Epita at SemEval-2018 Task 1: Sentiment Analysis Using Transfer Learning Approach

Guillaume Daval-Frerot **Abdessalam Bouchekif** **Anatole Moreau**

Graduate school of computer science, EPITA, France

firstname.lastname@epita.fr

Abstract

In this paper we present our system for detecting valence task. The major issue was to apply a state-of-the-art system despite the small dataset provided : the system would quickly overfit. The main idea of our proposal is to use transfer learning, which allows to avoid learning from scratch. Indeed, we start to train a first model to predict if a tweet is positive, negative or neutral. For this we use an external dataset which is larger and similar to the target dataset. Then, the pre-trained model is re-used as the starting point to train a new model that classifies a tweet into one of the seven various levels of sentiment intensity.

Our system, trained using transfer learning, achieves 0.776 and 0.763 respectively for Pearson correlation coefficient and weighted quadratic kappa metrics on the subtask evaluation dataset.

1 Introduction

The goal of detecting valence task is to classify a given tweet into one of seven classes, corresponding to various levels of positive and negative sentiment intensity, that best represents the mental state of the tweeter. This can be seen as a multiclass classification problem, in which each tweet must be classified in one of the following classes : very negative (-3), moderately negative (-2), slightly negative (-1), neutral/mixed (0), slightly positive (1), moderately positive (2) and very positive (3) (Mohammad et al., 2018).

Several companies have been interested in customer opinion for a given product or service. Sentiment analysis is one approach to automatically detect their emotions from comments posted in social networks.

With the recent advances in deep learning, the ability to analyse sentiments has considerably improved. Indeed, many experiments have used state-of-the-art systems to achieve high performance. For example, (Baziotis et al., 2017) use Bidirectional Long Short-Term Memory (B-LSTM) with attention mechanisms while (Deriu et al., 2016) use Convolutional Neural Networks (CNN). Both systems obtained the best performance at the the 2016 and 2017 SemEval 4-A task respectively.

The amount of data is argued to be the main condition to train a reliable deep neural network. However, the dataset provided to build our system is limited. To address this issue, two solutions can be considered. The first solution consists in extending our dataset by either manually labeling new data, which can be very time consuming, or by using over-sampling approaches. The second solution consists in applying a transfer learning, which allows to avoid learning the model from scratch.

In this paper, we apply a transfer learning approach, from a model trained on a similar task : we propose to pre-train a model to predict if a tweet is positive, negative or neutral. Precisely, we apply a B-LSTM on an external dataset. Then, the pre-trained model is re-used to classify a tweet according to the seven-point scale of positive and negative sentiment intensity.

The rest of the paper is organized as follows. Section 2 presents a brief definition of transfer learning. The description of our proposed system is presented in Section 3. The experimental setup and results are described in Section 4. Finally, a conclusion is given with a discussion of future works in Section 5.

2 Transfer Learning

Transfer Learning (TL) consists in transferring the knowledge learned on one task to a second related task. In other words, the TL is about training a base network and then copy its first n layers to the first n layers of a target network (Yosinski et al., 2014). Usually the first n layers of a pre-

Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018), pages 151–155
New Orleans, Louisiana, June 5–6, 2018.

trained model (or source model) are frozen when training the new model. This means that weights are not changed during training on the new task. TL should not be confused with fine-tuning where the back-propagation error affects the entire neural network (including the first n layers).

For a limited number of training examples, TL allows to provide more precise predictions than the traditional supervised learning approaches. Moreover, TL significantly speeds up the learning process as training does not start from scratch. For example, (Cirean et al., 2012) use a CNN trained to recognize the Latin handwritten characters for the detection of Chinese characters. In natural language processing, TL has improved the performance of several systems from various domains such as : sentiment classification (Glorot et al., 2011), automatic translation (Zoph et al., 2016), speech recognition and document classification (Wang and Zheng, 2015).

3 Proposed System

In this section, we present the four main steps of our approach : (1) **Text processing** to filter the noise from the raw text data, (2) **Feature extraction** to represent words in tweets as vectors of length 426 by concatenating several features, (3) **Pre-training model** to predict the tweet polarity (positive, negative or neutral) based on external data and (4) **Learning** a new model where the pre-trained model is adapted to our task by removing the last layer and adding a fully-connected layer followed by an output layer.

3.1 Text processing

Tweets are processed using *ekphrasis* [1] tool which allows to perform the following tasks : tokenization, word normalization, word segmentation (for splitting hashtags) and spell correction (*i.e* replace a misspelled word with the most probable candidate word). All words are lowercase. E-mails, URLs and user handles are normalized. A detailed description of this tool is given in (Baziotis et al., 2017).

