SSN_NLP at SemEval-2019 Task 3: Contextual Emotion Identification from Textual Conversation using Seq2Seq Deep Neural Network

B. Senthil Kumar, D. Thenmozhi, Chandrabose Aravindan, S. Srinethe
Department of CSE, SSN College of Engineering, India
{senthil,theni_d,aravindanc}@ssn.edu.in
srinethe16108@cse.ssn.edu.in

Abstract

Emotion identification is a process of identifying the emotions automatically from text, speech or images. Emotion identification from textual conversations is a challenging problem due to absence of gestures, vocal intonation and facial expressions. It enables conversational agents, chat bots and messengers to detect and report the emotions to the user instantly for a healthy conversation by avoiding emotional cues and miscommunications. We have adopted a Seq2Seq deep neural network to identify the emotions present in the text sequences. Several layers namely embedding layer, encoding-decoding layer, softmax layer and a loss layer are used to map the sequences from textual conversations to the emotions namely Angry, Happy, Sad and Others. We have evaluated our approach on the EmoContext@SemEval2019 dataset and we have obtained the micro-averaged F1 scores as 0.595 and 0.6568 for the pre-evaluation dataset and final evaluation test set respectively. Our approach improved the base line score by 7% for final evaluation test set.

1 Introduction

Emotion identification is a process of identifying the emotions automatically from different modalities. Several research work have been presented on detecting emotions from text (Rao, 2016; Abdul-Mageed and Ungar, 2017; Samy et al., 2018; Al-Balooshi et al., 2018; Gaind et al., 2019), speech (Arias et al., 2014; Amer et al., 2014; Lim et al., 2016), images (Shan et al., 2009; Ko, 2018; Ayvaz et al., 2017; Faria et al., 2017; Mohammadpour et al., 2017) and video (Matsuda et al., 2018; Hossain and Muhammad, 2019; Kahou et al., 2016). Emotion understanding from video may be easier by analyzing the body language, speech variations and facial expressions. However, identification of emotions from textual conversations is a challenging problem due to absence of above factors. Emotions in text are not only identified by its cue words such as *happy, good, bore, hurt, hate* and *fun*, but also the presence of interjections (e.g. "whoops"), emoticons (e.g. ":)"), idiomatic expressions (e.g. "am in cloud nine"), metaphors (e.g. "sending clouds") and other descriptors mark the existence of emotions in the conversational text. Recently, the growth of text messaging applications for communications require emotion detection from conversation transcripts. This helps conversational agents, chat bots and messengers to avoid emotional cues and miscommunications by detecting the emotions during conversation. EmoContext@SemEval2019 shared task (Chatterjee et al., 2019) goal is to encourage more research in the field of contextual emotion detection in textual conversations. The shared task focuses on identifying emotions namely Angry, Happy, Sad and Others from conversation with three turns. Since, emotion detection is a classification problem, research works have been carried out by using machine learning with lexical features (Sharma et al., 2017) and deep learning with deep neural network (Phan et al., 2016) and convolutional neural network (Zahiri and Choi, 2018) to detect the emotions from text. However, we have adopted Seq2Seq deep neural network for detecting the emotions from textual conversations which include sequence of phrases. This paper elaborates our Seq2Seq approach for identifying emotions from text sequences.

2 Related Work

This section reviews the research work reported for emotion detection from text / tweets (Perikos and Hatzilygeroudis, 2013; Rao, 2016; Abdul-Mageed and Ungar, 2017; Samy et al., 2018; Al-Balooshi et al., 2018; Gaind et al., 2019) and text

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 318–323
Minneapolis, Minnesota, USA, June 6–7, 2019.

conversations (Phan et al., 2016; Sharma et al., 2017; Zahiri and Choi, 2018).

