Information Processing and Management xxx (xxxx) xxx–xxx

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Neural opinion dynamics model for the prediction of user-level stance dynamics

Lixing Zhu[a], Yulan He[b], Deyu Zhou[*,a]

[a] School of Computer Science and Engineering, Key Laboratory of Computer Network and Information Integration, Ministry of Education, Southeast University, China
[b] Department of Computer Science, University of Warwick, UK

ARTICLE INFO

Keywords:
Natural language processing
Opinion mining
Social networks
Dynamic modelling

ABSTRACT

Social media platforms allow users to express their opinions towards various topics online. Oftentimes, users' opinions are not static, but might be changed over time due to the influences from their neighbors in social networks or updated based on arguments encountered that undermine their beliefs. In this paper, we propose to use a Recurrent Neural Network (RNN) to model each user's posting behaviors on Twitter and incorporate their neighbors' topic-associated context as attention signals using an attention mechanism for user-level stance prediction. Moreover, our proposed model operates in an online setting in that its parameters are continuously updated with the Twitter stream data and can be used to predict user's topic-dependent stance. Detailed evaluation on two Twitter datasets, related to Brexit and US General Election, justifies the superior performance of our neural opinion dynamics model over both static and dynamic alternatives for user-level stance prediction.

1. Introduction

The proliferation of social media platforms such as Twitter enables users to express themselves in various ways. A large proportion of users manifest themselves in following, commenting on or forwarding others' texts (Wang, Zhao, He, & Li, 2014). This poses several new challenges to the field of opinion mining. Firstly, users may not explicitly express their opinions in text. Instead, they could show endorsements of their support of others' opinions through social interactions such as *retweet* on Twitter or clicking on the "*Like*" button on Facebook (Thonet, Cabanac, Boughanem, & Pinel-Sauvagnat, 2017). Secondly, users are not necessarily influenced by all the surrounding contexts (Qiu, Sim, Smith, & Jiang, 2015). For example, users tend to ignore those tweets containing topics falling outside of their interests. It is difficult to identify which post a user pays more attention to. Thirdly, users might update their opinions over time. An individual's opinion is an outcome of a combination of their predisposition and the influence from their neighbors over a short period of time (Cha, Haddadi, Benevenuto, & Gummadi, 2010).

To tackle the aforementioned challenges, existing literatures have utilized user relations (Thonet et al., 2017), contextual information about topics (Ren, Zhang, Zhang, & Ji, 2016; Xing et al., 2017) and temporally-ordered documents (Chen, Wang, & Li, 2018; De, Valera, Ganguly, Bhattacharya, & Rodriguez, 2016; He, Lin, Gao, & Wong, 2013) for opinion/stance detection. However, none of them integrated the three factors of social relations, context and temporality of documents into a unified framework. For example, the SNVDM in (Thonet et al., 2017) improved the performance of VODUM (Thonet, Cabanac, Boughanem, & Pinel-Sauvagnat, 2016) with relationships brought in, whilst they did not consider the temporal order of tweets. (Ren et al., 2016)

* Corresponding author.
E-mail addresses: zhulixing@seu.edu.cn (L. Zhu), Yulan.He@warwick.ac.uk (Y. He), d.zhou@seu.edu.cn (D. Zhou).

https://doi.org/10.1016/j.ipm.2019.03.010
Received 15 October 2018; Received in revised form 23 March 2019; Accepted 26 March 2019
0306-4573/

introduced a topic-based context to aggregate tweets for more accurate prediction and achieved the state of the art, but they ignored to leverage relationships. The dJST proposed in (He et al., 2013) extended JST (Lin & He, 2009) to dynamically detect topic and stance shifts over time on product reviews without considering user relations. Chen et al. (2018) used an attention mechanism (Bahdanau, Cho, & Bengio, 2014) to weigh the importance of a user's present tweet, previously published tweets and their neighbors' tweets and successfully predicted the tweet-level stance. However, they ignored the fact that users' tweets are topic-dependent, hence simply aggregating one's previous tweets and neighbors' tweets for stance prediction may not give good results.

In contrast, we assume that a user's opinion[1] is topic-dependent and is formed based on one's *a priori* perspective and social influence from their friends. Also, users' opinions are not static and could change over time. While previous work mostly focused on tweet-level stance prediction, we instead work on user-level stance prediction. Our rationale is that an individual tweet may be ambiguous in expressing opinions (e.g., sarcastic tweets), however, user-level stance prediction by considering tweets at the global level would have a smoothing effect which will potentially generate more accurate results. Moreover, user-level stance prediction has an advantage that it is able to predict a user's stance towards a specific topic even though the user did not publish any tweets about this topic in the current time window.

We thus propose a Neural Opinion Dynamics (NOD) model built on Recurrent Neural Networks (RNNs) to capture the three key factors for the prediction of user-level opinion dynamics: the user's past opinions, the user's neighborhood opinions and contextual information about topics. In our proposed framework, we split temporally-ordered tweets into sequential epochs and capture the representations of both users' and their neighbors' topical contextual information using an attention mechanism. Users' sequential posting behavior is simulated by a Gated Recurrent Unit (GRU) (Cho et al., 2014) network. We train the model through online learning that data stream is split into epochs temporally and the model is updated at each epoch sequentially to allow for the prediction of user-level topic-related stances in the following epoch. Our source code is made available at https://github.com/somethingx01/TopicalAttentionBrexit.

The contributions of this paper are summarized as follows.

- We propose a novel Neural Opinion Dynamics (NOD) model that jointly models users' posts, neighborhood and topical contextual information using an attention mechanism.
- Unlike previous approaches, a user's sequential posting behavior is simulated by a GRU, which fully exploits the temporal contextual information for online learning.
- Experimental results on two Twitter datasets show that the proposed approach outperforms the state-of-the-art approaches and is effective in tracking the user-level opinion dynamics.