3.2 Feature extraction

Each word in each tweet is represented by a vector of 426 dimensions which are obtained by the concatenation of the following features :

— *AFINN* and *Emoji Valence* [2] are two lists of english words and emojis rated for valence scoring range from -5 (very negative) to $+5$ (very positive) (Nielsen, 2011).

— *Depeche Mood* is a lexicon of $37k$ words associated with emotion scores (afraid, amused, angry, annoyed, sad, happy, inspired and don't care) (Staiano and Guerini, 2014).

— *Emoji Sentiment Lexicon* is a lexicon of the 969 most frequent emojis. The emojis sentiment is computed from the sentiment (positive, negative or neutral) of tweets in which they occur. Each emoji is associated with a unicode, number of occurrences, position in the tweet $[0, 1]$ (0 : start of the tweet, 1 : end of the tweet), probabilities of negativity, neutrality, and positivity of the emoji (Novak et al., 2015).

— *Linguistic Inquiry and Word Count* is a dictionary containing 5,690 stems associated with 64 categories, from linguistic dimensions to psychological processes (Tausczik and Pennebaker, 2010).

— *NRC Word-Emotion Association*, *Hashtag Emotion/Sentiment* and *Affect Intensity Lexicons* are lists of english words and their associations with eight emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and two sentiments (positive, negative), each with specificities detailed in (Mohammad and Turney, 2013), (Mohammad and Kiritchenko, 2015) and (Mohammad, 2017). The intensity score for both emotions and sentiments takes a value between 0 and 1.

— *Opinion Lexicon English* contains around $7k$ positive and negative sentiment words for the english language (Hu and Liu, 2004).

— *Sentiment140* is a list of words and their associations with positive and negative sentiment (Mohammad et al., 2013).

— *Words embeddings* are dense vectors of real numbers capturing the semantic meanings of words. We use datastories embeddings (Baziotis et al., 2017) which were trained on 330M english twitter messages posted from 12/2012 to 07/2016. The embeddings used in this work are 300 dimensional.

1. https://github.com/cbaziotis/ekphrasis

2. https://github.com/words/emoji-emotion

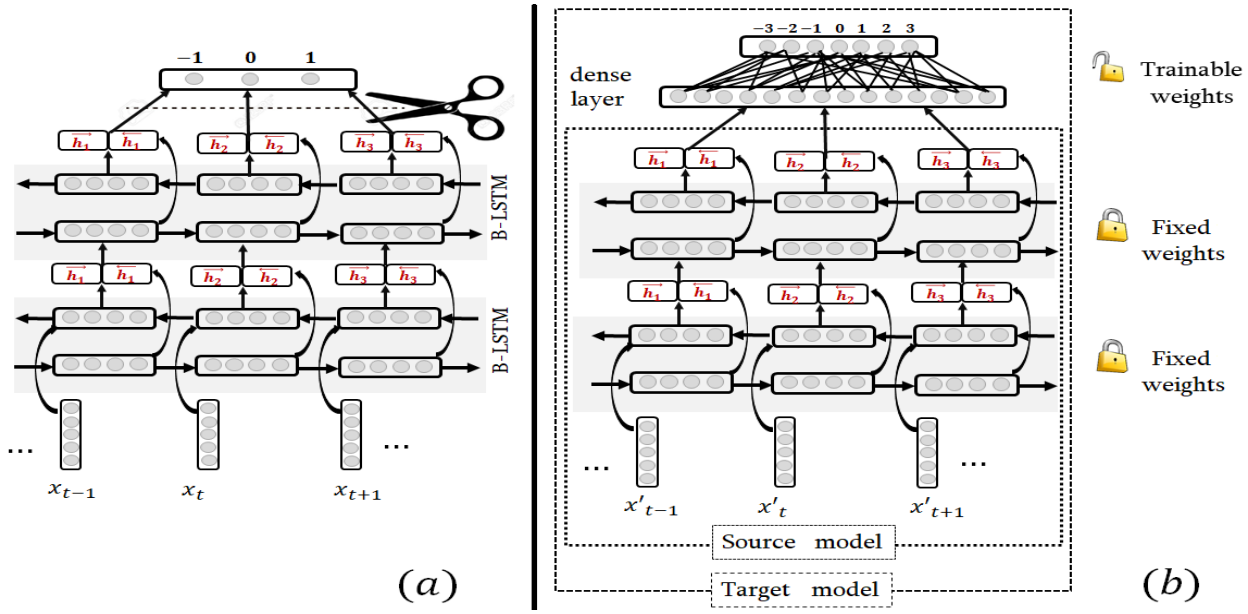

Figure 1 – Our transfer learning approach for sentiment analysis. (a) Pre-trained model learned with B-LSTM network with 2 layers of 150 neurons each to predict if a tweet is positive, negative or neutral. (b) The first layers of pre-traind model are locked and re-purposed to predict various levels of positive and negative sentiment intensity.

