Using Word Embeddings for Unsupervised Acronym Disambiguation

Jean Charbonnier
Hochschule Hannover
Expo Plaza 12
D-30539 Hannover
jean.charbonnier
@hs-hannover.de

Christian Wartena
Hochschule Hannover
Expo Plaza 12
D-30539 Hannover
christian.wartena
@hs-hannover.de

Abstract

Scientific papers from all disciplines contain many abbreviations and acronyms. In many cases these acronyms are ambiguous. We present a method to choose the contextual correct definition of an acronym that does not require training for each acronym and thus can be applied to a large number of different acronyms with only few instances. We constructed a set of 19,954 examples of 4,365 ambiguous acronyms from image captions in scientific papers along with their contextually correct definition from different domains. We learn word embeddings for all words in the corpus and compare the averaged context vector of the words in the expansion of an acronym with the weighted average vector of the words in the context of the acronym. We show that this method clearly outperforms (classical) cosine similarity. Furthermore, we show that word embeddings learned from a 1 billion word corpus of scientific texts outperform word embeddings learned from much larger general corpora.

1 Motivation

Scientific papers usually contain a high number abbreviations and acronyms that might be very general or very specific for a certain domain and that might be very common in that domain or that are constructed ad hoc. For various tasks, like retrieval or information extraction we need the expanded form of the acronyms.[1]

In most cases we can find the definition for an acronym in the paper. Finding this definition automatically is not as trivial as it might sound and there is a lot of literature on this topic. However, a lot of cases remain where we cannot find such a definition, because the extraction method fails or because there is no definition in the paper. In these cases we can use a definition found elsewhere. As already noted by Chang et al. (2002) in these cases acronyms often turn out to be ambiguous and the correct definition has to be chosen by considering the context of the acronym. Exactly this task is the topic of the present paper. Thus, although the goal of our work is acronym expansion, the work is more related to word sense disambiguation (WSD) than to typical work on acronym resolution. The main difference with WSD is that we do not have dictionaries with description of possible senses. Instead the possible expansions of the acronym directly represent the different senses.

In the Project NOA (Reuse of Open Access Images, http://noa.wp-hs-hannover.de) we analyze captions from images in scientific papers to collect appropriate metadata for those images (Charbonnier et al., 2018). In order to get maximal information from the usually short image captions we need to expand the acronyms. Moreover, the acronyms are often very specific technical terms that give a lot of information on the displayed image. Since the focus of the project is on image captions, in the present paper we consider only acronyms from image captions, but this is not relevant to our method that can be used to disambiguate any acronym in a scientific paper. Since the image captions in some cases are very short, we also use the text from the sentences referring to the image to disambiguate the acronyms in the caption.

Proceedings of the 27th International Conference on Computational Linguistics, pages 2610–2619
Santa Fe, New Mexico, USA, August 20-26, 2018.

2 Related Work

Typically definitions for acronyms are found using a number of patterns to find candidates in the context of the acronyms and a number of rules to select the most likely candidate. Patterns and rules are defined manually or learned in a supervised (Nadeau and Turney, 2005) or unsupervised (Kirchhoff and Turner, 2016) way. The work of Schwartz and Hearst (2011) focuses on finding and extracting abbreviations from medical publications by identifying <"short form", "long form"> pairs and was used in a majority of the work discussed here. They argue that many abbreviations in the biomedical domain follow a predictable pattern where the first letter of each word is a letter of its short form, but that there are cases where this is not true. Therefore they describe a simple and fast algorithm with a high recall and precision to find those pairs using string operations to verify that the acronym letters appear in the right order in the long form.

Much research on acronym resolution and disambiguation was done in the biomedical domain (Okazaki and Ananiadou, 2006; Finley et al., 2016; Vo et al., 2016; Wu et al., 2017). The most reliable approach for disambiguation is to train a classifier that chooses the right definition based on features representing the context of the acronym (Wu et al., 2015; Wang et al., 2016; Finley et al., 2016; Antunes and Matos, 2017). Typical features are the word frequencies but also morphemes and position features. Wu et al. (2015) and Antunes and Matos (2017) also include features from word embeddings. Kirchhoff and Turner (2016) learn the similarity between a possible expansion and the context in a neural network using word embedding features. Wu et al. (2015) use the sum of all pmi (pointwise mutual information) values between a definition and a context as a measure of their similarity. Thus they also use a more advanced similarity measure than just word overlap based on word co-occurrence in a large corpus.

