Related Tasks can Share! A Multi-task Framework for Affective language

Kumar Shikhar Deep, Md Shad Akhtar, Asif Ekbal, and Pushpak Bhattacharyya

Department of Computer Science and Engineering,
Indian Institute of Technology Patna, India
{shikhar.mtcs17,shad.pcs15,asif,pb}@iitp.ac.in

Abstract. Expressing the polarity of sentiment as '*positive*' and '*negative*' usually have limited scope compared with the intensity/degree of polarity. These two tasks (i.e. sentiment classification and sentiment intensity prediction) are closely related and may offer assistance to each other during the learning process. In this paper, we propose to leverage the relatedness of multiple tasks in a multi-task learning framework. Our multi-task model is based on convolutional-Gated Recurrent Unit (GRU) framework, which is further assisted by a diverse hand-crafted feature set. Evaluation and analysis suggest that joint-learning of the related tasks in a multi-task framework can outperform each of the individual tasks in the single-task frameworks.

Keywords: Multi-task learning, Single-task learning, Sentiment Classification , Sentiment Intensity Prediction.

1 Introduction

In general, people are always interested in what other people are thinking and what opinions they hold for a number of topics like product, politics, news, sports etc. The number of people expressing their opinions on various social media platforms such as Twitter, Facebook, LinkedIn etc. are being continuously growing. These social media platforms have made it possible for the researchers to gauge the public opinion on their topics of interest- and that too on demand. With the increase of contents on social media, the process of automation of Sentiment Analysis [20] is very much required and is in huge demand. User's opinions extracted from these social media platform are being used as inputs to assist in decision making for a number of applications such as businesses analysis, market research, stock market prediction etc.

Coarse-grained sentiment classification (i.e. classifying a text into either *positive* or *negative* sentiment) is a well-established and well-studied task [10]. However, such binary/ternary classification studies do not always reveal the exact state of human mind. We use language to communicate not only our sentiments but also the intensity of those sentiments, e.g. one could judge that we are very angry, slightly sad, very much elated, etc. through our utterances. Intensity

Table 1: Example sentences with their sentiment classes and intensity scores from SemEval-2018 dataset on Affect in Tweets [15].

Tweet	**Valence**	**Intensity**
@LoveMyFFAJacket FaceTime - we can still annoy you	*Pos-S*	0.677
and i shouldve cut them off the moment i started hurting myself over them	*Neg-M*	0.283
@VescioDiana You forgot #laughter as well	*Pos-S*	0.700

refers to the degree of sentiment a person may express through his text. It also facilitates us to analyze the sentiment on much finer level rather than only expressing the polarity of the sentiments as positive or negative. In recent times, studies on the amount of positiveness and negativeness of a sentence (i.e. how positive/negative a sentence is or the degree of positiveness/negativeness) has gained attention due to its potential applications in various fields. Few example sentences are depicted in Table 1. In this work, we focus on the fine-grained sentiment analysis [29]. Further, we aim to solve the fine-grained analysis with two different lenses i.e. fine-grained sentiment classification and sentiment intensity prediction.

- **Sentiment or Valence[1] Classification:** In this task, we classify each tweet into one of the seven possible fine-grained classes -corresponding to various levels of positive and negative sentiment intensity- that best represents the mental state of the tweeter, i.e. very positive (***Pos-V***), moderately positive (***Pos-M***), slightly positive (***Pos-S***), neutral (***Neu***), slightly negative (***Neg-S***), moderately negative (***Neg-M***), and very negative (***Neg-V***).
- **Sentiment or Valence Intensity Prediction:** Unlike the discrete labels in the classification task, in intensity prediction, we determine the degree or arousal of sentiment that best represents the sentimental state of the user. The scores are a real-valued number in the range 0 & 1, with 1 representing the highest intensity or arousal.

The two tasks i.e. sentiment classification and their intensity predictions are related and have inter-dependence on each other. Building separate system for each task is often less economical and more complex than a single multi-task system that handles both the tasks together. Further, joint-learning of two (or more) related tasks provides a great assistance to each other and also offers generalization of multiple tasks.

