Multi-task Learning for Detecting Stance in Tweets

Devamanyu Hazarika*[1], Gangeshwar Krishnamurthy*[2], Soujanya Poria[3], and Roger Zimmermann[1]

[1] National University of Singapore, Singapore
hazarika@comp.nus.edu.sg, rogerz@comp.nus.edu.sg
[2] Artificial Intelligence Initiative,
Agency for Science Technology and Research (A*STAR), Singapore
gangeshwark@ihpc.a-star.edu.sg
[3] Nanyang Technological University, Singapore
soujanya.poria@gmail.com

Abstract. Detecting stance of online posts is a crucial task to understand online content and trends. Existing approaches augment models with complex linguistic features, target-dependent properties, or increase complexity with attention-based modules or pipeline-based architectures. In this work, we propose a simpler multi-task learning framework with auxiliary tasks of subjectivity and sentiment classification. We also analyze the effect of regularization against inconsistent outputs. Our simple model achieves competitive performance with the state of the art in *micro-F1* metric and surpasses existing approaches in *macro-F1* metric across targets. We are able to show that multi-tasking with a simple architecture is indeed useful for the task of stance classification.

1 Introduction

Automatic detection of stance over text is an emerging task of opinion mining. In recent times, its importance has increased due to its role in practical applications. It is used in information retrieval systems to filter content based on the authors' stance, to analyze trends in politics and policies [14], in summarization systems to understand online controversies [13]. It also finds its use in modern day problems that plague the Internet, such as identification of rumor or hate speech [21].

The task involves determining whether a piece of text, such as a tweet or debate post is `FOR`, `AGAINST`, or `NONE` towards an entity which can be persons, organizations, products, policies, etc. (see Table 1). This task is challenging due to the use of informal language and literary devices such as sarcasm. For example, in the second sample in Table 1, the phrase *Thank you God!* can mislead a trained model to consider it as a favoring stance. Challenges also amplify as in many tweets the target of the stance may or may not be mentioned. In the third sample in Table 1, the tweet doesn't talk about feminism in particular but rather mocks

* Equal contribution.

	Target	Tweet	Stance
1	Climate Change is a Real Concern	Incredibly moving as a scientist weeps on @BBCRadio4 for the #ocean and for our grandchildren's future	For
2	Atheism	I still remember the days when I prayed God for strength.. then suddenly God gave me difficulties to make me strong. Thank you God!	Against
3	Feminist	When I go up the steps of my house I feel like the @ussoccer_wnt .. I too have won the Women's World Cup. #brokenlegprobs #USA	Against

Table 1: Sample tweets representing stances against target topics.

indirectly using *Women's World Cup*. Present state-of-the-art networks in this task majorly follow the neural approach. These models increase their complexity either by adding extra complex features – such as linguistic features [26] – as input or through complex networks with attention modules or pipeline-based architectures [9, 10].

In this paper, we restrict ourselves from increasing complexity and search for simple solutions for this task. To this end, we propose a simple convolutional neural network, named *MTL-Stance*, that adopts multi-task learning (MTL) for stance classification (Task A) by jointly training with the related tasks of subjectivity (Task B) and sentiment prediction (Task C). For subjectivity analysis, we categorize both `For` and `Against` stances to be *subjective* while `None` stance as *objective*. It is important to note that unlike traditional opinion mining, subjectivity here refers to the presence of stance towards a target in a tweet. Conversely, objectivity contains both tweets which have either no stance or are subjective by their stance is indeterminable. To tackle inconsistent predictions (e.g. Task A predicts `For` stance while Task B predicts *objective*), we explore a regularization term that penalizes the network for inconsistent outputs. Overall, subjectivity represents a coarse-grained version of stance classification and is thus expected to aid the task at hand.

We also consider sentiment prediction (Task C) in the MTL framework to allow the model learn common relations (if any). [18] mentions how sentiment and stance detection are related tasks. However, both the tasks are not same as a person might express same stance towards a target either by positive or negative opinion. A clear relationship is also often missing since the opinion expressed in text might not be directed towards the target. Nevertheless, both the tasks do tend to rely on some related parameters which motivates us to investigate their joint training.

The contributions of this work can be summarized as follows:

– We propose a multi-task learning algorithm for stance classification by associating the related tasks of subjectivity and sentiment detection.

