Participation of UC3M in SDU@AAAI-21: A Hybrid Approach to Disambiguate Scientific Acronyms

Areej Jaber,[1,2] **Paloma Martínez**[1]

[1]Computer Science Department, Universidad Carlos III de Madrid ,
[2]Applied Computer Science Department, PTUK University
[1]30 Av. de la Universidad, 28911 Leganés, Madrid, Spain
[2]Jaffa street, 7 Tulkarem, Palestine,
a.jabir@ptuk.edu.ps, pmf@inf.uc3m.es

Abstract

Acronyms disambiguation is considered a word sense disambiguation (WSD) task which consists on determining the correct expansion of an acronym based on a given context. This paper describes three hybrid systems to disambiguate acronyms in scientific documents, which combine three supervised machine learning (ML) models (Support Vector Machine, Naive Bayes and K-Nearest Neighbor) with cosine similarity on SciAD corpus. Our system achieved it's best performance on the independent test set on Naive Bayes and cosine similarity with 92.15% of precision, 77.97% of recall and 84.47% of F1-macro measure.

Introduction

Acronyms are defined as 'a short form of multiple words or phrases' which are used in various type of documents. Normally its meaning is represented the first time is used in each document. But there are many cases that it is used alone without its meaning like in the case of clinical documents.

There is no standard rules to create acronyms. Usually each acronym has more than one meaning which is called 'expansion' or 'long form'. Writing acronyms without their expansions in the same sentence makes it ambiguous. Determining the correct expansion for an acronym depends on many factors like the domain it is used in. For example; acronym 'ED' could mean 'Emergency Department' if it is used in documents related to medical domain, or it could mean 'Euclidean Distance' if it is used in documents related to mathematics domain. Furthermore, acronyms could have many expansions even in the same domain like 'RNN' which has two possible expansions 'Recurrent Neural Network' and 'Random Neural Networks' which both of them are used in computer science domain.

Word Sense Disambiguation (WSD) is a Natural Language Processing (NLP) task which is applied to determine the right expansion of acronyms based on it's context. There are two types of WSD; all_words WSD which disambiguates all words in the given context. The second type is Lexical_sample WSD which disambiguates specific word in the context. Disambiguate acronyms are considered a special case of Lexical_sample WSD.

Three main approaches are applied extensively on WSD. The first one is Knowledge based approach that integrates lexical knowledge bases and exploits semantic similarity and graph-based approaches. In similarity-based methods each expansion of the ambiguous acronym is compared to those of the content words appearing near it (context words) and the expansion with the highest similarity (for instance, using cosine distance) is supposed to be the right one. (Billami 2017).

Unsupervised ML approaches disambiguate by finding hidden structure in unlabelled data, for instance, clustering documents or sentences in groups each one representing an expansion (Charbonnier and Wartena 2018).

Finally, supervised ML approaches which require tagged corpora. WSD based on this approach, is considered as a text classification problem where the objective is to predict the correct expansion of an acronym among its different expansions (Melacci, Globo, and Rigutini 2018). Supervised approaches achieved high performance in this type of task, but it requires annotated data that is considered expensive to generate. To face this problem, semi-supervised approaches are applied. In semi-supervised approaches training data are automatically generated from few annotated examples (da Silva Sousa, Milios, and Berton 2020).

We explore word embeddings in this work as features to be used in ML algorithms; a preliminary analysis is done in (Jaber and Martínez 2021). A word embedding is a real-value vector that represents a single word based on the context in which it appears (Khattak et al. 2019). These numerical word representations could be built using different models like (Mikolov et al. 2013), (Peters et al. 2018) and (Devlin et al. 2019) based on different neural networks architectures. Fortunately, these embeddings could be trained on large data set, saved and used in solving other tasks; they are called pre-trained word embeddings or pre-trained models.

In this paper three supervised ML models combined with a knowledge based model are used to disambiguate scientific acronyms for SDU@AAAI-21 shared task (Veyseh et al. 2020a). The rest of the paper is organized as follows: Method section describes the data set which is used in this study, the features and the different models that we applied.

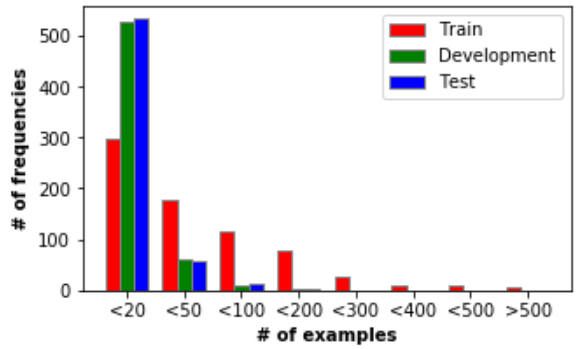

Figure 1: Frequency of each number of examples per acronym across train, development and test data sets.