In this paper, we propose a hybrid neural network based multi-task learning framework for sentiment classification and intensity prediction for tweets. Our network utilizes bidirectional gated recurrent unit (Bi-GRU) [28] network in cascade with convolutional neural network (CNN) [13]. The max-pooled features and a diverse set of hand-crafted features are then concatenated, and subsequently fed to the task-specific softmax layer for the final prediction. We evaluate

[1] Valence signifies the pleasant/unpleasant scenarios.

our approach on the benchmark dataset of SemEval-2018 shared task on Affect in Tweets [15]. We observe that, our proposed multi-task framework attains better performance when both the tasks are learned jointly.

The rest of the paper are organized as follows. In Section 2, we furnish the related work. We present our proposed approach in Section 3. In Section 4, we describe our experimental results and analysis. Finally, we conclude in Section 5.

2 Related Work

The motivation behind applying multi-task model for sentiment analysis comes from [27] which gives a general overview of multi-task learning using deep learning techniques. Multitask learning (MTL) is not only applied to Natural Language Processing [4] tasks, but it has also shown success in the areas of computer vision [9], drug discovery [24] and many other. The authors in [5] used stacking ensemble technique to merge the results of classifiers/regressors through which the handcrafted features were passed individually and finally fed those results to a meta classifier/regressor to produce the final prediction. This has reported to have achieved the state-of-the-art performance.

The authors in [8] used bidirectional Long Short Term Memory (biLSTM) and LSTM with attention mechanism and performed transfer learning by first pre-training the LSTM networks on sentiment data. Later, the penultimate layers of these networks are concatenated to form a single vector which is fed as an input to the dense layers. There was a gated recurrent units (GRU) based model proposed by [26] with a convolution neural network (CNN) attention mechanism and training stacking-based ensembles. In [14] they combined three different features generated using deep learning models and traditional methods in support vector machines (SVMs) to create an unified ensemble system. In [21] they used neural network model for extracting the features by transferring the emotional knowledge into it and passed these features through machine learning models like support vector regression (SVR) and logistic regression. In [2] authors have used a Bi-LSTM in their architecture. In order to improve the model performance they applied a multi-layer self attention mechanism in Bi-LSTM which is capable of identifying salient words in tweets, as well as gain insight into the models making them more interpretable.

Our proposed model differs from previous models in the sense that we propose an end to end neural network based approach that performs both sentiment analysis and sentiment intensity prediction simultaneously. We use gated recurrent units (GRU) along with convolutional neural network (CNN) inspired by [26]. We fed the hidden states of GRU to CNN layer in order to get a fixed size vector representation of each sentence. We also use various features extracted from the pre-trained resources like DeepMoji [7], Skip-Thought Vectors [12], Unsupervised Sentiment Neuron [23] and EmoInt [5].