Stevenson et al. (2009) created a biomedical evaluation corpus consisting of overall 55,655 documents using MEDLINE and MEDSTRACT. They counted the occurrence of 21 three letter abbreviations and of their expansions in the abstracts of those documents. The occurrence of these hand selected abbreviations range from $71 - 14,871$ times in the corpus. Jimeno-Yepes et al. (2011) created the widely used MSH-WSD dataset consisting of 203 ambiguous entities, where 106 are abbreviations. Each of those has a maximum of 100 instances from MEDLINE where they occur.

Disambiguation of acronyms is a special case of the more general problem of word sense disambiguation (Navigli, 2009; Yarowsky, 2010; Mihalcea, 2011). Word sense disambiguation is typically done by comparing the possible senses of an ambiguous word, expressed e.g. by their glosses in a dictionary, with the senses of the surrounding words (Lesk, 1986; Patwardhan et al., 2003). The gloss overlap can be improved, when the glosses are extended with related words (Banerjee and Pedersen, 2003). We will follow this idea below, but the extension of the glosses is done implicitly using distributional similarity, as also was done by Aga et al. (2016) and Oele and Noord (2017).

Henry et al. (2017) and Tulkens et al. (2016) specifically worked on disambiguation of acronyms. They used the MSH-WSD dataset and UMLS to disambiguate words/concepts in the domain of biomedicine. Tulkens et al. (2016) combined word representations and definitions from UMLS to create concept representations. Those concepts are then used to disambiguate terms using cosine similarity and the context of the unambiguous term. As features they used word embeddings as described by Mikolov et al. (2013) learned with MEDLINE abstracts and the MIMIC-III corpus. Henry et al. (2017) evaluate different feature extraction methods using 2-MRD (2nd Order Co-occurrence Machine Readable Dictionary) to disambiguate biomedical word senses. Among these were Singular Value Decomposition, Principal Component Analysis and Neural Word Embeddings trained with Bag of Words and Skip Gram. They used extended definitions like parent/children, narrow/broader relations and associated synonymous terms from UMLS to create those vectors. Pakhomov et al. (2016) evaluated the effect of domain specific corpora for disambiguating medical terms while using word embeddings. They showed that biomedical articles can be used to a degree to represent the semantics of clinical reports.

3 Data

Most available data sets with acronyms are quite small, or focus specifically on the biomedical domain, or do not contain ambiguous acronyms, or only ambiguous acronyms with a few very frequent resolutions. To develop a disambiguation method that is suited for our practical problem, we need a set of ambiguous

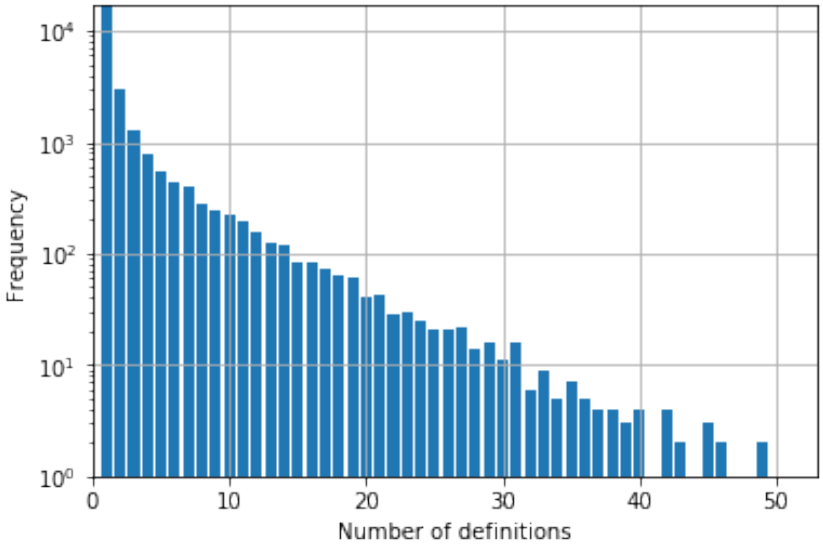

Figure 1: Number of definitions (after clustering) per acronym in the NOA corpus.

acronyms from different disciplines with possibly very infrequent and unusual definitions. In absence of such a test set, we created our own set of ambiguous acronyms.