The rest of the paper is organized as follows. We first review in Section 2 the three lines of research on incorporating social relations for stance detection, neural networks for stance prediction and stance/opinion dynamics detection using dynamic approaches. We then present our proposed neural network architecture in Section 3 followed by experimental results in comparison with baselines on two datasets relating to Brexit and US General Election in Section 4. Finally, we conclude the paper and outline directions for future research.

2. Related work

Our work is related to three lines of research: incorporating social relations and context for sentiment/stance detection, neural networks for stance detection and stance/opinion dynamics tracking.

2.1. Incorporating social relations and context for sentiment/stance detection

Early attempt of using social relations for user-level sentiment detection was made by Tan et al. (2011) who proposed a probabilistic graphical model in which the observed sentiment labels of users were propagated to users without sentiment labels along the edges in their social network. Hu, Tang, Tang, and Liu (2013) observed that connected individuals are more likely to hold similar opinion and incorporated this as a constraint into a probabilistic optimization model for sentiment classification on microblogs. More recently, Thonet et al. (2017) took the re-tweet and re-tweeted status as the observed variables, which were in turn sampled from viewpoint-specific distributions. They ran the proposed model on two political corpora for topic-specific stance classification.

Apart from social networks, contexts such as topics and opinion targets have also been proven useful in stance classification (Ren et al., 2016; Tang, Zhang, He, Lin, & Zhou, 2019; Vo & Zhang, 2015; Xing et al., 2017). The integration of context is typically achieved by the attention mechanism, which was originally proposed in (Mnih, Heess, & Graves, 2014), which can be viewed as a selector for the most relevant part of the input. For example, in (Ren et al., 2016), hashtags were viewed as topics and attention signals were allocated to neighborhood tweets which share similar hashtags. Neighborhood tweets with higher attention signals contribute more towards the sentiment of the tweet of interest. Ma, Peng, and Cambria (2018) considered stance as aspect-associated. For each sentence they computed the aspect representation from aspect words and used it together with the sentence representation to generate an aspect-stance vector.

[1] In our work here, we aim to detect a user's stance (suppport, be neutral or go against) towards a certain topic on Twitter and use *opinion* and *stance* interchangeably throughout the paper.

2.2. Neural networks for stance detection

In recent years, neural networks such as Long-Short Term Memory (LSTM) networks (Hochreiter & Schmidhuber, 1997) and Convolutional Neural Networks (CNNs) have been used widely for stance classification (Du, Gui, He, Xu, & Wang, 2019; Wang, Huang, & Zhao, 2016). In SemEval2016 Task 6 (Detecting Stance in Tweets), the top ranked system (Zarrella & Marsh, 2016) was built based on a transfer learning framework with two RNNs. A similar network architecture was proposed in (Baziotis, Pelekis, & Doulkeridis, 2017) where an attention layer was added on top of the BiLSTM to generalize the hidden states. Chen, Sun, Tu, Lin, and Liu (2016) introduced a sentence-level attention to an LSTM-based classifier and demonstrated its effectiveness in user-level stance classification. Yang, Deyu, and He (2018) incorporated topics for relevant emotion ranking. Some other works considered transferring external sentiment knowledge via sentimental word embeddings (Bandhakavi, Wiratunga, Massie, & Padmanabhan, 2017; Li et al., 2017). More recently, Majumder et al. (2018) employed a memory neural network to model the inter-aspect dependency, which showed effectiveness on classifying multi-aspect sentences.

2.3. Opinion/stance dynamics tracking

Early work to opinion/stance dynamics tracking utilized variants of topic models such as Latent Dirichlet Allocation (LDA) (Blei, Ng, & Jordan, 2003) or stochastic processes. For example, He et al. (2013) proposed the dynamic Joint Sentiment-Topic (dJST) model built upon LDA by adding a sentiment layer between the topic layer and words and assuming that the Dirichlet prior of the sentiment-topic-word distribution in the current epoch is influenced by the statistics collected in the previous n epochs. Their approach can track the evolution of both topics and topic-associated stances from reviews. Sasaki, Yoshikawa, and Furuhashi (2014) combined Twitter-LDA with a variant of Dynamic-LDA to detect the topic trend over time. Dermouche, Velcin, Khouas, and Loudcher (2014) also modified LDA by additionally adding documents' timestamps as observed variables. Volkova and Van Durme (2015) explored an active learning setup for user-level stance classification. De et al. (2016) forecast user-level opinion dynamics via stochastic processes based on the intuitive idea that opinions are hidden variables conditioned on neighbors' shared opinions. Another stochastic influential model was proposed in (Jia, Friedkin, & Bullo, 2017). More recently, Chen, Wang, Lei, and Li (2016) and Chen et al. (2018) leveraged temporally-ordered tweets by introducing an attention layer to weigh the importance of a user's previous tweets, their current tweet and their neighbors' tweets for tweet-level stance prediction. They also used an LSTM layer to capture historical influence from past epochs. Tweets in their model were represented by the bag-of-opinion-words where opinion words were detected using a sentiment lexicon.

Our proposed model is different from the aforementioned models in the following aspects: (1) our model considers topical information and generates the representations of tweets using an attention mechanism; (2) our model incorporates contextual information from neighbors to capture the social influence; (3) as opposing to (Chen, Wang et al., 2016; Chen et al., 2018) which apply an attention layer to distinguish between a user's previous tweets and current tweets, we model the user's tweets as a posting sequence which naturally captures the user's posting behavior. As will be shown in the experiments section, our model outperforms the existing approaches in stance dynamics tracking.

