NTUA-ISLab at SemEval-2019 Task 3: Determining emotions in contextual conversations with deep learning

Rolandos Alexandros Potamias, Georgios Siolas
Intelligent Systems Laboratory,
National Technical University of Athens,
Zografou, Athens , Greece
rolpotamias@gmail.com , gsiolas@islab.ntua.gr

Abstract

Sentiment analysis (SA) in texts is a well-studied Natural Language Processing task, which in nowadays gains popularity due to the explosion of social media, and the subsequent accumulation of huge amounts of related data. However, capturing emotional states and the sentiment polarity of written excerpts requires knowledge on the events triggering them. Towards this goal, we present a computational end-to-end context-aware SA methodology, which was competed in the context of the SemEval-2019 / EmoContext task (Task 3). The proposed system is founded on the combination of two neural architectures, a deep recurrent neural network, structured by an attentive Bidirectional LSTM, and a deep dense network (DNN). The system achieved 0.745 micro f1-score, and ranked 26/165 (top 20%) teams among the official task submissions.

1 Introduction and Related Work

One of the most challenging fields of Natural Language Processing (NLP) and Computational Linguistics is Sentiment Analysis (SA) that aims towards the automated extraction of a writer's sentiment or emotion as conveyed in text excerpts (Liu, 2015). Relevant efforts focus on tracking the sentiment polarity of single utterances, which in most of the cases is loaded with a lot of subjectivity and a degree of vagueness (Thelwall et al., 2010). Contemporary research in the field utilizes data from social media resources (e.g., Facebook, Twitter) as well as other short text references. However, users of social media tend to violate common grammar and vocabulary rules and even use various figurative language forms to communicate their message. In such situations, the sentiment polarity underlying the literal content of the conveyed concept may significantly differ from its figurative context, making SA tasks even more puzzling (Patra et al., 2016). Evidently, single turn text lacks in detecting sentiment polarity on sarcastic and ironic expressions (Potamias et al., 2019), as already indicatd in the relevant SemEval-2014 Task-9 *Sentiment Analysis in Twitter* (Sara et al., 2014). As sentiment reflects the emotion behind customer engagement, SA finds its realization in automated customer aware services (Kurniawati et al., 2013). A lot of research has already been devoted on capturing the sentiment polarity of contextual conversations of various utterances. Most of the relevant studies utilize single turn texts from topic related sources (e.g., Twitter). Hand-crafted and sentiment-oriented features, indicative of emotion polarity, are utilized to represent respective excerpt cases. The formed data were used as input to various traditional machine learning classifiers (e.g. SVM, Random Forests etc.) or deep learning architectures (e.g. recurrent neural networks, CNNs) in order to induce analytical models that are able to capture the underlying sentiment content of passages (Singh et al., 2018; Jianqiang et al., 2018; Hangya and Farkas, 2017).

The work presented in this study considers the recognition of the three fundamental emotions, Happy, Sad and Angry, specified in the SemEval-2019 / EmoContext task (Task 3), as a context sensitive task. In order to capture emotional categories, we settled two-layered bidirectional LSTM units that share weights over three embedded utterances resembling a siamese like architecture (Mueller and Thyagarajan, 2016). Pretrained word embeddings were utilized, Standford GloVe (Pennington et al., 2014), in order to represent single turn text input, resulting into an attentive and context-aware model. We also extended word embeddings with appropriate handling of emojis, utilizing the pretrained emoji2vec vectors (Eisner et al., 2016), and sentiment lexicons, NCR and De-

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 277–281
Minneapolis, Minnesota, USA, June 6–7, 2019.

pecheMood (Mohammad and Turney, 2013), resulting into an enhanced representation of single turn text cases. The overall complex presents a combined neural architecture to serve SA tasks.

2 Experimental Setup: Data & Preprocessing

The SemEval-2019 / EmoContext shared task (Task 3) targets the emotion classification of user utterances in three classes, namely Happy, Sad, Angry and Other (Chatterjee et al., 2019). It provides a 33K training textual dialogue dataset in the form of three contextual turns. The distribution of data demands the elaboration of techniques that are able to cope with class imbalances in order to capture correctly the less frequent emotions, i.e., Happy, Sad and Angry (13%, 17%, 17% and 53% for Happy, Sad, Angry and Other, respectively). To handle imbalance among classes we applied a penalty weight to the loss function, proportional to the respective class frequencies.

Given that the provided data do not contain any Twitter-specific informative sign, such as hashtags and user mentions, we tried to keep the preprocessing step as simple as possible. Thus, we replaced repeated emojis and punctuation with single ones, and substituted slang abbreviations to their full expression (e.g. "bcz" is substituted by "because").