3.1 Acronyms in the NOA Corpus

Our database contained 712,438 papers from open access journals by the end of 2017. The collection contains journals from almost all scientific disciplines with a focus on technical and biomedical research areas. We refer to Sohmen et al. (2018) for more details.

In this collection we found 2,838,713 occurrences of words written in all capitals of at least 3 letters that thus are likely to be acronyms. For 25,336 acronyms with a total of 628,470 occurrences we found a definition in the paper using a simple regular expression. For 379,509 additional acronyms we could find an unambiguous definition in another paper from the same journal. In total about 36% of all potential acronyms could be expanded. For 67 % of the remaining acronyms we have multiple possible expansions from other papers.

Since authors sometimes seem to be uncertain about the exact definition of acronyms we build clusters of definitions, such that for each pair (a, b) in a cluster either a is a permutation of b (like *annual average daily traffic* and *average annual daily traffic* for AADT) or if the Levenshtein Distance of a and b is smaller than 5 (like *anterior acetabular sector angle* and *Anterior acetabular section angle* for AASA.) Finally, we take the most frequent definition as the correct one. The average cluster size is 1.36. The average number of definitions was reduced from 3.6 to 2.7. Since we used simple heuristics for cluster building, we have a few cases in which two different definitions are put into one cluster and many cases of equivalent definitions that are still treated as different definitions: the overwhelming majority of the clusters only contain spelling variants, different capitalization and typos. The distribution of the number of definitions per acronym for all acronyms in image captions in the NOA corpus is given in Figure 1.

The acronym with the highest number of definitions is SSC with 52 definitions that is used for concepts like *Surgical Safety Checklist*, *symmetric similarity coefficients*, *secondary sclerosing cholangitis* and *Swedish Space Cooperation*.

3.2 Test Set

From our database we collected a set acronyms that have at least two and at most five definitions that are not too similar. As a criterion for similarity we used trigram overlap and we required that the Jaccard-Index of the sets of trigrams was smaller than 0.4. This gave us 4,365 acronyms. In this set we still have 140 acronyms with at least two definitions that only differ in one word, like *DNA Damage Induced Sumoylation*