3. Neural Opinion Dynamics (NOD) model

In this section, we present the Neural Opinion Dynamics (NOD) model which leverages content and context information for the tracking of user-level stance dynamics over time. We assume that tweets arrived in a temporal order and can be split into epochs. Each epoch stretches for a fixed window size which can be either time-based (e.g., a day) or count-based (e.g., 10k tweets). In each epoch e, a user u posted a sequence of tweets $\{d_1^{e,u}, d_2^{e,u}, \cdots, d_T^{e,u}\}$. Each tweet $d_t^{e,u}$ has a tweet representation $c_{t,0}^{e,u}$ derived from its content and the associated topic embedding $z_{t,0}^{e,u}$. We will discuss in Section 3.3 how to generate topic embeddings for tweets. In addition, we assume that when user u posts a tweet $d_t^{e,u}$ at epoch e, their opinion is also influenced by the most recent N tweets posted by their neighbors in their social network. Therefore, the context information of the tweet $d_t^{e,u}$ is captured by their neighbors' tweets in the form of a tweet representation sequence $\{c_{t,1}^{e,u}, c_{t,2}^{e,u}, \cdots, c_{t,N}^{e,u}\}$ along with the corresponding topic embeddings $\{z_{t,1}^{e,u}, z_{t,2}^{e,u}, \cdots, z_{t,N}^{e,u}\}$.

In our problem setup here, the parameters of NOD are updated online, i.e., NOD is updated using the data in the current epoch and is used for predicting a user's topic-related stance in the subsequent epoch. By doing so, NOD can be used for tracking the user-level stance dynamics. The problem setup is depicted in Fig. 1. In what follows, we first describe the overall architecture of NOD model and then present each of its components in details.

3.1. Overall architecture

The overall architecture of the NOD model is shown in Fig. 2. Here, the input is a user's timeline of tweets or posts. An LSTM layer is first used to generate the representation for each tweet. The tweet representation and the corresponding topic embedding is concatenated and subsequently combined with the context information captured by the neighbors' tweets using an attention mechanism to generate the integrated representation for tweet $d_t^{e,u}$ posted by user u at time t in epoch e. Each user u has a sequence of posts $\{d_1^{e,u}, d_2^{e,u}, \cdots, d_T^{e,u}\}$ in epoch e, which are passed to a GRU layer to obtain the user representation. Finally the user representation is used as features to predict the topic-stance distributions using a Softmax layer. To model the influence of the training data from previous epochs, the last state of the GRU in epoch $e-1$ is used to initialize the GRU for the current epoch e.

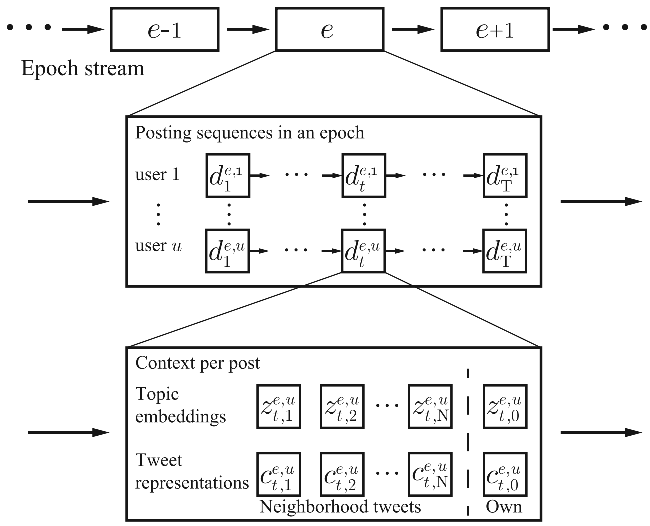

Fig. 1. The problem setup for opinion dynamics prediction.

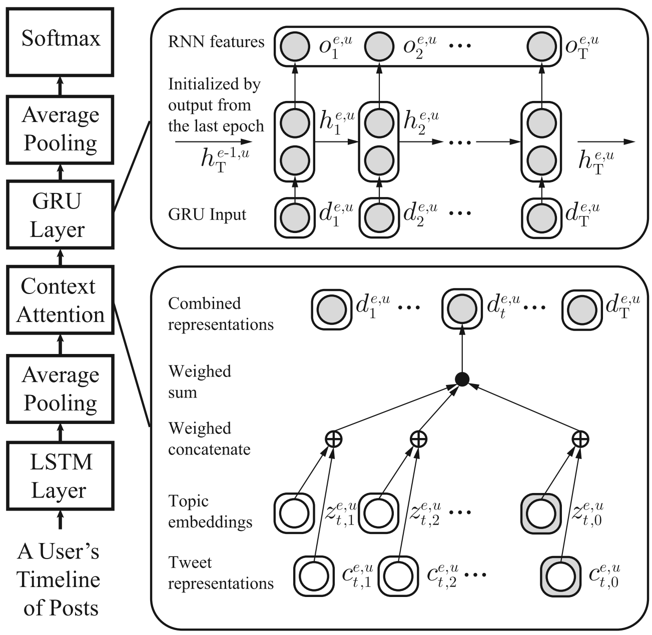

Fig. 2. The overall architecture of the proposed Neural Opinion Dynamics (NOD) model.

3.2. *Tweet representation*

We generate the representation of a tweet using an extensively exploited LSTM (Baziotis et al., 2017; Sutskever, Vinyals, & Le, 2014; Tang, Qin, & Liu, 2015; Wang et al., 2016), which corresponds to the *LSTM Layer* in Fig. 2. LSTM is more effective dealing with language modelling compared with other architectures (Jozefowicz, Zaremba, & Sutskever, 2015). We use the 300-dimensional word embeddings pre-trained on 20 million tweets (Barbieri, Kruszewski, Ronzano, & Saggion, 2016).[2]

Suppose the lth word $w_{t,nl}^{e,u}$ is mapped into its embedding $\mathbf{w}_{t,nl}^{e,u} \in \mathbb{R}^{300}$, given the previous cell state $C_{t,nl-1}^{e,u}$, hidden state $h_{t,nl-1}^{e,u}$ and the current input word $\mathbf{w}_{t,nl}^{e,u}$, the transition function of the LSTM is calculated as follows:

$$f_{t,nl}^{e,u} = \sigma\left(W_f \cdot \left[h_{t,nl-1}^{e,u}, \mathbf{w}_{t,nl}^{e,u}\right] + b_f\right) \tag{1}$$