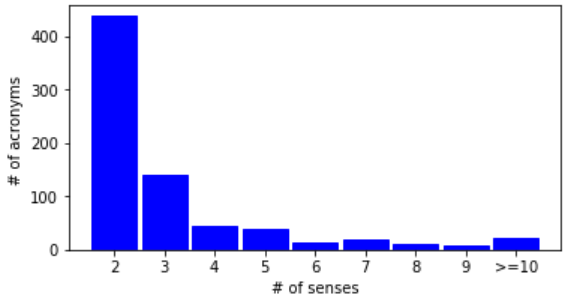

Figure 2: Number of senses per acronym in the dictionary. E.g. we see that there are 437 acronyms with two expansions.

In strategies section we describe how the proposed methods are experimentally conducted. Finally, we present our results compared to the baseline system.

Method

Acronyms are ambiguous because they could have multiple expansions. Determining the correct expansion of an acronym is a WSD problem. Since SciAD contains an small set of exampled for some acronyms, we combined supervised machine with knowledge based approaches to tackle this problem.

Data Set

SciAD (Veyseh et al. 2020b) corpus is used in this task, which is created by AAAI-21 shared task 2 organizers. SciAD was generated from 6,786 English papers from arXiv with 2,031,592 sentences. Table 1 shows the detailed numbers of annotated samples on three data set, training, development and test training data set.

	Training	Development	Test
Sentences	50034	6189	6218
Tokens	1548278	190654	190111
Acronyms	731	611	618
Expansions	2150	1233	-

Table 1: Description of training, development and test data sets.

Figure 1 shows frequencies of annotated examples per each acronym; 299 acronyms have less than 20 annotated examples in the training data set.

Additionally, the organizers provide the participants with an acronyms dictionary which contains 732 acronyms and 2308 senses with average of 3.15 senses per acronym. Figure 2 shows the distributions of senses for acronyms contained in the dictionary.

Model

Baseline In order to familiarize the participants with the task, the organizers provided a rule-based baseline in code directory. This baseline computes the frequency of the long forms in the training data set. Afterwards, to make prediction for each acronym in the development data set, it selects the long form with the highest frequency as the final prediction. If there is a tie, the long form that appears the first among all tied long forms in the dictionary is selected as the final prediction.

Supervised ML Three Supervised ML algorithms are implemented:

- Support Vector Machine (SVM): which separates positive samples from negative ones based on the idea of linear hyper-plane from labeled data set differentiating between samples into true or false categories. SVM is adapted to multi-class classification to be used in WSD.
- Naive Bayes (NB): a probabilistic approach to estimate probabilistic parameters which has a long history of success in WSD. This approach is based on Bayes theorem to compute the conditional probability for each sense of an abbreviation from a set of features.
- K-Nearest Neighbor (KNN): the classification is done by computing the Euclidean distance for each test vector with the most k similar training vectors.

Knowledge Based Approach For acronyms with few examples in the data set, which are insufficient to train a supervised ML method, a knowledge based approach is implemented. This method is based on expansion's dictionary provided by organizers; cosine similarity was applied in the test examples. Two vectors were said to be similar when the cosine similarity was close to 1, and they were said to be dissimilar when it is close to 0 (Singhal 2001).

Features

Features play an important rule in WSD system, two types of features were used. WSD Features: Several lexical features were used to disambiguate acronyms considering both left and right contexts of the target Acronym. Our system adopted a set of lexical features that have been used successful in WSD. Given a sentence s formed by a set of words $[...w_{-2}, w_{-1}, w_0, w_t, w_{+1}, w_{+2}...]$ where w_k is the targeted ambiguous acronym, we extracted the following features:

