# Connecting Targets to Tweets: Semantic Attention-Based Model for Target-Specific Stance Detection

Yiwei Zhou[1(✉)], Alexandra I. Cristea[1], and Lei Shi[2]

[1] Department of Computer Science, University of Warwick, Coventry, UK
{Yiwei.Zhou,A.I.Cristea}@warwick.ac.uk
[2] University of Liverpool, Liverpool, UK
Lei.Shi@liverpool.ac.uk

**Abstract.** Understanding what people say and really mean in tweets is still a wide open research question. In particular, understanding the *stance* of a tweet, which is determined not only by its content, but also by the given *target*, is a very recent research aim of the community. It still remains a challenge to construct a tweet's vector representation with respect to the target, especially when the target is only *implicitly mentioned*, or *not mentioned at all* in the tweet. We believe that better performance can be obtained by incorporating the information of the target into the tweet's vector representation. In this paper, we thus propose to embed a *novel attention mechanism at the semantic level* in the bi-directional GRU-CNN structure, which is more fine-grained than the existing token-level attention mechanism. This novel attention mechanism allows the model to automatically attend to useful semantic features of informative tokens in deciding the target-specific stance, which further results in a conditional vector representation of the tweet, with respect to the given target. We evaluate our proposed model on a recent, widely applied benchmark Stance Detection dataset from Twitter for the SemEval-2016 Task 6.A. Experimental results demonstrate that the proposed model substantially outperforms several strong baselines, which include the state-of-the-art token-level attention mechanism on bi-directional GRU outputs and the SVM classifier.

**Keywords:** Target-specific Stance Detection · Text classification · Neural network · Attention mechanism

## 1 Introduction

Target-specific Stance Detection is a problem that can be formulated as follows: given a tweet $X$ and a target $Y$, the aim is to classify the stance of $X$ towards $Y$ into three categories, *Favour*, *None* or *Against*. The target may be a

Y. Zhou—Work performed while at The Alan Turing Institute.


A. Bouguettaya et al. (Eds.): WISE 2017, Part I, LNCS 10569, pp. 18–32, 2017.
DOI: 10.1007/978-3-319-68783-4_2

person, an organisation, a government policy, a movement, a product, etc. [8]. Target-specific Stance Detection is a different problem from Aspect-level Sentiment Analysis [11,15] in the following ways: the same stance can be expressed through positive, negative or neutral sentiment [9]; the target of interest of the Stance Detection does not necessarily have to occur in the tweet, as the target-specific stance can be expressed by mentioning the target implicitly, or by talking about other relevant targets. Besides typical tweets characteristics, such as being short and noisy, the main challenge in this task is that the decision made by the classifier has to be target-specific, *whilst having very little contextual information or supervision provided*. Example training data from the benchmark target-specific Stance Detection dataset for SemEval-2016 Task 6 [8] can be found in Table 1. Deep neural networks enable the continuous vector representations of underlying semantic and syntactic information in natural language texts, and save researchers the efforts of feature engineering [14,15]. Recently, they have achieved significant improvements in various natural language processing tasks, such as Machine Translation [2,3], Question Answering [14], Sentiment Analysis [6,11,15,18], etc. However, applying deep neural networks on target-specific Stance Detection has not been successful, as their performances have, up to now, been slightly worse than traditional machine learning algorithms with manual feature engineering, such as Support Vector Machines (SVM) [8].

**Table 1.** Examples of target-specific stance detection.

| Target | Tweet | Stance |
|---|---|---|
| Donald Trump | #DonaldTrump my tell it like it is but his comments speaks to a prejudice and cold heart | *Against* |
| Hillary Clinton | I love the smell of Hillary in the morning. It smells like Republican Victory | *Against* |
| Hillary Clinton | Just think how many emails Hillary Clinton can delete with today's #leapsecond | *Against* |
| Climate Change | Coldest and wettest summer in memory | *Favour* |

In this work, the above challenges are tackled, based on our intuition that the target information is vital for the Stance Detection, and that the vector representations for the tweets should be "aware" of the given targets. Since not all parts in the tweet are equally helpful for the Stance Detection task towards the specified target, we firstly apply the state-of-the-art token-level attention mechanism [2]. This allows neural networks to automatically pay more attention to the tokens that are more relevant to the target and more informative for detecting the target-specific stance. Importantly, a given token can be interpreted differently, according to different targets, and the semantic features in the token's vector representation can be of different levels of importance, conditional on the given target. We propose a novel attention mechanism, which extends the current attention mechanism, from the *token level*, to the *semantic*

*level*, through a *gated structure*, whereby the tokens can be encoded adaptively, according to the target. We compare the models we propose based on the token-level attention mechanism and the novel semantic-level attention mechanism with several baselines, on the target-specific Stance Detection dataset for the SemEval-2016 Task 6.A [8], which is currently the most widely applied dataset on target-specific Stance Detection in tweets. The experimental results show that substantial improvements can be achieved on this task, compared with all previous neural network-based models, by inferencing conditional tweet vector representations with respect to the given targets; the neural network model with semantic-level attention also outperforms the SVM algorithm, which achieved the previous best performance in this task [8]. Additionally, it should be noted that our results are obtained with a *minimum of supervision*, with *no external domain corpus collected* to pre-train target-specific word embeddings, and *no extra sentiment information annotated*. Moreover, there are *no target-specific configurations or hand-engineered features involved*, thus *the proposed models can be easily generalised to other targets*, with no additional efforts.

