Medical NER Detection and Semantic Type Classification Using MedMentions Corpus with Flair

Javier Aguirre
University of the Basque Country
javiagu13@gmail.com

Abstract

Medical tagging has been a common task for a long time, nowadays computers are starting to solve this tasks automatically instead of humans. The aim of medical tagging is to be able to create medical electronic registers, classification tools, dictionaries, translators and so on. In this paper the detection of medical terms and the semantic types of the terms are aimed to be found using MedMentions corpus and the corresponding analysis of performance will be made in comparison with the state of the art. It has been proved that PubMed embeddings provide a better performance than GloVe embeddings for this specific task.

1 Introduction

Nowadays different kind of medical tagging tasks are common. UMLS tagging is one of the most important types of tags where each medical term has its corresponding tag. Also tagging in semantic types is another important task in order to know which kind of semantic type each term belongs to. Finally, knowing if a term is medical or non medical is also a common task.

In this paper we will be using the last two kinds of tags: finding medical terms and their semantic types. This way provinding a tool for creating medical electronic registers, classification tools, dictionaries and so on.

MedMentions (Mohan and Li, 2019) corpus will be used in order to get a sample of different medical terms and semantic types, also different embeddings will be tested such as gloVe and PubMed. For the training Flair (Akbik et al., 2019) will be used.

2 State of the Art

Currently biomedical name entity recognition state of the art with Flair is around 85%. Actually (Sharma and Daniel Jr, 2019) achieved 82.44% of accuracy and also (Patel, 2020) has shown high accuracy in different biomedical NER tasks.

Also, it has been shown a high accuracy of 93.09% for different nlp general tasks using Flair embeddings (Akbik et al., 2018). Which gives a hint of the effectiveness of using flair embeddings.

3 Named Entity Recognition

Named-entity recognition (NER) is a subtask of information extraction that seeks to locate and classify named entities mentioned in unstructured text into pre-defined categories such as person names, organizations, locations, medical codes, time expressions, quantities, monetary values, percentages, etc.

- In the case of medical field, let us suppose the NER for medical term detection will be applied:

  DCTN4 as a modifier of chronic Pseudomonas aeruginosa infection in cystic fibrosis.

- The aim of NER is to identify which are the medical terms in that senctence as follows:

  [DCTN4] as a modifier of [chronic Pseudomonas aeruginosa infection] in [cystic fibrosis].

- In the case of semantic type detection it will be defined like this:

  [DCTN4, T116] as a modifier of [chronic Pseudomonas aeruginosa infection, T047] in [cystic fibrosis, T047]

Flair (Akbik et al., 2019) is an open NLP framework that allows multiple tasks, in this case named entity recognition will be performed.

4 Contribution

It has been proved that PubMed embeddings make the model more accurate for medical name entity recognition tasks. Also, it can be seen that medical entity recognition is an easier task than semantic type recognition. It also seems that as high the number of semantic tags are used in a corpus, the more accurate the model is. This may occur because if medical entities are let without a tag, the model will predict that some medical terms are indeed out of the medical field.

A possible solution for the last problem is tagging all the tokens belonging to semantic types with their corresponding tags, and the rest of tokens instead of labeling them as ”out” if they belong to the medical field labeling them with ”begining” and ”inside”.

5 Experimentation

Taking into account state of the art performance of NER in medicine (Patel, 2020) we will also use similar hiperparameters in order to get the best out of the performance of our task. Due to the lack of RAM and GPU, the closest hiperparameters will be chosen.

5.1 Medical Term Detection

Nowadays as seen in (Patel, 2020), the state of the art is around 85% in finding several medical entities like: disease, chemical, protein and so on. Therefore, for just medical term detection a good performance is expected.

Experimental setup

For this task GloVe embeddings and PubMed embeddings will be compared.

As in the state of the art (Patel, 2020) is described, the same hiperparameters will be used. Bi directional LSTM approach will be made, learning rate of 0.1, minibatch size of 32, patience of 3, anneal factor of 0.5 and max epoch of 30. The main difference is the number of epoch used and the dataset which in this case is MedMentions.

Also, the tasks are slightly different, in this case medical term detection and in the state of the art multiple medical entities detection.

The accuracy of the algorithm will be measured.

Hypothesis

A good performance is expected and it is also expected that PubMed embeddings will outperform GloVe embeddings since they are medical specific embeddings.

Results

PubMed WordEmbeddings	GloVe WordEmbeddings
65,13	60,20

Figure 1: PubMed Embeddings outperform GloVe by around 5 points.