Sharma et al. (2017) proposed a methodology to create a lexicon - a vocabulary consisting of positive and negative expressions. This lexicon is used to assign an emotional value which is derived from a fuzzy set function. Gaind et al. (2019) classified twitter text into emotion by using textual and syntactic features with SMO and decision tree classifiers. The tweets are annotated manually by Liew and Turtle (2016) with 28 fine-grained emotion categories and experimented with different machine learning algorithms. Results show that SVM and BayesNet classifiers produce consistently good performance for fine-grained emotion classification. Phan et al. (2016) developed an emotion lexicon from WordNet. The conversation utterances are mapped to the lexicons and 22 features are extracted using rule-based algorithm. They used fully connected deep neural network to train and classify the emotions. TF-IDF with handcrafted NLP features were used by Al-Balooshi et al. (2018) in logistic regression, XGBClassifier and CNN+LSTM for emotion classification. The authors found that the logistic regression performed better than the deep neural network model. All the models discussed above considered the fine-grained emotion categories and used the twitter data to create a manually annotated corpus. These models used the rule-based or machine learning based algorithms to classify the emotion category.

A new C-GRU (Context-aware Gated Recurrent Units) a variant of LSTM was proposed by Samy et al. (2018) which extracts the contextual information (topics) from tweets and uses them as an extra layer to determine sentiments conveyed by the tweet. The topic vectors resembling an image are fed to CNN to learn the contextual information. Abdul-Mageed and Ungar (2017) built a very large dataset with 24 fine-grained types of emotions and classified the emotions using gated RNN. Instead of using basic CNN, a new recurrent sequential CNN is used by Zahiri and Choi (2018). They proposed several sequence-based convolution neural network (SCNN) models with attention to facilitate sequential dependencies among utterances. All the models discussed above show that the emotion prediction can be handled using variants of deep neural network such as C-GRU, G-RNN and Sequential-CNN. The commonality between the above models are the variations of RNN or LSTM. This motivated us to use the Sequence-to-Sequence (Seq2Seq) model which consists of stacked LSTMs to predic the emotion labels conditioned on the given utterance sequences.

3 Data and Preprocessing

We have used the dataset provided by EmoContext@SemEval2019 shared task in our approach. The dataset consists of training set, development set and test set with 30160, 2755 and 5509 instances respectively. The dataset contains sequence id, text sequences with three turns which include user utterance along with the context, followed by emotion class label. The task is to label the user utterance as one of emotion class: happy, sad, angry or others. The textual sequences contain many short words. In preprocessing, these words are replaced with original or full word. We resort to build a look-up table which replace ‘’m’, with ‘am’, ‘’re’ with ‘are’, ‘’ere’ with ‘were’, ‘n’t’ with ‘not’, ‘’ll’ with ‘will’, ‘’d’ with ‘would’, ‘what’s’ with ‘what is’ and ‘it’s’ with ‘it is’. The sequences are converted to lower case. Also, the three turns/sentences are delimited with “eos” in the input sequences.

4 Methodology

Seq2Seq model is the most popular model in learning the target sequence conditioned on the source sequence. The Seq2Seq model is adopted to map the sequences of n words with a target label (n:1 mapping). This model has an embedding layer, an encoder, a decoder and a projection layer as shown in Figure 1.

Once the dialogue sentences are preprocessed, the first three turns of each instance are considered as the input sequences w_1, w_2,..,w_n, and the corresponding label e is considered as the target sequence. For example, the given instance “13 Bad Bad bad! That’s the bad kind of bad. I have no gf sad” is converted into input sequence “bad eos bad bad that is the bad kind of bad eos i have no gf” and target label ”sad”. The input sequences and the target label are converted into its corresponding word embeddings by the embedding layer. The vector representation for each word is derived at embedding layer by choosing a fixed vocabulary of size V for input sequences and target labels.

Now, the encoder which uses Bi-LSTM, encode these embeddings into a fixed vector representa-