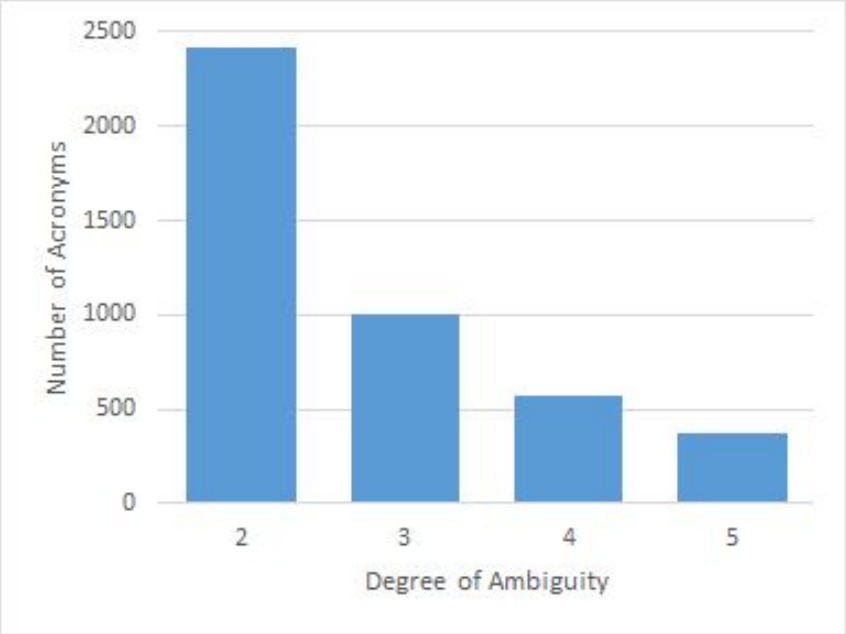

Figure 2: Number of acronyms for each degree of ambiguity in the test set. E.g. we see that there are 1000 acronyms with 3 possible expansions in the test set.

Table 1: Example of the data set. The contextual right definition is typeset in bold.

Acronym	Definitions	DOI of Source	Context/Caption
EBG	Electro-burnt Graphene **Electromagnetic Band Gap**	10.1155/2008/790358	*The EBG leaky wave antenna is fed with a patch array. It must radiate in the direction* $[\Theta = 30^o; \phi = 0^o]$.

and *DNA Damage Induced Senescence* or *very low calorie diet* and *very low carbohydrate diet*. The numbers of acronyms for each degree of ambiguity is given in Figure 2.

For each sense of an acronym we selected at least 1 and at most 5 image captions, in which that acronym was used, from papers in which it was unambiguously defined. This resulted in a set of 19,954 instances of acronyms in a specific context that have to be disambiguated. The data set is provided as supplementary material to this paper. A typical example of a such an instance is given in Table 1. The average length of the captions is 531.3 characters (105.7 tokens), the median is 385 characters (77 tokens). Figure 3 shows the number of expansions with one, two, etc. examples in the test set. Given the low number of examples, training a classifier for each ambiguous acronym is not possible.

3.3 Wordembeddings

We use the full text of all 672,157 papers in our database with a total of 1.662×10^9 tokens to compute word embeddings for all words in the definitions of the acronyms and in the image captions. For training of the word embeddings we only use the body texts and excluded the captions. The 12,002 definitions in our test set consist of 116,603 different tokens. For 92 of those tokens we do not have a word embedding. These are mainly stop words, typos and single letters (as in *Downstream of Kinases (DOK)* and *Vitamin K Antagonist (VKA)*, resp.). In the Google and GloVe dictionaries 821 and 189 tokens, respectively, are not captured. In theses dictionaries in addition many technical terms are missing.

The best results were achieved with a window size of 5 using CBOW, an embedding size of 300. We removed all tokens from our data that are in the NLTK stopword list, that have less than 2 characters or that occur less than 5 times in the corpus.

For comparison we also used two well known pre-trained general models: GoogleNews (Mikolov et al., 2013) and GloVe (Pennington et al., 2014).

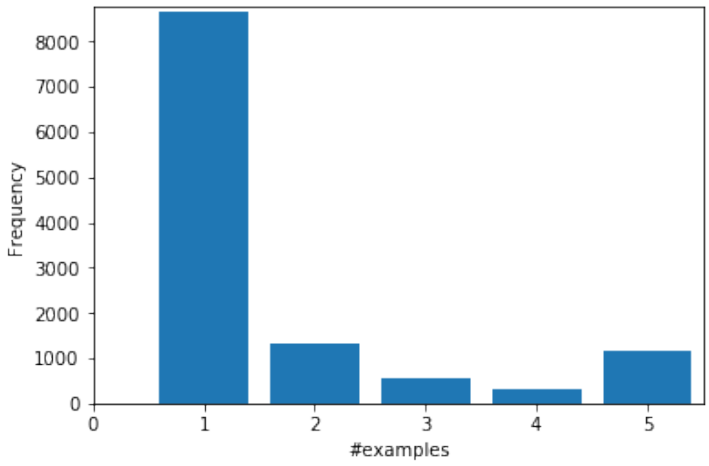

Figure 3: Frequency of each number of examples per definition: For the majority of the possible definitions of an acronym the test set only contains one example; there are about 1200 definitions for which there are 2 examples in the test set, etc.