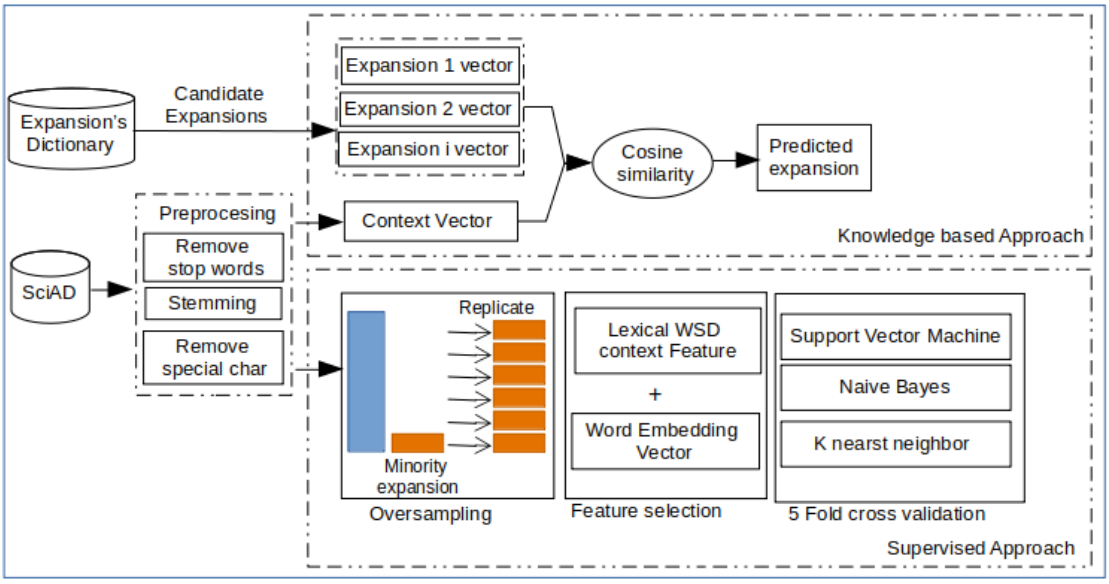

Figure 3: Overview of proposed approach to disambiguate acronyms.

1. Word Features: stemmed words for each tokens on both side of the target acronym.
2. Word features with direction: The relative direction (left or right side) of stemmed words.
3. POS (Part-Of-Speech) Tag: POS tag feature for each token on both sides.
4. Position features: The distance between the feature word and the target acronym.
5. Word formation features from the acronym itself including special characters, capital letters and numbers.

Pre-trained word embedding features: A pre-trained word embedding model with 300 dimension vectors was built used FastText (Joulin et al. 2016) generated from several English resources such as the Wikipedia and data from the common crawl project, (Mikolov et al. 2018).

Strategies

Pre processing data & Features extracting

Several pre-processing steps were conducted on the dataset including remove stop words, special characters and stemming the words before extracting the features. For supervised ML approaches, features are formed by combining WSD lexical features and the summation strategy from the pre trained word embeddings which are generated based on the following equation:

$$S = \sum_{i=0}^{|W|} v(W(i)), i \neq k \quad (1)$$

where W is a list of words which surrounding the targeted acronym. $|W|$is the length of the list and v(.) is a Fasttext pre trained word embedding as mentioned in previous subsection and k is the position of the target acronym.

On the other hand, for the knowledge based approach, just the summation strategy of pre trained word embedding vectors were generated for each example and for the candidate expansions which were extracted from expansions dictionary.

Training Phase

In this phase, training and development data sets were combined to increase the size of data set for each acronyms. Our goal was to build a model to predict acronym's expansion based on a context for each acronym that has more than 20 annotated examples. To achieve this goal the training data was separated based on each acronym data set. Table 2 shows the distribution of the whole data set for ML and Knowledge-based (KB) approaches, 450 acronyms with 53702 annotated examples, are disambiguated by three ML models, SVM, NB, and KNN. While 282 acronyms with 2521 annotated examples disambiguated by cosine similarity method.

	Data set	Acronyms	Expansions
ML	53702	450	1601
KB	2521	282	594
Total	56223	732	2195

Table 2: Distribution of data sets, acronyms over two proposed models in the training phase.

Testing Phase

When the testing data set was released by the organizers, the testing data set was divided based on the training data we had previously (see Figure 4). Table 3 shows the distribution of testing data set over the two models; 444 acronyms with

5876 annotated examples in the testing data set, are disambiguated through the three ML models. 174 acronyms in 342 annotated testing examples were disambiguated with cosine similarity method. Figure 3 summarizes the overall process for the proposed system.

	Data set size	# of acronyms
Machine Learning	5876	444
Knowledge based	342	174
Total	6218	618

Table 3: Distribution of Data sets, Acronyms over two proposed models in the testing phase.