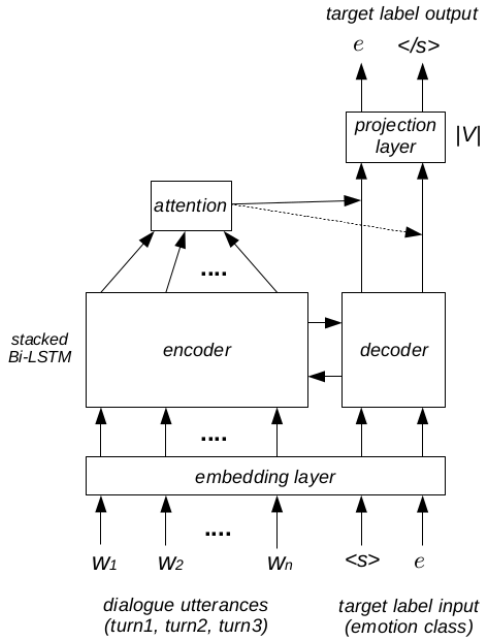

Figure 1: System Architecture

tion s which also represents the summary of input sequences. Once the source sequences are encoded, the last hidden state of the encoder is used to initialize the decoder. The projection layer is fed with the tensors of the target output label. Given the hidden state h_t , the decoder predicts the label e_t. However, h_t and e_t are conditioned on the previous output e_{t-1} and on the summary s of the input sequence. The projection layer is a dense matrix to turn the top hidden states of decoder to logit vectors of dimension V. Given the logit values, the training loss is easily minimized by using standard SGD optimizer with a learning rate. The model is also trained with the attention mechanism, which computes the attention weight by comparing the current decoder hidden state with all encoder states. The detailed description of working principle about Seq2Seq model is described in (Sutskever et al., 2014).

We have adopted Neural Machine Translation[1] code to implement our Seq2Seq deep neural network. Several variations have been implemented by varying the number of layers, units and attention mechanisms. It is evident from the earlier experiments (Sutskever et al., 2014; Thenmozhi et al., 2018) that bi-directional LSTM performs better for short text sequences. Hence, we have used it for encoding and decoding processes. The models were trained for 30000 steps with drop out

[1]https://github.com/tensorflow/nmt

Models	F1 μ Score
8L_SL_No_split	0.523
8L_NB_No_split	0.527
8L_NB_TV_split	0.5296
8L_NB_EOS_TV_split	0.5499
16L_NB_No_split	0.510
16L_NB_TV_split	0.526
16L_NB_EOS_TV_split	0.547
32L_NB_EOS_No_split	0.531
32L_NB_EOS_TV_split	0.5398
2L_NB_EOS_No_split	0.544
2L_NB_EOS_TV_split	**0.595**

Table 1: Development Set Micro-avereged F1 Score.

of 0.2. We have utilized two attention wrappers namely Normed_Bahdanau (NB) (Sutskever et al., 2014; Bahdanau et al., 2014) and Scaled_Luong (SL) (Luong et al., 2015, 2017).

Since, the model was developed using deep learning technique, it does not require much of linguistic features such as stemming, case normalization and PoS in identifying the emotion cue words. These linguistic phenomena could be captured by the encoder RNNs in sequence-to-sequence (Seq2Seq) model. The other statistical features such as the word frequency are also not considered as input to the model, because the presence of particular cue alone does not guarantee to detect emotions in the text.

5 Results

Our approach is evaluated on EmoContext@SemEval2019 data set. During development, we have implemented our variations with and without end of sentence (EOS) delimiter. We have built the models using entire training set (No_split) and train-validation splits (TV_split). 27160 and 3000 instances from training data were considered as training and validation set in TV_split. The performance was measured in terms of micro-averaged F1 score (F1μ) for the three emotion classes namely Angry, Happy and Sad.

We have submitted eleven runs for EmoContext@SemEval2019 shared task on pre-evaluation dataset. The results obtained for pre-evaluation dataset are given in Table 1.

We observe from Table 1 that Normed_Bahdanau attention mechanism performs better than Scaled_Luong. Model building

Models	F1 μ Score
16U_TV_split_1	0.649422
32U_TV_split_1	0.416399
64U_TV_split_1	**0.656752**
128U_TV_split_1	0.626124
256U_TV_split_1	0.581599
16U_TV_split_2	0.59668
32U_TV_split_2	0.617944
64U_TV_split_2	0.618201
128U_TV_split_2	0.622652
256U_TV_split_3	0.642144
16U_TV_split_3	0.611716
32U_TV_split_3	0.567093
64U_TV_split_3	0.624924
128U_TV_split_3	0.655106
256U_TV_split_3	0.612288

Table 2: Final Evaluation Test Data Micro-avereged F1 Score .

