# ELiRF-UPV at SemEval-2017 Task 4: Sentiment Analysis using Deep Learning

**José-Ángel González, Ferran Pla, Lluís-F. Hurtado**
Universitat Politècnica de València
Camí de Vera sn, 46022, València
{jogonba2|fpla|lhurtado}@dsic.upv.es

## Abstract

This paper describes the participation of ELiRF-UPV team at task 4 of SemEval2017. Our approach is based on the use of convolutional and recurrent neural networks and the combination of general and specific word embeddings with polarity lexicons. We participated in all of the proposed subtasks both for English and Arabic languages using the same system with small variations.

## 1 Introduction

Twitter has become a source of a huge amount of information which introduces great possibilities of research in the field of Sentiment Analysis. Sentiment Analysis or Opinion Mining, is a research area within Natural Language Processing whose aim is to identify the underlying emotion of a certain document, sentence or aspect (Liu, 2012). Sentiment Analysis systems has been applied, among other, for classifying reviews (Turney, 2002; Pang et al., 2002), for generating aspect-based summaries (Hu and Liu, 2004), or political tendency identification (Pla and Hurtado, 2014).

SemEval-2017 task 4 organizers proposed five different subtasks. All five subtasks are related to sentiment analysis at global level in Twitter, but each one of them has significant differences. Additionally, in the 2017 edition, the five subtasks were also proposed in Arabic. Altogether, the participants could address ten different challenges.

Subtask A consists in predicting the message polarity as positive, negative, or neutral. In subtasks B and C, given a message and a topic systems should assign the message in a two-point scale or in a five-point scale respectively.

Subtasks D and E address the problem of tweet quantification, that is, given a set of tweets about a given topic, estimate the distribution of the tweets across two-point scale or in a five-point scale respectively.

The rest of this paper is organized as follows. Section 2 describes the general system architecture proposed in this work. Section 3 presents both the variations on the general system introduced to address the different subtasks and the results obtained in the subtasks. Finally, section 4 presents some conclusions and the future work.

## 2 System description

In this section, we describe the system architecture we used for all the Sentiment Analysis subtasks. This system is based on the use of convolutional and recurrent neural networks and the combination of general and specific word embeddings (Mikolov et al., 2013b,a) with polarity lexicons.

Slight modifications of the system have been applied to adapt it to each subtask. These modifications are motivated by the characteristics of each subtask and the available resources.

The system combines three Convolutional Recurrent Neural Network (CRNN) (Zhou et al., 2002) in order to learn high level abstractions (Lecun et al., 2015) from noisy representations (Jim et al., 1994). The input of these three networks are: out-domain embeddings, in-domain embeddings, and sequences of the polarities of the words. The output of the CRNNs is concatenated and used as input for a discriminating model implemented by a fully-connected Multilayer Perceptron (MLP). Figure 1 summarizes the proposed approach.

The CRNNs used have as a first layer a unidimensional convolutional layer that allows to extract spatial relations among the words of a sentence (Kim, 2014). In some subtasks, a down-



*Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval-2017)*, pages 723–727,
Vancouver, Canada, August 3 - 4, 2017. 

