot displays two side-by-side web pages from the MaintNet platform. Both pages have a header with the MaintNet logo and navigation links: Home, Documentation, Dataset, Download, and Forum.

Discussion Forum: This panel shows a form for posting a new forum topic. It includes fields for Dataset, Annotation, First name, Last name, Username, User Affiliation (XXX University, Area of Expertise NLP), and a file upload field for choosing a file. There is also a Description text area and a checkbox for Agree to terms and conditions, followed by a Submit button.

Posted Forum: This panel shows a post made by a user (@User) on January 2, 2020 at 12:00 PM. The post asks if pre-processing technical text data with general NLP packages is losing some important information. Below the post, there is a "Leave a Comment:" input field, a "Submit" button, and a "USER XXX Comment" section. The comment section contains an "Answer:" field with a note about technical data containing domain-specific words and the potential loss of important information. It also lists a GitHub page for further reading. Below this, there is another "USER Comment" section with a note about developing an algorithm for pre-processing technical data.

Figure 2: A screenshot of MaintNet’s discussion webpages.

3 Conclusions and Future Work

In this paper we presented MaintNet, a collaborative open-source library for predictive maintenance language resources. MaintNet provides raw technical logbook data as well as several language resources such as abbreviation lists, morphosyntactic information lists, and termbanks from the aviation, automotive and facilities domains. Tools developed in Python are also made available for pre-processing, such as spell checking, POS tagging, and document clustering. In addition to these tools, the collaborative aspects of MaintNet should be emphasized. We welcome the community to contribute with new datasets that can be processed using the tools available at MaintNet, or share new and improved tools developed with MaintNet’s open source data.

MaintNet is also expanding as current work involves processing data from additional domains such as healthcare and power systems (*e.g.*, wind turbines). These datasets will be made available on MaintNet in upcoming months. We also aim to collect and release datasets and tools for languages other than English in the near future.

References

- Charu C. Aggarwal and ChengXiang Zhai. 2012. A survey of text clustering algorithms. In *Mining Text Data*.
- M. Tarik Altuncu, Erik Mayer, Sophia N. Yaliraki, and Mauricio Barahona. 2018. From text to topics in healthcare records: An unsupervised graph partitioning methodology. *ArXiv*, abs/1807.02599.
- Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python*. O'Reilly.
- Hady ElSahar, Elena Demidova, Simon Gottschalk, Christophe Gravier, and Frédérique Laforest. 2017. Unsupervised open relation extraction. *ArXiv*, abs/1801.07174.
- Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*.
- Chris Fraley and Adrian E. Raftery. 1998. How many clusters? which clustering method? answers via model-based cluster analysis. *Comput. J.*, 41:578–588.
- Anil Kumar Jain. 2010. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31:651–666.
- Gabriel Jarry, Daniel Delahaye, Florence Nicol, and Eric Feron. 2018. Aircraft atypical approach detection using functional principal component analysis. In *SID*.
- George A. Miller. 1992. Wordnet: A lexical database for english. *Commun. ACM*, 38:39–41.
- Radim Rehurek and Petr Sojka. 2010. Software framework for topic modelling with large corpora. In *LREC*.
- Ludovic Tanguy, Nikola Tulechki, Assaf Urieli, Eric Hermann, and Céline Raynal. 2016. Natural language processing for aviation safety reports: From classification to interactive analysis. *Computers in Industry*, 78:80–95.
- Konstantin Vorontsov, Oleksandr Frei, Murat Apishev, Peter Romov, and Marina Dudarenko. 2015. Bigartm: Open source library for regularized multimodal topic modeling of large collections. In *AIST*.