Models	F1 Score		
	Happy	**Sad**	**Angry**
16U_TV_split_1	0.619	0.645	0.686
32U_TV_split_1	0.299	0.550	0.384
64U_TV_split_1	**0.633**	0.638	**0.695**
128U_TV_split_1	0.606	0.583	0.689
256U_TV_split_1	0.553	0.566	0.626
16U_TV_split_2	0.525	0.584	0.684
32U_TV_split_2	0.537	0.637	0.677
64U_TV_split_2	0.585	0.615	0.657
128U_TV_split_2	0.596	0.595	0.676
256U_TV_split_3	0.609	0.637	0.681
16U_TV_split_3	0.513	0.641	0.679
32U_TV_split_3	0.507	0.588	0.607
64U_TV_split_3	0.552	0.637	0.685
128U_TV_split_3	0.612	**0.664**	0.692
256U_TV_split_3	0.559	0.618	0.657

Table 3: Class-wise F1 Score for Final Evaluation Test Data.

with TV_split performs better than the model without split. The incorporation of delimiter text EOS also improved the performance of our approach. Further, the performance degrades with the increase in number of layers. Thus, 2 layered LSTM with TVsplit, EOS delimiter and Normed_Bahdanau attention mechanism perform better on the pre-evaluation dataset of EmoContext@SemEval2019 and this architecture is considered for evaluating the final-evaluation test set. The final evaluation submissions are based upon the variations in TV_split ratio and the number of units as 16, 32, 64, 128 and 256. For TV_split_1, the development set (2755 instances) given by EmoContext@SemEval2019 was considered as a validation set. The other two TV_splits are by keeping the validation set as 1/5 (TV_split_2) and 1/3 (TV_split_3) of training set. The results of our submissions on final evaluation test data are given in Table 2. It is observed from Table 2 that 64U_TV_split_1 model outperforms all the other models with 0.656752 F1μ score. This score is higher than the base line score with 7% improvement. Table 3 shows the class-wise performance of our models on final evaluation set. Our models perform better for Angry class than the other two classes namely Happy and Sad.

6 Conclusion

We have adopted a Seq2Seq deep neural network to identify the emotions present in the text sequences. Our approach is evaluated on the EmoContext@SemEval2019 dataset. The input sequences are pre-processed by replacing the short hand notations and by introducing a delimiter string. The sequence is vectorized using word embeddings and given to bi-directional LSTM for encoding and decoding. We have implemented several variations by changing the parameters namely, number of layers, units, attention wrappers, with and without delimiter string and train-validation split. The performance is measured using micro-averaged F1 score on three emotion class labels namely Angry, Happy and Sad. Our experiments on development set show that 2 layered LSTM with Normed_Bahdanau attention mechanism with delimiter string and train-validation split performs better than all the other variations. Three variations of train-validation split ratio were experimented on final evaluation test data by varying the number of units with the best parameter values that are learnt during the development phase. 64U_TV_split_1 model performs better than all the other runs we have submitted to the task. This model shows 7% improvement than the base line on final evaluation test set. Our Seq2Seq model can be improved further by incorporating the soft attention mechanism which uses joint distribution between attention and output layer (Shankar et al., 2018).

References

Muhammad Abdul-Mageed and Lyle Ungar. 2017. Emonet: Fine-grained emotion detection with gated recurrent neural networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 718–728.

Hessa AlBalooshi, Shahram Rahmanian, and Rahul Venkatesh Kumar. 2018. Emotionx-smartdubai_nlp: Detecting user emotions in social media text. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*, pages 45–49.

Mohamed R Amer, Behjat Siddiquie, Colleen Richey, and Ajay Divakaran. 2014. Emotion detection in speech using deep networks. In *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 3724–3728. IEEE.

Juan Pablo Arias, Carlos Busso, and Nestor Becerra Yoma. 2014. Shape-based modeling of the fundamental frequency contour for emotion detection in speech. *Computer Speech & Language*, 28(1):278–294.

Uğur Ayvaz, Hüseyin Gürüler, and Mehmet Osman Devrim. 2017. Use of facial emotion recognition in e-learning systems. *Information Technologies and Learning Tools*, 60(4):95–104.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

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

Ana Raquel Faria, Ana Almeida, Constantino Martins, Ramiro Gonçalves, José Martins, and Frederico Branco. 2017. A global perspective on an emotional learning model proposal. *Telematics and Informatics*, 34(6):824–837.