– We demonstrate that a simple CNN-based classifier trained in an end-to-end fashion can supersede models having extra linguistic information or pipeline-based architectures.
– Our proposed model achieves competitive results to the state-of-the-art performance on the SemEval 2016 benchmark dataset whilst having a simpler architecture with a single-phase end-to-end mechanism [17].

The paper is organized as follows: Section 2 presents the related works in the literature and compares them to our proposed work; Section 3 presents the proposed model and explains the MTL framework utilized for training; Section 4 details the experimental setup and the results on the dataset. Finally, Section 5 provides concluding remarks.

2 Related work

The analysis of stance detection has been performed on various forms of text such as debates in congress or online platforms [12,24,27,30], student essays [20], company discussions [1], etc. With the popularity of social media, there is also a surge of opinionated text in microblogs [17]. Traditional approaches involve linguistic features into their models such as sentiment [24], lexicon-based and dependency-based features [2], argument features [12]. Many works also use structural information from online user graphs or retweet links [19, 22].

With the proliferation of deep-learning, several neural approaches have been attempted on this task with state-of-the-art performance. Most of the works utilize either recurrent or convolutional networks to model the text and the target. In [3], the authors use a bi-directional recurrent network to jointly model the text along with the target by initializing the text network with the output of the target network. On the other hand, convolutional neural networks also have been used for encoding text in this task [29]. Apart from basic networks, existing works also incorporate extra information as auxiliary input into their neural models. These features include user background information such as user tastes, comment history, etc. [5].

We focus on some recent works that have attained state-of-the-art performance. Specifically, we look at *Target-specific Attentional Network* (TAN) [10], *Two-phase Attention-embedded LSTM* (T-PAN) [9] and *Hierarchical Attention Network* (HAN) [26]. Similar to [3], TAN uses a bi-directional LSTM scheme to model the task. It includes the target embeddings into the network by using an attention mechanism [4]. We follow similar motivations to use target-specific information. However, aiming to minimize network complexity, we opt for a simple concatenation-based fusion scheme.

T-PAN stands closest to our proposed model as it too incorporates information from classifying subjectivity of tweets. It is achieved by following a two-phase model where in first phase the subjectivity is decided and in second phase, only the subjective tweets from first phase are used to be classified as favoring or non-favoring stances. In contrast to this approach, we do not use a pipeline-based

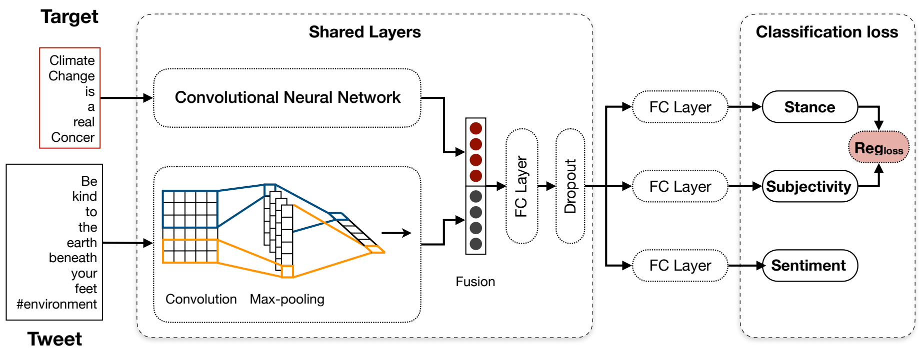

Fig. 1: Stance-MTL: Multi-task framework for stance detection.

approach as it bears an higher possibility of error propagation. Instead, in our MTL framework, both the classifications are done simultaneously.

The Hierarchical Attention Network (HAN), proposed by [26] contains of a hierarchical framework of attention-based layers which includes extra information from sentiment, dependency and argument representations. Unlike HAN, our model is not dependent on complex linguistic features. Rather, we enforce a simple CNN-based model that trains end-to-end under the multi-task regime.

3 Proposed Approach

3.1 Task Formulation

The task of stance classification can be defined as follows: given a tweet text and its associated target entity, the aim of the model is to determine the author's stance towards the target. The possible stances could be favoring (`FOR`), against (`AGAINST`) or inconclusive (`NONE`). The `NONE` class consists of tweets that could either have a neutral stance or be the case whether determining the stance is not easy.

