CoAStaL at SemEval-2019 Task 3: Affect Classification in Dialogue using Attentive BiLSTMs

Ana Valeria González* and **Victor Petrén Bach Hansen*** and **Joachim Bingel** and **Isabelle Augenstein** and **Anders Søgaard**
University of Copenhagen, Dept. of Computer Science
Copenhagen, Denmark
ana|victor.petren|bingel|augenstein|soegaard@di.ku.dk

Abstract

This work describes the system presented by the CoAStaL Natural Language Processing group at University of Copenhagen. The main system we present uses the same attention mechanism presented in (Yang et al., 2016). Our overall model architecture is also inspired by their hierarchical classification model and adapted to deal with classification in dialogue by encoding information at the turn level. We use different encodings for each turn to create a more expressive representation of dialogue context which is then fed into our classifier. We also define a custom preprocessing step in order to deal with language commonly used in interactions across many social media outlets. Our proposed system achieves a micro F1 score of 0.7340 on the test set and shows significant gains in performance compared to a system using dialogue level encoding.

1 Introduction

Recognizing emotion is crucial to human-human communication and has for a long time been a goal in human-machine interaction. Although there has been growing interest in emotion detection across many fields (Liscombe et al., 2005; Agrafioti et al., 2012; Craggs and Wood, 2004), much of the work has focused on developing empathetic systems using multimodal approaches i.e. speech and gestures as well as text (Hazarika et al., 2018). Approaching emotion detection as a multimodal problem certainly makes sense, as face-face human communication involves many modalities, however, this fails to consider all the communication that is increasingly happening solely via chat, or written means. Detecting emotion in textual dialogue without the other modalities, such as work done by Gupta et al., can allow us to improve a number of applications dealing with social media interactions, opinion mining, and customer interactions, unfortunately, this is a great challenge that has remained largely unexplored. SemEval 2019 Task 3 attempts to encourage research in this direction. Given a user utterance and the previous two turns of context, the task consists in classifying the user utterance according to one of four emotion classes: happy, sad, angry or other. For a full description of the task see (Chatterjee et al., 2019). In this paper, we describe our turn-level attention model used to tackle this task, specifically using the attention mechanism presented in (Yang et al., 2016). Our model encodes turns in a conversation separately using an Attentive Bidirectional LSTM encoder. In the model presented in the shared task, the turn encoders do not share parameters, achieving a micro F1 score of 0.7340. The code for all experiments presented here is available. [1]

2 Related Work

Due to its many potential applications across many fields, detection of speaker emotional state in spoken dialogue systems has been studied extensively. Early studies showed that the use of prosody as well as speaking style features leads to increases in accuracy for emotion prediction (Ang et al., 2002; Devillers et al., 2002). Researchers have also shown that using domain specific features as well as speech signals can improve performance (Ai et al., 2006). Furthermore, there is plenty of work that tries to improve detection of affect in text using multiple modalities such as video or an embodied conversational agent (Alonso-Martin et al., 2013; Dmello and Graesser, 2010), however, detection of emotion solely based on text conversation data has not seen the same breakthroughs.

*Authors contributed equally

[1] https://github.com/coastalcph/emocontext

Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019), pages 169–174
Minneapolis, Minnesota, USA, June 6–7, 2019.

Very recently, work has started to emerge dealing with emotion detection in textual dialogue only. Majumder et al. introduced an attentive RNN model that treats each party of a conversation independently in order to provide specific representations for speaker and listener at a given point in a conversation. This model assumes that in a dialogue, the emotion of speaker A will be influenced by the utterances expressed by speaker B. Using this approach, the model achieves state of the art performance in two affect datasets (Schuller et al., 2011; Busso et al., 2008)). The model presented by us falls in this line of research, as we also attempt to exploit turn level information independently, however, our work differs in the fact that we create representations for each turn as opposed to creating representations for each speaker.

3 Data Preprocessing

Due to the casual text language used in many of the dialogues in the dataset, properly preprocessing these is an essential part of the classification process. The preprocessing pipeline consists of multiple steps that we describe in more depth below.

Text Normalization We use a custom normalization function which takes commonly used contractions in social media and maps them to a normalized version by unpacking them i.e. *idk* $\rightarrow$ *i don't know* and *plz* $\rightarrow$ *please*.

Spell Correction We normalize elongated words and use a spell correction tool which replaces misspelled words with the most probable candidate based on 2 corpora (Wikipedia and Twitter) (Baziotis et al., 2017a).

Tokenization We employ a tokenizer which emphasizes expressions and words typically used in social media. These include: 1) censored words, 2) words with emphasis, 3) elongated words, 4) splitting emoticons etc. All words are also lowercased when tokenized.