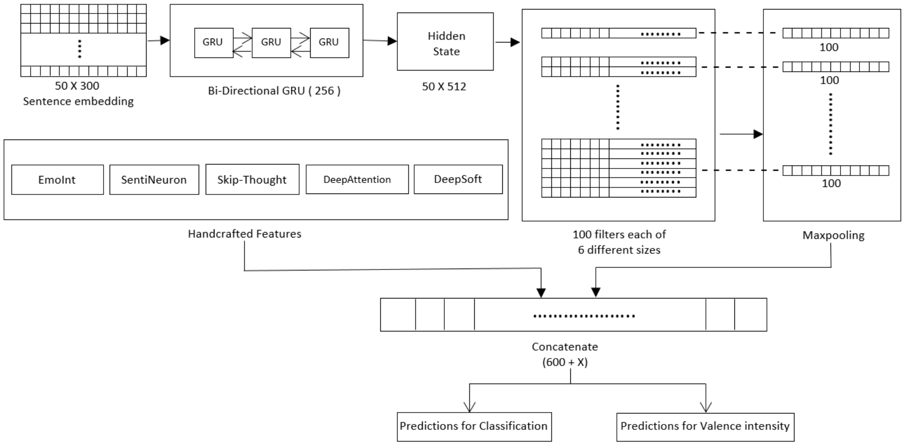


Fig. 1: Proposed Architecture

3 Proposed Methodology

In this section, we describe our proposed multi-task framework in details. Our model consists of a recurrent layer (biGRU) followed by a CNN module. Given a tweet, the GRU learns contextual representation of each word in the sentence, i.e. the representation of each word is learnt based on the sequence of words in the sentence. This representation is then used as input to the CNN module for the sentence representation. Subsequently, we apply max-pooling over the the convoluted features of each filter and concatenated them. The hidden representation, as obtained from the CNN module, is shared across multiple tasks (here, two tasks i.e. sentiment classification and intensity prediction). Further, the hidden representation is assisted by a diverse set of hand-crafted features (c.f. section 3.1) for the final prediction.

In our work, we experiment with two different paradigms of predictors i.e. a) the first model is the traditional deep learning framework that makes use of softmax (or sigmoid) function in the output layer, and b) the second model is developed by replacing softmax classifier using support vector machine (SVM) [31] (or support vector regressor (SVR)). In the first model we feed the concatenated representation to two separate fully-connected layers with softmax (classification) and sigmoid (intensity) functions for the two tasks. In the second model, we feed hidden representations as feature vectors to the SVM and SVR respectively, for the prediction. A high-level block diagram of the proposed methodology is depicted in Figure 1.

3.1 Hand-crafted Features

We perform transfer learning from various state-of-the-art deep learning techniques. Following sub-sections explains these models in detail:

- **DeepMoji** [7]: DeepMoji performs distant supervision on a very large dataset [32] [19] (1.2 billion tweets) comprising of noisy labels (emojis). By incorporating transfer learning on various downstream tasks, they were able to outperform the state-of-the-art results of 8 benchmark datasets on 3 NLP tasks across 5 domains. Since our target task is closely related to this, we adapt this for our domain. We extract 2 different feature sets:
  - the embeddings from the softmax layer which is of 64 dimensions.
  - the embeddings from the attention layer which is of 2304 dimensions.
- **Skip-Thought Vectors** [12]: Skip-thought is a kind of model that is trained to reconstruct the surrounding sentences to map sentences that share semantic and syntactic properties into similar vectors. It has the capability to produce highly generic semantic representation of sentence. The skip-thought model has two parts:
  - **Encoder :** It is generally a Gated Recurrent Unit (GRU) whose final hidden state is passed to the dense layers to get the fixed length vector representation of each sentence.
  - **Decoder :** It takes this vector representation as input and tries to generate the previous and next sentence. For this two different GRUs are needed.

  Due to its fixed length vector representation, skip-thought could be helpful to us. The feature extracted from skip-thought model is of dimension 4800.
- **Unsupervised Sentiment Neuron**: [23] developed an unsupervised system which learned an excellent representation of sentiment. Actually the model was designed to produce Amazon product reviews, but the data scientists discovered that one single unit of network was able to give high predictions for sentiments of texts. It was able to classify the reviews as positive or negative, and its performance was found to be better than some popular models. They even got encouraging results on applying their model on the dataset of Yelp reviews and binary subset of the Stanford Sentiment Treebank. Thus the sentiment neuron model could be used to extract features by transfer learning. The features extracted from Sentiment Neuron model are of dimension 4096.
- **EmoInt** [5]: We intended to use various lexical features apart from using some pre-trained embeddings. EmoInt [5] is a package which provides a high level wrapper to combine various word embeddings. The lexical features includes the following:
  - AFINN [17] contains list of words which are manually rated for valence between -5 to +5 where -5 indicates very negative sentiment and +5 indicates very positive sentiment.
  - SentiWordNet [1] is lexical resource for opinion mining. It assigns to each synset of WordNet three sentiment scores: positivity, negativity, objectivity.
  - SentiStrength [32] gives estimation of strength of positivity and negativity of sentiment.
  - NRC Hashtag Emotion Lexicon [17] consists of emotion word associations computed via Hashtags on twitter texts labelled by emotions.

