NLP@UNED at SMM4H 2019: Neural Networks Applied to Automatic Classifications of Adverse Effects Mentions in Tweets

Javier Cortes-Tejada
NLP & IR Group
UNED
28040 Madrid, Spain
jcortes@lsi.uned.es

Juan Martinez-Romo
NLP & IR Group (UNED)
IMIENS
28040 Madrid, Spain
juaner@lsi.uned.es

Lourdes Araujo
NLP & IR Group (UNED)
IMIENS
28040 Madrid, Spain
lurdes@lsi.uned.es

Abstract

This paper describes a system for automatically classifying adverse effects mentions in tweets developed for the task 1 at Social Media Mining for Health Applications (SMM4H) Shared Task 2019. We have developed a system based on LSTM neural networks inspired by the excellent results obtained by deep learning classifiers in the last edition of this task. The network is trained along with Twitter GloVe pre-trained word embeddings.

1 Introduction

The Shared Task (Weissenbacher et al., 2019) of the 2019 Social Media Mining for Health Applications (SMM4H) Workshop proposed several Natural Language Processing (NLP) tasks using social media mining for health monitoring. Since these tasks involve NLP techniques, they are as interesting as difficult to solve because these systems should be able to work with many linguistics variations and model the different ways people express medical-related concepts in social media. In addition, we must take into account the level of noise caused by creative sentences, misspellings or ambiguous and sarcastic expressions which makes hard to tackle these tasks.

For this shared task we decided to participate in the first task. This task proposes to find tweets mentioning Adverse Drug Reactions (ADR), taking into account the linguistic variations between ADRs and indications (the reason to use the medication). We have developed a system based on LSTM networks due to their latest achievements in the last edition of this task (Xherija, 2018).

2 Dataset

In this section we describe the dataset of the task 1 and the applied pre-processing. This task proposes to find tweets mentioning ADRs, therefore we have to deal with raw text extracted from Twitter.

The publicly available dataset contains for each tweet: (i) the user ID, (ii) the tweet ID, and (iii) the binary annotation indicating the presence or absence of ADRs. The dataset contains 24606 tweets manually tagged, being around 10% (2358) of tweets mentioning ADRs, and around the remaining 90% (22248) are tweets without ADRs.

2.1 Pre-processing

Regarding the dataset we normalized typical Twitter strings such as @user by <USER>, #hashtag by <HASHTAG> or https://... by <URL> to decrease the vocabulary size and reduce the dataset variability by grouping several tokens under the same meaning.

We also handle several elongated words such as “my goooood”. In these cases we replaced each token by a unique representation, for example “aaargh” and “arrggggh” by “argh”.

Finally the last step was to replace several constructions like “it’s” by “it is” or “OMG” by “Oh my god” and tokenize the text. For this step we used regular expressions and *NLTK* (Loper and Bird, 2002) to tokenize the text. We used specifically the class *TweetTokenizer* which is especially useful processing tweets since it splits the text into tokens, as others tokenizers, but also it takes into account some text elements like emojis or exclamatory particles, which are correctly separated into new tokens.

We didn’t remove any stop-word or convert to lowercase the text because that might change the meaning of a tweet drastically.

3 System architecture

We used a model based on a Bi-LSTM network due to its high performance in NLP tasks being

Proceedings of the 4^{th} Social Media Mining for Health Applications (#SMM4H) Workshop & Shared Task, pages 93–95
Florence, Italy, August 2, 2019. ©2019 Association for Computational Linguistics

used along with Twitter GloVe (Pennington et al., 2014) embeddings. The input of the system is a tweet (a sequence of words) which is used by the Embedding Layer with a fixed input size, while the weights of this layer are given by the GloVe word embeddings trained with 2 billion tweets. We have chosen these embeddings instead of others like word2vec (Mikolov et al., 2013), godin (Godin et al., 2015) or shin (Shin et al., 2016) because Twitter GloVe is trained with tweets, what is very useful since it allows us to have a greater vocabulary and also more similar to the text provided by the task.

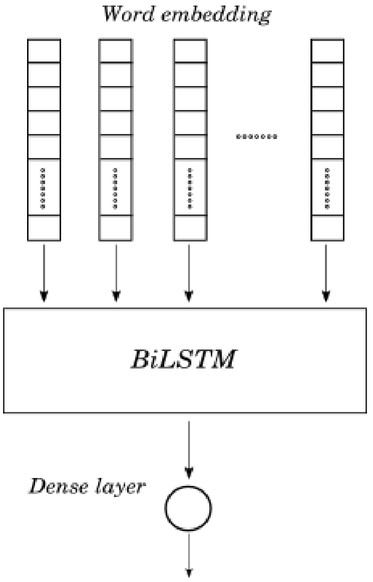

Figure 1: System architecture based on Twitter GloVe embeddings and a Bi-LSTM network.