Emoji Descriptions As the dialogues contain a wide variety of emojis, which can contain a great deal of information about a users emotions (Felbo et al., 2017), we replace the emojis found in the utterances with their textual description. We used the emoji descriptions utilized for training Emoji2Vec (Eisner et al., 2016) which can be found in the Unicode emoji standard [2].

For most of the preprocessing steps described above, we relied on the Ekphrasis[3] text processing tool (Baziotis et al., 2017b).

4 Model Description

This section describes our conversational sentiment classification model as was used in the EmoContext shared task. Our architecture is illustrated in figure 1.

Embedding Layer We initiate the embedding layer with an embedding matrix computed using pretrained GloVe embeddings trained on 2 Billion tweets[4]. We do not finetune the weights during training.

Turn Encoder We use bidirectional LSTMs to encode a single turn in the conversation. Given a turn T_k made up of N_k words i.e. $T_k = (w_{1_k}, w_{2_k}, ..., w_{N_k})$, the representation of a given word w_{N_k} consists of the concatenation of the forward hidden state $\overrightarrow{h}_{N_k}$ and backward hidden state $\overleftarrow{h}_{N_k}$, i.e. $h_{N_k} = [\overleftarrow{h}_{N_k}, \overrightarrow{h}_{N_k}]$. This bidirectional representation is then fed through a batch normalization layer and then into the attention layer. The different turn encoders do not share their weights.

Turn Attention We use the attention mechanism introduced by (Yang et al., 2016) in order to extract important words in a single turn. The representation h_{N_k} is fed into a one-layer MLP to obtain the representation u_{N_k}. The similarity between u_{N_k} and a randomly initialized word context vector u_c is computed using the dot product and then the normalized weights α_{N_k} are obtained through a softmax function:

$$\alpha_{N_k} = \frac{exp(u_{N_k} \cdot u_c)}{\sum_k exp(u_{N_k} \cdot u_c)}$$

The final turn representation T_k is the weighted sum of the word vectors based on α_{N_k}.

$$T_k = \sum_k \alpha_{N_k} h_{N_k}$$

[2] http://www.unicode.org/emoji/charts/full-emoji-list.html

[3] https://github.com/cbaziotis/ekphrasis

[4] https://nlp.stanford.edu/projects/glove/

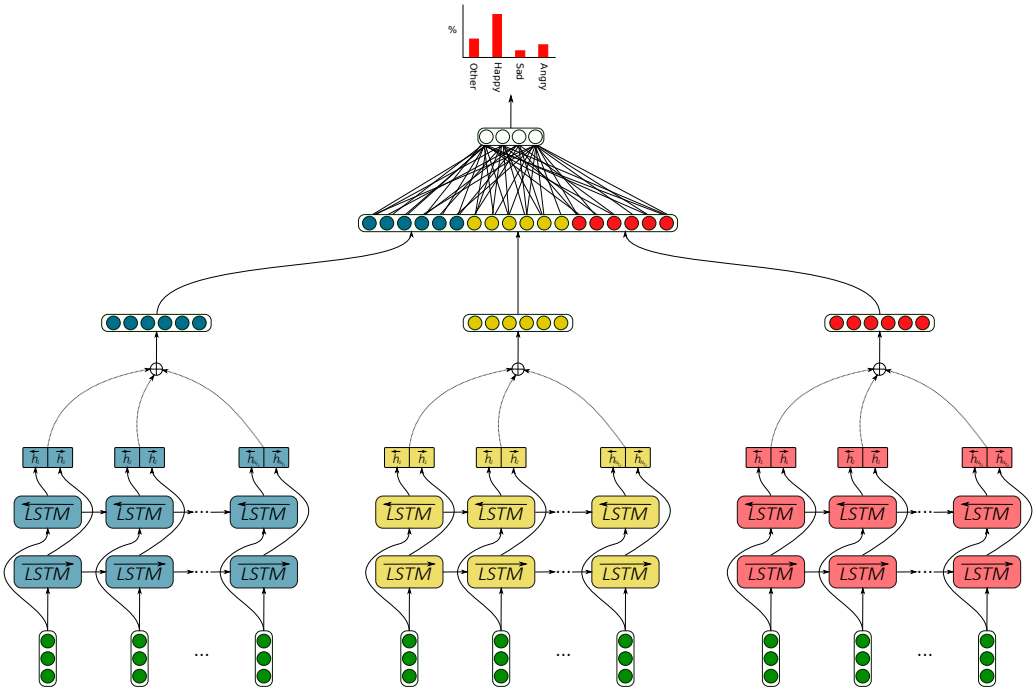

Figure 1: Our proposed model architecture. The turns are preprocessed and embedded using pretrain GloVe embeddings (trained on Twitter data) and fed into their respective BiLSTMs, which are attended over, and combined into a dialogue representation that is classified into one of the 4 classes.