[2] https://drive.google.com/drive/folders/0B13VF_-CUsHPd3FqdVJ2c1ZJaXc.

$$i_{t,nl}^{e,u} = \sigma\left(W_i\cdot\left[h_{t,nl-1}^{e,u}, \mathbf{w}_{t,nl}^{e,u}\right] + b_i\right) \tag{2}$$

$$o_{t,nl}^{e,u} = \sigma\left(W_o\cdot\left[h_{t,nl-1}^{e,u}, \mathbf{w}_{t,nl}^{e,u}\right] + b_o\right) \tag{3}$$

$$\tilde{C}_{t,nl}^{e,u} = \tanh\left(W_C\cdot\left[h_{t,nl-1}^{e,u}, \mathbf{w}_{t,nl}^{e,u}\right] + b_C\right) \tag{4}$$

$$C_{t,nl}^{e,u} = f_{t,nl}^{e,u} \odot C_{t,nl-1}^{e,u} + i_{t,nl}^{e,u} \odot \tilde{C}_{t,nl}^{e,u} \tag{5}$$

$$h_{t,nl}^{e,u} = o_{t,nl}^{e,u} \odot \tanh\left(C_{t,nl}^{e,u}\right) \tag{6}$$

where $\cdot$ stands for the matrix multiplication and $\odot$ is an element-wise multiplication operator. W and b are the parameters to be trained. The LSTM layer outputs a sequence of hidden states $\{h_{t,n_1}^{e,u}, h_{t,n_2}^{e,u}, \cdots, h_{t,n_L}^{e,u}\}$, which are conveyed to an average pooling layer to obtain a tweet representation $c_{t,n}^{e,u}$.

3.3. Topic embedding generation

The *topic embedding* in Fig. 2 captures the topic information of each tweet. The generation of the topic embedding is separated from the learning process. We use the Hierarchical LDA (HLDA)[3] for hierarchical topic detection.[4] In HLDA, each document is allocated with a set of topics which is essentially a path from the root to a leaf in a topic tree, where higher level nodes are more general topics. The topic tree is generated using a nested Chinese Restaurant Process (nCRP) (Blei, Griffiths, & Jordan, 2010). Each document is generated by drawing words from its assigned topics. For each tweet, we chose its level-2 topic in order to achieve balanced topic granularity. In HLDA, each topic is represented as a distribution over a fixed vocabulary. We sort the vocabulary by word frequency, keeping the normalized first 100 dimensions as the representation for each topic. Finally, each tweet is associated with a topic represented as a vector, which is denoted as z in Fig. 2.

3.4. Context attention

We assume that when a user posts a tweet, they would pay more attention to their neighbors' tweets carrying topics of their interests. Context-based attention mechanism has been demonstrated effective in a variety of tasks including topic aware response generation (Xing et al., 2017), sentiment analysis (Ren et al., 2016) and geolocation prediction (Miura, Taniguchi, Taniguchi, & Ohkuma, 2017). In the context attention layer of NOD shown in Fig. 2, the input consists of a fixed number of neighborhood tweets $\{c_{t,1}^{e,u}, c_{t,2}^{e,u}, \cdots, c_{t,\mathrm{N}}^{e,u}\}$, their respective topic embeddings $\{z_{t,1}^{e,u}, z_{t,2}^{e,u}, \cdots, z_{t,\mathrm{N}}^{e,u}\}$, the user's current tweet $c_{t,0}^{e,u}$ and topic $z_{t,0}^{e,u}$. First, for each tweet, its final representation is generated by combining the tweet representation with its corresponding topic embedding by:

$$g_{t,n}^{e,u} = \frac{\alpha_1 z_{t,n}^{e,u} \oplus \alpha_2 c_{t,n}^{e,u}}{\alpha_1 + \alpha_2} \tag{7}$$

where α_i captures the importance between topic and content. Note that if $\alpha_1 \ll \alpha_2$ then the model degenerates to the one entirely relying on the textual information. The combined representations are then passed to a user attention layer whose output $d_t^{e,u}$ is a normalized weighed sum of $\{g_{t,1}^{e,u}, g_{t,2}^{e,u}, \cdots, g_{t,\mathrm{N}}^{e,u}, g_{t,0}^{e,u}\}$:

$$d_t^{e,u} = \sum_{n=0}^{\mathrm{N}} \beta_n g_{t,n}^{e,u} \tag{8}$$

$$\beta_n = \frac{\exp(\mathbf{v}^{\mathrm{T}} u_{t,n}^{e,u})}{\sum_{n=0}^{\mathrm{N}} \exp(\mathbf{v}^{\mathrm{T}} u_{t,n}^{e,u})} \tag{9}$$

$$u_{t,n}^{e,u} = \tanh(\mathbf{W} g_{t,n}^{e,u} + \mathbf{b}) \tag{10}$$

where β_n, $(n > 0)$, represents the user's attention towards nth neighborhood tweet. In other words, β_n essentially measures the degree of influence from the nth neighborhood tweet and β_0 is the attention signal on the user's current tweet. $\mathbf{v}$ is the weight vector, $u_{t,n}^{e,u}$ functions as a smoothing factor calculated from $g_{t,n}^{e,u}$ by a fully connected layer.

3.5. User-level stance prediction on streaming data

It was previously demonstrated that GRU outperforms LSTM in all tasks except for language modelling (Jozefowicz et al., 2015). As such, we feed the integrated representations of user e's post sequence in epoch e, $\{d_1^{e,u}, d_2^{e,u}, \cdots, d_{\mathrm{T}}^{e,u}\}$, to a GRU layer for user-level stance prediction. Let $h_t^{e,u}$ denote the hidden state, the current state $h_t^{e,u}$ is updated based upon the previous state $h_{t-1}^{e,u}$ as follows:

[3] http://www.cs.columbia.edu/~blei/downloads/hlda-c.tgz.