4 Method

The general idea to decide which sense of an acronym is the contextual correct one, is to compare each definition of the acronym with its context. Now we can vary both the context and the method to compare the definition and context. As the context we take either the image caption from which the acronym was taken, or we expand the caption with sentences referring to the image.

4.1 Baselines

As a baseline we use an unsupervised classifier based on the cosine similarity between tf-idf-vectors of the definition and the context.

As a further baseline we use a majority classifier, that always assigns the definition that was found most frequently in the whole database.

4.2 Prediction

Since the simple cosine similarity used as a baseline captures only literal word overlap and not semantic similarities we also use word embeddings to represent the word in the texts and compare the average word embedding vector for both texts.

To decide which of the given candidates for a solution should be selected, we construct a representing vector for the example text. This vector is the weighted average of all word-embeddings found in the text. We simply discard any token that is not in our dictionary during the average vector calculation. We also calculate an average weighted vector representing the alternative definitions. In both cases we use idf-values as weights for the components. We calculate the cosine similarity and select the candidate with the higher similarity as the right solution.

Formally, let $t = s_0, s_1, \ldots, s_n$ be any sequence of tokens. Now we define the *context vector* of t as

$$cv(t) \quad = \quad \frac{1}{N} \sum_{i=0}^{n} idf(s_i) \cdot cv(s_i) \tag{1}$$

where $cv(s_i)$ is the word embedding or context vector of s_i. Now, for some acronym a let $\mathcal{D} = \{D^0, D^1, \ldots D^n\}$ be the set of definitions for a, with $D^i = d^i_0 \ldots d^i_{k^i}$ the sequence of tokens from the definitions for each $0 \leq i \leq n$ and $C = c_0 \ldots c_l$ the tokens in the context. Now the predicted definition $\text{pred}(a, C)$ of a in context C is:

$$\text{pred}(a, C) \quad = \quad \max_{\forall D \in \mathcal{D}} \cos\left(cv(D), cv(C)\right) \tag{2}$$

Table 2: Accuracy of acronym disambiguation

Method	*small Context*	*larger Context*
Majority Classifier	0.44	
Vector Space Model (Cosine)	0.74	0.77
Distr. Sim. – GloVe	0.70	0.71
Distr. Sim. – Google News	0.71	0.73
Distr. Sim. – Own vectors	0.84	0.86

Table 3: Acronyms incorrectly resolved by our method. The contextual right definition is typeset in bold.

Acronym	**Definitions**	**DOI of Source**	**Context/Caption**
DOW	Day of Week **Deep Ocean Water**	10.3390/md15060168	..... *DOW: TAA-induced mice fed with DOW (0.603 mL/kg/day), AS: TAA-induced mice fed with adenosine (0.329 mg/kg/day), CC: TAA-induced mice fed with cordycepin (0.615 mg/kg/day).*
AIHT	**Adaptive Inverse Hyperbolic Tangent** Analysis Iterative Hard Thresholding	10.1155/2014/825169	*AIHT function image enhancement algorithm procedures.*

4.3 Adding more Context

For very short captions we can extend the context of the acronym with text from paragraphs referring to the image. To be precise, if there are less then 15 tokens in the caption, we add the referring sentence to the context and keep adding left and right neighbor sentences until we have 35 tokens.

5 Results

The results are shown in Table 2. We see that all methods using the context of an acronym outperform the majority classifier. In all cases adding more context for short captions results in a small improvement. The results of the conditions using the pre-trained vectors are slightly worse than those obtained by simple word overlap. The similarity based classifier using the corpus specific word embeddings clearly outperforms all other variants.

6 Discussion

One would expect that using word embeddings to compare the acronym definition with the context of the abbreviation always has a big advantage over a simple word overlap measure, like cosine similarity. Surprisingly, this is not the case if we use pre-trained vectors. When we train a model on the same corpus from which we have collected our examples (excluding the parts, however, used for the examples), we get much better results indeed. This shows in the first place, that unsupervised disambiguation of acronyms is feasible, if we use word embeddings to represent the alternative definitions and the context. In the second place we have shown that training word embeddings on a specialized corpus with text comparable to those in the task to be carried out, has a big advantage over using available embeddings trained on a general corpus, even if the used corpus is much smaller than the general corpus.