## 2 Neural Network Models for Target-Specific Stance Detection in Tweets

In this section, we first describe two baseline models, the bi-directional Gated Recurrent Unit (biGRU) model, and the model that stacks a Convolutional Neural Network (CNN) structure on the outputs of the biGRU (biGRU-CNN) model. We then show how we extend these two baseline models, by incorporating the target information through *token-level* and *semantic-level attention mechanisms*, obtaining the AT-biGRU model and the AS-biGRU-CNN model, respectively. Finally, we demonstrate methods to generate the target embedding, and how to obtain the stance detection result based on the tweet vector representation, as well as other model training details.

### 2.1 biGRU Model

GRU [3] aims at solving the gradient vanishing or exploding problems, by introducing a gating mechanism. It adaptively captures dependencies in sequences, without introducing extra memory cells. GRU maps an input sequence of length $N$, $[x_1, x_2, \cdots, x_N]$ into a set of hidden states $[h_1, h_2, \cdots, h_N]$ as follows:

$$r_n = \sigma(W_r x_n + U_r h_{n-1} + b_r) \tag{1}$$

$$z_n = \sigma(W_z x_n + U_z h_{n-1} + b_z) \tag{2}$$

$$\tilde{h_n} = \tanh(W_h x_n + U_h(r_n \odot h_{n-1}) + b_h) \tag{3}$$

$$h_n = (1 - z_n) \odot h_{n-1} + z_n \odot \tilde{h_n}. \tag{4}$$

where $n \in \{1, \ldots, N\}$; $r_n$ is the reset gate and $z_n$ is the update gate; $\tilde{h_n} \in \mathbb{R}^{d_1}$ represents the "candidate" hidden state generated by the GRU; $h_n \in \mathbb{R}^{d_1}$

represents the real hidden state generated by the GRU; $x_n \in \mathbb{R}^{d_0}$ represents the word embedding vector of a token in the tweet; $W_r$, $W_z$, $W_h \in \mathbb{R}^{d_1 \times d_0}$ and $U_r$, $U_z$, $U_h \in \mathbb{R}^{d_1 \times d_1}$ represent the weight matrices; $b_r$, $b_z$, $b_h \in \mathbb{R}^{d_1}$ represent the bias terms; $\sigma(\cdot)$ represents the sigmoid function; $\odot$ represents the Hadamard product operation (element-wise multiplication).

To capture the information from both the past and the future sequence, the bi-directional GRU (biGRU), which processes the sequence in both the forward and backward directions, has proven to be successful in various applications [2,18]. In biGRU, the hidden states generated by processing the sequence in opposite directions are concatenated as the new output: $[\overrightarrow{h_1} \| \overleftarrow{h_1}, \overrightarrow{h_2} \| \overleftarrow{h_2}, \cdots, \overrightarrow{h_N} \| \overleftarrow{h_N}]$, where $\overrightarrow{h_n} \| \overleftarrow{h_n} \in \mathbb{R}^{2d_1}$, and the arrow represents the direction of the processing.

In the biGRU model, the final hidden states of the input sequence, when processing it in opposite directions, are concatenated, to form the vector representation of the tweet $s$:

$$s = \overrightarrow{h_N} \| \overleftarrow{h_1}. \tag{5}$$

### 2.2 biGRU-CNN Model

The biGRU model attempts to propagate all the semantic and syntactic information in a tweet into two fixed hidden state vectors, which could become a bottleneck, when there exist some long-distance dependencies in the tweet. In [14], Recurrent Neural Network (RNN) outputs were fed into a CNN structure, to generate a vector representation, based on all the hidden states of the RNN, rather than just the final hidden state. Specifically, a filter $w_f \in \mathbb{R}^{2kd_1}$ is applied to $k$ concatenated consecutive hidden states $h_{i:i+k-1} \in \mathbb{R}^{2kd_1}$ to compute $c_i$, one value in the feature map corresponding to this filter:

$$c_i = f(w_f^T h_{i:i+k-1} + b_f), \tag{6}$$

where $f$ is the rectified linear unit function and $b_f \in \mathbb{R}$ is a bias term. A max-pooling operation is further applied over the feature map $\mathbf{c} = (c_1, c_2, \cdots, c_{N-k+1})$, to capture the most important semantic feature $\hat{c}$ in each feature map:

$$\hat{c} = \max\{\mathbf{c}\}. \tag{7}$$

$\hat{c}$ is the feature generated by filter $w_f$. Filters with varying sliding window sizes $k$ can be applied, to obtain multiple features. The features generated by different filters are concatenated, to form the vector representation of the tweet $s$.