Dialogue Representation For each turn of the conversation we use separate turn encoders and turn attention mechanisms and concatenate the final representations. So for a dialogue of k turns, we end up with a dialogue vector $D = [T_1, T_2, T_3, ..., T_k]$. This representation was chosen as the dialogue length was fixed to 3, but in a variable turn number setting, an LSTM could be utilized to create the final dialogue representation.

Emotion Classification The representation of dialogue D is fed into a softmax layer in order to estimate the probability distribution over the four possible emotion classes.

5 Baselines

Dialogue-level LSTM The main baseline model we compare performance to is the one provided by the task organizers. The system consists of one LSTM encoder, which encodes all turns in the conversation in the same sequence, separated by an end-of-turn token. We show the performance of the baseline given the provided preprocessing script by the organizers. In addition, we include the results of the baseline model using our custom preprocessor.

Dialogue-level TF-IDF with no added features For comparison with the dialogue-level LSTM, we include the performance of a SVM model with stochastic gradient descent. As input we use TF-IDF features computed over all turns. We encode the entire dialogue into a vector of the top 5k features.

Dialogue-level TF-IDF with added features In order to investigate the effect of additional information beyond TF-IDF features, we compute the ratio of words that are 1) elongated and 2) capitalized at the dialogue-level. In addition, we compute the average embeddings of the emojis (Eisner et al., 2016) occurring in the dialogue and concatenate all features with the dialogue level TF-IDF features.

Turn-level TF-IDF with no added features As our main system is a turn encoder, for comparison we also include the performance of an SVM classifier using stochastic gradient descent. using turn level TF-IDF vectors, concatenated into a final dialogue representation.

Turn-level TF-IDF with added features In order to quickly investigate the effect of additional information at the turn level, we compute the same features as mentioned earlier: 1) the ratio of words

that are capitalized in a given turn, 2) the ratio of words that are elongated, and 3) the average embeddings of the emojis (Eisner et al., 2016) occurring in a given turn. All features are concatenated.

6 Setup and Results

In addition to our proposed system that is described in Section 4 (BILSTM-ATT) and the baselines in Section 5, we also report results for two other variants of BILSTM-ATT. The first model (BILSTM-ATT-SHARED) shares weights between the RNNs, ie. we encode all turns individually with the same BiLSTM, and a model that simply encodes the entire dialogue as a single turn (BILSTM-ATT-DIA). We train the models with BiLSTM hidden state size of 256, a dropout rate between the LSTM layer and attention layer of 0.5 , a batch size of 200, and we use a word embedding dimension of 200. We optimize using the ADADELTA algorithm (Zeiler, 2012) with a learning rate of 1.0, $\rho = 0.95$ and $\varepsilon = 10^{-6}$.

System	Micro F1	Precision	Recall
LSTM	0.5613	0.4743	0.6862
DIALOGUE-TFIDF	0.5528	0.6202	0.4986
DIALOGUE-TFIDF-ADDED	0.6064	0.7127	0.5276
TURN-TFIDF	0.5918	0.6394	0.5507
TURN-TFIDF-ADDED	0.6955	**0.7632**	0.6388
BILSTM-ATT	**0.7340**	0.7132	0.7560
BILSTM-ATT-SHARED	0.7243	0.6715	**0.7861**
BILSTM-ATT-DIA	0.6789	0.6039	0.7752

Table 1: The table shows the results of our models on the EmoContext shared task test set.

Our results are shown in Table 1. From the results we can observe that our proposed attentive turn-level BiLSTM outperforms all baselines, including the task organizers LSTM model, with a Micro F1 score of 0.7340. What is interesting to note is that almost all of our proposed simple SVM baselines also outperforms the baseline LSTM, with even TURN-TFIDF-ADDED by a significant margin. In general we see that encoding the dialogue on the turn level achieves better performance than its dialogue level counterparts.

7 Discussion

We saw that in all cases, encoding information at the turn level led to improvements in classifier performance over the dialogue level encoding. This observation is in line with work that exists trying to encode conversational context beyond the single turn or the dialogue level (Majumder et al., 2019; Webb et al., 2005). In addition, in the results shown in Table 1, we can observe that the dialogue level attention LSTM achieves a high recall but low precision. In contrast, the differences in all metrics for our proposed model are much smaller and more balanced. This suggests that without the turn level encoding, the classifier becomes more biased towards a specific class. In Table 2, we show the scores of the individual emotion classes for our turn and dialogue-level models. We can see that across all classes the models have a harder time when it comes to classifying dialogues labeled as Happy, suggesting that the happy conversations might have a tendency to be more neutral in language, resulting in a higher mislabelling rate with the Other class. This becomes more apparent when inspecting the data itself. What is also noteworthy is that the BILSTM-ATT-SHARED, which shared a turn encoder between turns, achieves a lower F1 score than BILSTM-ATT, which used separate turn encoders. This could indicate that the different turns carry different weights in the context when it comes to determining the sentiment of the most recent speaker.