3.2 Multi-task learning

Multi-task learning (MTL) is a framework that requires optimizing a network towards multiple tasks [23]. The motivation arises from the belief that features learnt for a particular task can be useful for related tasks [7]. In particular, MTL exploits the synergies between related tasks through *joint learning*, where supervision from the related/auxiliary tasks provides an *inductive bias* into the network that allows it to generalize better. Effectiveness of MTL framework is evident in the literature of various fields such as speech recognition [8], computer vision [11], natural language processing [6], and others.

MTL algorithms can be realized by two primary methods. First is to train individual models for each task with a common regularization that enforces the models to be similar. Second way is to follow a stronger association by sharing common weights across tasks. In this work, we take influence from both these approaches by using a shared model along with explicit regularization against inconsistent output combinations. Below we provide the details of our model: MTL-Stance.

3.3 Model details

The overall architecture of the MTL-Stance is shown in the figure 1. It consists of the input tweet and its target. The inputs are processed by shared convolutional layers whose outputs are concatenated. The further layers are separated into the three mentioned tasks. Concrete network details are mentioned below.

Input Representation A training example consists of a tweet text: $\{Tw_i\}_{i=0}^{n}$, a target entity: $\{Tr_i\}_{i=0}^{m}$, stance label: $y_1 \in \{\texttt{For}, \texttt{Against}, \texttt{None}\}$, the derived subjectivity label: $y_2 \in \{\texttt{Subjective}, \texttt{Objective}\}$ and sentiment label: $y_3 \in \{\texttt{Positive}, \texttt{Negative}, \texttt{Neither}\}$. Both $Tw \in \mathbb{R}^{n\times k}$ and $Tr \in \mathbb{R}^{m\times k}$ are sequences of words represented in a matrix form with each word corresponding to its k-dimensional word vector [15].

Shared Parameters To both the tweet and target representations, we apply a shared convolutional layer to extract higher-level features. We use multiple filter of different sizes. The width of each filter is fixed to k but the height, h, is varied (as hyper-parameter). For example, let $w \in \mathbb{R}^{h\times k}$ be a filter which can extract a feature vector $\mathbf{z}$ of size $\mathbb{R}^{L-h+1}$ where L is the length of the input. Each entry of vector $\mathbf{z}$ is given by:

$$z_i = g(w \star T_{i:i+h-1} + b)$$

here, $\star$ is the convolution operation, $b \in \mathbb{R}$ is a bias term, and, g is a non-linear function. We then apply a max-over-time pooling operation over the feature vector $\mathbf{z}$ to get the maximum value of $\hat{z} = max(\mathbf{z})$.

The above convolution layer with F_l filters is applied M times on both tweet and target representations to get an output of $M \cdot F_l$ features. These feature representations of tweet text and target text are given by F_{Tw} and F_{Tr}. Next, we obtain the joint representation by concatenating them, i.e., $F_T = [F_{Tw}; F_{Tr}] \in \mathbb{R}^{2\cdot M\cdot F_l}$. This representation is fed to a non-linear fully-connected layer fc_{inter} coupled with Dropout [25].

$$h_{inter} = fc_{inter}(F_T)$$

This layer is also the last shared layer before task-specific layers are applied.

Task-specific Layers For each of the three tasks, i.e., stance, subjectivity, and sentiment classification, we use three different non-linear fully-connected layers. The layer weights are not shared among them so that they can individually learn task specific features.

$$h_i = fc_i(h_{inter}) \quad \forall i \in \{1, 2, 3\}$$

Finally, we pass these features through another projection layer with softmax normalization to get the probability distribution over the labels for each task.

$$\mathbf{\hat{y}_i} = softmax(W_i \cdot h_i + b_i) \quad \forall i \in \{1, 2, 3\}$$

Loss Function We use the categorical cross-entropy on each of the outputs to calculate the loss. We also add a joint regularization loss (Reg_{loss}) to the total loss function:

$$Loss = \frac{-1}{N}\sum_{i=1}^{N}\left\{\sum_{k=1}^{3}\left(\sum_{j=1}^{C_k} y_k^{i,j}\ log_2(\hat{y}_k{}^{i,j})\right) + Reg_{loss}\right\} \tag{1}$$

where N is the number of mini-batch samples, C_k is the number of categories for each of k^{th} task (in the order: stance, subjectivity and sentiment), $y_k^{i,j}$ is the probability of the i^{th} sample of k^{th} task for the j^{th} label, and similarly $\hat{y}_k^{i,j}$ is its predicted counterpart. In our setup, $C_1 = 3$, $C_2 = 2$ and $C_3 = 3$. The regularization term Reg_{loss} is dependent on the output of the first two tasks, and defined as:

$$Reg_{loss} = \alpha \cdot (\text{sgn}|\ \text{argmax}(\mathbf{\hat{y}}_1^i)\ | \oplus \text{sgn}|\ \text{argmax}(\mathbf{\hat{y}}_2^i)\ |) \tag{2}$$

where α is a weighting term (hyper-parameter), sgn|.| is the sign function and $\oplus$ is a logical XOR operation used to penalize the instances where both subjectivity and stance are predicted with contradiction.

