Heteronym Sense Linking

Lenka Bajčetić[1], Thierry Declerck[1,2], John P. McCrae[3]

[1]Austrian Centre for Digital Humanities and Cultural Heritage
Sonnenfelsgasse 19, Wien 1010, Austria
[2]German Research Center for Artificial Intelligence (DFKI)
Saarland Informatics Campus D3 2, Saarbrücken, Germany
[3] Data Science Insitute, NUI Galway, Ireland
E-mail: lenka.bajcetic@oeaw.ac.at, declerck@dfki.de, john@mccr.ae

Abstract

In this paper we present ongoing work which aims to semi-automatically connect pronunciation information to lexical semantic resources which currently lack such information, with a focus on WordNet. This is particularly relevant for the cases of heteronyms — homographs that have different meanings associated with different pronunciations — as this is a factor that implies a re-design and adaptation of the formal representation of the targeted lexical semantic resources: in the case of heteronyms it is not enough to just add a slot for pronunciation information to each WordNet entry. Also, there are numerous tools and resources which rely on WordNet, so we hope that enriching WordNet with valuable pronunciation information can prove beneficial for many applications in the future. Our work consists of compiling a small gold standard dataset of heteronymous words, which contains short documents created for each WordNet sense, in total 136 senses matched with their pronunciation from Wiktionary. For the task of matching WordNet senses with their corresponding Wiktionary entries, we train several supervised classifiers which rely on various similarity metrics, and we explore whether these metrics can serve as useful features as well as the quality of the different classifiers tested on our dataset. Finally, we explain in what way these results could be stored in OntoLex-Lemon and integrated to the Open English WordNet.

Keywords: Sense Linking; Heteronyms; Wordnets; Wiktionary

1. Introduction

There are many types of ambiguity in language, and one interesting example are homographs. These are words that are spelled the same, but they have different pronunciations. Specifically, homographs that have different meanings associated with different pronunciations, are called heteronyms (Martin et al., 1981).

Heteronyms can cause great challenges for speech-to-text and text-to-speech systems. They also provide an interesting use-case for our endeavour to enrich WordNet with pronunciation information.

Recently, the Global WordNet Association (GWA) updated its Global Wordnet Formats (McCrae et al., 2021)[1], which have been introduced to enable wordnets to have a common representation. One of the updates performed by GWA concerns the possibility to add pronunciation information to the entries of wordnets. GWA decided to “support the use of IETF language tags to indicate dialect”. This update is a great step towards integrating pronunciation information in wordnets.

As a complementary task to the representation of heteronymy in wordnets, we start with the task of supporting an automated linking between the senses of the heteronyms we extracted from Wiktionary and those included in the Open English WordNet (McCrae et al., 2020). While the sense linking task is in itself interesting Ahmadi & McCrae (2021), it can lead to an automated addition of the pronunciation information to the heteronyms included in English WordNet. Since English WordNet is a manually curated gold standard resource, this would lead to the possibility to get an evaluation of the linking work for this

[1] https://globalwordnet.github.io/schemas/

specific type of phenomenon and also to the building of a training set for an extension of the linking work.

2. Related Work

In order to be valuable, language resources should be accessible and legal to use, sufficient in terms of quality and size, and ideally with a documented interface (Ishida, 2006). According to these aspects, both WordNet and Wiktionary are language resources of the highest value, and it is no surprise there are many endeavours aimed at connecting the two. For instance, the work of Meyer & Gurevych (2011) shows that automatic alignments between Wiktionary senses and Princeton WordNet can be established by combining several text similarity scores to compare a bag of words based on several pieces of information linked to a WordNet sense with another bag of words obtained from a Wiktionary entry. This is quite similar to the approach we have followed also, as explained in the Method section. A large part of this work is also harnessing the multilingualism of the two resources, in an attempt to create very large multilingual corpora by aligning several Wiktionaries and WordNets.

Our previous work on heteronyms is presented in (Declerck & Bajčetić, 2021). This work consisted of extracting entries from Wiktionary that carry pronunciation information (following suggestions made by Schlippe et al. (2010)), for the four categories that are relevant for WordNet: nouns, verbs, adjectives, and adverbs. The result of this procedure consisted of listing each heteronymous word, together with its pronunciations, associated with their respective meanings and related example sentences. Declerck & Bajčetić (2021) propose a first representation of such entries using the OntoLex-Lemon model (Cimiano et al., 2016), suggesting a deduplication of lexical entries on the base of their different pronunciations, if those are related with specific meanings.