[4] We did not use LDA here as it requires pre-setting the topic number while HLDA can automatically infer the topic number from data.

$$s_t^{e,u} = \sigma(W_s \cdot [h_{t-1}^{e,u}, d_t^{e,u}] + b_s) \tag{11}$$

$$r_t^{e,u} = \sigma(W_r \cdot [h_{t-1}^{e,u}, d_t^{e,u}] + b_r) \tag{12}$$

$$\tilde{h}_t^{e,u} = \tanh(W_h \cdot [r_t^{e,u} \odot h_{t-1}^{e,u}, d_t^{e,u}] + b_h) \tag{13}$$

$$h_t^{e,u} = (1 - s_t^{e,u}) \odot h_{t-1}^{e,u} + s_t^{e,u} \odot \tilde{h}_t^{e,u} \tag{14}$$

where $s_t^{e,u}$ is the update gate, $r_t^{e,u}$ is the reset gate, W_s, W_r, W_h are the weight matrices and b_s, b_r, b_h are the biases. The hidden states are then passed to an average pooling layer to obtain a user representation $h^{e,u}$, which is transformed with the following equation:

$$o^{e,u} = W_{\text{out}} \cdot h^{e,u} \tag{15}$$

where W_{out} maps the user representation $h^{e,u}$ into a dimension of C categories. The output is normalized by a Softmax layer:

$$y_c^{e,u} = \frac{\exp(o_c^{e,u})}{\sum_{i=1}^{\mathrm{C}} \exp(o_i^{e,u})} \tag{16}$$

which is a distribution over C categories. The loss function is the KL-divergence (Kullback, 1997) loss described as:

$$\mathcal{L} = \sum_{u=1}^{\mathrm{U}} \sum_{i=1}^{\mathrm{C}} \mathbf{KL}(y_i^{e,u} \| \mathrm{g}_i^{e,u}) \tag{17}$$

where $\mathrm{g}^{e,u}$ is the ground truth distribution over categories. Assuming there are K topics and each topic has S stance labels, then there are essentially $\mathrm{C} = \mathrm{K} \times \mathrm{S}$ categories. We can map the C-length vector $y^{e,u}$ into a $K \times S$ matrix to obtain the topic-stance distribution.

To model the impact from previous epochs, we initialize the state of GRU for user u in epoch e with the final state of GRU for user u in epoch $e - 1$, that is, $h_0^{e,u} = h_{\mathrm{T}}^{e-1,u}$. The process is repeated for each epoch and this results in a long RNN transition chain.

3.6. A variant of the proposed model

The NOD model described so far outputs topic-associated stance labels for each user. We also perform coarse-level stance classification in which the output is a three-class stance label (‘*oppose*’, ‘*neutral*’ or ‘*support*’) with topics ignored. This is achieved by modifying the output layer and the loss function of NOD. Concretely, the output $o^{e,u}$ is a 3-dimensional vector and the loss function is changed to cross-entropy loss defined as:

$$\mathcal{L} = -\sum_{u=1}^{\mathrm{U}} \log\left(\frac{\exp(o_{\mathrm{g}^{e,u}}^{e,u})}{\sum_{i=1}^{3} \exp(o_i^{e,u})}\right) \tag{18}$$

where $\mathrm{g}^{e,u}$ is the ground truth three-class stance label. We call such a variant **NOD_Coarse**.

4. Experiments

4.1. Experimental setup

Data. We constructed two datasets by crawling tweets related to Brexit and US General Election 2016 using the Twitter Streaming API with relevant hashtags. For Brexit, tweets were crawled between 2nd and 21st of June 2016 using hashtags *#EURef, #EU, #Referendum, #Brexit, #VoteRemain* and *#VoteLeave*. For US General Election, tweets were crawled between November 6th and 7th 2016 using keywords *Trump, Clinton* and *Hillary*. Only English tweets were kept and duplicate tweets including re-tweets were removed. Users' social networks (*following* and *follower* relations) were also collected. The BREXIT dataset was split into epochs every other day, which resulted in a total of 9 epochs with each epoch consisting of 40,440 tweets on average. The ELECTION dataset contains a total of 452,128 tweets and were split into 11 count-based epochs with the epoch size set to 40,000 and on average 16,019 users per epoch. The statistics of the two datasets are summarized in Table 1. The distribution of opposed, neutral and supportive users per epoch is shown in Fig. 3. It can be observed that the ELECTION dataset has more polarized users compared to the BREXIT dataset.

Table 1
Statistics of the two Twitter datasets. ‘user/epoch’ is the average number of users per epoch and ‘tweet/user/epoch’ is the averaged number of tweets per user in an epoch.

	BREXIT	ELECTION
#user	38,335	108,689
#tweet	363,961	452,128
#epoch	9	11
user/epoch	10,802	16,019
tweet/epoch	40,440	40,000
tweet/user/epoch	3.7	2.5

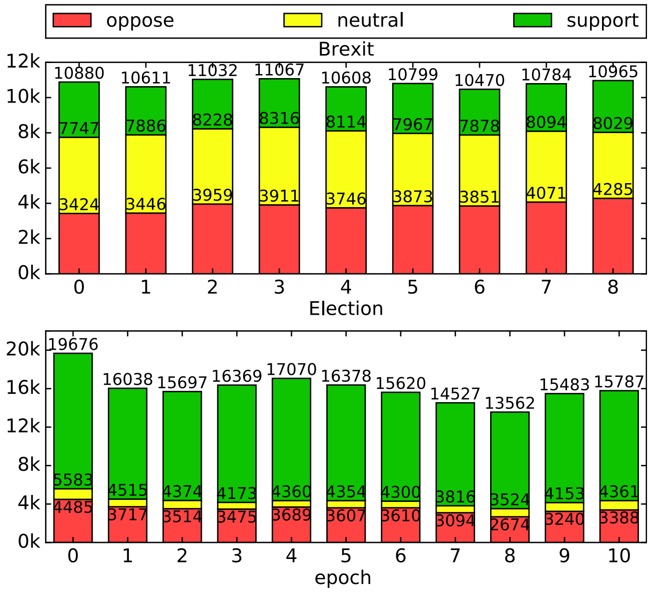

Fig. 3. The number of opposed, neutral and supportive users versus epochs.