### 2.3 AT-biGRU Model

Whilst they solve specific problems as above, neither the biGRU model nor the biGRU-CNN model takes into account the target information. However, when human annotators are asked to label the stance of a tweet towards a given target, they are likely to keep the information about the target in their mind, and pay

more attention to the parts relevant to the target. The *token-level attention mechanism*, firstly proposed in [2] for Machine Translation, allowed the neural network to automatically search for tokens of a source sentence that were relevant to predicting a target word, and mask irrelevant tokens; it released the burden on RNN in compressing the entire source sentence into a static, fixed representation. The attention mechanism has been successfully applied in Question Answering [14], Caption Generation [17], Sentiment Analysis [18], etc.

In this paper, we propose to apply the attention mechanism to the biGRU model, to enable the model to automatically compute proper alignments in the tweet, which reflect the importance levels of different tokens in deciding the tweet's stance towards the given target, as shown in Fig. 1.

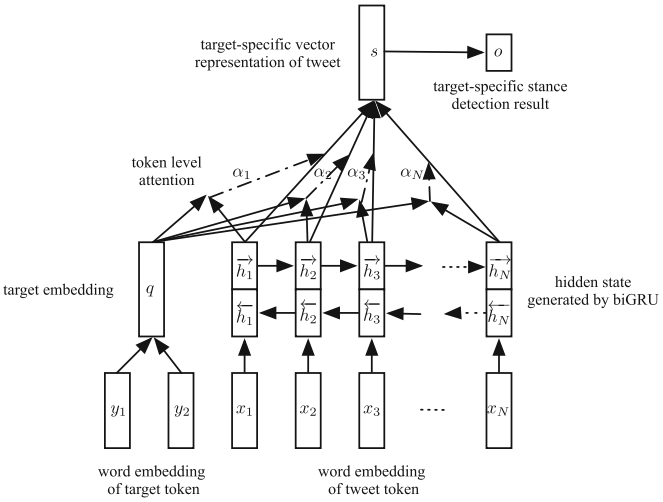


**Fig. 1.** The AT-biGRU model for target-specific stance detection.

In the AT-biGRU model, the vector representation $s$ of the tweet is calculated as the weighted sum of the hidden states:

$$s = \sum_{n=1}^{N} \alpha_n h_n. \tag{8}$$

In the above equation, the weight $\alpha_n$ of each hidden state $h_n$ is computed by:

$$\alpha_n = \frac{\exp(e_n)}{\sum_{n=1}^{N} \exp(e_n)}, \tag{9}$$

where $e_n \in \mathbb{R}$ is calculated through a multi-layer perceptron that takes $h_n$ and the target embedding $q$ as input, specifically:

$$e_n = att(h_n, q) = w_m^T(\tanh(W_{ah}h_n + W_{aq}q + b_a)) + b_m. \tag{10}$$

where $W_{ah} \in \mathbb{R}^{2d_1 \times 2d_1}$; $W_{aq} \in \mathbb{R}^{2d_1 \times d_2}$; $b_a$, $w_m \in \mathbb{R}^{2d_1}$; $b_m \in \mathbb{R}$ are token-level attention parameters to optimise. In Sect. 2.5, we explore various ways to generate the target embedding $q \in \mathbb{R}^{d_2}$, based on the embeddings of the tokens in the target $Y$, denoted by $y_1$, $y_2 \in \mathbb{R}^{d_0}$. The weight $\alpha_n$ can be interpreted as the degree to which the model attends to token $x_n$ in the tweet, while deciding the stance of the tweet towards the given target.

### 2.4 AS-biGRU-CNN Model

The model we propose above is an improvement on prior research. However, it can be further refined, as follows. The AT-biGRU model applies the attention mechanism at the token level, which enables the model to pay more attention to the tokens that have contributed to the stance decision towards specified targets. However, in the AT-biGRU model, the vector representations of the tokens do not have direct interaction with the vector representation of the target, which is against the intuition that the target can influence the human annotators' interpretation of each token. For example, the token 'email' in Table 1 implies an *Against* stance towards the target "Hillary Clinton", but has no obvious influence on stances towards other targets; the token "cold" can either reveal the user's *Favour* stance towards the target "Climate Change is a Real Concern", or suggest the user's *Against* stance towards the target "Donald Trump".

Thus, we use a gated structure to extend the current token-level attention mechanism to a *more fine-grained semantic level*, by introducing the direct interaction between the hidden states and the vector representation of the target. The gated structure can be embedded into the biGRU-CNN model, which results in the AS-biGRU-CNN model, as shown in Fig. 2.