3. Method

When designing our linking approach[2], we have decided to pose our task as a classification problem. First we have created a dataset which consisted of the correct matches from the gold standard, and added their incorrect counterparts. In the end, we are left with a dataset of 272 examples labelled ‘True’ or ‘False’ depending on the matching. This means we have effectively transformed our sentence similarity task into a binary classification problem. While binary classification can be tackled in many ways, we have decided to experiment with supervised classifiers which were trained using various sentence similarity metrics.

3.1 Gold standard

In order to test and train our classifiers, we have compiled a small dataset which covers 10 examples of heteronymous words. The dataset consists of 136 WordNet senses matched with their pronunciation as stored in Wiktionary.

In the future we consider using the lists compiled by Martin et al. (1981). The authors have compiled an extensive list of 54 strong and 62 weak heteronyms. They came up

[2] The code and data are available here: https://github.com/acdh-oeaw/heteronym_sl

Word	Pronunciation 1	Pronunciation 2	N° of senses
bass	bæs	beɪs	9
bow	baʊ	boʊ	14
desert	dɪˈzɛːt	ˈdɛzət	4
house	haʊs	haʊz	14
lead	lɛd	liːd	31
live	lɪv	laɪv	19
raven	ˈɹeɪvən	ˈɹævən	5
row	ɹaʊ	ɹəʊ	10
subject	ˈsʌb.dʒɛkt	səbˈdʒɛkt	15
wind	wɪnd	waɪnd	15

Table 1: Gold standard

with this classification to reflect the distinctiveness of meaning between two senses which have different pronunciations. For example, "subject" is considered an example of weak heteronym, because the differently pronounced senses denote the same concept in verb and noun form. On the other hand, "row" is considered a strong heteronym, since the meanings it conveys are completely different. According to this classification, our list has examples of 3 weak heteronyms: "live", "house", and "subject".

3.2 Sense Linking

In order to parse Wiktionary files we extract headwords, parts of speech, definitions, examples and of course the pronunciation info from the XML Wiktionary database dumps as provided by the Wikimedia Foundation. The main body of Wiktionary articles are stored in a Wikitext format, which is a semi-structured format. Each article is centred around the "Etymology" section, and words which have several meanings have several etymologies. After extracting and packing all the relevant information from the Wiktionary article, we are left with several documents — one for each of the etymologies. For simplicity, we have chosen to work with those examples that have two possible etymologies, which in our case translates to two possible pronunciations, so our task can be understood as binary classification.

For each of the words, we retrieve all the senses from WordNet. Then, for each sense we extract the synonyms with their definitions, examples, and the hypernym hierarchy. By combining this information we create a short document for each sense. Finally, for each of the pairings of WordNet senses and their two corresponding Wiktionary articles, we have provided a final label of True or False. This means that our training dataset consists of 272 examples, half of which are correctly linked.

3.3 Features

The classifiers rely on five features:

- Wiktionary POS
- WordNet POS

- S-BERT similarity score
- Laser similarity score
- TFIDF similarity score

We have decided to use the POS tags because this is an intuitive and easy idea. This feature has proven useful, but less so in comparison with the similarity metrics.

In order to get the S-BERT similarity score, we use the cosine distance of the sentence embeddings from a transformer model which is pre-trained for paraphrase identification (paraphrase-distilroberta-base-v1) and a model which is pre-trained for semantic textual similarity (stsb-roberta-base). Sentence-BERT is a modification of the pretrained BERT network which uses Siamese and triplet network structures to derive semantically meaningful sentence embeddings that can be easily compared using cosine-similarity (Reimers & Gurevych, 2019). Sentence transformers are the current state-of-the-art approach in Semantic Textual Similarity (STS) tasks, and they perform very competitively in all sentence similarity tasks. The performance of S-BERT is evaluated for common STS tasks using the cosine-similarity to compare the similarity between two sentence embeddings, which is exactly the approach we have followed also.