4 Experiments

4.1 Dataset

We utilize the benchmark *Twitter Stance Detection* corpus for stance classification originally proposed by [16] and later used in a SemEval task 6 [4] [17]. The dataset presents the task of identifying the stance of a tweet's author towards a target, determining whether the author is favoring (`For`) or is against (`Against`) a target or whether neither of the inference is likely (`None`). The dataset comprises of English tweets spanning five primary targets: *Atheism*, *Climate Change is Concern*, *Feminist Movement*, *Hillary Clinton*, *Legalization of Abortion* with pre-defined training and testing splits. The distribution statistics are provided in Table 2. Along with stance labels, sentiment labels are also provided which we use for the joint training (see Table 2).

[4] `http://alt.qcri.org/semeval2016/task6/`

Target	Stance								Sentiment					
	Train				Test				Train			Test		
	#	for	against	neither	#	for	against	neither	pos	neg	none	pos	neg	none
Atheism	513	17.9	59.3	22.8	220	14.5	72.7	12.7	60.4	35.0	4.4	59.0	35.4	5.4
Climate	395	53.7	3.8	42.5	169	72.8	6.5	20.7	60.4	35.0	4.4	59.0	35.4	5.4
Feminism	664	31.6	49.4	19.0	285	20.4	64.2	15.4	17.9	77.2	4.8	19.3	76.1	4.5
Hillary	689	17.1	57.0	25.8	295	15.3	58.3	26.4	32.0	64.0	3.9	25.7	70.1	4.0
Abortion	653	18.5	54.4	27.1	280	16.4	67.5	16.1	28.7	66.1	5.0	20.3	72.1	7.5
All	2914	25.8	47.9	26.3	1249	24.3	57.3	18.4	33.0	60.4	6.4	29.4	63.3	7.2

Table 2: Percentage distribution of stance and sentiment labels for instances (tweets) in the dataset across targets and splits

4.2 Training details

We use the standard training and testing set provided in the dataset. Hyper-parameters are tuned using a held out validation data: 10% of the training data. To optimize the parameters, we use RMSProp [28] optimizer with an inital learning rate of $1e^{-4}$. The hyper-parameter are $f_l = 128$, $M = 3$. And for each of the 3 filter size the window size (h) is 2, 3 and 4. We fix the tweet length n to 30 and target length m to 6. The number of hidden units in task-specific layers $fc_{[1/2/3]}$ is 300. We initialize the word vectors with the 300-dimensional pre-trained word2vec embeddings [15] which are optimized during training. Following the previous works, we train different models for different targets but with the same hyperparameters. And the final result is the concatenation of predicted result of these models.

4.3 Baselines

We compare MTL-Stance with the following baseline methods:

- **SVM:** This model accounts for a non-neural baseline that has been widely used in previous works [17]. The model uses simple bag-of-words features for stance classification.
- **LSTM:** A simple LSTM model without target features for classification.
- **TAN:** is an RNN-based architecture that uses an target-specific attention-module to focus on parts of the tweet that is related to the target topic [10].
- **T-PAN:** is a two-phase model for classifying the stance [9]. The first phase classifies subjectivity and the second phase classifies the stance based on first phase. Concretely, utterances classified as objective in the first-phase are dropped out from the second phase and assigned the `None` label.
- **HAN:** is a hierarchical attention model which uses linguistic features that include sentiment, dependency and argument features [26].

Model	Atheism	Climate	Feminism	Hillary	Abortion	$MacF_{avg}$	$MicF_{avg}$
SVM	62.16	42.91	56.43	55.68	60.38	55.51	67.01
LSTM	58.18	40.05	49.06	61.84	51.03	52.03	63.21
TAN	59.33	53.59	55.77	65.38	63.72	59.56	68.79
T-PAN	61.19	**66.27**	58.45	57.48	60.21	60.72	68.84
HAN	**70.53**	49.56	57.50	61.23	66.16	61.00	69.79
MTL-Stance	66.15	64.66	**58.82**	**66.27**	**67.54**	**64.69**	**69.88**

Table 3: Comparision of MTL-Stance with state-of-the-art models on Twitter Stance Detection corpus. MTL-Stance results are the average of 5 runs with different initializations.

