CMPUT 600 Project Report: Classifying Words into Homonyms or Polysemes Using WordNet Sense Definitions

Afia Anjum
Department of Computing Science
University of Alberta

Abstract

Humour recognition or pun disambiguation systems depend upon the performance of a homonym classification task. We utilize WordNet synsets and their definitions for a given word to classify it into either Homonym or Polyseme. Two counter approach techniques such as removal of similar synset definitions and grouping of similar synsets of a word are applied to evaluate the performance of such a task. Besides, WordNet synonymy and hypernymy relations are also leveraged to check the performance. The errors from these models are investigated with the help of some random samples. It is evident from our analysis that, a model utilizing WordNet path similarity values will hold significant advantages in classifying a word into a homonym or a polyseme. Moreover, the fine-grained nature of WordNet makes this task very complex even if the synonymy and hypernymy relations of WordNet are utilized.

1 Introduction

Homonym detection is the task of identifying a word, having multiple semantically unrelated senses. Consequently, all these senses depict the same set of words that are identical in spelling but vary in origins and meanings. The relationship between two such semantically unrelated senses of a word is defined as homonymy, otherwise, it is known as polysemy (Jurasky and Martin, 2009). The units of language that are represented in the lexicon are called Lexemes (Murphy and Koskela, 2010). Sometimes, the association between lexemes is also used to distinguish between homonymous and polysemous words. A homonymous word (e.g. noun sense of 'yard' in Table 1) represents more than one lexeme, each of which is a homonym. Semantically related senses (e.g. #1 and #5 in Table 1) of a homonymous word will belong to the same lexeme and are thus, polysemous, while, semantically unrelated senses (e.g. #1 and #2 in Table 1) of such word will belong to a more than one lexeme and are thus, homonymous (Hauer and Kondrak, 2020a).

YARD_n^1	YARD_n^2
#1 unit of length	#2 land around a house
#5 unit of volume	#3 tract of land (for repair)
#8 long horizontal spar	#6 tract of land (gather logs)
	#7 railway yard
	#9 enclosure for livestock

Table 1: Senses of the noun 'yard' grouped by its two homonym utilizing WordNet 3.0

Our study, in particular, focuses on extracting WordNet (Miller, 1998) sense definitions for a word and then utilizes these definitions for our homonym-polyseme classification task. It can also enhance the performance of other Natural Language Processing (NLP) systems, such as machine-assisted interpretation and translation system (Grif and Manueva, 2018), humor recognition system (van den Beukel and Aroyo, 2018) and pun disambiguation system (Miller and Gurevych, 2015).

There have been various research done so far in this field, however, the existing methods are insufficient to enhance the performance of the aforementioned NLP systems. Additionally, the standard resources that are used to identify homonyms suffer from various drawbacks (e.g. fine-grained nature of WordNet, etc). Hence, the goal of this research is to contribute to this task primarily by replicating and validating the homonym detection technique used in (van den Beukel and Aroyo, 2018) with similar dataset and settings. In addition to replication, we propose various improvements in the settings as well as leverage WordNet synonymy and hypernymy relations to enhance the performance for homonym classification tasks on our own dataset. Finally, we investigate and analyze the errors from all these methods to understand their limitations.

2 Related work

Advanced research has been done in creating a list or dictionary of English homonyms (Parent, 2012; Rothwell, 2007) or a list of homonyms based on their etymological origins (Hauer and Kondrak, 2020a). Besides, some researchers (van den Beukel and Aroyo, 2018) have tried to enhance the performance of a humor recognition task in one-liners by utilizing the WordNet sense definitions. Moreover, some researchers (Miller and Gurevych, 2015) proposed automatic disambiguation using the Lesk algorithm to overcome the issue of ambiguity in English puns.

Observing all of this research, it is evident that the performance of the homonym classification task can be amplified if resources like WordNet are associated with complex similarity measurement techniques to distinguish between the extracted definitions.