The LASER similarity score is obtained in the same way, with the distinction of using LASER sentence embeddings[3]. LASER stands for Language-Agnostic SEntence Representations, and it uses a single pre-trained BiLSTM encoder for 93 languages, obtaining very strong results in various scenarios without any fine-tuning, including cross-lingual textual similarity (Artetxe & Schwenk, 2018). Since we intend to expand this research to other languages, we have decided it is important to explore multilingual options as well as English language specific ones, despite the fact LASER scores on monolingual tasks are usually not as good as the ones obtained using monolingual BERT-based sentence transformers (Artetxe & Schwenk, 2018).

Finally, the TFIDF similarity score is created by following the approach laid out by Meyer & Gurevych (2011). In their work, they utilize cosine distance between bag-of-words vectors as a similarity measure. The cosine similarity calculates the cosine of the angle between a vector representation of the two senses s1 and s2:

$$COS(s1, s2) = \frac{BoW(s1) \cdot BoW(s2)}{||BoW(s1)||||BoW(s2)||}$$

Following this approach, for each word in the gold standard we simply create a corpus of short documents explaining senses from WordNet, and we create pairs of Wiktionary documents which explain the two different pronunciations. Then, we calculate the value of the cosine distance for all document combinations.

3.4 Classifiers

After feature extraction, data is split into training and testing subsets with 2:1 ratio, and we use it to train several simple classifiers from sklearn[4]:

[3] https://github.com/yannvgn/laserembeddings

[4] All the classifiers can be found here: https://scikit-learn.org/stable/supervised_learning.html

- Naive Bayes
- Decision Tree
- Random Forest

Naive Bayes methods are a set of supervised learning algorithms based on applying Bayes' theorem with the "naive" assumption of conditional independence between every pair of features given the value of the class variable. Despite this over-simplified assumption, naive Bayes classifiers have performed quite well in many tasks, especially document classification and spam filtering. They require a small amount of training data to estimate the necessary parameters, and for this reason we have decided to try it. However, this model has not proven so good in our task, and consistently achieved scores lower than other classifiers.

Decision Trees are a non-parametric supervised learning method used for classification and regression. The model aims to predict the value of a target variable by learning simple decision rules inferred from the data features. Decision trees are simple to understand and to interpret, and they require little data preparation. As we can see in the Results section, this model has shown good results but not as good as the random forest classifier.

A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. Since the random forest classifier has proven to have the best results, we have decided to fine-tune it by trying different sets of parameters. More specifically, we explored different values for the number of trees in the forest (estimators) and the maximum number of levels in each decision tree (max depth). First, we used GridSearch from sklearn library to determine the best set of parameters from Table 2, and then we trained several classifiers with those parameters, but experimenting with different number of estimators and max depth. A graph of the classifiers' accuracy depending on the hyperparameters value is shown in Figure 1 and Figure 2.

The results so far do not show clearly which is the best parameter set. This is most likely due to the small training set which is the biggest limitation of our model. Before obtaining a larger set it is hard to get definitive results which is the best model, we can only notice some trends regarding the potential shown by some features or classifiers.

Parameter	Values
bootstrap	True, False
max features	auto, sqrt
min samples leaf	1, 2, 4
min samples split	2, 5, 10
max depth	None, 2, 4, 6, 8, 10, 15, 20, 30, 40, 50, 60, 70, 80, 90, 100
n estimators	5, 10, 15, 20, 50, 100, 150, 200, 400, 600, 800, 1000

Table 2: Different parameters tried for Random Forest

3.5 Connecting to Open English WordNet

Open English WordNet (McCrae et al., 2020) is an open-source fork of the Princeton WordNet (Miller, 1995). The wordnet is freely available on GitHub and is formatted according to the XML schemas defined at https://globalwordnet.github.io/schemas/ (McCrae et al., 2021). Currently, there are no distinctions between heteronyms in WordNet, so these would need to be introduced. As a result of adopting the XML schema, it is now possible to define two lexical entries with the same lemma that was not possible in the previous formats used by Princeton WordNet. In addition, recently support was added for indicating the pronunciation in the schema files. An example of this new modelling is as follows:

```
<LexicalEntry id="ewn-bass-n-1">
  <Lemma writtenForm="bass" partOfSpeech="n">
    <Pronunciation notation="fonipa">bæs</Pronunciation>
  </Lemma>
  <Sense id="bass%1:05:00::"
         synset="ewn-02568204-n"/>
  ...
</LexicalEntry>
<LexicalEntry id="ewn-bass-n-2">
  <Lemma writtenForm="bass" partOfSpeech="n">
    <Pronunciation notation="fonipa">beIs</Pronunciation>
  </Lemma>
  <Sense id="bass%1:06:02::"
         synset="ewn-02806515-n"/>
  ...
</LexicalEntry>
```