3.3 Pre-training model

The objective is to build a model which allows to predict the tweeter's attitude (positive, negative or neutral). Bidirectional Long Short-Term Memory networks (B-LSTM) (Schuster and Paliwal, 1997) have become a standard for sentiment analysis (Baziotis et al., 2017) (Mousa and Schuller, 2017) (Moore and Rayson, 2017). B-LSTM consists in two LSTMs in different directions running in parallel : the first forward network reads the input sequence from left to right and the second backward network reads the sequence from right to left. Each LSTM yields a hidden representation : $\vec{h}$ (left to right vector) and $\overleftarrow{h}$ (right-to-left vector) which are then combined to compute the output sequence. For our problem, capturing the context of words from both directions allows to better understand the tweet semantic. We here use a B-LSTM network with 2 layers of 150 neurons each. The architecture is shown in Figure 1 (a).

For training, we use the external dataset [3] composed of 50333 tweets (7840 negatives, 19903 positives and 22590 neutrals).

3. https://github.com/cbaziotis/datastories-semeval2017-task4/tree/master/dataset/Subtask_A/downloaded.

3.4 Learning model

Let us note that our final objective is to train a model to classify a tweet into seven classes (very negative, moderately negative, slightly negative, neutral, slightly positive, moderately positive and very positive). To train the model, we use the dataset provided for the target task (Mohammad and Kiritchenko, 2018). The training and development dataset contain respectively 1180 and 448 tweets. Since the dataset is small, fine-tuning may result in overfitting. Therefore, we propose to freeze the network layers except the final dense layer that is associated with the three classes sentiment analysis, which is removed after pre-training. Then, we add a fully-connected layer of 150 neurons followed by an output layer of 7 neurons, as illustrated on Figure 1 (b).

4 Results and Analysis

The official [4] evaluation metric is *Pearson Correlation Coefficient* (P). Submited systems are also evaluated with the weighted quadratic kappa (W). However, the pre-trained model was evaluated using classification accuracy. We implemented our system using Keras tool with the Tensorflow backend.

4. https://github.com/felipebravom/SemEval_2018_Task_1_Eval

4.1 Pre-trained model evaluation

As proposed in (Baziotis et al., 2017), we used B-LSTM with the following parameters : size of LSTM layers is 150 (300 for B-LSTM), 2 layers of B-LSTM, with a dropout of 0.3 and 0.5 for embedding and LSTM layers respectively. Other hyper-parameters used are : Gaussian noise with σ of 0.3, and L_2 regularization of 0.0001. We trained the B-LSTM over 18 epochs with a learning rate of 0.01 and batch size of 128 sequences.

We trained our model with external data (more details in section 3.3) but for the evaluation we adapted the training and development sets provided for the target task. The various levels of positive sentiments (*i.e* slightly, moderately and very positive) were regrouped in the same class. The same goes for the various levels of negative sentiments. Our model achieves 69.4% of accuracy.

4.2 Model evaluation

We adapted the pre-trained model described above by removing the last fully-connected layer, and added a dense layer of 150 neurons followed by an output layer of 7 neurons. As a reminder, the pre-trained layers are frozen. We used the training and development sets to train our system, and evaluated by predicting the valence on the evaluation set. We trained our model over 8 epochs with a learning rate of 0.01 and batch size of 50 sequences. Our model achieves 0.776 and 0.763 respectively on P and W.

4.3 Other experiments

Finally, we conducted a set of experiments to validate our system and approach. We evaluated more commonly used systems, with and without transfer learning. These new systems are built by :

— using similar number of layers, parameters and hyper-parameters.
— replacing B-LSTM layers by LSTM, CNN and dense layers.
— for the DNN, computing predictions using the mean of each word-vector of tweets, since it can not use sequences as input.
— for the CNN, using multiple convolutional filters of sizes 3, 4 and 5.
— for the combinations of systems, averaging the output probabilities.

The results are presented on Table 1.

We can observe that TL approach achieves better scores, and that B-LSTM is leading the score on both approaches as a single system. Moreover, combining systems enhances greatly the prediction without TL, but decreases the score with TL : the combination of independent systems compensates a small lack of data, but becomes useless with enough training.

