EmoDet at SemEval-2019 Task 3: Emotion Detection in Text using Deep Learning

Hani Al-Omari
alomarihani1997@gmail.com

Malak Abdullah
mabdullah@just.edu.jo

Nabeel Bassam
nabeelbassam98@gmail.com

Department of Computer Science
Jordan University of Science and Technology
Irbid, Jordan

Abstract

Task 3, EmoContext, in the International Workshop SemEval 2019 provides training and testing datasets for the participant teams to detect emotion classes (Happy, Sad, Angry, or Others). This paper proposes a participating system (EmoDet) to detect emotions using deep learning architecture. The main input to the system is a combination of Word2Vec word embeddings and a set of semantic features (e.g. from AffectiveTweets Weka-package). The proposed system (EmoDet) ensembles a fully connected neural network architecture and LSTM neural network to obtain performance results that show substantial improvements (F1-Score 0.67) over the baseline model provided by Task 3 organizers (F1-score 0.58).

1 Introduction

The past decades have seen an explosive growth of user-generated content through social media platforms. People are expressing online their feelings and opinions on a variety of topics on a daily basis. Tracking and analyzing public opinions from social media can help to predict certain political events or predicting people's attitude towards certain products. Therefore, detecting sentiments and emotions in text have gained a considerable amount of attention(Mohammad et al., 2018). Researchers and scientists in different fields considered this a promising topic (Abdullah et al., 2018; Liu, 2012). Many machine learning approaches have been used to detect and predict emotions and sentiments. Recently, the deep neural network (DNN) is attracting more researchers as they have been benefited from the high-performance graphics processing unit (GPU) power (Abdullah et al., 2018; Dos Santos and Gatti, 2014).

The shared task (Task 3: "EmoContext") in SemEval-2019 workshop has been designed for understanding emotions in textual conversations (Chatterjee et al., 2019). In this task, the participants are given a textual dialogue i.e. a user utterance along with three turns of context. The participant teams have to classify the emotion of user utterance as one of the emotion classes: Happy, Sad, Angry or Others. Further details about Task 3 and the datasets appear in Section 3.

This paper describes our team approach to detect and classify emotions. The input has been represented as word vectors (Mikolov et al., 2013b) and a set of different features which are applied to different neural network architecture to obtain the results. The performance of the system shows substantial improvements F1-Score over the baseline model provided by Task 3 organizers.

The remainder of this research paper is organized as follows: Section 2 gives a brief overview of existing work on social media emotion and sentiment analyses. Section 3 presents the requirements of SemEval Task3 and examines our proposed system to determine the presence of emotion in conversational text. Section 4 summarizes the key findings of the study and the evaluations and concludes with future directions for this research.

2 Related Work

Defining and theorizing emotions had been investigated by several psychology researchers (Plutchik, 1990; Ekman and Keltner, 1997). The basic emotions according to Ekman (Ekman and Keltner, 1997) had been identified as anger, disgust, fear, happiness, sadness, and surprise. A little corpus exists for emotion labeling with text. Recently, several shared tasks and challenges had been introduced for detecting the intensity of emo-

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 200–204
Minneapolis, Minnesota, USA, June 6–7, 2019.

tion felt by the speaker of a tweet (Mohammad et al., 2018; Strapparava and Mihalcea, 2007). A group of researchers (Mohammad and Bravo-Marquez, 2017) introduced the WASSA- 2017 shared task of detecting the intensity of emotion felt by the speaker of a tweet. The previous Semeval Task1 (Mohammad et al., 2018) also introduced a dataset (annotated tweets) for emotion detection. The state-of- the-art systems in the previous competitions used different approaches of ensembling different deep neural network-based models, representing tweets as word2vec/doc2vec embedding vectors and extracting semantic features. Our system is using word2vec embedding vectors (Mikolov et al., 2013a) and extracted features using a Weka package, AffectiveTweet, (Bravo-Marquez et al., 2014), also extracting embedding from the text using deeMoji model (Felbo et al., 2017).

3 Our Approach

Our system has the ability to determine the emotion (Happy, Sad, Angry and Other) in English textual dialogue with F1-Score over 0.67. Figure 1 shows the general structure of the system. More details for the systems components are shown in the following subsections: Section 3.1 describes the systems input and preprocessing step. Section 3.2 lists the extracted feature vectors, and Section 3.3 details the system's architecture of neural networks. Section 3.4 discusses the output details.

3.1 Input and Preprocessing

The shared task (Task 3: EmoContext) provides training and testing datasets to be used by all participants. The number of training and testing datasets for each emotion can be shown in Table 1.

	Train Data	Test Data
Anger	5656	298
Happy	4385	284
Sad	5588	250
Other	17286	4677
Total	32915	5509

Table 1: Training and Testing Datasets

The training dataset contains 5 columns:

ID - Contains a unique number to identify each training sample.

Turn 1 - Contains the first turn in the three turn conversation, written by User 1.

Turn 2 - Contains the second turn, which is a reply to the first turn in conversation and is written by User 2.