4.4 Evaluation Metrics

We use both micro-average and macro-average of F1-score across targets as our evaluation metric as defined by [26]. The F1-score for *Favour* and *Against* categories for all instances is calculated as:

$$F_{[favor/against]} = \frac{2 \times P_{[favor/against]} \times R_{[favor/against]}}{P_{[favor/against]} + R_{[favor/against]}} \tag{3}$$

where P and R are precision and recall. Then the final metric, $MicF_{avg}$ is the average of F_{favor} and $F_{against}$.

$$MicF_{avg} = \frac{F_{favor} + F_{against}}{2} \tag{4}$$

4.5 Results

Table 3 shows the performance results on Twitter Stance Detection corpus. Our model, MTL-Stance, performs significantly better than the state-of-the-art models across most targets. The SVM model does not perform well since it only uses bag of words features of tweet text only. LSTM model also does not exploit the information from target text; hence its performance is significantly lower, though it uses a neural architecture. On the other hand, neural models such as TAN, T-PAN, and HAN use both tweet and target text which outperforms both SVM and LSTM. This indicates that target information is a useful feature for stance classification.

4.6 Ablation Study

We further experiment on different variations of the MTL-Stance model to analyze the extent to which various features of our model contribute to the performance. The variants of the model are as follows:

– **Single:** This model does not use multi-task learning framework. The model is trained with only stance labels.
– **Single + subj.:** This model uses multi-task framework and uses subjectivity labels along with the stance labels.
– **Single + subj. + $\mathbf{reg_{loss}}$:** This model further adds regularization loss (see Section 3.3) to add penalty to mismatched output.
– **Single + sent.:** This model uses multi-task framework and uses sentiment labels along with the stance labels.
– **MTL-Stance:** Our final model that uses multi-task learning with regularization loss. This model uses all the three labels: subjectivity, sentiment and stance.

Table 4 provides the performance results of these models. As seen, understanding the subjectivity of a tweet towards the target helps the model make better judgment about its stance. Intuitively, a tweet that has no stance towards the target tends to be objective while the one with opinion tends to be subjective. Addition of regularization penalty further improves the overall performance.

Analyzing the confusion matrix between the sentiment and stance labels reveals that stance and sentiment are not correlated [18]. Yet, addition of sentiment classification task in MTL improves performance of the model. This indicates the presence of common underlying relationships that the model is able to learn and exploit.

Also note that our single model consists of a very simple architecture and does not beat the state-of-the-art models described in Table 3. But the same architecture outperforms them with a multi-task learning objective and regularization loss. This indicates that the performance can be significantly improved if complex neural architectures are combined with the multi-task learning for stance classification.

Model: stance	Atheism	Climate	Feminism	Hillary	Abortion	$MacF_{avg}$	$MicF_{avg}$
	63.71	43.89	58.75	63.12	63.05	58.50	67.40
+ subj.	66.27	51.70	56.70	62.57	65.03	60.45	67.41
+ subj. + $\mathrm{reg_{loss}}$	64.87	51.53	**60.09**	64.25	65.49	61.25	68.40
+ sent.	**66.30**	62.06	56.19	63.11	64.44	62.42	67.76
MTL-Stance	66.15	**64.66**	58.82	**66.27**	**67.54**	**64.69**	**69.88**

Table 4: Ablation across auxiliary tasks. Note: *subj.* = subjectivity (Task B) , *sent.* = sentiment (Task C)

4.7 Importance of regularization

Table 5 compares the effect of regularization loss that we have introduced in this paper. The regularization loss allows the model to learn the correlation between

Reg_{Loss}	Atheism	Climate	Feminism	Hillary	Abortion	$MacF_{avg}$	$MicF_{avg}$
No	64.03	63.75	58.46	64.21	68.58	63.81	68.13
Yes	66.15	64.66	58.82	66.27	67.54	64.69	69.88