- NRC Word-Emotion Association Lexicon [17] consists of 8 sense level associations (anger, fear, joy, sadness, anticipation, trust, disgust and surprise) and 2 sentiment level associations(positive and negative)
- The NRC Affect Intensity [16] are the lexicons which provides real values of affect intensity.

The final feature vector is the concatenation of all the individual features. This feature vector is of size (133, 1).

3.2 Word Embeddings

Embedding matrix is generated from the pre-processed text using a combination of three pre-trained embeddings:

1. **Pre-trained GloVe embeddings for tweets** [22]: We use 200-dimensional pre-trained GloVe word embeddings, trained on the Twitter corpus, for the experiments. To make it compatible with the other embeddings, we pad 100-dimensional zero vector to each embedding.
2. **Emoji2Vec** [6]: Emoji2Vec provides 300 dimension vectors for most commonly used emojis in twitter platform (in case any emoji is not replaced with its corresponding meaning).
3. **Character-level embeddings**[2]: Character-level embeddings are trained over common crawl glove corpus providing 300 dimensional vectors for each character (used in case if word is not present in other two embeddings).

Procedure to generate representations for a tweet using all these embeddings is described in Algorithm 1.

Algorithm 1 Procedure to generate representations

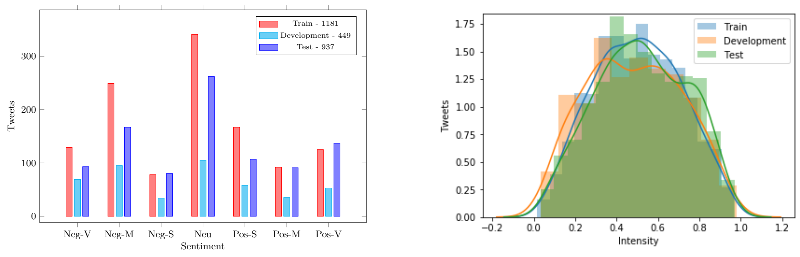
```
for word in tweet do
    if word in GloVe then
        word_vector = get_vector(GloVe, word)
    else if word in Emoji2Vec then
        word_vector = get_vector(Emoji2Vec, word)
    else
        /*n = Number of characters in word*/
        word_vector = $\frac{1}{n} * \sum_{1}^{n}$ get_vector(CharEmbed, chars[n])
    end if
end for
```

4 Experiments and Results

4.1 Dataset

We evaluate our proposed model on the datasets of SemEval-2018 shared task on Affect in Tweets [15]. There are approximately 1181, 449 & 937 tweets for

[2] https://github.com/minimaxir/char-embeddings



(a) Sentiment class distribution.

(b) Sentiment intensity distribution.

Fig. 2: Sentiment distribution for SemEval-2018 task on Affect in Tweets [15]

training, development and testing. For each tweet, two labels are given: a) sentiment class (one of the seven class on sentiment scale i.e. very positive (***Pos-V***), moderately positive (***Pos-M***), slightly positive (***Pos-S***), neutral (***Neu***), slightly negative (***Neg-S***), moderately negative (***Neg-M***), and very negative (***Neg-V***)); and b) an intensity score in the range 0 to 1. We treat prediction of these two labels as two separate tasks. In our multi-task learning framework, we intend to solve these two tasks together. A brief statistics of the datasets is depicted in Figure 2.

4.2 Preprocessing

Tweets in raw form are noisy because of the use of irregular, short form of text (e.g. *hlo*, *whtsgoin* etc.), emojis and slangs and are prone to many distortions in terms of semantic and syntactic structures. The preprocessing step modifies the raw tweets to prepare for feature extraction. We use Ekphrasis tool [3] for tokenization, word normalization, word segmentation (for splitting hashtags) and spell correction. Ekphrasis is a text processing tool, geared towards text from social networks, such as Twitter or Facebook. They used word statistics from 2 big corpora i.e. English Wikipedia and Twitter (330 million English tweets). Ekphrasis was developed as a part of text processing pipeline for SemEval-2017 shared task on Sentiment Analysis in Twitter [25]. We list the preprocessing steps that have been carried out below.