In figure 1 it can be seen that PubMed word embeddings make the algorithm perform better than with GloVe.

Conclusions

Let us discuss why the state of the art algorithm outperforms ours. First of all the number of epoch trained is smaller in this case. Also, while creating the embeddings, Flair allows stacked embedding where multiple embeddings can be stucked, in this case just one embedding was used therefore affecting the performance of the algorithm.

While training with GloVe apart from GloVe word embeddings, news-forward and news-backward flair embeddings could have been stacked. While training with PubMed, PubMed-forward and PubMed-backward could have been also stacked.

Ultimately, due to the lack of RAM and GPU just 30 epoch have been trained and embedding stacking has not been done. Also it can be seen that PubMed outperformed GloVe.

5.2 Semantic Type Detection

As explained above due to the lack of RAM and GPU not all semantic types will be chosen, just the most recurrent ones. And a comparison between both medical term detection and semantic type detection will be made.

Experimental setup

For this task GloVe embeddings and PubMed embeddings will be compared. First executed with 5 different semantic types, then with 10 semantic types and finally with 15 semantic types.

As in the state of the art (Patel, 2020) is described, the same hiperparameters will be used. Bi directional LSTM approach will be made, learning rate of 0.1, minibatch size of 16 (when number of entities to recognise grow minibatch size has been reduced for memory and GPU issues), patience of 3, anneal factor of 0.5 and max epoch of 30. The main difference is the number of epoch used and the dataset which in this case is PubMed.

The accuracy of the algorithm will be measured for each of the executions.

Hypothesis

It is expected that the accuracy with lesser semantic types will be the best. Also punctuations are expected to be worse in general than in the first experiment where only medical terms are aimed to be found.

It is also expected that pubmed embeddings will outperform glove embeddings since they are medical specific embeddings.

Results

Number of Semantic Types	PubMed WordEmbeddings	GloVe WordEmbeddings
5	42,80	38
10	46,00	41,45
15	46,33	41,83

Figure 2: PubMed Embeddings outperform GloVe by around 4 points.

In Figure 2 it can be seen that PubMed word embeddings make the algorithm perform better than with GloVe. It also seems that as high the number of semantic tags are used in a corpus, the more accurate the model is. This may occur because if medical entities are let without a tag, the model will predict that some medical terms are indeed out of the medical field.

A possible solution for the last problem is tagging all the tokens belonging to semantic types with their corresponding tags, and the rest of tokens instead of labeling them as ”out” if they belong to the medical field labeling them with ”begining” and ”inside”.

Conclusions

Let us discuss why the state of the art algorithm outperforms ours. As explained before the number of epoch trained is smaller in this case. Also, while creating the embeddings, Flair allows stacked embedding where multiple embeddings can be stucked, in this case just one embedding was used therefore affecting the performance of the algorithm.

While training with GloVe apart from GloVe word embeddings, news-forward and news-backward flair embeddings could have been stacked. While training with PubMed, PubMed-forward and PubMed-backward could have been also stacked.

Ultimately, due to the lack of RAM and GPU just 30 epoch have been trained and embedding stacking has not been done. Also it can be seen that PubMed outperformed GloVe.

Also it must be mentioned that unexpectedly the higher the number of semantic types, the more accurate the model. This may occur because if medical entities are let without a tag, the model will predict that some medical terms are indeed out of the medical field.

A possible solution for the last problem is tagging all the tokens belonging to semantic types with their corresponding tags, and the rest of tokens instead of labeling them as ”out” if they belong to the medical field labeling them with ”begining” and ”inside”.

6 Conclusions

The difference in performance between semantic type detection and medical term detection has been analysed, different experiments have been done in order to see the difference on performance of the embeddings and to compare our algorithm with the state of the art. PubMed embeddings have clearly raised the performance of the algorithm. As future work, stacked embeddings should be added to the model and other state of the art datasets should be tested apart from MedMentions.

References

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. Flair: An easy-to-use framework for state-of-the-art nlp. In *NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59.

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Sunil Mohan and Donghui Li. 2019. Medmentions: A large biomedical corpus annotated with umls concepts. Amherst, Massachusetts, USA. In Proceedings of the 2019 Conference on Automated Knowledge Base Construction (AKBC 2019).

Harsh Patel. 2020. Bionerflair: biomedical named entity recognition using flair embedding and sequence tagger.

Shreyas Sharma and Ron Daniel Jr. 2019. Bioflair: Pretrained pooled contextualized embeddings for biomedical sequence labeling tasks. *arXiv preprint arXiv:1908.05760*.