Approach	Systems	*Pearson*
Without TL	DNN	0.683
	CNN	0.702
	LSTM	0.721
	B-LSTM	0.735
	CNN + LSTM	**0.742**
With TL	CNN	0.741
	B-LSTM	**0.776**
	CNN + B-LSTM	0.755

Table 1 – Pearson scores on test set with different systems and combinations.

5 Conclusion

In this paper, we propose to use a transfer learning approach for sentiment analysis (SemEval2018 task 1). Using B-LSTM networks, we pre-trained a model to predict the tweet polarity (positive, negative or neutral) based on an external dataset of $50k$ tweets. To avoid the overfitting, layers (except the last one) of the pre-trained model were frozen. A dense layer was then added followed by a seven neurones output layer. Finally, the new network was trained on the small target dataset. The system achieves a score of 0.776 on Pearson Correlation Coefficient.

Improvements could be made concerning the features, and by using attention mechanisms. However, the future work will focus on multiple transfers, to increase the amount of data used in the process. We will perform transfers from two classes (positive and negative) to three classes (adding neutral), then five classes and finally seven classes. Numerous datasets [5] are currently available to deploy such a system.

Acknowledgments

We thank Dr. Yassine Nair Benrekia for interesting scientific discussions.

5. http://alt.qcri.org/semeval2016/task4/

References

Christos Baziotis, Nikos Pelekis, and Christos Doulkeridis. 2017. Datastories at semeval-2017 task 6 : Siamese LSTM with attention for humorous text comparison. In *Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval@ACL 2017, Vancouver, Canada*.

D. C. Cirean, U. Meier, and J. Schmidhuber. 2012. Transfer learning for latin and chinese characters with deep neural networks. In *The 2012 International Joint Conference on Neural Networks (IJCNN)*.

Jan Deriu, Maurice Gonzenbach, Fatih Uzdilli, Aurélien Lucchi, Valeria De Luca, and Martin Jaggi. 2016. Swisscheese at semeval-2016 task 4 : Sentiment classification using an ensemble of convolutional neural networks with distant supervision. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT, USA*.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Domain adaptation for large-scale sentiment classification : A deep learning approach. In *Proceedings of the 28th International Conference on Machine Learning, USA*.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, Washington, USA*, pages 168–177.

Saif Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. Nrc-canada : Building the state-of-the-art in sentiment analysis of tweets. In *Proceedings of the seventh international workshop on Semantic Evaluation Exercises (SemEval-2013)*.

Saif Mohammad and Peter D. Turney. 2013. Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, 29(3) :436–465.

Saif M. Mohammad. 2017. Word affect intensities. *CoRR*, abs/1704.08798.

Saif M. Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. Semeval-2018 Task 1 : Affect in tweets. In *Proceedings of International Workshop on Semantic Evaluation (SemEval-2018)*, New Orleans, LA, USA.

Saif M. Mohammad and Svetlana Kiritchenko. 2015. Using hashtags to capture fine emotion categories from tweets. *Computational Intelligence*, 31(2) :301–326.

Saif M. Mohammad and Svetlana Kiritchenko. 2018. Understanding emotions : A dataset of tweets to study interactions between affect categories. In *Proceedings of the 11th Edition of the Language Resources and Evaluation Conference*, Miyazaki, Japan.

Andrew Moore and Paul Rayson. 2017. Lancaster A at semeval-2017 task 5 : Evaluation metrics matter : predicting sentiment from financial news headlines. In *Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval@ACL 2017, Canada*.

Amr Mousa and Bjrn Schuller. 2017. Contextual bidirectional long short term memory recurrent neural network language models : A generative approach to sentiment analysis. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*.

Finn Årup Nielsen. 2011. A new evaluation of a word list for sentiment analysis in microblogs. In *Proceedings of the ESWC2011 Workshop on 'Making Sense of Microposts' : Big things come in small packages Crete, Greece*, pages 93–98.

Petra Kralj Novak, Jasmina Smailovic, Borut Sluban, and Igor Mozetic. 2015. Sentiment of emojis. *CoRR*, abs/1509.07761.

Mike Schuster and Kuldip K. Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Trans. Signal Processing*.

Jacopo Staiano and Marco Guerini. 2014. Depeche mood : a lexicon for emotion analysis from crowd annotated news. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL, Baltimore, USA*, pages 427–433.

Yla R. Tausczik and James W. Pennebaker. 2010. The psychological meaning of words : Liwc and computerized text analysis methods. *Journal of Language and Social Psychology*, 29 :24–54.

Dong Wang and Thomas Fang Zheng. 2015. Transfer learning for speech and language processing. In *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks ? In *Proceedings of the 27th International Conference on Neural Information Processing Systems*.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, USA*.