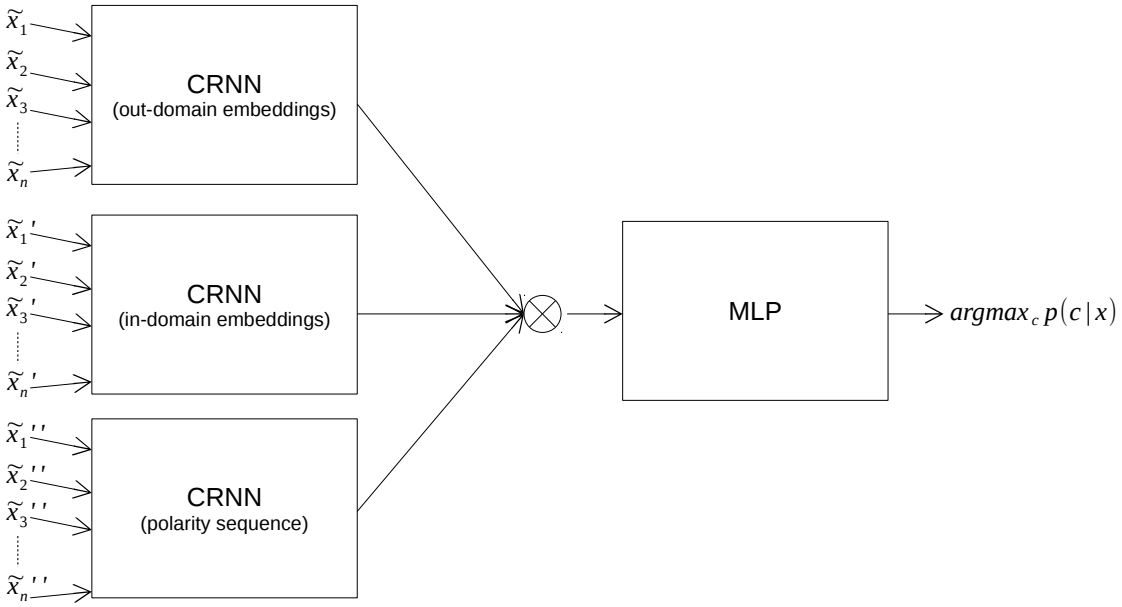


Figure 1: General system architecture.

sampling process by means of a *max pooling* layer was applied.

Then, the output of the convolutional layers (including *max pooling* in some subtasks) is used as input for a recurrent neural network (LSTM). Moreover, because the polarity of a subsequence of the sentence not only depends on the previous words but also depends on the next words, we used a Bidirectional Long-Short-Term Memory (BLSTM) network (Hochreiter and Schmidhuber, 1997; Schuster and Paliwal, 1997). In most subtasks, only one BLSTM layer has been used. The dimension for the output vector has been fixed between 32 and 256.

Figure 2 shows a graphical representation of the CRNN layers, where $\tilde{x}_i$ is a noisy version of the input, $c_i$ are the kernels of the convolutional layer, $p_i$ represent the operations of max pooling, and $y_s$ is the output of the CRNNs.

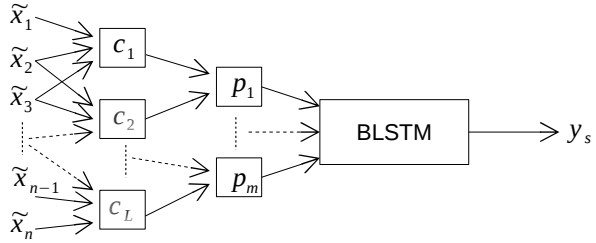


Figure 2: Implementation of the Convolutional Recurrent Neural Network.

The last network used in our system is a fully connected Multilayer Perceptron. Depending on the subtasks, we used between 1 and 3 hidden layers. The number of neurons also depended on the subtask. Softmax activation function was used in the output layer to estimate $p(c|x)$ (the number of neurons in that layer depends on the number of classes in the task).

A graphical representation of the MLP used can be seen in Figure 3, where $y_i$ are the outputs of the CRNNs, which are used as input for the MLP. Note that, in this case, no noise is applied to the input because the chosen setup obtained better results during the tuning phase.

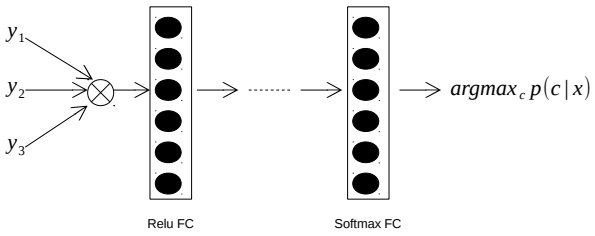


Figure 3: Implementation of the Multilayer Perceptron.

### 2.1 Resources

As we stated above, we used two different kind of embeddings (in-domain and out-domain) as input to the system for all the Arabic and English subtasks.

We used these two embeddings models in order to reduce the number of unseen words. In this way, we combined a specific representation that only considers the words seen in the training set (in-domain embeddings) with a more general one that has a great amount of words unseen in the training set but that can appear in the test set (out-domain embeddings).

For the English subtasks, we used as out-domain model the word2vec model learned by Fréderic Godin (Godin et al., 2015; Ritter et al., 2011) from 400 million tweets in English. For the Arabic subtasks, we learned a 400-dimensional word2vec model using the articles of the Wikipedia in Arabic (Wikipedia, 2017). With respect to the in-domain models, a word2vec model was trained for each subtask from the provided training corpus.

In addition to the two representations based on embeddings, we added polarity information to the input layer. To include this information, we considered a representation of tweets based on a sequence of C-dimensional one-hot vectors, where C is the number of sentiment classes. Each vector indicates the polarity of one word according to certain polarity lexicon. This way, a tweet is a sequence of C-dimensional vectors. Once again, the resources used depended on the language. We used the NRC lexicon (Mohammad et al., 2013) both for the Arabic and English subtasks and the *Afinn* lexicon (Hansen et al., 2011) only for the English subtasks.

## 3 Results

In this section, we present the modifications we made on the general schema for all the subtasks in which we participated. We also report and discuss the results we achieved in the different subtasks.