6.1 Error analysis

Our data set is generated automatically and not manually corrected. Therefore, we have some errors in our data set that cause misclassifications. E.g., there are two definitions for DJF: *December January February* and *Dec Jan Feb*. Obviously, the first one is the correct one, but the classifier might find either. In some cases different but equivalent definitions are found for an acronym, like *Gap Junction Channel* and *Gap Junctional Communication* for GJC, or the same acronym is used for slightly different, but not

Table 4: Accuracy of acronym disambiguation using small context for complete models and models with restricted set of word embeddings.

Model	*all words*	*intersection*
Distr. Sim. – Google News	0.71	0.64
Distr. Sim. – Own vectors	0.84	0.83

identical concepts, like IPSO for *Improved Particle Swarm Optimization* and *Iteration Particle Swarm Optimization*.

Table 3 gives two typical examples of errors. The first example has a very misleading context. While words like *water* and *ocean* are not very typical for cell experiments, at the same time the word day, that is part of the alternative definition, occurs several times in the context. The second example is a typical case for acronyms from related disciplines and a context that is too short and general to find the correct definition.

Our impression is, that about 2% to 5% of the examples are erroneous, problematic or simply do not have enough information to disambiguate the acronym. Thus the upper bound for the accuracy would be around 0.95. Our result of 0.86 is already very close to this.

6.2 Quality of embeddings vs. number of embeddings

Finally we wanted to see whether we get better results because we have word embeddings for more (corpus specific) words, or because the word embeddings trained on the specialized corpus are better suited for our purpose. To get more insight in this question, we repeated the tests in Table 2 using only words that have an embedding in our model and in the Google News model. There are only 60,940 tokens that are shared among both models. The huge drop in the amount of words in our model (originally embeddings for 1,367,648 tokens) can be explained by the fact that we computed embeddings for many number, symbols, etc. The results for the repeated experiment using only words common to both models are given in Table 4. We see, that there is hardly any change in the results for our model. Surprisingly, the Google News Model seems to loose a number of important words. Thus we can conclude that the better results can be explained by the fact that the word embeddings in the specialized model better reflect the meaning of the words in our corpus of scientific papers.

6.3 Results with high confidence

Figure 4 suggests that we can achieve an accuracy of over 0.95 if we only consider cases in which the difference between the cosine similarities is at least 0.1. (Of course the exact value should be determined using an independent data set). This way we can still classify over 76% of all examples. Thus we can resolve a large number of acronyms in our database. However, we have to keep in mind that for most definitions just one example could be found. Thus, the most likely definition of any unresolved acronym is a definition that we have not seen up to now.

7 Conclusion and Future Work

We started our experiment with a very simple method to find definitions of acronyms in scientific papers. Using definitions from other papers we can resolve many unresolved definitions. Nevertheless, an obvious way to improve the overall results of acronym expansion would be to improve the methods to find definitions.

From our point of view the presented method has two aspects that should be improved in future work. While we use advanced methods to compute word representations, for a text we simply take the (weighted) average of the word representations. We will learn text representations with neural nets in future as well. A consequent next step then would be to learn text similarity as well.

The present result encourages us to follow this way, since we have shown (1) that training word embeddings on our own corpus gives much better results than using standard word vectors and that