- All characters in text are converted to lower case
- Remove punctuation except ! and ? because '!' and '?' may contribute to better result of valence detection.
- Remove extra space and newline character
- Group similar emoji, replace them with their meaning in words using Emojipedia
- Named Entity recognition and replace with keyword or token (*@shikhar* → *username*, *https://www.iitp.ac.in* → *url*)
- Split the hashtags (*#iamcool* → *i am cool*)
- Correct the misspelled words (*facbok* → *facebook*)

Table 2: Pearson correlation for STL and MTL frameworks for sentiment classification and intensity prediction. $^{+}$Reported in [5]; *Reproduced by us.

Framework	**Sentiment Classification**		**Intensity Prediction**	
	DL (Softmax)	ML (SVM)	DL (Sigmoid)	ML (SVR)
Single-task Learning (STL)	0.361	0.745	0.821	0.818
Multi-task Learning (MTL)	**0.408**	**0.772**	**0.825**	**0.830**
State-of-the-art [5]	**0.836^{+} (0.776*)**		**0.873^{+} (0.829*)**	

4.3 Experiments

We pad each tweet to a maximum length of 50 words. We employ 300-dimensional word embedding for the experiments (c.f. Section 3.2). The GRU dimension is set to 256. We use 100 different filters of varying sizes (i.e. 2-gram, 3-gram, 4-gram, 5-gram and 6-gram filters) with max-pool layer in the CNN module. We use *ReLU* [18] activation and set the *Dropout* [30] as 0.5 . We optimize our model using *Adam* [11] with cross-entropy and *mean-squared-error (MSE)* loss functions for sentiment classification and intensity prediction, respectively.

For experiments, we employ python based deep learning library Keras with TensorFlow as the backend. We adopt the official evaluation metric of SemEval-2018 shared task on Affect in Tweets [15], i.e. Pearson correlation coefficient, for measuring the performance of both tasks. We train our model for the maximum 100 epochs with early stopping criteria having patience=20.

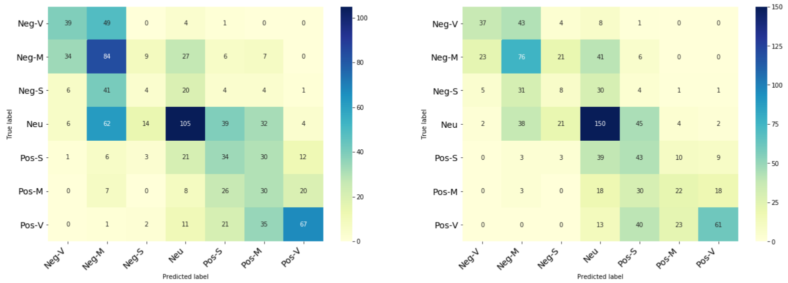
In single-task learning (STL) framework, we build separate systems for both sentiment classification and intensity prediction. We pass the normalized tweet to our Convolutional-GRU framework for learning. Since the number of training samples are considerably few to effectively learn a deep learning model, we assist the model with various hand-crafted features. The concatenated representations are fed to the softmax layer (or sigmoid) for the sentiment (intensity) prediction. We obtain 0.361 Pearson coefficient for sentiment classification and 0.821 for intensity prediction. Further, we also try to exploit the traditional machine learning algorithms for prediction. We extract the concatenated representations and feed them as an input to SVM for sentiment classification and SVR for intensity prediction. Consequently, SVM reports increased Pearson score of 0.745 for sentiment classification, whereas we observe comparable results (i.e. 0.818 Pearson score) for intensity prediction.