3 System Overview

The proposed emotion analysis methodology is composed by (i) the formulation of the suitable representation schemes for the input data, and (ii) the implementation of an elaborative deep neural network architecture to map these schemes to their associated labels and the appropriate neural layers.

3.1 Embedding Layer

Word Embeddings tend to become a necessary component of deep learning approaches, with the mapping of single dimensional words to their dense vector encodings to be a critical part of the job (Mikolov et al., 2013). Vector representations are exhaustively trained on large corpora to capture the semantic content of each word. One of the most utilized vector representation is offered by Standford GloVe pretrained word vectors. However, GloVe vectors do not handle one of the most important factors in sentiment analysis, the emojis. To expand their capabilities, we append GloVe embeddings with pretrained emoji2vec, as proposed by Eisner et al. (2016). In addition, we utilized 23 extra features to enhance our pretrained word embeddings. Specifically, we elaborated 13 mood-oriented emotions, provided by the DepecheMood lexicon, as proposed by Staiano and Guerini (2014), in order to capture words mood intentions, as well as 10 NRC emotion relation scores. The utility of emotion scores is manifested in various studies (Mohammad and Turney, 2013; Kiritchenko et al., 2014). The enhancement led to a dense 323 dimensions vector representation for each word (300d-GloVe + 13d-DepecheMood emotions + 10d-NRC sentiment scores), which after re-normalization fed a recurrent neural network.

3.2 Bidirectional LSTMs and Siamese Network Architecture

In the recent years, deep learning models and in particular convolutional neural network (CNN) architectures, have become a popular and a favorable choice for several artificial intelligence tasks. However, the recurrent nature of textual data implies the need for architectures that are able to capture data of sequential nature information, which CNNs are unable to manage. To cope with the recurrent nature of textual data as well as with the sentiment contradictions that may occur in both text directions we utilize a bidirectional LSTM network architecture. In particular, we used three bidirectional LSTMs (Hochreiter and Schmidhuber, 1997) in a siamese like architecture (Bromley et al., 1994) in order to map all turns into the same vector space (Figure 1). To determine the sentiment impact of the last utterance, given the previous two, we introduce bidirectional LSTM hidden states, for each time-step and utterance. The hidden states are calculated using the same weights for each input.

3.3 Attention layer

To focus on the most significant time-sample, an attention layer (Bahdanau et al., 2014) is added on top of the regular LSTMs in order to capture and assign an importance attention factor to the hidden states, forming the so-called attentive vector $\vec{s}$:

$$r_t = tanh(W_h h_t + b_t) \quad (1)$$

$$a_t = \frac{e^{r_t}}{\sum_{j=0}^{T} e^{r_t}}, \quad \sum_{t=1}^{n} a_t = 1 \quad (2)$$

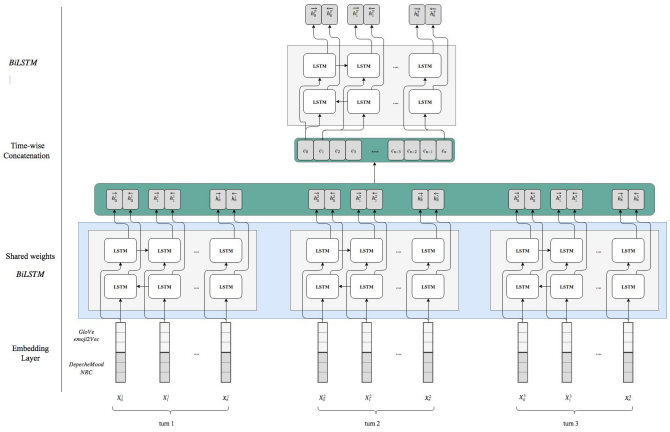

Figure 1: Siamese alike architecture, containing two layers of bidirectional LSTMs.

$$\vec{s} = \sum_{t=0}^{T} a_t h_t \tag{3}$$

where W_h and b_t are the LSTM model weights, to be optimized during training.

3.4 Dense network

To unfold and map the devised feature set we implemented a four-layered deep dense neural network, equiped with unigram and bigram Tf-idf weights of each utterance. Each neuron is activated by a ReLu function, and the final layer is concatenated with the attentive vector as defined in sub-section 3.3.