3 WordNet

WordNet is a lexical database that consists of words with different parts of speech (pos) tags and are grouped into sets of synonyms called the synsets. Each synset expresses a distinct concept. WordNet has around 1,17,000 synsets which are linked by semantic and lexical relations. In this way, it forms a network of meaningfully related words and concepts. Additionally, a synset contains a brief definition (gloss) and a short example sentence to demonstrate the use of synset members. Besides, every word sense belongs to exactly one synset and distinct synsets will not contain synonymous senses (Hauer and Kondrak, 2020b). Relations like hypernymy, hyponymy, meronymy, etc are also present in WordNet in addition to synonymy. IS-A relation is used to express the hypernym and hyponym relation between the synsets. For example, the noun form of the word 'yard' has many synsets (e.g. yard.n.01, c_yard.n.01, yard.n.02, etc) in WordNet. The hypernym of the synset 'yard.n.01' is another synset expressed by 'linear_unit.n.01', while the former is a hyponym of the later. Moreover, each synset can contain one or more lemmas that can represent a particular sense of a particular word. It is to be noted that, our homonym classification method is based on a simple assumption that, for any two given tokens, a similarity exists if they have any synset in common or if they share the same hypernym or both.

4 Methodology

Our approach to solve the homonym classification task is based upon the homograph detection technique of (van den Beukel and Aroyo, 2018). Unless otherwise specified, we will refer to this work as BA18 in the entire paper.

4.1 BA18 Task

The initial step of BA18 includes the retrieval of all WordNet synset definitions for a given word. The definitions are tokenized and stop words are removed. Then, the similarity is measured between every two synsets in terms of path similarity (PS) in the WordNet graph and by calculating word overlap (with Jaccard similarity coefficient (JC)) utilizing the tokenized definition lists. A similarity score greater than a particular threshold is used to remove the corresponding definition or gloss for the second synset. All the above steps are repeated for all the retrieved synsets. Finally, if the remaining number of definitions is more than one then the word is classified as homonym. An annotated dataset is used to measure the performance of the homonym recognition approach. All the above steps were applied and similar results were obtained, which also suggests the replicability of BA18.

4.2 Caveats of BA18

While implementing the BA18 task with its original settings, we observed some drawbacks. BA18 method looks at the total number of remaining definitions by removing them one by one. However, similar synsets should be grouped rather than removing definitions all together. Additionally, pos tags are not taken into consideration while classifying a word as a homonym. Besides, the implementation only considers the lemmas of a synset which has an exact match with the actual form of the candidate word to classify. This, in turn, will ignore many synsets, which might affect the overall performance. Finally, there is no clarification regarding the selected threshold which is used in order to measure similarity.

4.3 Our approach

Inspired by the homograph detection technique mentioned above, we have designed our methodology in the following ways:

1. In this method, we follow the exact steps mentioned in section 4.1 taking the pos tag for a word into consideration. Additionally, instead of removing definitions, we draw a graph where synsets represent vertices and an edge between two synsets represents the similarity between them. If the number of the connected components in the graph is greater than one, then the word is classified as a homonym, otherwise polyseme.

2. The main idea of the second method is to modify the BA18 methodology using complicated similarity techniques. Given two tokenized and stop word removed definition lists, we check for shared words between them. If they do not have any words in common, we will look for any synsets in common. If such synsets are not found, we will look if they share the same hypernym. Similar to step 1, a graph is obtained connecting the synsets if they hold a similarity. The number of the connected components from the resulting graph is used to classify the word as either homonym or polyseme.

For example, the tokenized and stop word removed WordNet glosses for the synsets 'yard.n.02' and 'yard.n.03' are shown below:

#2	enclosed,land,house,building
#3	tract,land,enclosed,particular,activities

For every two tokens in between every two synsets, we will follow the above-mentioned steps and evaluate our algorithm performance.

4.4 Input & Output

The input to system is a ***list of words tagged with its pos***, while the output is it's classification either into ***homonym*** or ***polyseme*** and a ***graph depicting the relations between the retrieved WordNet synsets for that word*** with the help of vertices and edges.

