A method for automatic detection of acronyms in texts and building a dataset for acronym disambiguation

Sasan Azimi
University of Tehran
Tehran, Iran
azimi.s@ut.ac.ir

Hadi Veisi
University of Tehran
Tehran, Iran
h.veisi@ut.ac.ir

Reyhaneh Amouie
University of Tehran
Tehran, Iran
reyhaneh.amouie@ut.ac.ir

Abstract
Nowadays, there is an increasing tendency for using acronyms in technical texts, which has led to ambiguous acronyms with different possible expansions. Diversity of expansions of a single acronym makes recognizing its expansion a challenging task. Replacing acronyms with incorrect expansions will lead to problems in text mining procedures, namely text normalization, summarization, machine translation, and tech-mining. Tech-mining involves exploring and analyzing technical texts to recognize the relations between technologies. This paper is aimed at proposing a method for building a dataset that meets the requirements for training acronym disambiguation models in technical texts. In this paper, challenges in automatic acronym disambiguation are presented. We have proposed a method for building the dataset and the accuracy of the acronym disambiguation model is 86%.

Keywords: Acronym disambiguation, Tech-mining, Text Mining, Natural Language Processing

I. INTRODUCTION

Technical texts are plentiful of acronyms and predicting their meanings forms an important subject in the automatic extraction of information from texts [1][2]. In natural language processing, acronym disambiguation can be seen as a particular case of word sense disambiguation. Humans use contextual information, topics, and background knowledge to predict the meaning of ambiguous words [3]. The intrinsic ability of humans to disambiguate and recognize the meaning of words makes comprehension possible.

Acronym disambiguation for enterprises is challenging for different reasons. First, acronyms may be highly ambiguous because an acronym used in the enterprise could have multiple meanings. Second, there is usually no universal knowledge to predict the expansion of a given acronym. Finally, the system should be capable of working for any enterprise [4].

As an instance, DMS has 55 expansions in IT field out of 177 general expansions. Some of them are listed in table 1.

Table 1- Sample acronyms and ambiguity in their expansions

DMS (55 IT expansions out of 171 general expansions)
Data Management System
Database migration service
Digital Marketing Solutions
Docket Management System
Dynamic Message Sign
SRS (55 IT expansions out of 171 general expansions)
Software Requirements Specification
Sample Registration System
Sequence Retrieval System
Service Request System
MMS (46 IT expansions out of 150 general expansions)
Multimedia Messaging Service
Microsoft Management Summit
Mobile Marketing Solutions
Maximum Message Size
Memory Management System

In order to analyze, understand, and exploit knowledge from technical texts, text mining, and natural language processing algorithms are deployed [5]. Also, artificial intelligence algorithms are used to model, understand, learn, and extract knowledge from scientific and technical texts. Recently, there has been a significant development in the applications mentioned above [6]. The input to the intelligent text processing systems is a massive number of texts in which ambiguities has a negative impact on their performance. Since the texts are of high volume, the disambiguation process must be done automatically.

Various techniques of text mining are exploited to perform tech-mining tasks. In tech-mining, the relation among technologies is analyzed and extracted [7]. The ambiguity of acronyms represents a challenge to the process of recognizing the relations among technologies [7]. The input to the systems in these tasks is of the order of millions of words. Hence, disambiguation must be performed automatically. Nowadays,

978-1-7281-5350-6/19/$31.00 ©2019 IEEE

thanks to the advances in artificial neural networks, more complex tasks can be assigned to machines [8].

Every day, 37 new human-edited acronyms are defined and there are 379,918 acronyms with 4,766,899 expansions available on the Internet[1]. On average, there are 12.5 expansions for each acronym on this site [9]. The figure shows the scale of the challenge and emphasizes the importance of the issue.

Today, there are methods for retrieving all the possible expansions of an acronym from Wikipedia and AcronymFinder. These expansions were used to collect the right expansions for the acronym by using a deep learning technique called Doc2Vec [10].

In order to train expansion recognition models, it is necessary to have a tagged set of data. In this paper, we define acronyms as the initials of a multi-token phrase. In other words, other types of abbreviations, e.g., MathML, which stands for Mathematical Markup Language, are not of interest.

This paper aims to propose a method for building and releasing the appropriate dataset. The idea of the method is to automatically mine the texts in the dataset, extract acronym-expansion pairs based on special characters and patterns within the text, and store them in the database. In section 2, the proposed methods as well as its challenges are presented. Section 3 and 4 provide a conclusion and suggestions for further works.

II. Background and Related Works

According to our knowledge of the previous works, methods proposed for acronym disambiguation are not numerous. Disambiguation is done using adjacent words. In [11], a method is presented by using surrounding based embedding feature (SBE). In [12], an unsupervised approach for period character disambiguation in German clinical narratives is presented which is evaluated for the task of abbreviation detection. This task used two approaches: statistical approach and dictionary-based approach. In a similar task, word embedding method was exploited to create vectors of contexts over a set of 19,954 examples of 4,365 ambiguous acronyms. The results show that the proposed approach has performed better than majority classifier [13].