Turn 3 - Contains the third turn, which is a reply to the second turn in the conversation, which is written by User 1.

Label - Contains the human-judged label of Emotion of Turn 3 based on the conversation for the given training sample. It is always one of the four values - 'happy', 'sad, 'angry' and 'others'.

For testing dataset, the 5th column - 'Label' is absent. See Table 2 for more clarification.

The prepossessing methods applied for the data include converting the text into lower case, stemming the words and removing of extraneous white spaces. Punctuation has been treated as individual words (".,?!:;()[]#@'). It's worth mentioning that removing stop-words dissolved the meaning of the sentence, therefore we didn't remove them.

3.2 Feature Vector

We have explored different features to represent each turn and the concatenated turns. Our approach extracts feature vectors from texts with a total of 2753 dimensions (Check Table 3). We have applied the same methods for each turn and the concatenated turns.

Each turn is represented as a 300-dimensional vector using the pretrained word2vec embedding model that is trained on Google News (Mikolov et al., 2013a). We have used the summation technique to represent every turn or conversation. In addition to that, each turn/conversation is represented as 145 dimensional vectors by concatenating three vectors obtained from the AffectiveTweets Weka-package bravo2014meta, mohammad2017wassa, 43 features have been extracted using the TweetToLexiconFeatureVectorattribute that calculates attributes for a tweet using a variety of lexical resources; two-dimensional vector using the sentiments strength feature from the same package, and the final 100 dimensional vectors is obtained by vectorizing the tweets to embeddings attribute also from the same package. We have also extracted 2302 dimensions vector using the attention layer of DeepMoji model (Felbo et al., 2017). Finally, we have used the NRC Valence, Arousal, and Dominance Lexicon to extract the last 4-dimensional

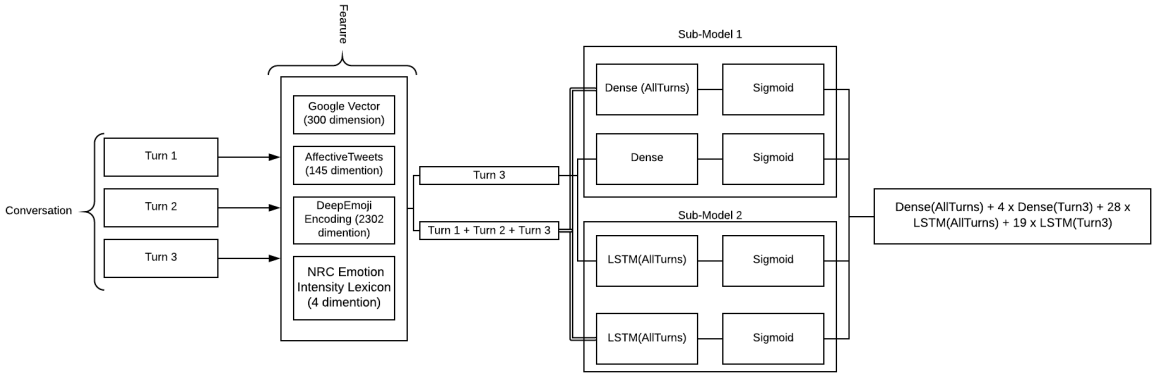

Figure 1: The architecture of our approach

id	Turn1	Turn2	Turn3	label
156	You are funny	LOL I konw that.	:)	happy
187	Yeah exactly	Like you said, like brother like sister ;)	Not in the least	others

Table 2: Examples of datasets format

vector to represent Anger, fear, sadness, and joy (Mohammad, 2018).

	Dimension
Word2Vec	300
AffectiveTweets	145
DeepMoji	2302
NRC	4
Total	2753

Table 3: Feature vectors

3.3 Network Architecture

Knowing that Deep Neural Networks (DNN) is showing significant improvements over traditional Machine Learning (ML) based approaches on classification tasks(LeCun et al., 2015). This derives more researchers to apply it recently for detecting sentiments and emotions. The standard Recurrent Neural Network (RNN) is distinguished from Feed-forward network with a memory. A special kind of RNNs are Long Short-Term Memory Network (LSTM) (Hochreiter and Schmidhuber, 1997), which is composed of a memory cell, an input gate, an output gate and a forget gate.

The architecture of our system consists of two sub-models that use both: feed-forward (Dense) and LSTM. For our first sub-Model, the Input 2753-dimensional vector feeds a fully connected neural network with three dense hidden layers of 500, 200 and 80 neurons for each layer. The activation function for each layer is ReLU (Goodfellow et al., 2013). Two dropouts have been added to this sub-model, which are 0.3 and 0.2 after the first and the second layers. The output layer consists of 4 sigmoid neurons to predict the class of emotion in each conversation. For optimization, we use Stochastic Gradient Descent (SGD) optimizer (lr=.001, decay=1×10^{-6}, and momentum = 0.9) augmenting for MSE loss function and ACCURACY metrics. We have also saved the output predictions weights to predict the testing data sets. The fit function uses number of epochs = 60, batch size=32, validation split= 33%.