As it can be seen in Figure 1, the next layer of our system is a Bi-LSTM layer. We decided to use it because a single LSTM network have not access to further tokens as they have not been seen. A Bi-LSTM has access to past tokens and future tokens, so this layer will give us a complete knowledge about the tweet; one LSTM will scan the sentence in one direction and the other will scan in the reverse direction. After these two layers we set a Dropout layer to prevent overfitting (Peng et al., 2015) with a rate of 0.3 for the Embedding layer and 0.5 for the Bi-LSTM layer. Finally we added a Dense layer with a sigmoid activation function at the end of the network to get the final results.

Regarding hyper parameters we used some configurations before we submitted the runs. For these tests we have tuned the epochs, the size of the batch (32, 64 and 128), the size of the embedding (vector of 50 and 100 dimensions in both embeddings), and the optimizer by considering a couple of them as Adam (Kingma and Ba, 2014) and AdaGrad (Duchi et al., 2011). We also handle the vocabulary tokens by adding pad right. At the end we chose the 3 configurations that reported the best results, whose hyper parameters are shown in Table 1.

4 Experiments and Results

For the implementation of the system we chose *Keras* and *Tensorflow* (Abadi et al., 2016) while for the pre-processing of the data we used *Scikit-learn* (Pedregosa et al., 2011), in particular for padding and split the dataset into validation, train and test sets.

In order to test the functioning of our system we used the evaluation script provided by the organizers. Several experiments are shown in Table 2. In these experiments we used a network without embeddings (Base) and with two types of embeddings, one pre-trained on Wikipedia pages (Wikipedia GloVe) and the other one based on tweets (Twitter GloVe). Due to the better performance shown by the configuration that used Twitter GloVe pre-trained embeddings, we decided to use it for the runs that we submitted to the task.

Table 3 shows the official results for the three runs that we submitted to the task 1 and the task average score provided by the organizers. According to the results obtained, it could be said that a greater number of epochs provides better results although the recall begins to fall.

5 Conclusions

Taking into account the experiments carried out on the training set and the results obtained, we can say that the use of embeddings pre-trained on tweets has been positive, that a greater number of epochs has provide us a better performance and that the best feature of our system is the recall as it obtains a value above the average.

In the future, we will try to create a more complex system to improve its performance. For this task we will add new features such as POS tagging and char embeddings as well as an attention mechanism.

Agreements

This work has been partially supported by the projects PROSA-MED (TIN2016-77820-C3-2-R) and EXTRAE (IMIENS 2017).

	Run 1	**Run 2**	**Run 3**
Epochs	40	30	20
Embedding	Twitter Glove		
Batch size	64	64	32
Embedding size	100	100	50
Optimizer	AdaGrad	AdaGrad	Adam

Table 1: Hyper parameter tunning used in the 3 runs submitted for task 1.

System	**P**	**R**	**F1**
Base	0.408	0.430	0.419
Base + Wikipedia G	0.450	0.512	0.483
Base + Twitter G	**0.458**	**0.590**	**0.510**

Table 2: System results according the Precision (P), Recall (R) and F-Measure (F1) scores.

References

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. 2016. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159.

Fréderic Godin, Baptist Vandersmissen, Wesley De Neve, and Rik Van de Walle. 2015. Multimedia lab @ acl wnut ner shared task: Named entity recognition for twitter microposts using distributed word representations. In *Proceedings of the Workshop on Noisy User-generated Text*, pages 146–153.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Edward Loper and Steven Bird. 2002. Nltk: the natural language toolkit. *arXiv preprint cs/0205028*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830.

Hao Peng, Lili Mou, Ge Li, Yunchuan Chen, Yangyang Lu, and Zhi Jin. 2015. A comparative study on regularization strategies for embedding-based neural networks. *arXiv preprint arXiv:1508.03721*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Bonggun Shin, Timothy Lee, and Jinho D Choi. 2016. Lexicon integrated cnn models with attention for sentiment analysis. *arXiv preprint arXiv:1610.06272*.

Davy Weissenbacher, Abeed Sarker, Arjun Magge, Ashlynn Daughton, Karen O'Connor, Michael Paul, and Graciela Gonzalez-Hernandez. 2019. Overview of the fourth Social Media Mining for Health (SMM4H) shared task at ACL 2019. In *Proceedings of the 2019 ACL Workshop SMM4H: The 4th Social Media Mining for Health Applications Workshop Shared Task*.

Orest Xherija. 2018. Classification of medication-related tweets using stacked bidirectional lstms with context-aware attention. In *Proceedings of the 2018 EMNLP Workshop SMM4H: The 3rd Social Media Mining for Health Applications Workshop and Shared Task*, pages 38–42. Association for Computational Linguistics.

Runs	**P**	**R**	**F1**
Run 1 (30 epochs)	0.463	**0.535**	0.408
Run 2 (40 epochs)	**0.472**	0.524	**0.429**
Run 3 (20 epochs)	0.431	0.491	0.385
Task average score	0.535	0.505	0.501

Table 3: Official results for the three runs that participated in task 1 and task average score provided by organizers.