The MTL framework yields an improved performance for both the tasks in both the scenarios. In the first model, MTL reports 0.408 and 0.825 Pearson scores as compared with the Pearson scores of 0.361 & 0.821 in STL framework for the sentiment classification and intensity prediction, respectively. Similarly, the MTL framework reports 3 and 2 points improved Pearson scores in the second model for the two tasks, respectively. These improvements clearly suggest that the MTL framework, indeed, exploit the inter-relatedness of multiple tasks in order to enhance the individual performance through a joint-model. Further, we observe the improvement of MTL models to be statistically significant with 95% confidence i.e. p-value < 0.05 for paired *T-test*.

On same dataset, Duppada et al. [5] (winning system of SemEval-2018 task on Affect in Tweets [15]) reports Pearson scores of 0.836 and 0.873 for sentiment classification and intensity prediction, respectively. The authors in [5] passed the same handcrafted features individually through XGBost and Random Forest classifier/regressor and combined the results of all the classifiers/regressors using stacking ensemble technique. After that they passed the results from the models to a meta classifier/regressor as input. They used Ordinal Logistic Classifier and Ridge Regressor as meta classifier/regressor. In comparison, our proposed system (i.e. MTL for ML framework) obtains Pearson scores of 0.772 and 0.830 for sentiment classification and intensity prediction, respectively. It should be noted that we tried to reproduce the works of Duppada et al. [5], but obtained Pearson scores of only 0.776 and 0.829, respectively. Further, our proposed MTL model offers lesser complexity compared to the state-of-the-art systems. Unlike the state-of-the-art systems we do not require separate system for each task, rather an end-to-end single model addresses both the tasks simultaneously.

4.4 Error Analysis

In Figure 3, we present the confusion matrices for both the models (first and second, based on DL and ML paradigms). It is evident from the confusion matrices that most of the mis-classifications are within the close proximity of the actual labels, and our systems occasionally confuse with '*positive*' and '*negative*' polarities (i.e. only 43 and 22 mis-classifications for the first model-DL based and second model-ML based, respectively).



	Neg-V	Neg-M	Neg-S	Neu	Pos-S	Pos-M	Pos-V
Neg-V	39	49	0	4	1	0	0
Neg-M	34	84	9	27	6	7	0
Neg-S	6	41	4	20	4	4	1
Neu	6	62	14	105	39	32	4
Pos-S	1	6	3	21	34	30	12
Pos-M	0	7	0	8	26	30	20
Pos-V	0	1	2	11	21	35	67

(a) Multi-task:DL

	Neg-V	Neg-M	Neg-S	Neu	Pos-S	Pos-M	Pos-V
Neg-V	37	43	4	8	1	0	0
Neg-M	23	76	21	41	6	0	0
Neg-S	5	31	8	30	4	1	1
Neu	2	38	21	150	45	4	2
Pos-S	0	3	3	39	43	10	9
Pos-M	0	3	0	18	30	22	18
Pos-V	0	0	0	13	40	23	61

(b) Multi-task:ML

Fig. 3: Confusion matrices for sentiment classification.

We also perform error analysis on the obtained results. Few frequently occurring error cases are presented below:

– **Metaphoric expressions:** Presence of metaphoric/ironic/sarcastic expressions in the tweets makes it challenging for the systems in correct predictions.

Table 3: MTL vs STL for Sentiment Classification and Intensity Prediction

Sentence	**Actual**	**DL**		**ML**	
		MTL	**STL**	**MTL**	**STL**
Maybe he was partly right. THESE emails might lead to impeachment and 'lock him up' #ironic #ImpeachTrump	***Neg-M***	*Neg-M*	*Neg-V*	*Neg-M*	*Neg-S*
#Laughter strengthens #relationships. #Women are more attracted to someone with the ability to make them #laugh.	***Pos-M***	*Pos-M*	*Pos-S*	*Pos-M*	*Pos-S*

a) Sentiment Classification

I graduated yesterday and already had 8 family members asking what job I've got now #nightmare	**0.55**	0.57 (+0.02)	0.51 (-0.04)	0.59 (+0.04)	0.64 (+0.09)
@rohandes Lets see how this goes. We falter in SL and this goes downhill.	**0.49**	0.48 (-0.01)	0.35 (-0.14)	0.49 (+0.00)	0.29 (-0.20)
It's kind of shocking how amazing your rodeo family is when the time comes that you need someone	**0.52**	0.53 (+0.01)	0.55 (+0.03)	0.52 (+0.00)	0.51 (-0.01)