In the second sub-model, the same 2753-dimensional vector feeds an LSTM by using an embedding layer of 500-dimensions. The LSTM layer consists of 300 neurons with using Dropout 0.3 after the LSTM layer to avoid over-fitting. A dense layer with 200 neurons is added and followed by four sigmoid neurons to predict the emotion class in each conversation. For optimization, we use the same method as the first sub-model. We have also used early stopping technique to get the best result. finally, we have saved the output prediction weights to predict the testing data sets. The fit function uses number of epochs = 80, batch size=8, validation split= 33%.

Formula	micro-F1
$Dense(Turn1) + 2 \times Dense(Turn3)$	0.605355
$Dense(Turn1) + 2 \times Dense(Turn3) + 2 \times Dense(All)$	0.618162
$2 \times Dense(All) + 3 \times Dense(Turn3) + 3 \times LSTM(All)$	0.626
$Dense(All) + part3 + 3 \times LSTM(All)$	0.636
$Dense(All) + part3 + 10 \times LSTM(All)$	0.656165
$Dense(All) + 4 \times Dense(Turn3) + 28 \times LSTM(All) + 19 \times LSTM(Turn3)$	0.6714

Table 4: Weight Ensembling

Conversation	Turn 1	Turn 2	Turn 3
micro-F1	0.26379	0.08435	0.58920

Table 5: Using Sub-model 1 - Dense Layer

Conversation	Turn 1	Turn 2	Turn 3
micro-F1	0.20734	0.13543	0.44364

Table 6: Using Sub-model 1 - Dense Layer plus removing 70% of others randomly

System	Epoch	micro-F1
LSTM (All Conversation)	40	0.5376
LSTM (All Conversation)	80	0.6094
LSTM (All Conversation)	114	0.5096
Dense (All Conversation)	60	4677

Table 7: Best Epoch for both LSTM and Dense

3.4 Output and result

In the beginning, we have analyzed all the conversation (turn1 + turn2 + turn3) using both sub-model systems. We have noticed that the third turn of the conversation provides better predictions of emotion's class, see in Table 5. Removing 70% of the others randomly in the training data set led to bad predictions so we didn't apply this method, see Table 6. One of the key findings is noticing that LSTM gives better prediction than the feed-forward system for the whole conversation, see the result in Table 7. For the final stage, we have combined both sub-models results to produce a real value number between 0 and 1. It has shown that the second sub-model gives higher accuracy than the first sub-model. Applying different amount of weights for four prediction led us to find out that the correct formula for our system using turn3 alone as a sub-model and All-Conversation prediction and from the second sub-Model turn3 and All-Conversation and combining them together in a formula, It showed a higher F1-Score equal to 0.67. Also, it's worth mentioning that we used Grid Search to find the best parameters for the formula (Check Table 4).

4 Conclusion

In this paper, we have presented our system EmoDet that uses deep learning architectures for detecting the existence of emotions in a text. The performance of the system surpasses the performance of the baselines model indicating that our approach is promising. In this system, we uses word embedding models with feature vectors extracted using the AffectiveTweets package and Deepmoji model. These vectors feed different deep neural network architectures, feed-forward and LSTM, to obtain the predictions. We use the SemEval-2019 Task 3s datasets as input for our system and show that EmoDet has a high proficiency in detecting emotions in a conversational text and surpasses the F1-score Baseline models performance, which is provided by the SemEval-Task 3 organizers.

References

Malak Abdullah, Mirsad Hadzikadicy, and Samira Shaikhz. 2018. Sedat: Sentiment and emotion detection in arabic text using cnn-lstm deep learning. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 835–840. IEEE.

Felipe Bravo-Marquez, Marcelo Mendoza, and Barbara Poblete. 2014. Meta-level sentiment models for big social data analysis. *Knowledge-Based Systems*, 69:86–99.

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.

Cicero Dos Santos and Maira Gatti. 2014. Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 69–78.

Paul Ekman and Dacher Keltner. 1997. Universal facial expressions of emotion. *Segerstrale U, P. Molnar P, eds. Nonverbal communication: Where nature meets culture*, pages 27–46.

Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. *arXiv preprint arXiv:1708.00524*.

Ian J Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. 2013. Maxout networks. *arXiv preprint arXiv:1302.4389*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature*, 521(7553):436.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

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

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

Saif Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. Semeval-2018 task 1: Affect in tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 1–17.

Saif M. Mohammad. 2018. Word affect intensities. In *Proceedings of the 11th Edition of the Language Resources and Evaluation Conference (LREC-2018)*, Miyazaki, Japan.

Saif M Mohammad and Felipe Bravo-Marquez. 2017. Wassa-2017 shared task on emotion intensity. *arXiv preprint arXiv:1708.03700*.

Robert Plutchik. 1990. Emotions and psychotherapy: A psychoevolutionary perspective. In *Emotion, psychopathology, and psychotherapy*, pages 3–41. Elsevier.

Carlo Strapparava and Rada Mihalcea. 2007. Semeval-2007 task 14: Affective text. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 70–74.