Table 2
Top ten words in the discovered topics on two datasets.

	BREXIT
Sovereignty	leave EU vote UK stay Europe country control borders independence
Economy	EU UK economy jobs brexit trade free NHS money tax
Boris&Farage	Boris gove voteleave Johnson Farage brexit Michael Cameron David Geldof
Immigration	brexit UK EU EURef leaveEU voteleave England migrants refugees muslim
Campaign	brexit campaign remain racist political MP JoCox murder working class
BBCdebate	debate remain BBCdebate voteremain watching voteleave Boris ITVEURef argument tonight
Polls	referendum EU brexit EURef UK remain news debate poll polls
Vote	EURef vote referendum Thursday week today debate voting days June
	ELECTION
Vote	voting people win president candidate supporters supporter voted vote America
Email scandal	Hillary rigged emails FBI Comey director Trump talking guy things
Jobs	election polls vote state signs jobs tax plan steel China
Email scandal	Hillary Clinton emails FBI wikileaks email foundation state money server
Slogans	Trump Donald MAGA Clinton president vote election final Hillary IMWITHHER
Campaign	rally Donald campaign Trump Nugent Ted Clinton sign Reno protester
Election	Donald USA president world Clinton vote united win states campaign

Pre-processing. We pre-process tweets by tokenization, lowercasing, removing URLs, user handles (@user) and email addresses. Recall that in our model each user is a training instance consisting of a sequence of posts. In our experiments, we set each user's post sequence to a fixed length by either padding null tweets for users with few tweets or trimming extra tweets for users with too many tweets. The same rule was also applied when dealing with a user's neighborhood context. In our datasets, we observe that less than 11% of the users have posted more than 3 tweets in any epoch. As such, we set the length of a user's posting sequence in any epoch to 3. We also observe that users on average possess a neighborhood context of 5 tweets in each epoch. Therefore, we set the neighborhood context size to 5 in our experiments. The maximum length of each tweet was set to 24 since over 88% tweets have less than 24 words.

Topic Setting. As mentioned before, we used HLDA to extract topics from tweets. As there might be potentially a large variety of topics discussed in tweets, we only kept the most prominent topics and merged topics with fewer than 10,000 associated tweets as '*Others*'. The prominent topics are illustrated in Table 2, where we removed common stop words[5] and manually assigned a label to each topic for easy understanding. We further introduced a '*Null*' topic for those null tweets added to the training data. We ended up with 10 topics for the BREXIT data and 9 topics for the ELECTION data.

Ground Truth Stance Labels. As it is very difficult to manually annotate over 800k tweets in our datasets, we resorted to training a stance classifier using distant supervision. In particular, for BREXIT, we collected over 4 million tweets[6] between May 16th and June

[5] https://github.com/somethingx01/TopicalAttentionBrexit/blob/master/postCommonStopWords.txt.
[6] Only the original tweets were accounted for the user-level ground truth acquisition. Retweets and duplicate tweets were removed.

2nd 2016 with hashtags clearly indicating stances. Concretely, we assign stance label '*support*' to tweets if they exclusively carry hashtags '*#voteleave, #euistheproblem, #leaveeu, #betteroffout, #takecontrol* or *#votebrexit*', but do not contain '*#voteremain, #strongertogether, #stayintheeu, #strongerin, #bettertogether* or *#remainineu*'. We annotate opposite tweets vice versa. Tweets without stance-indicative hashtags and do not contain any polarity words are assumed to be '*neutral*'. This practice was inspired by the observation that most users use their own hashtags to manifest their political preferences, when we were manually annotating 1000 validation tweets in the initial pass. For example, both tweets '*#TakeControl on 23rd in #EUref*' and '*Great Britain is happy to trade with its neighbours, it does not need their laws imposed upon #VoteLeave*' show a '*support*' stance, while tweet '*Gove had enough of experts. Would rather listen to experts as won't make as many errors as gobby knowalls. #StrongerIn*' exhibits an '*oppose*' stance. Similarly, for ELECTION, we collected over 17 million tweets in the first week of November 2016 with stance-indicative hashtags such as '*#votetrump, #trumpwillwin, #neverhillary, #crookedhillary, #draintheswamp* and *#notwithher*' for '*support*' and '*#votehillary, #nevertrump, #trumptapes, #imwithher, #voteblue* and *#trumpdown*' for '*oppose*', and sampled neutral tweets in a similar way. We then trained an off-the-shelf stance classifier DataStories[7] (Baziotis et al., 2017), which gave the best result in the SemEval-2017 Task 4 on "Sentiment Analysis in Twitter", using these external corpora. To assess the accuracy of such externally-trained stance classifier, we manually annotate 1000 tweets, and report the results in Table 3. Since some studies rely on sentiment lexicons e.g., emoticons, for the acquisition of ground truth labels (Chen et al., 2018; Lim & Buntine, 2014; Marchetti-Bowick & Chambers, 2012), we also compare the results with those obtained using Sentiment140Lex (Mohammad, Kiritchenko, & Zhu, 2013), a lexicon specially designed for Twitter sentiment analysis, and Vader (Gilbert, 2014), a lexicon-based sentiment classifier. It can be observed that DataStories significantly outperforms Sentiment140Lex and Vader by a large margin. This is not surprising since sentiment lexicons are primarily used for sentiment classification, which is, however, quite different from stance prediction since tweets containing positive words could express an '*opposing*' stance. To further obtain the user-level topic-stance distributions, for each user we calculate the number of tweets under each topic with different stance labels and normalize the counts to obtain the topic-stance distributions.