Due to the different sizes of the corpora used in every subtask, we made some changes from the generic model in order to reduce or increase the number of parameters to be estimated. These changes had been fixed for each subtask by means of a tuning process.

### 3.1 Subtask A: Message Polarity Classification

Subtask A consists in classifying the message as positive, negative, or neutral. Our model for this subtask consists of three CRNN merged with a three layer MLP, see general schema in Figure 1.

The results achieved by our system in Subtask A are shown in Table 1. The measure used to range the participants was macroaveraged recall ($\rho$). Two additional measures were also considered: $F_1$ averaged across the positives and the negatives ($F_1^{PN}$) and Accuracy ($Acc$). We have also included, for each measure, the position reached by our system compared with the other participants.

| Subtask A | English | | Arabic | |
|---|---|---|---|---|
| $\rho$ | 0.632 | (14/38) | 0.478 | (3/8) |
| $F_1^{PN}$ | 0.619 | (12/38) | 0.467 | (4/8) |
| $Acc$ | 0.599 | (24/38) | 0.508 | (3/8) |

Table 1: Results for Subtask A: Message Polarity Classification, English and Arabic.

Note the different ranking position achieved by our system considering $\rho$ and $Acc$ measures for English. $\rho$ achieved the 14th position while $Acc$ achieved the 24th position. We think this is due to the way we tackled with the imbalanced classes in the corpus. The decision was to balance the training set by eliminating some samples of those classes that appeared more times in the corpus.

In contrast, for the Arabic subtask, Accuracy results are not influenced by the way we managed the imbalanced problem, achieving similar position in all the measures considered.

### 3.2 Subtask B: Tweet classification according to a two-point scale

In subtask B, given a message and a topic, the participants must classify the message on two-point scale (positive and negative) towards that topic. Unfortunately, we did not include information of the topic in the model and, in consequence, our model consists of a variation of the generic model. In this case, max pooling layers were replaced with another convolutional layer, the number of neurons in MLP layers was reduced and we used Gaussian noise over MLP layers activations because better results are obtained over the validation set. For the Arabic language, we used the same topology, but we reduced the number of parameters due to the size of the training corpus.

The results achieved by our system in Subtask B are shown in Table 2. The measures considered were the same as in Subtask A.

The scores achieved in all measures are better than those obtained in task A. Perhaps, this sub-

| Subtask B | English | | Arabic | |
|---|---|---|---|---|
| $\rho$ | 0.766 | (17/23) | 0.721 | (2/4) |
| $F_1^{PN}$ | 0.773 | (16/23) | 0.724 | (2/4) |
| $Acc$ | 0.790 | (13/23) | 0.734 | (2/4) |

Table 2: Results for Subtask B: Tweet classification according to a two-point scale, English and Arabic.

task is easier because only two classes are considered. But, compared with the other participants, our system ranked lower in this subtask. We think this is because no information of the topic was included in the model. For this subtask, the behavior of the system for both languages is similar.

### 3.3 Subtask C: Tweet classification according to a five-point scale

In this subtask, given a message and a topic, participants must classify the message on a five-point scale towards that topic. As in Subtask B, we did not include topic information to the model. Our model was an extension of the generic model, with two convolutional layers and two max pooling layers in each CRNN. For the Arabic version, we used the generic model with less parameters because of the available data.

The results achieved by our system in Subtask C are shown in Table 3. The measure used to range the participants was macroaveraged Mean Absolute Error ($MAE^M$). An extension of macroaveraged recall for ordinal regression ($MAE^\mu$) was also considered.

| Subtask C | English | | Arabic | |
|---|---|---|---|---|
| $MAE^M$ | 0.806 | (7/15) | 1.264 | (2/2) |
| $MAE^\mu$ | 0.586 | (11/15) | 0.787 | (2/2) |

Table 3: Results for Subtask C: Tweet classification according to a five-point scale, English and Arabic.

For the English language, our system achieved the 7th position (0.806), with big difference respect to the team that obtained the best results (0.481). Once again, not including information about the topic could be decisive in the performance of the system.

### 3.4 Subtask D: Tweet quantification according to a two-point scale

Subtask D consists of tweet quantification in a two-point scale. Given a set of tweets about a given topic, participants must estimate the distribution of the tweets across two-point scale (positive and negative). We used the output of Subtask B to estimate, by maximum likelihood, the distribution of the tweets.

The results achieved by our system in Subtask D are shown in Table 4. The measure used to range the participants was Kullback-Leibler Divergence ($KLD$). Two additional measures were also considered: absolute error ($AE$) and relative absolute error ($RAE$).

| Subtask D | English | | Arabic | |
|---|---|---|---|---|
| $KLD$ | 1.060 | (14/15) | 1.183 | (3/3) |
| $AE$ | 0.593 | (15/15) | 0.537 | (3/3) |
| $RAE$ | 7.991 | (15/15) | 11.434 | (3/3) |

Table 4: Results for Subtask D: Tweet quantification according to a two-point scale, English and Arabic.