In Fig. 2, we introduce the *target-specific hidden state* $h_n^{'}$, to replace the original hidden state $h_n$ generated by biGRU. The target-specific hidden state is calculated as follows:

$$h_n^{'} = a_n \odot h_n. \tag{11}$$

The attention vector $a_n \in \mathbb{R}^{2d_1}$ decides which semantic features in each hidden state are meaningful specifically towards the target, which is calculated through a gated structure, as follows:

$$a_n = \sigma(W_m(\tanh(W_{ah}h_n + W_{aq}q + b_a)) + b_m). \tag{12}$$

where $W_{ah}$, $W_m \in \mathbb{R}^{2d_1 \times 2d_1}$; $W_{aq} \in \mathbb{R}^{2d_1 \times d_2}$; $b_a$, $b_m \in \mathbb{R}^{2d_1}$ are semantic-level attention parameters, to optimise in the gated structure. The methods to derive the target embedding $q \in \mathbb{R}^{d_2}$ based on the embeddings of the tokens in the target $Y$, denoted by $y_1$, $y_2 \in \mathbb{R}^{d_0}$, will be explained in Sect. 2.5. The elements in the attention vector $a_n$ can be understood as the degrees to which the model attends to the semantic features of token $x_n$ in the tweet, while deciding the stance of the tweet towards the given target.

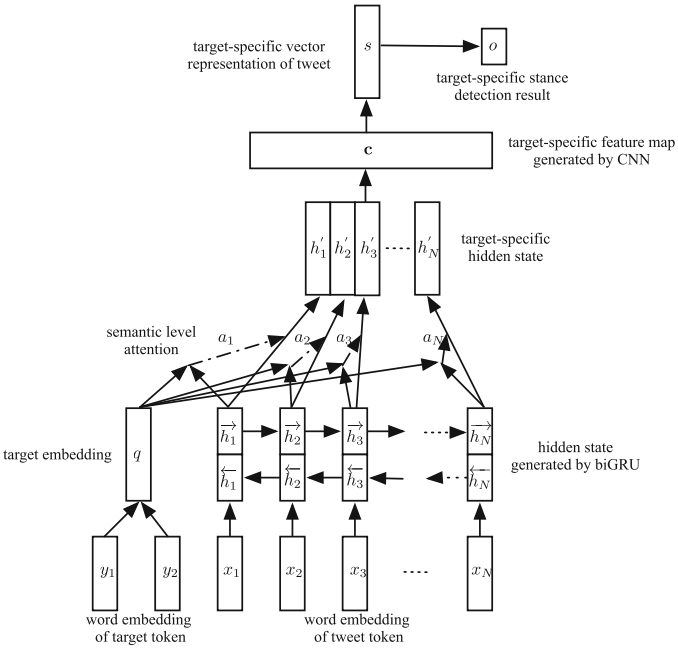


**Fig. 2.** The AS-biGRU-CNN model for target-specific stance detection.

### 2.5 Target Embedding

The models proposed in Sects. 2.3 and 2.4 employ the embedding of the given target $q \in \mathbb{R}^{d_2}$, derived from the embeddings of the tokens in the given target $y_1, y_2 \in \mathbb{R}^{d_0}$. Without loss of generality, here we use a target with two tokens, as an example. However, the methods can be directly applied on targets with any number of tokens. To generate target embeddings of the same dimensionality for the targets with different token numbers, we propose to use a separate biGRU model, described in Sect. 2.1, with the target token embeddings $y_1$ and $y_2$ as inputs. For this scenario, the dimensionality of $q$, denoted by $d_2$ in Sects. 2.3 and 2.4, equals to the dimensionality of the concatenated final hidden states of the biGRU model, denoted by $2d_1$. Results of the AT-biGRU model and the AS-biGRU-CNN model using the biGRU target embedding are reported in Sect. 3.4. In some aspect-level Sentiment Analysis works, researchers have been using the average of the aspect token embeddings to encode the aspect [11,15]. We also use the averaging method as a baseline target encoding approach to derive the target embedding $q$, by averaging the target token embeddings $y_1$ and $y_2$. For this scenario, $d_2$ equals to the dimensionality of the target token embeddings, denoted by $d_0$. Results of the AT-biGRU model and the AS-biGRU-CNN model using the averaging target embedding are reported in Sect. 3.5.

### 2.6 Model Training

The vector representation of the tweet $s$ is fed as input to a softmax layer, after a linear transformation step that transforms it into a vector, whose length is equal to the number of possible stance categories. The outputs of the softmax layer (denoted by $o$ in Figs. 1 and 2) are the probabilities of the tweet $X$ belonging to the stance category $z$, given the target $Y$, denoted by $P(z|X,Y)$. The stance category with the maximum probability is selected as the *predicted category*, $z^*$:

$$z^* = argmax_{z \in \mathbf{z}} P(z|X,Y). \tag{13}$$

All the models are smooth and differentiable, and they can be trained in an end-to-end manner, with standard back-propagation. We use the cross-entropy loss as the *objective function* $L(\theta)$, which is defined as follows:

$$L(\theta) = -\sum_{X \in \mathbf{X}} \sum_{z \in \mathbf{z}} P^{'}(z|X,Y) \cdot \log(P(z|X,Y)). \tag{14}$$

where $\mathbf{X}$ is the set of training data; $\mathbf{z}$ is the set of stance categories; $P^{'}(z|X,Y)$ denotes the target stance distribution $z$ given $X$ and $Y$; $\theta$ is the set of parameters.