Table 5: MTL-Stance with and without regularization loss

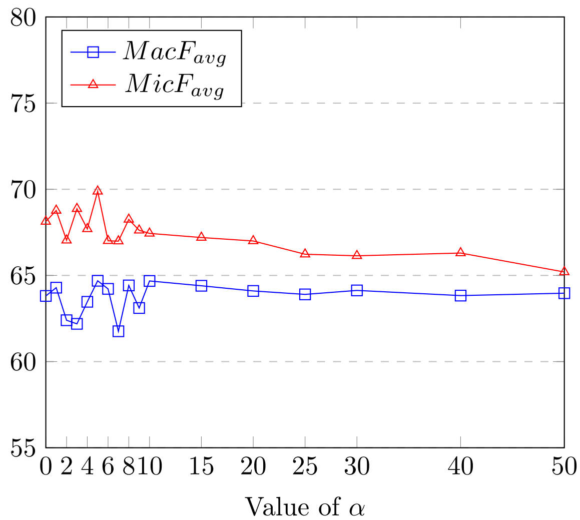

Fig. 2: Performance plot of the MTL-Stance model when α in regularization loss is varied.

subjectivity and stance more effectively by penalizing when the model predicts a tweet as subjective but a neutral stance (or vice-versa). The performance improvement shows the effectiveness of the regularization in our model.

4.8 Effect on regularization strength (α)

Figure 2 shows the performance trend of the model as α is varied in regularization loss. At $\alpha = 5$, the model reaches the highest performance with $MicF_{avg} = 69.88$ and $MacF_{avg} = 64.69$. As the value of α is increased, we observe that the performance of the model starts dropping. This is expected as the model starts under-fitting after it exceeds the α value of 10, and continues to drop in performance as it is increased.

4.9 Case-study and Error Analyses

We present an analysis of few instances where our model, MTL-Stance, succeeds and also fails in predicting the correct stance. We also include the subjectivity of the tweet that the model predicts for more insights into the model's behavior.

Tweet: @violencehurts @WomenCanSee The most fundamental right of them all, the right to life, is also a right of the unborn. #SemST
Target: Legalization of Abortion

Actual Stance: `Against`
Predicted Stance: `Against`
Actual Sentiment: `Positive`
Predicted Sentiment: `Positive`
Predicted Subjectivity: `Subjective`

This is an example of a tweet having positive sentiment while having an opposing opinion towards the target. Though the stance of the tweet towards the target is `Against`, the overall sentiment of the tweet without considering target is `Positive`. We observe that MTL-Stance is able to capture such complex relationships across many instances in the test set.

Tweet: @rhhhhh380 What we need to do is support all Republicans and criticize the opposition. #SemST
Target: Hillary Clinton

Actual Stance: `Against`
Predicted Stance: `None`
Predicted Subjectivity: `Subjective`

For the tweet above MTL-Stance predicts `None` whereas the true stance is `Against`. The tweet is targeted towards 'Hillary Clinton', but we observe that the author is referring to Republicans and not the target directly. This is a challenging example since it requires knowledge about the relation of both entities (Hillary and Republicans) to predict the stance label correctly. MTL-Stance. however, is able to correctly perdict the subjective label which demonstrates that it is able to capture some of these patterns in the coarse-grained classification.

Tweet: Please vote against the anti-choice amendment to the Scotland Bill on Monday @KevinBrennanMP - Thanks! #abortionrights #SemST
Target: Legalization of Abortion

Actual Stance: `Against`
Predicted Stance: `Favor`
Predicted Subjectivity: `Subjective`

The above instance suggests other challenges that models face in predicting the stance correctly. The tweet has multiple negations in it and requires multi-hop inference in order to come to the right conclusion about the stance. Handling such use cases demands rigorous design and fundamental reasoning capabilities.

5 Conclusion

In this paper, we introduced MTL-Stance, a novel model that leverages sentiment and subjectivity information for stance classification through a multi-task learning setting. We also propose a regularization loss that helps the model to learn the correlation between subjectivity and stance more effectively. MTL-Stance uses simple end-to-end model with CNN architecture for stance classification. In addition, it does not use any kind of extra linguistic features or pipeline methods. The experimental results shows that MTL-Stance outperforms state-of-the-art models on the Twitter Stance Detection benchmark dataset.

Acknowledgment

This research is supported by Singapore Ministry of Education Academic Research Fund Tier 1 under MOE's official grant number T1 251RES1820.

