Research better NLP spellchecking options

Lisa Hensens

December 2, 2020

Synonyms detection with NLP

- Look for acronyms, e.g. UB - Universiteits Bibliotheek/University Library.
- It is possible to create your own synonym generation algorithm which you can use for synonym detection. [1]
- WordNet is a large lexicon of the English language.
- Word2vec is another option of a database that can be used that contains synonyms, but then in the form of vectors. [2]
- Word2vec differs from WordNet in that it isn't concerned with grammar [3]. Word2vec can fall short as customers search for products because the word pairs are related, but not always interchangeable.
- GloVe contains pretrained models which you can use to represent text data and try to find synonyms and associations like "zipcode - city". [5]
- Your chatbot needs a preprocessing NLP pipeline to handle typical errors. It may include these steps: spellcheck, split into sentences, split into words, POS tagging, lemmatize words, entity recognition, find synonyms. [6]
- Lemmatization and stop word removal are both potentially useful steps in preprocessing text, but they are not necessarily necessary. [9]
- Word2vec can also be used to predict the next word in a sentence. [10]

Conclusion

At this point, it looks like that we maybe need to add a few extra steps to preprocessing, such as improving the spellchecker with information from [11], splitting the message into seperate sentences (if it contains multiple sentences). In addition, we can try to use a lexicon such as WordNet to find synonyms and maybe add a few synonyms of our own such as "UB - University Library" because they will probably not be in a standard lexicon.

Furthermore, I will try to implement lemmatization in addition to the preprocessing and see if it improves the performance. If not, I will remove it again, because then it is redundant.

Word2vec can be a useful tool for finding synonyms with word embedding (mapping words into vectors) because synonyms tend to seem the same as their synonyms. Example can be 'aeroplane', 'airplane', 'aircraft', 'plane' [7].

[02/12/2020] While working on the corresponding code to improve the natural language processing of our chatbot, I have found that it is better not to combine lemmatization and stemming and thus decided on only using stemming which is currently done in the code. Also, The spellchecker from [11] is the inspiration for the spellchecker class which is imported and already used, and thus will not be used or added in the implementation. I will try to try to use POS tagging for synonym detection, because the original plans for improvement have been declared redundant by further findings which were found between writing the conclusion up until today.

Future work

Skip gram predicts the surrounding context words within specific window given current word, which can be a good option for predicting 3 (standard) questions when having some input.

Make use of POS tagging, which tags a word with 'noun' or 'verb' which can increase the accuracy of classifying the input with a matching output.

To detect abbreviations (and thus including acronyms), there maybe needs to be more research done. It is possible that I will maybe do this in this research and implement task (task WAT-89 and WAT-130), otherwise I will create a new task specific for this (and other additional future work stuff mentioned in this part).

References

[1] https://medium.com/@nikhilbd/how-to-use-machine-learning-to-find-synonyms-6380c0c6106b [1]

[2] `https://en.wikipedia.org/wiki/Word2vec`

[3] `https://lucidworks.com/post/search-automatic-synonym-detection/`

[4] `https://medium.com/rasa-blog/do-it-yourself-nlp-for-bot-developers-2e2da2817f3d`

[1] Lol couldn't use the url command because then the url would be too long

[5] https://github.com/stanfordnlp/GloVe

[6] https://medium.com/@surmenok/natural-language-pipeline-for-chatbots-897bda41482

[7] https://livebook.manning.com/book/deep-learning-for-search/chapter-2/44

[8] https://www.researchgate.net/publication/338035783_Extracting_Word_Synonyms_from_Text_using_Neural_Approaches

[9] https://opendatagroup.github.io/data%20science/2019/03/21/preprocessing-text.html

[10] http://jalammar.github.io/illustrated-word2vec/

[11] http://norvig.com/spell-correct.html