Bharat Gaind, Varun Syal, and Sneha Padgalwar. 2019. Emotion detection and analysis on social media. *arXiv preprint arXiv:1901.08458*.

M Shamim Hossain and Ghulam Muhammad. 2019. Emotion recognition using deep learning approach from audio–visual emotional big data. *Information Fusion*, 49:69–78.

Samira Ebrahimi Kahou, Xavier Bouthillier, Pascal Lamblin, Caglar Gulcehre, Vincent Michalski, Kishore Konda, Sébastien Jean, Pierre Froumenty, Yann Dauphin, Nicolas Boulanger-Lewandowski, et al. 2016. Emonets: Multimodal deep learning approaches for emotion recognition in video. *Journal on Multimodal User Interfaces*, 10(2):99–111.

Byoung Ko. 2018. A brief review of facial emotion recognition based on visual information. *sensors*, 18(2):401.

Jasy Suet Yan Liew and Howard R Turtle. 2016. Exploring fine-grained emotion detection in tweets. In *Proceedings of the NAACL Student Research Workshop*, pages 73–80.

Wootaek Lim, Daeyoung Jang, and Taejin Lee. 2016. Speech emotion recognition using convolutional and recurrent neural networks. In *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pages 1–4. IEEE.

Minh-Thang Luong, Eugene Brevdo, and Rui Zhao. 2017. Neural machine translation (seq2seq) tutorial. *https://github.com/tensorflow/nmt*.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.

Yuki Matsuda, Dmitrii Fedotov, Yuta Takahashi, Yutaka Arakawa, Keiichi Yasumoto, and Wolfgang Minker. 2018. Emotour: Multimodal emotion recognition using physiological and audio-visual features. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*, pages 946–951. ACM.

Mostafa Mohammadpour, Hossein Khaliliardali, Seyyed Mohammad R Hashemi, and Mohammad M AlyanNezhadi. 2017. Facial emotion recognition using deep convolutional networks. In *2017 IEEE 4th International Conference on Knowledge-Based Engineering and Innovation (KBEI)*, pages 0017–0021. IEEE.

Isidoros Perikos and Ioannis Hatzilygeroudis. 2013. Recognizing emotion presence in natural language sentences. In *International conference on engineering applications of neural networks*, pages 30–39. Springer.

Duc Anh Phan, Hiroyuki Shindo, and Yuji Matsumoto. 2016. Multiple emotions detection in conversation transcripts. In *Proceedings of the 30th Pacific Asia Conference on Language, Information and Computation: Oral Papers*, pages 85–94.

Yanghui Rao. 2016. Contextual sentiment topic model for adaptive social emotion classification. *IEEE Intelligent Systems*, 31(1):41–47.

Ahmed E Samy, Samhaa R El-Beltagy, and Ehab Hassanien. 2018. A context integrated model for multi-label emotion detection. *Procedia computer science*, 142:61–71.

Caifeng Shan, Shaogang Gong, and Peter W McOwan. 2009. Facial expression recognition based on local binary patterns: A comprehensive study. *Image and vision Computing*, 27(6):803–816.

Shiv Shankar, Siddhant Garg, and Sunita Sarawagi. 2018. Surprisingly easy hard-attention for sequence to sequence learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 640–645.

Shikhar Sharma, Piyush Kumar, and Krishan Kumar. 2017. Lexer: Lexicon based emotion analyzer. In *International Conference on Pattern Recognition and Machine Intelligence*, pages 373–379. Springer.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

D Thenmozhi, B Senthil Kumar, and Chandrabose Aravindan. 2018. Ssn_nlp@ iecsil-fire-2018: Deep learning approach to named entity recognition and relation extraction for conversational systems in indian languages. *CEUR*, 2266:187–201.

Sayyed M Zahiri and Jinho D Choi. 2018. Emotion detection on tv show transcripts with sequence-based convolutional neural networks. In *Workshops at the Thirty-Second AAAI Conference on Artificial Intelligence*.