b) Intensity Prediction

  - "*@user But you have a lot of time for tweeting **#ironic***".
    Actual: *Neg-M* **Prediction:** *Neu*
- **Neutralizing effect of opposing words:** Presence of opposing phrases in a sentence neutralizes the effect of actual sentiments.
  - "*@user Macron slips up and has a moment of clarity & common sense... now he is a **raging racist**. Sounds right. **Liberal** logic*"
    Actual: *Neg-M* **Prediction:** *Neu*

We further analyze the predictions of our MTL models against STL models. Analysis suggests that our MTL model indeed improves the predictions of many examples that are mis-classified (or having larger error margins) than the STL models. In Table 3, we list a few examples showing the actual labels, MTL prediction and STL prediction for both sentiment classification and intensity prediction.

5 Conclusion

In this paper, we have presented a hybrid multi-task learning framework for affective language. We propose a convolutional-GRU network with the assistance of a diverse hand-crafted feature set for learning the shared hidden representations for multiple tasks. The learned representation is fed to SVM/SVR classifier for the predictions. We have evaluated our model on the benchmark datasets of SemEval-2018 shared on Affect in Tweets for the two tasks (i.e. sentiment classification and intensity prediction). Evaluation suggests that a single multi-task model obtains improved results against separate systems of single-task models.

6 Acknowledgement

Asif Ekbal acknowledges the Young Faculty Research Fellowship (YFRF), supported by Visvesvaraya PhD scheme for Electronics and IT, Ministry of Electronics and Information Technology (MeitY), Government of India, being implemented by Digital India Corporation (formerly Media Lab Asia).

References

1. Baccianella, S., Esuli, A., Sebastiani, F.: SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In: LREC. vol. 10, pp. 2200–2204 (2010)
2. Baziotis, C., Athanasiou, N., Chronopoulou, A., Kolovou, A., Paraskevopoulos, G., Ellinas, N., Narayanan, S., Potamianos, A.: NTUA-SLP at SemEval-2018 Task 1: Predicting Affective Content in Tweets with Deep Attentive RNNs and Transfer Learning. arXiv Preprint arXiv:1804.06658 (2018)
3. Baziotis, C., Pelekis, N., Doulkeridis, C.: 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). pp. 747–754. Vancouver, Canada (2017)
4. Collobert, R., Weston, J.: A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. In: Proceedings of the 25th International Conference on Machine Learning. pp. 160–167. NY, USA (2008)
5. Duppada, V., Jain, R., Hiray, S.: SeerNet at SemEval-2018 Task 1: Domain Adaptation for Affect in Tweets. arXiv Preprint arXiv:1804.06137 (2018)
6. Eisner, B., Rocktäschel, T., Augenstein, I., Bošnjak, M., Riedel, S.: Emoji2vec: Learning Emoji Representations from their Description. arXiv Preprint arXiv:1609.08359 (2016)
7. Felbo, B., Mislove, A., Søgaard, A., Rahwan, I., Lehmann, S.: Using-Millions of Emoji Occurrences to Learn Any-Domain Representations for Detecting Sentiment, Emotion and Sarcasm. In: Conference on Empirical Methods in Natural Language Processing (EMNLP) (2017)
8. Gee, G., Wang, E.: psyML at SemEval-2018 Task 1: Transfer Learning for Sentiment and Emotion Analysis. In: Proceedings of The 12th International Workshop on Semantic Evaluation. pp. 369–376 (2018)
9. Girshick, R.: Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1440–1448 (2015)
10. Kim, S.M., Hovy, E.: Determining the Sentiment of Opinions. In: Proceedings of the 20th International Conference on Computational Linguistics. p. 1367 (2004)
11. Kingma, D.P., Ba, J.: Adam: A Method for Stochastic Optimization. arXiv Preprint arXiv:1412.6980 (2014)
12. Kiros, R., Zhu, Y., Salakhutdinov, R., Zemel, R.S., Torralba, A., Urtasun, R., Fidler, S.: Skip-Thought Vectors. arXiv Preprint arXiv:1506.06726 (2015)
13. LeCun, Y., Bengio, Y., et al.: Convolutional Networks for Images, Speech, and Time Series. The Handbook of Brain Theory and Neural Networks **3361**(10) (1995)
14. Meisheri, H., Dey, L.: TCS Research at SemEval-2018 Task 1: Learning Robust Representations using Multi-Attention Architecture. In: Proceedings of The 12th International Workshop on Semantic Evaluation. pp. 291–299 (2018)