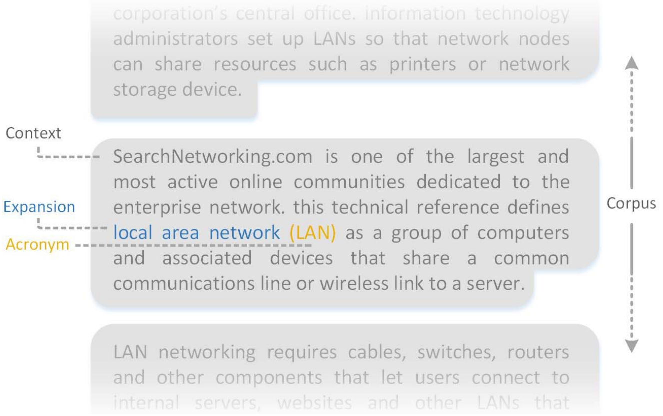
III. The Proposed Method

A. *The structure of the proposed method*

Similar to the natural ability of humans to predict the expansion of an acronym based on its surrounding text, the prepared texts are used to train models and recognize the correct expansions of acronyms. To this aim, it is necessary to tag the acronyms that are present in the corpus to train neural networks [8].

Therefore, there is a need for a dataset with three elements as follows: acronyms, expansions, and the surrounding text of the acronyms (figure 1).

The proposed method has two requirements. The first one is correctness and validity of the training data. Correctness of the training data implies the correct association of acronyms with their corresponding expansions. This confirms that the acronym belongs to the expansion. The second requirement is detecting the maximum number of acronyms in the corpus. In order to meet the first requirement, automatic tagging with an accuracy of ~100% was deployed. Manual checking did not reveal any inaccuracies.



corporation's central office. information technology administrators set up LANs so that network nodes can share resources such as printers or network storage device.

SearchNetworking.com is one of the largest and most active online communities dedicated to the enterprise network. this technical reference defines local area network (LAN) as a group of computers and associated devices that share a common communications line or wireless link to a server.

LAN networking requires cables, switches, routers and other components that let users connect to internal servers, websites and other LANs that

Figure 1- Sample of corpus, context, expansion and the acronym

The second requirement is met by defining and implementing acronym disambiguation algorithms.

Before implementing these algorithms, the raw corpus is preprocessed and normalized; punctuations and stop words are removed in order to prepare the corpus for the next step.

By mining the normalized texts in the database, two major patterns of acronyms were recognized (Table 2).

Table 2 – Patterns of acronyms in texts

Pattern one:	
Definition	Example
The acronym inside parentheses follows the expansion	Local area network (LAN)
Pattern two:	
Definition	Example
The acronym is prior to its expansion, which is embraced by parentheses	LAN (local area network)

In order to detect pattern one, the algorithm mines the corpus. As the algorithm detects a single parenthesis, the phrase inside parentheses is extracted. If the phrase consists of one token with capital letters, it is a potential acronym. If the length of

[1] (https://www.acronymfinder.com), best known reference for acronyms

978-1-7281-5350-6/19/$31.00 ©2019 IEEE

this candidate is n, n words prior to this phrase are extracted, and its initials are put together to make a potential acronym. If the initials from the potential expansion match the actual acronym, it can be derived that a correct pair of acronym and expansion has been detected. In practice, there are challenges in acronym-expansion pair detection as follows:

- Double quotations, dashes, stop words, and the like.
- Nested expansions, i.e., expansions that include acronyms, e.g., Independent BSS (Independent Base Station Subsystem)
- Two-letter acronyms, e.g., PGW: Payment Gateway

In order to address such issues, normalization is performed to remove punctuations and stop words. The expansions of common two-letter acronyms are split into two tokens in the normalization phase. Take the following pair of acronym and expansion as an example. GW is a two-letter acronym that stands for a single token (Gateway). In the normalization phase we replace Gateway with Gate Way.

Pattern two:

This pattern is less common than the first one. In this pattern, in order to detect the acronyms, the whole text is mined. If a token has a single parenthesis, it is considered as a potential expansion. The word before this token is presumed to be the acronym. If the acronym were of length n, n subsequent words after the acronym would be the desired expansion. The initials of the potential expansion are then compared with the acronym. If they matched, it would be derived that the correct pair of acronym and expansion has been detected.

B. Results

I. PREPARING RAW DATA

In order to collect data, we implemented crawlers aimed at collecting texts from scientific resources such as ACM, Arxiv, IEEE, and US patents. Within three months, 165420 paper abstracts, 98058 technical and scientific papers were extracted and stored in a database.