## 3 Experimental Results

### 3.1 Dataset Description

As said, we evaluated the effectiveness of the proposed models on the benchmark Stance Detection dataset for the SemEval-2016 Task 6.A [8]. We used the exact same data as provided to the contestants for this task, with no extra labelled data [4] or domain corpus [1,9] employed. The benchmark Stance Detection training dataset contained 2,914 tweets relevant to five targets: "Atheism" (**A**), "Climate Change is a Real Concern" (**CC**), "Feminist Movement" (**FM**), "Hillary Clinton" (**HC**) and "Legalisation of Abortion" (**LA**). Each tweet was annotated as *Favour*, *Neither* or *Against* towards one of the five targets. The benchmark Stance Detection test dataset contained 1,249 tweets, as well as the interested targets. Detailed statistics about the dataset can be found in Table 2, where "#" represents the number of tweets, "**%F**", "**%A**" and "**%N**" represent the percentages of tweets with *Favour*, *Against* and *Neither* stances towards the targets, respectively.

### 3.2 Comparison Models

We compared the proposed models with the two best performing models in the SemEval-2016 Task 6.A: (1) MITRE [19], which trained separate Long Short-Term Memory (LSTM) networks with a voting scheme for different targets—the LSTM networks were pre-trained, by an auxiliary hashtag prediction task on 298,973 self-collected tweets; (2) pkudblab [16], which also trained

**Table 2.** Statistics of the benchmark target-specific stance detection dataset.

| Target | Training | | | | Test | | | |
|---|---|---|---|---|---|---|---|---|
| | # | %F | %A | %N | # | %F | %A | %N |
| A | 513 | 17.9 | 59.3 | 22.8 | 220 | 14.5 | 72.7 | 12.7 |
| CC | 395 | 53.7 | 3.8 | 42.5 | 169 | 72.8 | 6.5 | 20.7 |
| FM | 664 | 31.6 | 49.4 | 19.0 | 285 | 20.4 | 64.2 | 15.4 |
| HC | 689 | 17.1 | 57.0 | 25.8 | 295 | 15.3 | 58.3 | 26.4 |
| LA | 653 | 18.5 | 54.4 | 27.1 | 280 | 16.4 | 67.5 | 16.1 |
| All | 2914 | 25.8 | 47.9 | 26.3 | 1249 | 24.3 | 57.3 | 18.4 |

separate CNN classifiers for different targets, with a voting scheme employed both in and out of each epoch, to improve the performance. We also compared against the SVM classifiers trained on the corresponding training datasets for the five targets, using word n-grams and character n-grams features, as reported in [8], representing the previous best performer for this task. Additionally, to illustrate the influence of the token-level and semantic-level attention mechanism, we included the performance comparison between the biGRU model (Sect. 2.1) and the AT-biGRU model (Sect. 2.3), the biGRU-CNN model (Sect. 2.2) and the AS-biGRU-CNN model (Sect. 2.4).

### 3.3 Experimental Settings and Model Configuration

In line with former works, we first trained separate classifiers for different targets. To obtain a fair comparison, we employed the *only* evaluation metric in the SemEval-2016 Task 6.A, which was the macro-average of the F1 scores for the *Favour* and *Against* stance categories. This evaluation metric will be referred to as "macro-average F1 score" in this paper for simplicity purpose. In the evaluation stage of SemEval-2016 Task 6.A, the target information of each tweet was ignored, in order to measure each team's overall performance, rather than performance on each separate target. This was because the training datasets for different targets had different percentages of tweets with Favour, Against and Neither stances, as well as different percentages of tweets expressing stances by mentioning the given target and by mentioning other targets. Thus, this evaluation metric can reflect each team's overall ability in dealing with different scenarios. It should be noted that even though separate classifiers were trained for different targets, we used the same configurations for target-specific classifiers, to make sure our proposed models can be easily applied to any other target, as well as effectively demonstrate the advantages of target-specific tweet vector representation, by eliminating the effects of target-specific model settings. Various methods were applied to avoid overfitting. We performed a standard 5-fold cross-validation. For each round of cross-validation, we experimentally set the maximum number of epochs to 50, and located the epoch that achieved the best performance on the validation dataset. The post-softmax probabilities of the 5

trained classifiers were averaged, to obtain the probabilities of a tweet in the test dataset belonging to the three stance categories.

We implemented the proposed models using Theano[1] and Keras[2].