References

1. Rakesh Agrawal, Sridhar Rajagopalan, Ramakrishnan Srikant, and Yirong Xu. Mining newsgroups using networks arising from social behavior. In *Proceedings of the 12th international conference on World Wide Web*, pages 529–535. ACM, 2003.
2. Pranav Anand, Marilyn Walker, Rob Abbott, Jean E Fox Tree, Robeson Bowmani, and Michael Minor. Cats rule and dogs drool!: Classifying stance in online debate. In *Proceedings of the 2nd workshop on computational approaches to subjectivity and sentiment analysis*, pages 1–9. Association for Computational Linguistics, 2011.
3. Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. Stance detection with bidirectional conditional encoding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 876–885, 2016.
4. Dzmitry Bahdanau, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio. End-to-end attention-based large vocabulary speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 4945–4949. IEEE, 2016.
5. Wei-Fan Chen and Lun-Wei Ku. Utcnn: a deep learning model of stance classification on social media text. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1635–1645, 2016.
6. Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM, 2008.
7. Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537, 2011.
8. Li Deng, Geoffrey Hinton, and Brian Kingsbury. New types of deep neural network learning for speech recognition and related applications: An overview. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8599–8603. IEEE, 2013.

9. Kuntal Dey, Ritvik Shrivastava, and Saroj Kaushik. Topical stance detection for twitter: A two-phase lstm model using attention. In *European Conference on Information Retrieval*, pages 529–536. Springer, 2018.
10. Jiachen Du, Ruifeng Xu, Yulan He, and Lin Gui. Stance classification with target-specific neural attention networks. International Joint Conferences on Artificial Intelligence, 2017.
11. Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.
12. Kazi Saidul Hasan and Vincent Ng. Why are you taking this stance? identifying and classifying reasons in ideological debates. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 751–762, 2014.
13. Myungha Jang and James Allan. Explaining controversy on social media via stance summarization. *arXiv preprint arXiv:1806.07942*, 2018.
14. Mirko Lai, Delia Irazú Hernández Farías, Viviana Patti, and Paolo Rosso. Friends and enemies of clinton and trump: using context for detecting stance in political tweets. In *Mexican International Conference on Artificial Intelligence*, pages 155–168. Springer, 2016.
15. Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.
16. Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiao-Dan Zhu, and Colin Cherry. A dataset for detecting stance in tweets. In *LREC*, 2016.
17. Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41, 2016.
18. Saif M Mohammad, Parinaz Sobhani, and Svetlana Kiritchenko. Stance and sentiment in tweets. *ACM Transactions on Internet Technology (TOIT)*, 17(3):26, 2017.
19. Akiko Murakami and Rudy Raymond. Support or oppose?: classifying positions in online debates from reply activities and opinion expressions. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 869–875. Association for Computational Linguistics, 2010.
20. Isaac Persing and Vincent Ng. Modeling stance in student essays. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 2174–2184, 2016.
21. Lahari Poddar, Wynne Hsu, Mong Li Lee, and Shruti Subramaniyam. Predicting stances in twitter conversations for detecting veracity of rumors: A neural approach. In *2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 65–72. IEEE, 2018.
22. Ashwin Rajadesingan and Huan Liu. Identifying users with opposing opinions in twitter debates. In *International conference on social computing, behavioral-cultural modeling, and prediction*, pages 153–160. Springer, 2014.
23. Sebastian Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.
24. Swapna Somasundaran and Janyce Wiebe. Recognizing stances in ideological on-line debates. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 116–124. Association for Computational Linguistics, 2010.

25. Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.
26. Qingying Sun, Zhongqing Wang, Qiaoming Zhu, and Guodong Zhou. Stance detection with hierarchical attention network. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2399–2409, 2018.
27. Matt Thomas, Bo Pang, and Lillian Lee. Get out the vote: Determining support or opposition from congressional floor-debate transcripts. In *Proceedings of the 2006 conference on empirical methods in natural language processing*, pages 327–335. Association for Computational Linguistics, 2006.
28. Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.
29. Prashanth Vijayaraghavan, Ivan Sysoev, Soroush Vosoughi, and Deb Roy. Deepstance at semeval-2016 task 6: Detecting stance in tweets using character and word-level cnns. *Proceedings of SemEval*, pages 413–419, 2016.
30. Marilyn A Walker, Pranav Anand, Robert Abbott, and Ricky Grant. Stance classification using dialogic properties of persuasion. In *Proceedings of the 2012 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 592–596. Association for Computational Linguistics, 2012.