15. Mohammad, S., Bravo-Marquez, F., Salameh, M., Kiritchenko, S.: Semeval-2018 Task 1: Affect in Tweets. In: Proceedings of The 12th International Workshop on Semantic Evaluation. pp. 1–17 (2018)
16. Mohammad, S.M., Bravo-Marquez, F.: Emotion Intensities in Tweets. arXiv Preprint arXiv:1708.03696 (2017)
17. Mohammad, S.M., Bravo-Marquez, F.: WASSA-2017 Shared Task on Emotion Intensity. arXiv Preprint arXiv:1708.03700 (2017)
18. Nair, V., Hinton, G.E.: Rectified Linear Units Improve Restricted Boltzmann Machines. In: Proceedings of the 27th International Conference on Machine Learning (ICML-10). pp. 807–814 (2010)
19. Nakov, P., Ritter, A., Rosenthal, S., Sebastiani, F., Stoyanov, V.: SemEval-2016 Task 4: Sentiment Analysis in Twitter. In: Proceedings of the 10th International Workshop on Semantic Evaluation (Semeval-2016). pp. 1–18 (2016)
20. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs Up?: Sentiment Classification Using Machine Learning Techniques. In: Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing-Volume 10. pp. 79–86 (2002)
21. Park, J.H., Xu, P., Fung, P.: PlusEmo2Vec at SemEval-2018 Task 1: Exploiting Emotion Knowledge from Emoji and Hashtags. arXiv Preprint arXiv:1804.08280 (2018)
22. Pennington, J., Socher, R., Manning, C.: Glove: Global Vectors For Word Representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1532–1543 (2014)
23. Radford, A., Jozefowicz, R., Sutskever, I.: Learning to Generate Reviews and Discovering Sentiment. arXiv Preprint arXiv:1704.01444 (2017)
24. Ramsundar, B., Kearnes, S., Riley, P., Webster, D., Konerding, D., Pande, V.: Massively Multitask Networks for Drug Discovery. arXiv Preprint arXiv:1502.02072 (2015)
25. Rosenthal, S., Farra, N., Nakov, P.: SemEval-2017 Task 4: Sentiment Analysis in Twitter. In: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017). pp. 502–518. Vancouver, Canada (August 2017)
26. Rozental, A., Fleischer, D.: Amobee at SemEval-2018 Task 1: GRU Neural Network with a CNN Attention Mechanism for Sentiment Classification. arXiv Preprint arXiv:1804.04380 (2018)
27. Ruder, S.: An Overview of Multi-Task Learning in Deep Neural Networks. arXiv Preprint arXiv:1706.05098 (2017)
28. Schuster, M., Paliwal, K.: Bidirectional Recurrent Neural Networks. Trans. Sig. Proc. **45**(11), 2673–2681 (Nov 1997)
29. Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C.D., Ng, A., Potts, C.: Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. pp. 1631–1642 (2013)
30. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research **15**, 1929–1958 (2014)
31. Suykens, J.A., Vandewalle, J.: Least Squares Support Vector Machine Classifiers. Neural Processing Letters **9**(3), 293–300 (1999)
32. Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., Kappas, A.: Sentiment Strength Detection in Short Informal Text. Journal of the American Society for Information Science and Technology **61**(12), 2544–2558 (2010)