In this example, we see two entries `ewn-bass-n-1` with a pronunciation to rhyme with 'mass' and `ewn-bass-n-2` with a pronunciation that rhymes with 'face', the senses are assigned to one of each of the entries. Note that, each of the entries are actually organized into distinct lexicographer files, so in this case it is merely the task of identifying which of the lexicographer files corresponds to which pronunciation, e.g., `noun.food` and `noun.animal` for the first pronunciation and `noun.attribute`, `noun.communication`, `noun.person`, `noun.artifact` and `adj.all` for the second.

3.6 Representation of heteronyms in OntoLex-Lemon

The work of Declerck & Bajčetić (2021) discusses the addition of pronunciation information in wordnets, with a focus on heteronyms. Those cases are particularly relevant for wordnets, as they do carry specific senses that need to be accounted for in such lexical semantics repositories. The authors make use of the OntoLex-Lemon representation model, as it has proven to be well adapted for linking the conceptual type of resources, as exemplified by wordnets, with the full lexicographic descriptions of the lemmas, which in wordnets are only minimally represented (just the written form and the associated part-of-speech). OntoLex-Lemon introduces form variants of lexical entries as full ontological objects, which can therefore carry information on a number of grammatical properties, like gender, case, and number. Those “form” objects also include the corresponding written and phonetic representations. So that (Declerck & Bajčetić, 2021) could propose a way to represent in OntoLex-Lemon the combination of wordnet entries and lexical entries, which are themselves pointing to form variants displaying the corresponding pronunciation information. The challenge would be now to extend this approach to compound words, and we are investigating for this the use of the *decomp* module of OntoLex-Lemon.[5]

4. Results

First we will compare the results of the four classifier models, namely naive Bayes, Decision Tree, and two versions of the random forest classifier. Then, we will take a closer look at the relevance of each feature for the classification. In the end, we will see how the variance in the parameter set for training the random forest classifier affects the results.

As we have previously mentioned, we employed two different pre-trained models for obtaining the S-Bert similarity feature, namely a model pre-trained on paraphrase detection and a model trained for semantic textual similarity task. As we can see in the results below, the similarity feature extracted using the paraphrase model has proven to give better results in our case. This makes sense, as our documents are usually not aligned with each other and consist of examples and definitions glued together, sometimes incoherently. It appears that for this kind of data, paraphrase detection serves as a better benchmark task than a typical STS task. Of course, we cannot know this for certain before we obtain a larger training set to experiment with.

Classifier	Accuracy	Precision	Recall	F1-score
Naive-Bayes	0.65	0.67	0.65	0.65
Decision Tree	0.71	0.70	0.71	0.71
Random Forest - STS [6]	0.79	0.80	0.79	0.79
Random Forest - Paraphrase[7]	0.84	0.85	0.84	0.84

Table 3: Performance of different classifiers on our gold standard test

In order to compare the benefits provided by different similarity metrics, we have tried using them as a basis for very simple classifiers with a threshold value. This is quite a simple, yet effective, way to investigate the capacity of each feature in our task. Since for this purpose you do not need to train a classifier, we have used the whole gold standard as the test set for this simple one-feature threshold-based classifiers. As we can see in Figure 1, both classifiers which are based on S-Bert similarity score can reach a score on our task of up to 0.7, with the right threshold value. This shows that S-Bert similarity score has great potential as a feature, even though it is not sufficient as a classifier by itself. On the other hand, the classifiers based on LASER and TFIDF similarity scores are not quite as useful to work on their own. As another way to check the usability of our features, we have also used the feature importance function from sklearn’s library. The results of this can be seen in Table 4.

[5] See https://www.w3.org/2016/05/ontolex/#decomposition-decomp for more details.