Baselines. Recall that in Section 1 we elaborated our motivation by analyzing six related models. Here, we choose SNVDM as a static baseline which incorporates user relationships. CbNNM is chosen considering the utilization of context attention. As the dynamic baselines we employ dJST, SLANT and CSIM_W. We further add DataStories since it has been used for distant supervision. For the methods which can only output tweet-level stance labels, we aggregate the stance classification results by users to obtain the user-level stance labels.

dJST (He et al., 2013) is a weakly-supervised LDA-based generative model for dynamic sentiment-topic detection. The model used the sentiment-topic-word statistics gathered in the previous n epochs to modify the Dirichlet prior for the sentiment-topic-word distribution in the current epoch. We incorporate into the model the word prior polarity information obtained from Sentiment140Lex.

SLANT (De et al., 2016) is a supervised probabilistic generative model which models each user's latent opinions over time as a multidimensional stochastic process. Users' extant messages and stance labels are considered as observations for parameter estimation.

CSIM_W (Chen et al., 2018) used an attention layer to weigh the importance of a given user's previously published tweets, their current tweet and their neighbors' tweets and employed an LSTM layer to capture the influence in the previous epochs. They represented each tweet using the bag-of-opinion-words which were identified using Vader (Gilbert, 2014). We also implemented a variant of CSIM_W, called CSIM_W_Rep, in which each tweet is represented by a sequence of word embeddings as in our model.

SNVDM (Thonet et al., 2017) is an unsupervised LDA-based generative model where the sender/receiver information is regarded as observed variables, which is generated by a hidden viewpoint variable. In our experiments, a user is a sender and their followers are receivers.

CbNNM (Ren et al., 2016) considered hashtags as topics. In their model, the contextual information of a tweet (i.e., the neighbors' tweets sharing the same hashtag) serve as features for tweet-level stance classification.

DataStories (Baziotis et al., 2017) is the state-of-art method in tweet-level stance classification.

Among the aforementioned baselines, dJST, SLANT, CSIM_W and CSIM_W_Rep are dynamic models whose parameters are updated with the continuously arriving data. The static models such as SNVDM, CbNNM and DataStories are trained from the data in the current epoch and used to predict the stance labels of tweets in the next epoch. Also, dJST and SNVDM are unsupervised models that do not use any tweet label for training.

4.2. Results

We evaluate the performance of various models in terms of accuracy and micro-F1 for coarse-grained (i.e., three-class) stance classification. For NOD, the output topic-stance distributions are aggregated across topics to obtain the user-level stance labels. For NOD_Coarse, the output is already one of the three stance classes.

4.2.1. Brexit

We report in Fig. 4 the accuracy and micro-F1 of our method against baselines over epochs on BREXIT. It can be observed that supervised models generally outperform unsupervised or weakly-supervised models. In addition, we observe an upward trend for dynamic models owing to their ability in capturing historical context in the previous epochs. Static models such as SNVDM and

[7] https://github.com/cbaziotis/datastories-semeval2017-task4.

Table 3
Accuracy of the ground truth acquisition methods.

	BREXIT	ELECTION
DataStories	0.906	0.895
Sentiment140Lex	0.579	0.562
Vader	0.538	0.481

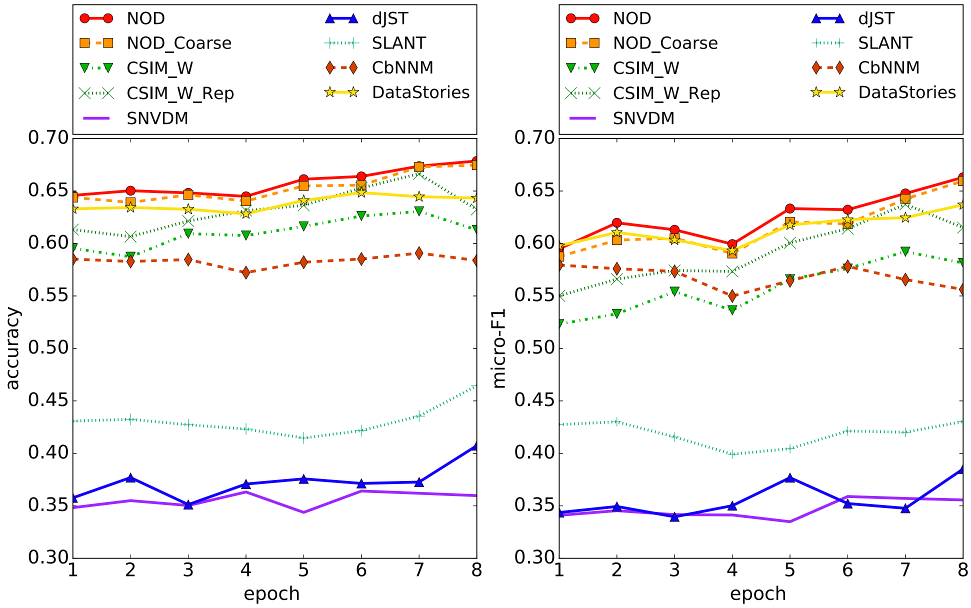

Fig. 4. Accuracy and micro-F1 versus epochs on BREXIT.

CbNNM give more constant results as they only have the access to a fix amount of training data in each epoch. We also observe that CSIM_W was beaten by CSIM_W_Rep which represents each tweet by word embeddings instead of bag-of-opinion-words. DataStories outperforms all the other baselines. Nevertheless, our proposed NOD and NOD_Coarse perform similarly and they both give superior results compared to DataStories in general across all the epochs.