3.5 Proposed method

As already mentioned, we utilize two different and combined schemes to represent and train processed data, (i) an embedded matrix that feeds the Embeddings layer (as described in sub-section 3.1), and (ii) a uni/bi-gram Tf-idf training schema to be processed by the a layered dense DNN. In addition, the Embedding layer is connected with a siamese like bidirectional LSTM architecture, containing 164 units each. As shown in Figure 2, on top of the shared weighted LSTMs we add another bidirectional LSTM layer, also containing 164 units. To boost LSTM performance, we apply an attention mechanism on top of it, concluding to an attentive vector, as described in sub-section 3.3. We will refer to this subnetwork as *S-LSTM*, which includes the siamese like layers extended with an attentive mechanism. The Tf-idf features, as extracted for each conversation turn, are mapped onto two-layered dense neural networks containing 84 neurons each, with a ReLu non-linear activation function. The output of these layers is then concatenated and followed by another dense 84-neuron layer, creating a vector $\vec{d}$. We refer to this dense sub-network as *F-DNN*. The output vectors, $\vec{s}$ and $\vec{d}$, from the respective *S-LSTM* and *F-DNN* sub-networks are then concatenated and feed a final softmax activated dense network. The whole setup presents a combination of different and heterogeneous deep learning models.

3.6 Training

To train our model we adopted several regularization techniques to prevent overfitting. Therefore, we applied Dropout (Srivastava et al., 2014) to randomly deactivate neurons during forward training pass. We empirically set dropout parameters to 0.3 for every model layer, as well as recurrent connections of LSTM units (Gal and Ghahramani, 2016). In addition, we utilized L_2 regularization penalty loss function to every LSTM unit exceeding weight limits. Finally, we apply early-stopping technique to terminate training when loss on the development phase stop decreasing. To optimize our network we adopted Adam optimizer (Kingma and Ba, 2014) using cross entropy loss.

4 Results

Our proposed system achieved a micro f1-score ($\mathbf{f1}_\mu$) of 0.743, ranked the 26^{th} in the SemEval 2019 EmoContext task. Results are presented in Table 1 and compared with different approaches. EmoContext organisers proposed a baseline classifier (referred as Baseline) that exhibits a f1-score of 0.587(Chatterjee et al., 2019). In Table 1, we compare the proposed method with the *S-LSTM* and *F-DNN* implementations, described in 3.4 and 3.5, respectively. Moreover, we present results for the SS-BED system, proposed by Gupta et al.

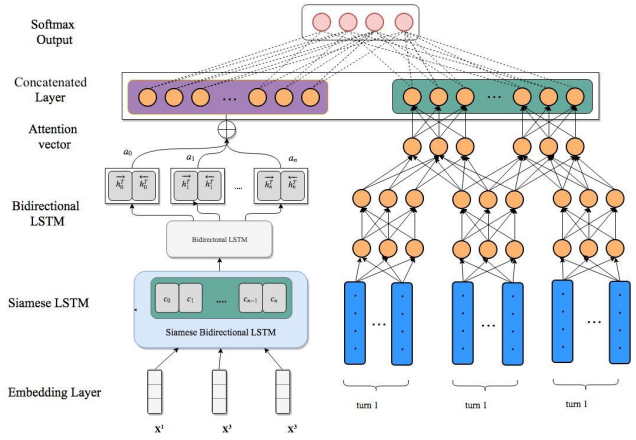

Figure 2: **Proposed Method**. the left model (S-LSTM) is composed by two LSTM layers followed by an attentive mechanism; the right model (F-DNN) is composed by two dense layers for each utterance followed by two additional dense layers

(2017). Compared with the other approaches, the proposed method exhibits a significantly higher f1-score for the Happy and Angry classes, 0.71 and 0.75, respectively. SS-BED system achieves a better performance for the Sad class (0.81) but it exhibits a poor performance for the Happy class (0.59).

To assess the importance of emoji embeddings and of the introduced mood and emotion feature sets which appended GloVe embeddings (G), we conducted additional experiments retaning the same siamese neural architecture (S-LSTM). In each experiment we extended GloVe embeddings with a respective feature set, i.e., the emoji2vec embeddings (E), the 13 DepecheMood (D) mood intensions, and the 10 NRC emotion relations (N). Compared to the S-LSTM$_G$ neural classifier, the use of additional embedding sets improve slightly the $f1_\mu$ performance, with the utilization of the emoji embeddings to achieve the highest increase (S-LSTM$_{G+E}$, from 0.67 to 0.70). But their yield remains lower compared to the respective architecture where all embedding sets are utilised (S-LSTM$_{all}$, 0.72).

In summary, the results demonstrate the superiority of the proposed method over all other approaches, and signifies its ability to succeed stable results over the different sentiment classes.

5 Conclusion

In this study we implemented a combination of two different representations and respective training schemes for the input data. First, we extended pretrained GloVe embedding vectors with emojis and appended 23 additional emotional features. In addition, we developed an S-LSTM model, containing a siamese alike bidirectional LSTM architecture with its output to feed another bidirectional LSTM layer followed by an attention layer. Furthermore, we transformed input data by their Tf-idf weight representations in order to feed a dense deep neural network (F-DNN).