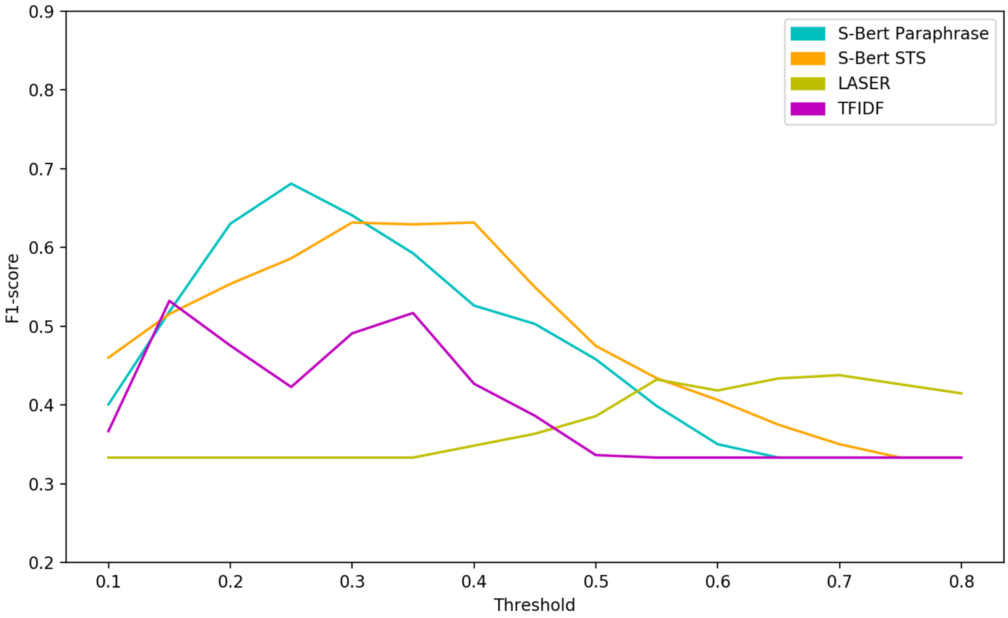

Figure 1: F1 score of different similarity metrics, depending on the threshold

Classifier	S-Bert	LASER	TFIDF	POS1	POS2
Decision Tree	0.36	0.15	0.43	0.05	0.02
Random Forest	0.39	0.23	0.28	0.06	0.04

Table 4: Relevance of different features for the classifiers

What we can see from the table is that similarity scores prove to be much more valuable predictors in comparison to the POS tags. In fact, it even looks like the POS tags can be considered irrelevant, but we have discovered that without them the results for all classifiers drop significantly (up to 10%). When it comes to similarity metrics, we see that all three of them are quite useful. Interestingly, the random forest classifier utilizes the S-Bert value most, while Decision Tree relies mostly on TFIDF. It is expected that LASER is the least useful of the three similarity metrics, due to the fact that these embeddings are multilingual, while other metrics are fine-tuned with English language in mind. Although our dataset is still quite small and the models are limited, we can assume that all three of the similarity metrics can provide valuable input to a future model.

Since we noticed that the parameters of maximum depth and the number of estimators affect the results the most, we have decided to explore them in greater length. First we use the GridSearch from sklearn to determine the best parameter set from Table 2, and then we trained several versions of the best classifier, while changing the desired two parameters. Results of this exploration can be seen in Figure 2 and Figure 3 below. We can conclude that there is no clear choice for the best value for the number of estimators nor maximum depth, but there is a distinctive trend. For maximum depth, lower values seem to perform better, which makes sense for a small dataset like ours. On the other hand, for the number of estimators very low values are not giving high performance, and neither are very high ones — the best choice are values around 200.

5. Conclusion

Since this is ongoing work, there is quite some space for future work. One of the most important things to be done next is to compile a bigger gold standard dataset, also in a multilingual setting. Another possibility to increase our dataset is to explore ways to up-sample data, or generate artificial data to increase the size of our corpus. Since the size of the data can negatively affect generalization and create difficulty in reaching the global optimum, this is an important issue when creating supervised classifiers.

A promising next step to increase the impact of our work includes handling compounds or phrasal entries in which a component is a heteronym, like for example "lead pencil". Ultimately, we hope this work will prove beneficial for handling heteronyms in text-to-speech systems as well (Henton & Naik, 2014) and (Wang et al., 2011), with the help of enriched wordnets.