For comparison fairness, all the neural network-based models in the experiments also used the same hyper-parameters (as illustrated below), which were selected using grid search on the baseline biGRU model. In the experiments, all the word embeddings were initialised by the Glove [10] 100-dimensional pre-trained embeddings on Wikipedia data, i.e., $d_0 = 100$. We applied dropout [13] with probability 0.2 on the embedding layer. The word embeddings were fine-tuned during the training process, to capture the stance information. From the preliminary experiments, we observed that the models that shared the embedding layer between the tweets and the targets performed significantly better than the models that did not. We chose the dimensionality of hidden states ($d_1$) of both the GRU encoding the tweet and the GRU encoding the target to be 64, and the GRU weights are initialised from a uniform distribution $U(-\epsilon, \epsilon)$. Following [5], we added a dropout level of 0.3 between each recurrent connection in the GRU that encoded the tweets. We further selected the hyper-parameters for the CNN structure on top of the fixed hyper-parameters of the biGRU model. Following [6], we used filters of $k \in \{3, 4, 5\}$, with widths equal to the dimensionality of the outputs of the biGRU, which was 128 in this case. There were 100 filters for each size. To increase the robustness of the models to overfitting, a dropout level of 0.5 was further applied before the softmax layer.

We used the Adam optimiser [7] for back-propagation with the two momentum parameters set to 0.9 and 0.999, respectively. The mini-batch size was set to 16. The code for the experiments is available at https://github.com/zhouyiwei/tsd.

### 3.4 Using the biGRU Target Embedding

The experimental results are shown in Table 3. Besides the evaluation metric of the SemEval-2016 Task6.A, we also provide the macro-average F1 scores of different targets, as references. From the comparison between the biGRU model and the biGRU-CNN model, it can be seen that the CNN structure on top of the biGRU model can help to generate more compact and abstract vector representations of the tweets for Stance Detection.

Both neural network-based models that incorporate target information when generating vector representations for the tweets, i.e., the AT-biGRU and AS-biGRU-CNN, outperform other neural network-based models that did not, i.e., MITRE, pkudblab, biGRU and biGRU-CNN. Specifically, the state-of-the-art token-level attention mechanism helps to increase the performance of the biGRU model by 0.32 in the overall macro-average F1 score. The injection of target information through the proposed semantic-level attention mechanism in the biGRU-CNN model, which results in the AS-biGRU-CNN model, leads to a

[1] http://deeplearning.net/software/theano/.
[2] https://keras.io/.

more significant improvement (1.71) on the basis of the biGRU-CNN model, which makes it the best performing model among all the neural network-based models. This demonstrates the effectiveness of attention mechanisms in constructing a composite vector representation between the target and contextual information provided in the tweet. The proposed AS-biGRU-CNN model with semantic-level attention, however, has stronger capability in modelling the complex interaction between the target and each token in the tweet, and generating an expressive conditional vector representation of the tweet, with respect to the target, compared with the AT-biGRU model with the token-level attention.

Moreover, the AS-biGRU-CNN model outperforms the traditional SVM algorithm, with word n-grams and character n-grams features reported in [8] by a substantial margin, in the absence of feature engineering and target-specific tuning, which justifies the motivation to automatically intensify the features that are essential to the target, and "dilute" the features that are not.

**Table 3.** Performance of target-specific stance detection based on the macro-average F1 score, using separate classifiers.

| Model | Target | | | | | Overall |
|---|---|---|---|---|---|---|
| | A | CC | FM | HC | LA | |
| SVM | 65.19 | 42.35 | 57.46 | 58.63 | 66.42 | 68.98 |
| MITRE | 61.47 | 41.63 | 62.09 | 57.67 | 57.28 | 67.82 |
| pkudblab | 63.34 | 52.69 | 51.33 | 64.41 | 61.09 | 67.33 |
| biGRU | 65.26 | 43.08 | 56.53 | 55.60 | 61.39 | 67.65 |
| biGRU-CNN | 63.42 | 42.91 | 58.69 | 55.11 | 60.55 | 67.71 |
| AT-biGRU | 62.32 | 43.89 | 54.15 | 57.94 | 64.05 | 67.97 |
| AS-biGRU-CNN | 66.76 | 43.40 | 58.83 | 57.12 | 65.45 | **69.42** |

### 3.5 Using the Averaging Target Embedding

In Table 3, we used biGRU to generate the vector representations for the targets. Additionally, we further experimented with the AT-biGRU and AS-biGRU-CNN models, using the averaging target embeddings. The overall macro-average F1 score of the AT-biGRU model increases from 67.97 to 68.30, while the macro-average F1 score of the AS-biGRU-CNN model decreases from 69.42 to 68.35. One possible explanation could be that a simple averaging approach is insufficient to capture the semantic meanings of the targets, thus for the biGRU-CNN model, which has stronger expressive power than the biGRU model in target-specific Stance Detection, it is helpful to use more flexible target embeddings to perform complex inference. However, for the AT-biGRU model, the target embeddings generated by biGRU surpass its capability to learn and generalise. *This is also the reason why stacking the CNN structure on top of the AT-biGRU model cannot help to improve the performance, as it does in the AS-biGRU-CNN model.*

### 3.6 Using Combined Classifiers

In the Stance Detection dataset for the SemEval-2016 Task 6.A, the training data for all the targets were of similar sizes, except for the target "Climate Change is a Real Concern". There were only 395 items in its training data and they were highly biased, with only 3.8% of them coming from the *Against* category. As a result of this, all the models in Table 3 cannot achieve a comparable performance on this target, when compared with other targets. When there was not enough training data for some targets, or the training data for some targets was highly biased, it was not possible to guarantee the performance of independent classifiers for these targets. For this case, we hypothesised that a combined classifier of all the targets can alleviate this problem, through jointly modelling the interaction between the stances and contexts of all the available targets. This way, when performing Stance Detection on the "Climate Change is a Real Concern" target, the classifier can employ—or even transfer—the knowledge about the intricate connection between the stances and contexts learnt from the training data of other targets. Motivated by this idea, we further trained combined classifiers based on the proposed models, using all the training data, rather than trained separate classifiers for different targets. The combined classifiers' performances are shown in Table 4.