	Precision	Recall	F1-macro
NB-KB	90.31%	87.16%	84.37%
SVM-KB	90.20%	86.78%	88.16%
KNN-KB	83.85%	79.59%	79.53%

Table 4: Averaged performance of the three proposed hybrid approaches implemented on the training phase.

	Precision	Recall	F1-macro
NB	92.15%	77.97%	84.47%
SVM	91.66%	73.33%	81.48%
KNN	90.26%	67.51%	77.25%

Table 5: Averaged performance of the three proposed hybrid approaches on testing data set.

Evaluation & Result

The system performance was evaluated by using three metrics, Precision which is defined as the percentage of the instances which actually have a class label X (True Positives) divided by all those which were classified as class label X as the following equation:

$$Precision = \frac{TruePositives}{Truepositives + FalsePositives} \quad (2)$$

Recall is defined as the percentage of the instances which were classified as class X, divided by all instances which correctly have class X as the following equation:

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives} \quad (3)$$

F1-macro is defined as the harmonic mean of Precision and Recall as the following equation:

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

The training data set includes 634 expansions with less than 10 annotated examples from different acronyms, to balance the data set, these expansions were replicated through oversampling techniques using sklearn library. Then 5 fold

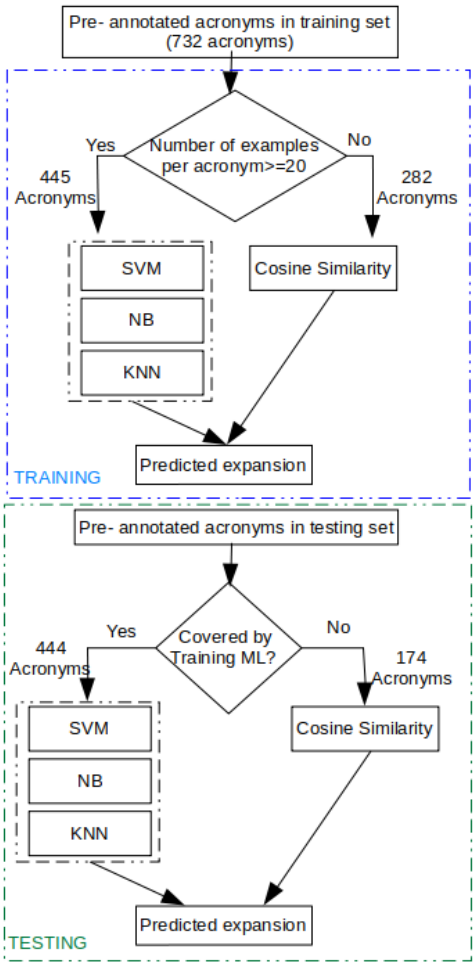

Figure 4: Data flow chart in training and testing phases.

cross validation was used for all acronyms in ML models. Furthermore, the training data set contains 10 non-ambiguous acronyms which their data set contain one expansion.

Table 4 shows our result on training phase, NB with cosine similarity achieved the highest performance with precision 90.31% , recall 87.16% and F1-macro 84.37%.

Table 5 shows the final scores for our systems were reported by the organizers. The best performance achieved precision 92.15%, recall 77.97% and F1-macro 84.47%, for a hybrid approach with NB and cosine similarity.

Preliminary Analysis of Errors

A sample of low performance on accuracy were achieved on a training phase shows how strongly imbalanced data set size affects on the model. We focus on Naive Bayes approach since the best result was achieved through this approach. Table 6 shows the accuracy of 4 acronyms, ARD acronym with 246 dataset is distributed between two expansions "accelerated robust distillation" with 46 training examples and "adversarially robust distillation" with 201 training examples,

Acronym	Data set size	Number of expansions	Accuracy	Expansion	Number of examples per expansion
MSE	501	3	52%	mean squared error	462
				minimum square error	10
				model selection eqn	29
GP	552	2	61%	gaussian process	466
				geometric programming	86
CNN	2973	4	58%	citation nearest neighbour	14
				complicated neural networks	1
				condensed nearest neighbor	33
				convolutional neural network	2925
ARD	247	2	38%	accelerated robust distillation	46
				adversarially robust distillation	201

Table 6: Distribution of data set size over expansions and the accuracy of Naive Bayes model on sample acronyms.

was achieved the lowest accuracy which is 38%.

Conclusion

In this paper, we introduced a system to disambiguate scientific acronyms. Our system best score was achieved by a hybrid approach combining supervised ML Naive Bayes and cosine similarity with precision 92.15%, recall 77.97% and F1-macro 84.47%.