The following example represents two of the retrieved WordNet synset glosses for a word (e.g. yard#n) in the input list:

1. yard - (the enclosed land around a house or other building).

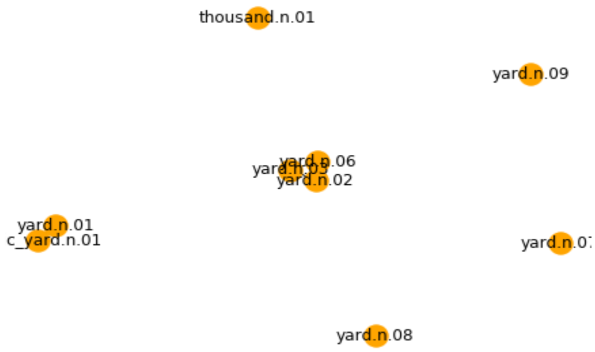

Figure 1: Similar synsets of the noun 'yard' grouped together according to similarity

2. yard - (a tract of land enclosed for particular activities).

Finally, the output for this classification task would be: **Homonym** and the graph in Figure 1.

5 Evaluation

5.1 Datasets

The dataset mentioned in BA18 paper is used in section 4.1. This annotated dataset consists of 301 sentences from numerous jokes websites, Reuters news headlines, English proverbs, and Wikipedia sentences. Every word of each sentence is annotated with two annotators. Their agreement score is calculated from which a total of 248 words are labeled as homonyms. In addition to this dataset, we created development and test dataset with an equal number of homonymous (100) and polysemous (100) words along with their parts of speech tag. The homonymous words are randomly collected from Type-A homonym list[1]. Whereas, the words, for which more than one clusters are present in OntoNotes and are also absent in the Type-A homonym list, are used to construct polysemous examples in our datasets. The development set is used during the model development and parameter tuning and the test set is used for measuring the final performance of our task.

5.2 Evaluation metric

The evaluation metric for the BA18 system are the Precision, Recall and F-score values. Whereas our methods discussed in section 4.3 utilize a balanced dataset and hence Accuracy score is utilized to evaluate the performance. Additionally,

[1]https://webdocs.cs.ualberta.ca/ kondrak/homonyms.html

a simple baseline, based on the number of WordNet synsets, is also created to compare the performance of our approach. If the number of synsets for a word is greater than four, then it is chosen as a homonym by the baseline, otherwise polyseme. The baseline accuracy score for our development set is 56.5%.

5.3 Parameter tuning

The performance of the homonym recognition task of BA18 on their annotated dataset is evaluated using the first method mentioned in section 4.3. We name the resultant system as M1. Comparing the F1 scores of BA18 and M1 in Table 2, we can say that the homonym detection technique which involves removal of the definition has similar performance to the technique which groups similar synsets in a graph. The performance of the BA18 task is also measured by varying different parameters of the original setting. The best performance (BA18best) is achieved by keeping PS=JC=0.1 and by putting a restriction on the lemmas of the synsets. A similar result is also observed when the system utilized our first method (M1best).

Table 3 depicts the settings selected by tuning the development set in order to run the test dataset. Here, M1 and M2 represent the two methods discussed in section 4.3. Additionally, the second method is performed by checking for shared words (SW), synonymy (S) and hypernymy (H) individually and with a combination. The letter 'N' means lemmas of a synset are not restricted (all lemmas) and 'Y' means they are restricted. For the second method, a shared word combined with the restriction on the lemma of the synsets technique performs the best.

5.4 Results on test set

The results of our test set (Table 4) is obtained following the settings which maximized the accuracy scores(in bold) in Table 3. For the second method, the setting selection remains unclear. However, in our case, shared word combined with synonymy and all lemmas of synsets technique performs the best. In most of the cases, they outperform the constructed baseline.