**Table 4.** Performance of target-specific stance detection based on the macro-average F1 score, using combined classifiers.

| Model | CC | Overall |
|---|---|---|
| SVM | 47.76 | 62.06 |
| biGRU | 54.14 | 62.82 |
| biGRU-CNN | 54.57 | 62.70 |
| AT-biGRU | 55.69 | 63.36 |
| AS-biGRU-CNN | **58.24** | **67.40** |

In Table 4, we use the combined SVM classifier reported in [8] as a baseline. For combined classifiers, richer semantic and syntactic information was needed in the tweets' vector representations, as it was necessary to additionally encode the relatedness and diversity of different targets in stance expressions. This was a much harder task, as the combined classifier had to employ useful knowledge from other targets and avoid the impairment of useless information. For this reason, we continued to employ the biGRU model to generate the target embeddings, which had stronger expressive power than the averaging method. The difficulty level of this task is illustrated by the significant diminished overall macro-average F1 score of the SVM combined classifier in Table 4, compared with the overall macro-average F1 score of the SVM separate classifiers in Table 3. We experimentally increased the dimensionality of the pre-trained word embedding vectors from 100 to 300, and the dimensionality of the hidden states of GRU from 64 to 256, to satisfy the above requirements. All the other hyper-parameters were kept the same, as illustrated in Sect. 3.3.

From Table 4, it can be observed that for the target "Climate Change is a Real Concern", it is helpful for all models to employ the training data from other targets. Comparatively, combined classifiers using models based on neural networks achieve much better macro-average F1 scores on this target than the combined classifiers using the traditional SVM algorithm. This is because the neural network-based models employed continuous vector representations of tweets, which allows them to more easily incorporate information from other domains, compared with the traditional SVM algorithm, which employs sparse and discrete vector representations, based on feature engineering. The combined classifier using the proposed AS-biGRU-CNN model yields the best performance so far on the "Climate Change is a Real Concern" target, which further illustrates the model's strong ability to capture the generality in stance expressions of different targets. However, the overall performance of the combined classifiers all decreases. This is because the performances for targets with sufficient training data can be negatively influenced by the redundant information from other targets. Nevertheless, the AS-biGRU-CNN model still yields the best overall performance, using only combined classifiers, which shows the model's power in modelling the differences in stance expressions of different targets.

## 4 Related Work

Very few recent researches attempted to tackle the target-specific Stance Detection task on tweets, such as [1,4,9,16,19]. [1] focused on *predicting the stances towards targets with no training data provided*, which was the SemEval-2016 Task 6.B, a different task to the one studied here. For the problem we tackled in this work, there was a training dataset for each specified target, to effectively update the states and memories of the encoders. [4] studied the correlation between sentiment and stance, and the sentiment labels of the tweets were additionally needed to train the model. Thus, *the settings of both above researches were different from the settings of the SemEval-2016 Task 6.A.* [16,19] ignored the target information while performing classification, whereas our experiments have clearly proven that the target-specific vector representation of tweets can substantially boost the performance. [9] relied on feature engineering and large domain corpus, to perform feature selection, which was hard to generalise to other targets; and the collection of domain corpus additionally added difficulty, because of the limitations of the Twitter API. *The attention-based models proposed in this work, on the contrary, are fully automatic, with minimum supervision. We did not collect any extra domain corpus or use any linguistic tools, and no feature engineering was needed. Since no target-specific configurations are involved, the proposed models can be directly applied to other targets.*

Another track of relevant research is aspect-level Sentiment Analysis on texts [11,12,15]. In this task, the text to be analysed, or at least part of the text, focuses on the aspects of interest, by explicitly mentioning the aspects, which renders the problem of modelling the importance and relatedness of tokens with respect to the aspects, easier. *However, this is not the case for the target-specific*

*Stance Detection task.* Thus, a deeper integration between the target and the tweet, and a more complex inference mechanism, are needed, as proposed in our research.