Acknowledgments.

Thanks to Palestine Technical University-Kadoorie (PTUK) and DeepEMR project (TIN2017-87548-C2-1-R) for partially funding this work.

References

Billami, M. 2017. A Knowledge-Based Approach to Word Sense Disambiguation by distributional selection and semantic features. *CoRR* abs/1702.08450. URL http://arxiv.org/abs/1702.08450.

Charbonnier, J.; and Wartena, C. 2018. Using Word Embeddings for Unsupervised Acronym Disambiguation. In Bender, E. M.; Derczynski, L.; and Isabelle, P., eds., *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, 2610–2619. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/C18-1221/.

da Silva Sousa, S. B.; Milios, E. E.; and Berton, L. 2020. Word sense disambiguation: an evaluation study of semi-supervised approaches with word embeddings. In *2020 International Joint Conference on Neural Networks, IJCNN 2020, Glasgow, United Kingdom, July 19-24, 2020*, 1–8. IEEE. doi:10.1109/IJCNN48605.2020.9207225. URL https://doi.org/10.1109/IJCNN48605.2020.9207225.

Devlin, J.; Chang, M. W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference* 1(Mlm): 4171–4186.

Jaber, A.; and Martínez, P. 2021. Disambiguating Clinical Abbreviations Using Pre-trained Word Embeddings. In *To appear in Proceedings of the 14th International Joint Conference on Biomedical Engineering Systems and Technologies: HEALTHINF,*. INSTICC.

Joulin, A.; Grave, E.; Bojanowski, P.; and Mikolov, T. 2016. Bag of Tricks for Efficient Text Classification. *arXiv preprint arXiv:1607.01759* .

Khattak, F. K.; Jeblee, S.; Pou-Prom, C.; Abdalla, M.; Meaney, C.; and Rudzicz, F. 2019. A survey of word embeddings for clinical text. *Journal of Biomedical Informatics: X* 4(October): 100057. ISSN 2590177X. doi:10.1016/j.yjbinx.2019.100057. URL https://doi.org/10.1016/j.yjbinx.2019.100057.

Melacci, S.; Globo, A.; and Rigutini, L. 2018. Enhancing Modern Supervised Word Sense Disambiguation Models by Semantic Lexical Resources. In Calzolari, N.; Choukri, K.; Cieri, C.; Declerck, T.; Goggi, S.; Hasida, K.; Isahara, H.; Maegaard, B.; Mariani, J.; Mazo, H.; Moreno, A.; Odijk, J.; Piperidis, S.; and Tokunaga, T., eds., *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2018/summaries/112.html.

Mikolov, T.; Grave, E.; Bojanowski, P.; Puhrsch, C.; and Joulin, A. 2018. Advances in Pre-Training Distributed Word Representations. In Calzolari, N.; Choukri, K.; Cieri, C.; Declerck, T.; Goggi, S.; Hasida, K.; Isahara, H.; Maegaard, B.; Mariani, J.; Mazo, H.; Moreno, A.; Odijk, J.; Piperidis, S.; and Tokunaga, T., eds., *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2018/summaries/721.html.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; and Dean, J. 2013. Distributed representations ofwords and phrases and their compositionality. *Advances in Neural Information Processing Systems* 1–9. ISSN 10495258.

Peters, M. E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; and Zettlemoyer, L. 2018. Deep Contextualized Word Representations. In Walker, M. A.; Ji, H.; and Stent, A., eds., *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, 2227–2237. Association for Computational Linguistics. doi:10.18653/v1/n18-1202. URL https://doi.org/10.18653/v1/n18-1202.

Singhal, A. 2001. Modern Information Retrieval: A Brief Overview. *IEEE Data Eng. Bull.* 24(4): 35–43. URL http://sites.computer.org/debull/A01DEC-CD.pdf.

Veyseh, A. P. B.; Dernoncourt, F.; Nguyen, T. H.; Chang, W.; and Celi, L. A. 2020a. Acronym Identification and Disambiguation Shared Tasks for Scientific Document Understanding. *CoRR* abs/2012.11760. URL https://arxiv.org/abs/2012.11760.

Veyseh, A. P. B.; Dernoncourt, F.; Tran, Q. H.; and Nguyen, T. H. 2020b. What Does This Acronym Mean? Introducing a New Dataset for Acronym Identification and Disambiguation. *CoRR* abs/2010.14678. URL https://arxiv.org/abs/2010.14678.