5.5 Error Analysis

In order to understand the performance of these models clearly, we looked at 10-15 incorrectly identified words that utilized the above-mentioned methods. It has been observed that BA18 and M1 cannot detect more than 20 homonymous words but only around 15 polysemous words suffer from misclassification. On the other hand, M2(SW+S) does a very good job in correctly classifying 92 homonymous words but around 58 polysemous words suffer from misclassification. Besides, the fine-grained nature of WordNet makes the correct identification of many words e.g. worship#v, etc. impossible. This is a polysemous word, but WordNet gloss for two of its synsets are totally different. This suggests that looking only at WordNet glosses might not be the best idea. Additionally, in different experiments on the development and test set, it has been observed that PS alone is able to provide the same accuracy score even when the JC calculation is removed. And the use of only JC without PS makes the model's performance worse than the constructed baseline. This suggests that the utilization of WordNet path similarity provides an advantage to the models BA18 and M1. And this is why our second method is unable to outperform them.

Table 2: Performance of BA18 task

	BA18	M1	BA18best	M1best
Pre.	0.354	0.349	0.388	0.351
Rec.	0.826	0.824	0.751	0.689
F1-sc.	0.496	0.490	**0.511**	**0.465**
Acc.	0.746	0.741	0.783	0.760

Table 3: Experimental settings using dev. set

	PS	JC	Lem. Rest.	Acc.
BA18	0.1	0.1	N	**0.73**
M1	0.1	0.1	N	**0.74**
M2(SW+S)	-	-	Y	**0.615**
M2(SW)	-	-	Y	**0.615**

6 Conclusion

We have investigated the potential drawbacks of a homonym detection task by means of replication and validation. We have performed various modifications to the task in terms of settings and

Table 4: Results on test set

	PS	JC	Lem. Rest.	Acc.
BA18	0.1	0.1	N	**0.80**
M1	0.1	0.1	N	**0.785**
M2(SW+S)	-	-	N	**0.67**
M2(S)	-	-	N	**0.665**

parameters to check it's efficiency on our dataset. Besides, this task serves as a foundation in designing our two other methodologies. We have performed evaluations of our methods on a test set and analyzed the errors by selecting some random instances. Future work includes determining the effectiveness of our selected methods by deriving the number of homonyms for a given homonymous word. Secondly, the fine-grained nature of WordNet makes the classification task skewed towards identifying most of the words as homonyms. Therefore, a course-grained inventory could also be used to analyze the outcome.

Acknowledgments

This report and the project itself was done by consulting Amir Ahmad Habibi. We would like to thank him for providing us with the data needed to develop and test our models. We would also like to thank Professor Greg Kondrak and his NLP team for their feedback and guidance on making our project better.

References

Mikhail G Grif and Julia S Manueva. 2018. The translation of sentences from russian language to russian sign language after homonymy removal. In *2018 XIV International Scientific-Technical Conference on Actual Problems of Electronics Instrument Engineering (APEIE)*. IEEE, pages 421–425.

Bradley Hauer and Grzegorz Kondrak. 2020a. One homonym per translation. *arXiv preprint arXiv:1904.08533v2[cs.CL]* .

Bradley Hauer and Grzegorz Kondrak. 2020b. Synonymy = translational equivalence .

D. Jurasky and J. H. Martin. 2009. Speech and language processing. *Prentice Hall, 2nd edition.* .

George A Miller. 1998. *WordNet: An electronic lexical database*. MIT press.

Tristan Miller and Iryna Gurevych. 2015. Automatic disambiguation of english puns. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. pages 719–729.

M Lynne Murphy and Anu Koskela. 2010. *Key terms in semantics*. A&C Black.

Kevin Parent. 2012. The most frequent english homonyms. *RELC Journal* 43(1):69–81.

David Rothwell. 2007. *Dictionary of homonyms*. Wordsworth Editions.

Sven van den Beukel and Lora Aroyo. 2018. Homonym detection for humor recognition in short text. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*. pages 286–291.