## 5 Conclusion

To the best of our knowledge, we are the first ones to effectively apply the traditional token-level attention mechanism to the problem of target-specific Stance Detection in tweets, which achieves better performance than other neural network-based models. Moreover, we propose to use a gated structure on the basis of the biGRU-CNN model, to embed target information into the tweet's vector representation, aiming at introducing the direct semantic interaction between the target and each token in the tweet, to perform *target-specific Stance Detection*. The proposed model employs a *semantic-level attention mechanism*, which is more fine-grained than the token-level attention mechanism. The proposed semantic-level attention mechanism searches for certain semantic features of each token in the tweet, based on the information contribution these semantic features have, in deciding the stance of the tweet, towards the given target. For the resulting AS-biGRU-CNN model, not only the tweet's representation vector, but also the representation vectors of the tokens are target-specific. The experimental results demonstrates that the proposed model outperforms several state-of-the-art baselines, in terms of macro-average F1 score, on the benchmark target-specific Stance Detection dataset of tweets, for both the scenario when separate classifiers are allowed for different targets and the scenario when only one combined classifier is allowed. Thus, the AS-biGRU-CNN model has stronger expressive power, and higher generalising capability, to extract target-specific knowledge from annotated datasets, to perform target-specific Stance Detection on tweets. Importantly, unlike previous works on target-specific detection in tweets, the models employed in this work do not rely on any extra annotation, domain corpus or feature engineering, and can be easily generalised to other targets of interest.

**Acknowledgments.** This work was supported by The Alan Turing Institute under the EPSRC grant EP/N510129/1.

## References

1. Augenstein, I., Rocktäschel, T., Vlachos, A., Bontcheva, K.: Stance detection with bidirectional conditional encoding. In: Proceedings of 2016 Conference on Empirical Methods in Natural Language Processing, pp. 876–885. ACL (2016)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. In: Proceedings of 3rd International Conference on Learning Representations (2015)
3. Cho, K., van Merrienboer, B., Gülçehre, Ç., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: Proceedings of 2014 Conference on Empirical Methods in Natural Language Processing, pp. 1724–1734. ACL (2014)

4. Ebrahimi, J., Dou, D., Lowd, D.: A joint sentiment-target-stance model for stance classification in tweets. In: Proceedings of 26th International Conference on Computational Linguistics, pp. 2656–2665. ACL (2016)
5. Gal, Y., Ghahramani, Z.: A theoretically grounded application of dropout in recurrent neural networks. In: Proceedings of Advances in Neural Information Processing Systems, vol. 29, pp. 1019–1027 (2016)
6. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of 2014 Conference on Empirical Methods in Natural Language Processing, pp. 1746–1751. ACL (2014)
7. Kingma, D., Ba, J.: Adam: a method for stochastic optimization. In: Proceedings of 3rd International Conference on Learning Representations: Poster Session (2015)
8. Mohammad, S.M., Kiritchenko, S., Sobhani, P., Zhu, X., Cherry, C.: SemEval-2016 task 6: detecting stance in tweets. In: Proceedings of 10th International Workshop on Semantic Evaluation (2016)
9. Mohammad, S.M., Sobhani, P., Kiritchenko, S.: Stance and sentiment in tweets. ACM Trans. Internet Technol. **17**(3), 26 (2017)
10. Pennington, J., Socher, R., Manning, C.D.: Glove: global vectors for word representation. In: Proceedings of 2014 Conference on Empirical Methods in Natural Language Processing, vol. 14, pp. 1532–1543. ACL (2014)
11. Ruder, S., Ghaffari, P., Breslin, J.G.: A hierarchical model of reviews for aspect-based sentiment analysis. In: Proceedings of 2016 Conference on Empirical Methods in Natural Language Processing, pp. 999–1005. ACL (2016)
12. Schouten, K., Baas, F., Bus, O., Osinga, A., van de Ven, N., van Loenhout, S., Vrolijk, L., Frasincar, F.: Aspect-based sentiment analysis using lexico-semantic patterns. In: Cellary, W., Mokbel, M.F., Wang, J., Wang, H., Zhou, R., Zhang, Y. (eds.) WISE 2016. LNCS, vol. 10042, pp. 35–42. Springer, Cham (2016). doi:10.1007/978-3-319-48743-4_3
13. Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
14. Tan, M., Xiang, B., Zhou, B.: LSTM-based deep learning models for non-factoid answer selection. In: Proceedings of 4th International Conference on Learning Representations: Workshop Track (2016)
15. Tang, D., Qin, B., Feng, X., Liu, T.: Effective LSTMs for target-dependent sentiment classification. In: Proceedings of 26th International Conference on Computational Linguistics, pp. 3298–3307. ACL (2016)
16. Wei, W., Zhang, X., Liu, X., Chen, W., Wang, T.: pkudblab at SemEval-2016 task 6: a specific convolutional neural network system for effective stance detection. In: Proceedings of 10th International Workshop on Semantic Evaluation (2016)
17. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A.C., Salakhutdinov, R., Zemel, R.S., Bengio, Y.: Show, attend and tell: neural image caption generation with visual attention. In: Proceedings of 32nd International Conference on Machine Learning, pp. 2048–2057. ACM (2015)
18. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A.J., Hovy, E.H.: Hierarchical attention networks for document classification. In: Proceedings of 2016 Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies, pp. 1480–1489. ACL (2016)
19. Zarrella, G., Marsh, A.: MITRE at SemEval-2016 task 6: transfer learning for stance detection. In: Proceedings of 10th International Workshop on Semantic Evaluation (2016)