System	**f1**$_{Happy}$	**f1**$_{Sad}$	**f1**$_{Angry}$	**f1**$_\mu$
Baseline	-	-	-	0.58
SS-BED	0.59	**0.81**	0.74	0.71
F-DNN	0.70	0.68	0.73	0.70
S-LSTM$_{all}$	0.69	0.76	0.71	0.72
S-LSTM$_G$	0.67	0.69	0.65	0.67
S-LSTM$_{G+E}$	0.65	0.71	0.74	0.70
S-LSTM$_{G+D}$	0.68	0.68	0.71	0.69
S-LSTM$_{G+N}$	0.66	0.71	0.68	0.68
Proposed	**0.71**	0.77	**0.75**	**0.74**

Table 1: Comparison results: Proposed method vs. other approaches (**f1**$_\mu$ refer to the f1-micro metric)

All model features are appropriately mapped and concatenated in order to feed the final dense softmax layer. Comparative results demonstrate the superiority of the proposed method over other approaches and single network models, as well as its robustness with regard the stability of performance over different emotional classes. Thus we could state that the proposed methodology defines an end-to-end solution on sentiment analysis and classification tasks, suited for imbalanced data, with the ability as well, to cope with huge amounts of data.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. 1994. Signature verification using a" siamese" time delay neural network. In *Advances in neural information processing systems*, pages 737–744.

Ankush Chatterjee, Kedhar Nath Narahari, Meghana Joshi, and Puneet Agrawal. 2019. Semeval-2019 task 3: Emocontext: Contextual emotion detection in text. In *Proceedings of The 13th International Workshop on Semantic Evaluation (SemEval-2019)*, Minneapolis, Minnesota.

Ben Eisner, Tim Rocktäschel, Isabelle Augenstein, Matko Bošnjak, and Sebastian Riedel. 2016. emoji2vec: Learning emoji representations from their description. *arXiv preprint arXiv:1609.08359*.

Yarin Gal and Zoubin Ghahramani. 2016. A theoretically grounded application of dropout in recurrent neural networks. In *Advances in neural information processing systems*, pages 1019–1027.

Umang Gupta, Ankush Chatterjee, Radhakrishnan Srikanth, and Puneet Agrawal. 2017. A sentiment-and-semantics-based approach for emotion detection in textual conversations. *CoRR*, abs/1707.06996.

Viktor Hangya and Richárd Farkas. 2017. A comparative empirical study on social media sentiment analysis over various genres and languages. *Artificial Intelligence Review*, 47(4):485–505.

Sepp Hochreiter and Jrgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.*, 9(8):1735–1780.

Zhao Jianqiang, Gui Xiaolin, and Zhang Xuejun. 2018. Deep convolution neural networks for twitter sentiment analysis. *IEEE Access*, 6:23253–23260.

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

Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif Mohammad. 2014. Nrc-canada-2014: Detecting aspects and sentiment in customer reviews. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 437–442.

Kurniawati Kurniawati, Graeme G Shanks, and Nargiza Bekmamedova. 2013. The business impact of social media analytics. In *ECIS*, volume 13, page 13.

Bing Liu. 2015. *Sentiment analysis: Mining opinions, sentiments, and emotions*. Cambridge University Press.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Saif M Mohammad and Peter D Turney. 2013. Nrc emotion lexicon. *National Research Council, Canada*.

Jonas Mueller and Aditya Thyagarajan. 2016. Siamese recurrent architectures for learning sentence similarity. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, pages 2786–2792. AAAI Press.

Braja Gopal Patra, Soumadeep Mazumdar, Dipankar Das, Paolo Rosso, and Sivaji Bandyopadhyay. 2016. A multilevel approach to sentiment analysis of figurative language in twitter. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 281–291. Springer.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation. In *EMNLP*, volume 14, pages 1532–1543.

Rolandos Alexandros Potamias, Georgios Siolas, and Andreas Stafylopatis. 2019. A robust deep ensemble classifier for figurative language detection. In *20th International Conference on Engineering Applications of Neural Networks*. Springer.

Rosenthal Sara, Ritter Alan, Nakov Preslav, and Stoyanov Veselin. 2014. Semeval-2014 task 9: Sentiment analysis in twitter. In *Proc. of the 8th International Workshop on Semantic Evaluation*, pages 73–80.

Nikhil Kumar Singh, Deepak Singh Tomar, and Arun Kumar Sangaiah. 2018. Sentiment analysis: a review and comparative analysis over social media. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–21.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Jacopo Staiano and Marco Guerini. 2014. Depechemood: a lexicon for emotion analysis from crowd-annotated news. *arXiv preprint arXiv:1405.1605*.

Mike Thelwall, Kevan Buckley, Georgios Paltoglou, Di Cai, and Arvid Kappas. 2010. Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology*, 61(12):2544–2558.