II. ACCURACY MEASUREMENT

From 165,420 abstracts and 98,058 papers, 1067 acronyms and their corresponding expansions from twenty randomly chosen papers were manually extracted and tagged; the zip file of the word documents is shared[2]. Also, the proposed method is implemented in python and is shared[3]. By using the proposed method, 919 acronyms and their expansions were automatically extracted with 86% accuracy (Table 3).

Due to volume limitations, it was not feasible to upload the generated dataset (it was more than 3GB), but the proposed method, which is implemented and shared, could tag the row text to create the desired dataset.

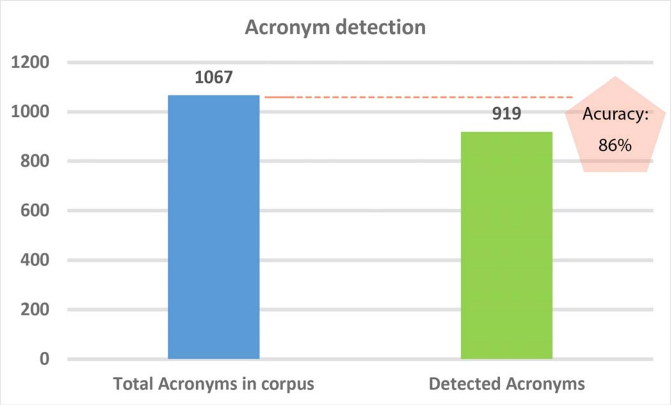


Figure 2 - Accuracy of the proposed method to detect correct acronyms and expansions

Table 3 – statistical results

Total number of texts	263478
Number of texts in the test set	20
Number of acronyms in the test set	1067
Number of detected acronyms	919
Accuracy	86%

IV. CONCLUSION

In this paper, a method for building a database of tagged acronyms and their corresponding expansions in English technical and scientific texts was proposed. The model yields an accuracy of 86 %.

V. FURTHER WORKS

In future works, methods for building recognition and disambiguation models based on machine learning methods will be presented. Accuracy of prediction using this model can be enhanced using bigger datasets, more efficient algorithms, and considering more diverse patterns.

REFERENCES

[1] Pustejovsky, James, et al. "Automatic extraction of acronym-meaning pairs from MEDLINE databases." *Studies in health technology and informatics* 1 (2001): 371-375.

[2] Pustejovsky, James, et al. "Extraction and disambiguation of acronym-meaning pairs in medline." *Medinfo* 10.2001 (2001): 371-375.

[3] Singh, Harsimran, and Vishal Gupta. "An Insight into Word Sense Disambiguation Techniques." *International Journal of Computer Applications* 118.23 (2015).

[2] https://github.com/S-Azimi/Acronym_detection/raw/master/word_files.zip
[3] https://github.com/S-Azimi/Acronym_detection/blob/master/Acronym_detection.py

978-1-7281-5350-6/19/$31.00 ©2019 IEEE

[4] Li, Yang, et al. "Guess Me if You Can: Acronym Disambiguation for Enterprises." Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2018.

[5] Kao, Anne, and Steve Poteet. "Text mining and natural language processing: introduction for the special issue." *ACM SIGKDD Explorations Newsletter* 7.1 (2005): 1-2.

[6] Mooney, Raymond J., and Razvan Bunescu. "Mining knowledge from text using information extraction." *ACM SIGKDD explorations newsletter* 7.1 (2005): 3-10.

[7] Azimi, S., H. Veisi, and R. Rahmani. "Automatic Discovery of Technology Networks for Industrial-Scale R&D IT Projects via Data Mining." *AUT Journal of Electrical Engineering* 47.1 (2015): 17-22.

[8] Moro, Andrea, Alessandro Raganato, and Roberto Navigli. "Entity linking meets word sense disambiguation: a unified approach." *Transactions of the Association for Computational Linguistics* 2 (2014): 231-244.

[9] Li, Chao, Lei Ji, and Jun Yan. "Acronym disambiguation using word embedding." *Twenty-Ninth AAAI Conference on Artificial Intelligence*. 2015.

[10] Thakker, Aditya, Suhail Barot, and Sudhir Bagul. "Acronym Disambiguation: A Domain Independent Approach." *arXiv preprint arXiv:1711.09271* (2017).

[11] Xu, Jun, Yaoyun Zhang, and Hua Xu. "Clinical abbreviation disambiguation using neural word embeddings." *Proceedings of BioNLP 15*. 2015.

[12] Kreuzthaler, Markus, et al. "Unsupervised abbreviation detection in clinical narratives." Proceedings of the clinical natural language processing workshop (ClinicalNLP). 2016

[13] Charbonnier, Jean, and Christian Wartena. "Using word embeddings for unsupervised acronym disambiguation." (2018).