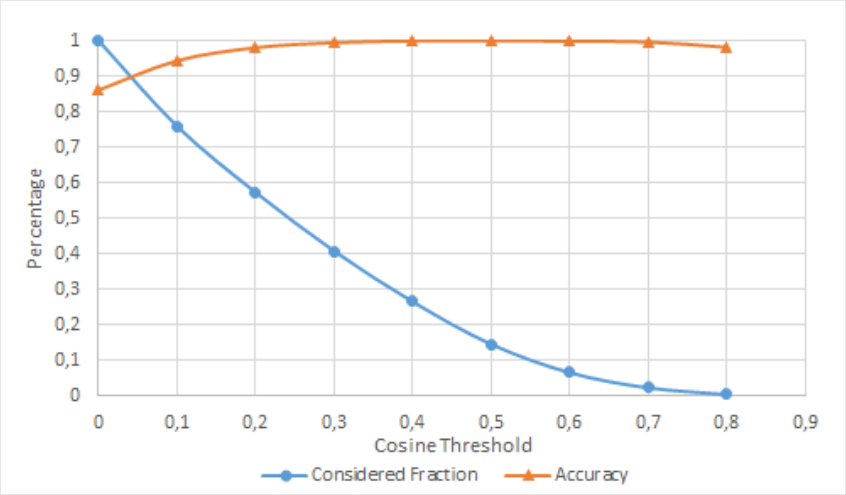

Figure 4: Fraction of classified pairs and accuracy when only examples with a difference between the cosine values above a given threshold are considered.

(2) we can obtain very good results for domain independent acronym disambiguation using these word embeddings.

Acknowledgments

The presented work was developed within the NOA Project - Automatic Harvesting, Indexing and Provision of Open Access Figures from the fields of Engineering and Technology Using the Infrastructure of Wikimedia Commons and Wikidata - funded by the DFG under grant number 315976924. NOA is a cooperative project of the Hochschule Hannover and the Technische Informationsbibliothek Hannover.

References

Rosa Tsegaye Aga, Christian Wartena, and Michael Franke-Maier. 2016. Automatic Recognition and Disambiguation of Library of Congress Subject Headings. In *Research and Advanced Technology for Digital Libraries - 20th International Conference on Theory and Practice of Digital Libraries, TPDL 2016, Hannover, Germany, September 5-9, 2016, Proceedings*, pages 442–446.

Rui Antunes and Sérgio Matos. 2017. Biomedical Word Sense Disambiguation with Word Embeddings. In Florentino Fdez-Riverola, Mohd Saberi Mohamad, Miguel Rocha, Juan F. De Paz, and Tiago Pinto, editors, *11th International Conference on Practical Applications of Computational Biology & Bioinformatics*, pages 273–279. Springer International Publishing, Cham. DOI: 10.1007/978-3-319-60816-7_33.

Satanjeev Banerjee and Ted Pedersen. 2003. Extended gloss overlaps as a measure of semantic relatedness. In *Ijcai*, volume 3, pages 805–810.

Jeffrey T Chang, Hinrich Schütze, and Russ B Altman. 2002. Creating an online dictionary of abbreviations from medline. *Journal of the American Medical Informatics Association*, 9(6):612–620.

Jean Charbonnier, Lucia Sohmen, John Rothman, Birte Rohden, and Christian Wartena. 2018. Noa: A search engine for reusable scientific images beyond the life sciences. In Gabriella Pasi, Benjamin Piwowarski, Leif Azzopardi, and Allan Hanbury, editors, *Advances in Information Retrieval*, pages 797–800, Cham. Springer International Publishing.

Gregory P Finley, Serguei VS Pakhomov, Reed McEwan, and Genevieve B Melton. 2016. Towards Comprehensive Clinical Abbreviation Disambiguation Using Machine-Labeled Training Data. *AMIA Annual Symposium Proceedings*, 2016:560–569.

Sam Henry, Clint Cuffy, and Bridget McInnes. 2017. Evaluating feature extraction methods for knowledge-based biomedical word sense disambiguation. In *BioNLP 2017*, pages 272–281. Association for Computational Linguistics.

Antonio J. Jimeno-Yepes, Bridget T. McInnes, and Alan R. Aronson. 2011. Exploiting MeSH indexing in MEDLINE to generate a data set for word sense disambiguation. *BMC Bioinformatics*, 12(1):223, Jun.

Katrin Kirchhoff and Anne M Turner. 2016. Unsupervised resolution of acronyms and abbreviations in nursing notes using document-level context models. *EMNLP 2016*, page 52.

Michael Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In *Proceedings of the 5th annual international conference on Systems documentation*, pages 24–26. ACM.