6. Acknowledgements

Contributions by the German Research Center for Artificial Intelligence (DFKI GmbH) were supported in part by the H2020 project Prêt-à-LLOD with Grant Agreement number 825182. Contributions by the Austrian Centre for Digital Humanities and Cultural Heritage at the Austrian Academy of Sciences were supported in part by the H2020 project "ELEXIS" with Grant Agreement number 731015. The work described in this paper was also pursued in part in the larger context of the COST Action CA18209 — NexusLinguarum "European network for Web-centred linguistic data science". We also thank the anonymous reviewers for their insightful comments.

7. References

Ahmadi, S. & McCrae, J.P. (2021). Monolingual Word Sense Alignment as a Classification Problem. In *Proceedings of the 11th Global Wordnet Conference*. University of South Africa (UNISA): Global Wordnet Association, pp. 73–80. URL https://www.aclweb.org/anthology/2021.gwc-1.9.

Artetxe, M. & Schwenk, H. (2018). Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond. *CoRR*, abs/1812.10464. URL http://arxiv.org/abs/1812.10464. 1812.10464.

Bond, F. & Foster, R. (2013). Linking and Extending an Open Multilingual Wordnet. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Sofia, Bulgaria: Association for Computational Linguistics, pp. 1352–1362. URL https://www.aclweb.org/anthology/P13-1133.

Cimiano, P., McCrae, J.P. & Buitelaar, P. (2016). Lexicon Model for Ontologies: Community Report.

Declerck, T. & Bajčetić, L. (2021). Towards the Addition of Pronunciation Information to Lexical Semantic Resources. In *Proceedings of the 11th Global Wordnet Conference*. University of South Africa (UNISA): Global Wordnet Association, pp. 284–291. URL https://www.aclweb.org/anthology/2021.gwc-1.33.

Devlin, J., Chang, M., Lee, K. & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, abs/1810.04805. URL http://arxiv.org/abs/1810.04805. 1810.04805.

Henton, C. & Naik, D. (2014). Disambiguating heteronyms in speech synthesis. URL https://patents.google.com/patent/US9711141B2/en.

Ishida, T. (2006). Language grid: an infrastructure for intercultural collaboration.

Martin, M., Jones, G., Nelson, D. & Nelson, L. (1981). Heteronyms and polyphones: Categories of words with multiple phonemic representations. *Behavior Research Methods & Instrumentation*, 13, pp. 299–307.

McCrae, J.P., Goodman, M.W., Bond, F., Rademaker, A., Rudnicka, E. & Costa, L.M.D. (2021). The GlobalWordNet Formats: Updates for 2020. In *Proceedings of the 11th Global Wordnet Conference*. University of South Africa (UNISA): Global Wordnet Association, pp. 91–99. URL https://www.aclweb.org/anthology/2021.gwc-1.11.

McCrae, J.P., Rademaker, A., Rudnicka, E. & Bond, F. (2020). English WordNet 2020: Improving and Extending a WordNet for English using an Open-Source Methodology. In T. Declerk, I. Gonzalez-Dios & G. Rigau (eds.) *Proceedings of the LREC 2020 Workshop on Multimodal Wordnets, MMW@LREC 2020, Marseille, France, May 2020*. The European Language Resources Association (ELRA), pp. 14–19. URL https://www.aclweb.org/anthology/2020.mmw-1.3/.

Meyer, C.M. & Gurevych, I. (2011). What Psycholinguists Know About Chemistry: Aligning Wiktionary and WordNet for Increased Domain Coverage. In *Proceedings of 5th International Joint Conference on Natural Language Processing*. Chiang Mai, Thailand: Asian Federation of Natural Language Processing, pp. 883–892. URL https://www.aclweb.org/anthology/I11-1099.

Miller, G.A. (1995). WordNet: a lexical database for English. *Communications of the ACM*, 38(11), pp. 39–41.

Reimers, N. & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *CoRR*, abs/1908.10084. URL http://arxiv.org/abs/1908.10084. 1908.10084.

Schlippe, T., Ochs, S. & Schultz, T. (2010). Wiktionary as a source for automatic pronunciation extraction. *INTERSPEECH-2010*, pp. 2290–2293.