Emotion class	Micro F1	Precision	Recall
BILSTM-ATT			
ANGRY	0.751	0.686	0.829
HAPPY	0.692	0.721	0.666
SAD	0.757	0.742	0.772
BILSTM-ATT-DIA			
ANGRY	0.699	0.590	0.859
HAPPY	0.648	0.601	0.704
SAD	0.687	0.687	0.760

Table 2: F1, precision and recall scores for each of the emotion classes for two of our proposed models, BILSTM-ATT and BILSTM-ATT-DIA

8 Conclusion

Overall, our very straight forward model shows the important effect that encoding turn level information separately has when it comes to classifying dialogues. Using the entire dialogue with an end-of-turn token, we see that the model is not able to capture important features of individual turns that might affect the overall sentiment of the conversation. Our results also shows that, although less sophisticated, simpler and more interpretable models does also give decent results, compared to the LSTM baseline model.

References

Foteini Agrafioti, Dimitris Hatzinakos, and Adam K Anderson. 2012. Ecg pattern analysis for emotion detection. *IEEE Transactions on Affective Computing*, 3(1):102–115.

Hua Ai, Diane J Litman, Kate Forbes-Riley, Mihai Rotaru, Joel Tetreault, and Amruta Purandare. 2006. Using system and user performance features to improve emotion detection in spoken tutoring dialogs. In *Ninth International Conference on Spoken Language Processing*.

Fernando Alonso-Martin, Maria Malfaz, Joao Sequeira, Javier Gorostiza, and Miguel Salichs. 2013. A multimodal emotion detection system during human–robot interaction. *Sensors*, 13(11):15549–15581.

Jeremy Ang, Rajdip Dhillon, Ashley Krupski, Elizabeth Shriberg, and Andreas Stolcke. 2002. Prosody-based automatic detection of annoyance and frustration in human-computer dialog. In *Seventh International Conference on Spoken Language Processing*.

Christos Baziotis, Nikos Pelekis, and Christos Doulkeridis. 2017a. Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 747–754.

Christos Baziotis, Nikos Pelekis, and Christos Doulkeridis. 2017b. Datastories at semeval-2017 task 4: Deep lstm with attention for message-level and topic-based sentiment analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 747–754, Vancouver, Canada. Association for Computational Linguistics.

Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N Chang, Sungbok Lee, and Shrikanth S Narayanan. 2008. Iemocap: Interactive emotional dyadic motion capture database. *Language resources and evaluation*, 42(4):335.

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.

Richard Craggs and Mary McGee Wood. 2004. A categorical annotation scheme for emotion in the linguistic content of dialogue. In *Tutorial and Research Workshop on Affective Dialogue Systems*, pages 89–100. Springer.

Laurence Devillers, Ioana Vasilescu, and Lori Lamel. 2002. Annotation and detection of emotion in a task-oriented human-human dialog corpus. In *proceedings of ISLE Workshop*.

Sidney K Dmello and Arthur Graesser. 2010. Multimodal semi-automated affect detection from conversational cues, gross body language, and facial features. *User Modeling and User-Adapted Interaction*, 20(2):147–187.

Ben Eisner, Tim Rocktäschel, Isabelle Augenstein, Matko Bosnjak, and Sebastian Riedel. 2016. emoji2vec: Learning emoji representations from their description. *CoRR*, abs/1609.08359.

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. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1615–1625. Association for Computational Linguistics.

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.

Devamanyu Hazarika, Soujanya Poria, Amir Zadeh, Erik Cambria, Louis-Philippe Morency, and Roger Zimmermann. 2018. Conversational memory network for emotion recognition in dyadic dialogue videos. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 2122–2132.

Jackson Liscombe, Giuseppe Riccardi, and Dilek Hakkani-Tur. 2005. Using context to improve emotion detection in spoken dialog systems.

Navonil Majumder, Soujanya Poria, Devamanyu Hazarika, Rada Mihalcea, Alexander Gelbukh, and Erik Cambria. 2019. Dialoguernn: An attentive rnn for emotion detection in conversations. In *proceedings of AAAI Conference*.

Björn Schuller, Michel Valstar, Florian Eyben, Gary McKeown, Roddy Cowie, and Maja Pantic. 2011. Avec 2011–the first international audio/visual emotion challenge. In *Affective Computing and Intelligent Interaction*, pages 415–424. Springer.

Nick Webb, Mark Hepple, and Yorick Wilks. 2005. Dialogue act classification based on intra-utterance features. In *Proceedings of the AAAI Workshop on Spoken Language Understanding*, volume 4, page 5. Citeseer.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489.

Matthew D. Zeiler. 2012. ADADELTA: an adaptive learning rate method. *CoRR*, abs/1212.5701.