Rada Mihalcea. 2011. Word sense disambiguation. In U. S. Springer, editor, *Encyclopedia of Machine Learning*, pages 1027–1030.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

David Nadeau and Peter D. Turney, 2005. *A Supervised Learning Approach to Acronym Identification*, pages 319–329. Springer Berlin Heidelberg, Berlin, Heidelberg.

Roberto Navigli. 2009. Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, 41(2):10.

Dieke Oele and Gertjan Van Noord. 2017. Distributional lesk: Effective knowledge-based word sense disambiguation. In *IWCS 2017 — 12th International Conference on Computational Semantics — Short papers*.

Naoaki Okazaki and Sophia Ananiadou. 2006. Building an abbreviation dictionary using a term recognition approach. *Bioinformatics*, 22(24):3089–3095, December.

Serguei VS Pakhomov, Greg Finley, Reed McEwan, Yan Wang, and Genevieve B Melton. 2016. Corpus domain effects on distributional semantic modeling of medical terms. *Bioinformatics*, 32(23):3635 – 3644.

Siddharth Patwardhan, Satanjeev Banerjee, and Ted Pedersen. 2003. Using measures of semantic relatedness for word sense disambiguation. In *Computational linguistics and intelligent text processing*, pages 241–257. Springer.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Ariel S. Schwartz and Marti A. Hearst, 2011. *A simple algorithm for identifying abbreviation definitions in biomedical text*, pages 451–462. World Scientific.

Lucia Sohmen, Jean Charbonnier, Ina Blümel, Christian Wartena, and Lambert Heller. 2018. Figures in open access scientific publications. In *Research and Advanced Technology for Digital Libraries - 22nd International Conference on Theory and Practice of Digital Libraries, TPDL 2018, Porto, Portugal, September 10-13, 2018, Proceedings*. to appear.

Mark Stevenson, Yikun Guo, Abdulaziz Alamri, and Robert Gaizauskas. 2009. Disambiguation of biomedical abbreviations. In *Proceedings of the BioNLP 2009 Workshop*, pages 71–79. Association for Computational Linguistics.

Stephan Tulkens, Simon Suster, and Walter Daelemans. 2016. Using distributed representations to disambiguate biomedical and clinical concepts. In *Proceedings of the 15th Workshop on Biomedical Natural Language Processing*, pages 77–82. Association for Computational Linguistics.

Thi Ngoc Chau Vo, Tru Hoang Cao, and Tu Bao Ho. 2016. Abbreviation Identification in Clinical Notes with Level-wise Feature Engineering and Supervised Learning. In Hayato Ohwada and Kenichi Yoshida, editors, *Knowledge Management and Acquisition for Intelligent Systems : 14th Pacific Rim Knowledge Acquisition Workshop, PKAW 2016, Phuket, Thailand, August 22-23, 2016, Proceedings*, pages 3–17. Springer International Publishing, Cham. DOI: 10.1007/978-3-319-42706-5_1.

Yue Wang, Kai Zheng, Hua Xu, and Qiaozhu Mei. 2016. Clinical word sense disambiguation with interactive search and classification. In *AMIA Annual Symposium Proceedings*, volume 2016, page 2062. American Medical Informatics Association.

Yonghui Wu, Jun Xu, Yaoyun Zhang, and Hua Xu. 2015. Clinical abbreviation disambiguation using neural word embeddings. In *Proceedings of the 2015 Workshop on Biomedical Natural Language Processing (BioNLP)*, pages 171–176.

Yonghui Wu, Joshua C Denny, S Trent Rosenbloom, Randolph A Miller, Dario A Giuse, Lulu Wang, Carmelo Blanquicett, Ergin Soysal, Jun Xu, and Hua Xu. 2017. A long journey to short abbreviations: developing an open-source framework for clinical abbreviation recognition and disambiguation (card). *Journal of the American Medical Informatics Association*, 24(e1):e79–e86.

David Yarowsky. 2010. Word Sense Disambiguation. In Nitin Indurkhya and Fred J. Damerau, editors, *Handbook of Natural Language Processing, Second Edition*, pages 315–338.