Wang, X., Lou, X. & Li, J. (2011). Speech synthesis with fuzzy heteronym prediction using decision trees. URL https://patents.google.com/patent/US9058811B2/en.

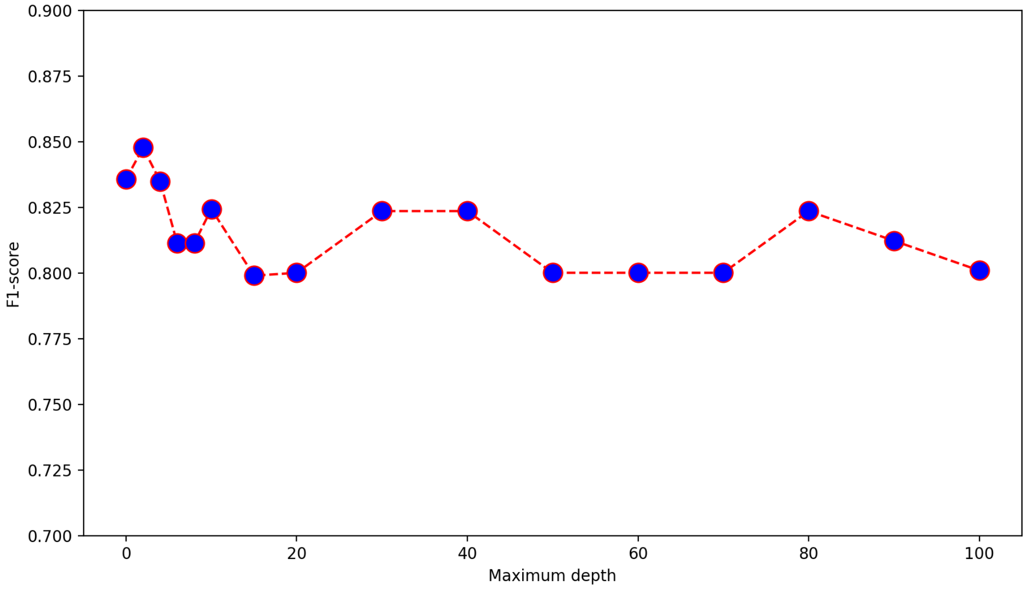

Figure 2: Random Forest Classifier F1-score depending on maximum depth

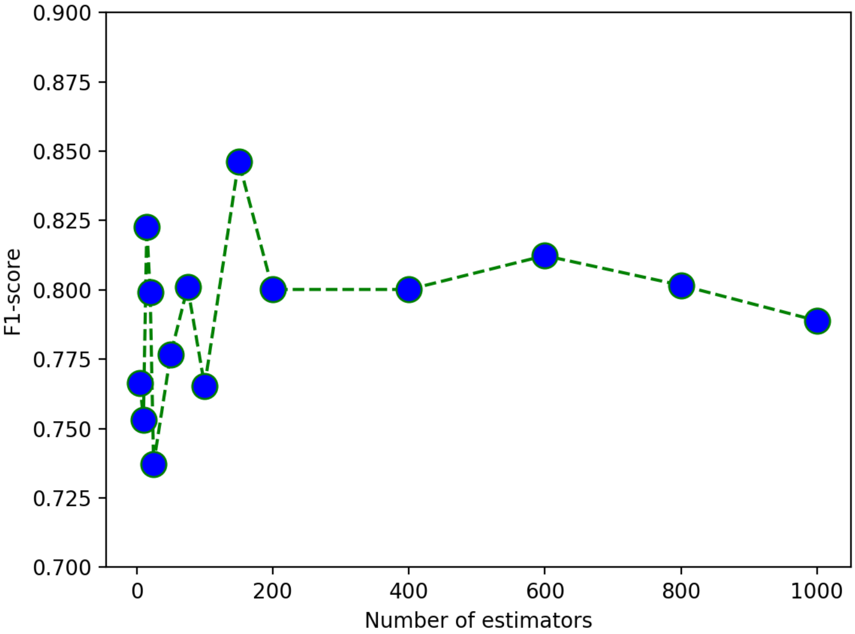

Figure 3: Random Forest Classifier accuracy depending on number of estimators