We can partially explain these poor results due to the simplicity of the method used to estimate the probability distribution and because the output of Subtask B also included errors.

### 3.5 Subsection E: Tweet quantification according to a five-point scale

In a similar way that Subtask D, Subtask E was a tweet quantification task, but in a five-point scale. For this subtask, we used the output of Subtask C to estimate, by maximum likelihood, the distribution of the tweets.

The results achieved by our system in Subtask E are shown in Table 5. The measure used to range the participants was Earth Mover's Distance ($EMD$).

| Subtask E | English | | Arabic | |
|---|---|---|---|---|
| $EMD$ | 0.306 | (4/12) | 0.564 | (2/2) |

Table 5: Results for Subtask E: Tweet quantification according to a five-point scale, English and Arabic.

Our system achieved the 4th position (0.306) for English, with slight difference respect to the first system (0.245).

## 4 Conclusions

In this work, we have presented the system developed by ELiRF-UPV team for participating in the task 4 of SemEval2017. We used a general system with small modifications to participate in all the subtasks. The system was based on the use of convolutional and recurrent neural networks and the combination of general and specific word embeddings with polarity lexicons. The results achieved by our system were competitive in many subtasks.

As future work, we plan to study some problems not addressed in this work such as tackle with the imbalance problem, address tweet quantification problem properly, add topic information in the model for B and C subtasks, and consider additional resources for tweet representation.

## Acknowledgements

This work has been funded by the MINECO and FEDER founds under TIN2014-54288-C4-3-R project: ASLP-MULAN: Audio, Speech and Language Processing for Multimedia Analytics.

## References

Fréderic Godin, Baptist Vandersmissen, Wesley De Neve, and Rik Van de Walle. 2015. Multimedia lab@ acl w-nut ner shared task: named entity recognition for twitter microposts using distributed word representations. *ACL-IJCNLP* 2015:146–153.

Lars Kai Hansen, Adam Arvidsson, Finn Årup Nielsen, Elanor Colleoni, and Michael Etter. 2011. Good friends, bad news-affect and virality in twitter. In *Future information technology*, Springer, pages 34–43.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.* 9(8):1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735.

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*. ACM, New York, NY, USA, KDD '04, pages 168–177. https://doi.org/10.1145/1014052.1014073.

Kam Jim, Bill G. Horne, and C. Lee Giles. 1994. Effects of noise on convergence and generalization in recurrent networks. In *Proceedings of the 7th International Conference on Neural Information Processing Systems*. MIT Press, Cambridge, MA, USA, NIPS'94, pages 649–656. http://dl.acm.org/citation.cfm?id=2998687.2998768.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *CoRR* abs/1408.5882. http://arxiv.org/abs/1408.5882.

Yann Lecun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature* 521(7553):436–444. https://doi.org/10.1038/nature14539.

Bing Liu. 2012. *Sentiment Analysis and Opinion Mining. A Comprehensive Introduction and Survey*. Morgan & Claypool Publishers.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *CoRR* abs/1301.3781. http://arxiv.org/abs/1301.3781.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. *CoRR* abs/1310.4546. http://arxiv.org/abs/1310.4546.

Saif M Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. *arXiv preprint arXiv:1308.6242* .

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *IN PROCEEDINGS OF EMNLP*. pages 79–86.

Ferran Pla and Lluís-F. Hurtado. 2014. Political tendency identification in twitter using sentiment analysis techniques. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*. Dublin City University and Association for Computational Linguistics, Dublin, Ireland, pages 183–192. http://www.aclweb.org/anthology/C14-1019.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named entity recognition in tweets: An experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Stroudsburg, PA, USA, EMNLP '11, pages 1524–1534. http://dl.acm.org/citation.cfm?id=2145432.2145595.

M. Schuster and K.K. Paliwal. 1997. Bidirectional recurrent neural networks. *Trans. Sig. Proc.* 45(11):2673–2681. https://doi.org/10.1109/78.650093.

Peter D. Turney. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *ACL*. pages 417–424. https://doi.org/http://www.aclweb.org/anthology/P02-1053.pdf.

Wikipedia. 2017. Wikipedia arabic dumps. [Online; accessed 22-February-2017]. https://dumps.wikimedia.org/arwiki/.

Zhi-Hua Zhou, Jianxin Wu, and Wei Tang. 2002. Ensembling neural networks: Many could be better than all. *Artificial Intelligence* 137(1):239 – 263. https://doi.org/http://dx.doi.org/10.1016/S0004-3702(02)00190-X.