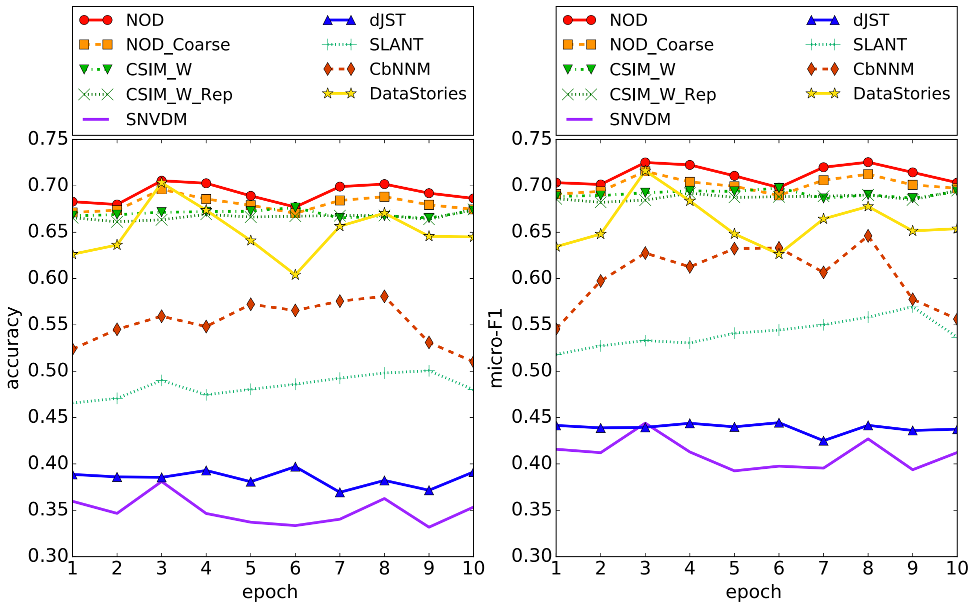

Fig. 5. Accuracy and micro-F1 versus epochs on ELECTION.

Table 4
Wilcoxon T test statistics of the Wilcoxon signed-rank test on the classification accuracy ($T_{acc.}$) and recall ($T_{rec.}$) between NOD and the top 3 baselines, respectively.

Baselines	BREXIT		ELECTION	
	$T_{acc.}$	$T_{rec.}$	$T_{acc.}$	$T_{rec.}$
CSIM_W	0.004	0.004	0.003	0.003
CSIM_W_Rep	0.008	0.008	0.003	0.003
DataStories	0.010	0.016	0.001	0.001

4.2.2. Election

Fig. 5 presents the accuracy and micro-F1 versus epochs on ELECTION. Again, we observe that supervised models are superior to unsupervised or weakly-supervised models. Static models exhibit drastic fluctuations over epochs. It is more noticeable for DataStories that it gives a significant drop at Epoch 6. This may partly due to the mismatch of the training data in the neighboring epochs. SNVDM and CbNNM are relatively steady compared to DataStories since they captured context information to some extent. Conversely, dynamic models give more consistent results across epochs. NOD slightly outperforms NOD_Coarse and they beat all the other baselines across all epochs except for epoch 6, where CSIM_W produced comparable results. Nevertheless, CSIM_W is inferior to NOD overall.

4.2.3. Statistical significance test

We notice in Fig. 4 that the gaps between NOD and two baselines, CSIM_W_rep and DataStories, are small at some points. To further verify that the proposed model significantly outperforms the baseline approaches, we measure the statistical significance between NOD and the 3 top-performing baselines. Some past research suggested using Wilcoxon signed-rank test rather than paired t-test for the prediction over time series data (Mozetič, Torgo, Cerqueira, & Smailović, 2018; Oliveira, Cortez, & Areal, 2016). Following the sequential validation setup in (Mozetič et al., 2018), we posit that models trained by a training set are used to predict the user-level stance in the subsequent epoch, which is in line with our original experimental setup. The null hypothesis is that the differences between NOD and a baseline follow a symmetric distribution around zero. The significance threshold is chosen as 5% in accordance with (Oliveira et al., 2016). We report the statistics in Table 4. It shows that all the test statistics T fall into the rejection region. Therefore, we accept the alternative hypothesis that the proposed model outperforms the baseline approaches, which is at the 1% significance level on ELECTION. When evaluated by recall on BREXIT, NOD exceeds Datastories at the 5% level.

4.2.4. Analysis of errors

We further analysed the wrongly predicted instances and found that these users typically have insufficient neighborhood context for the model to make correct prediction. For example, a user did not post any tweets before and was only linked to the author of the tweet "*BBC debate well said. Brexit, take back control, vote leave*". With such limited neighborhood context information, our model predicted that the user's stance is to '*support*' Brexit. However, this user then posted a tweet "*I will be voting remain. But I am not calling myself a remainder.*" in the next epoch showing the opposite stance. Due to the training setup of our model (trained on a user's previously posted tweets and their neighborhood context and to predict the user's stance in the next epoch without seeing their future posted tweets), there is no way for our model to make correct prediction in such cases.

4.2.5. Impact of contextual attention

We studied the effect of contextual attention by either excluding the topical attention or ignoring neighborhood context. More concretely, we set α_1 to 0 so that the topical attention is not accounted for. We also examined the effect of ignoring neighborhood tweets by tying up $\beta_{1:N}$ to 0. We tested these settings on the last epoch for both datasets and show the results in Table 5. It can be observed that neighborhood attention boosts the performance more significantly while topical attention also helps increase the accuracy. This shows that both topics and neighborhood context are indeed important for user-level stance prediction.

We argue that the neighborhood attention helps boosting the performance because of *homophily* (McPherson, Smith-Lovin, & Cook, 2001). But we also acknowledge that a user's stance could be totally different from most of their friends'. That is why our model simultaneously takes into account both the user's previous tweets and their neighborhood context and learns attention weights automatically in order to achieve an optimal decision for the user-level stance prediction. In our future work, we will explore a more principled way to study the major factors influencing the opinion formation.

Table 5
User-level stance prediction accuracy on the last epoch with or without contextual information.

	BREXIT	ELECTION
Original	**0.6786**	**0.6865**
- Topics	0.6679	0.6794
- Neighbor Context	0.6451	0.6592

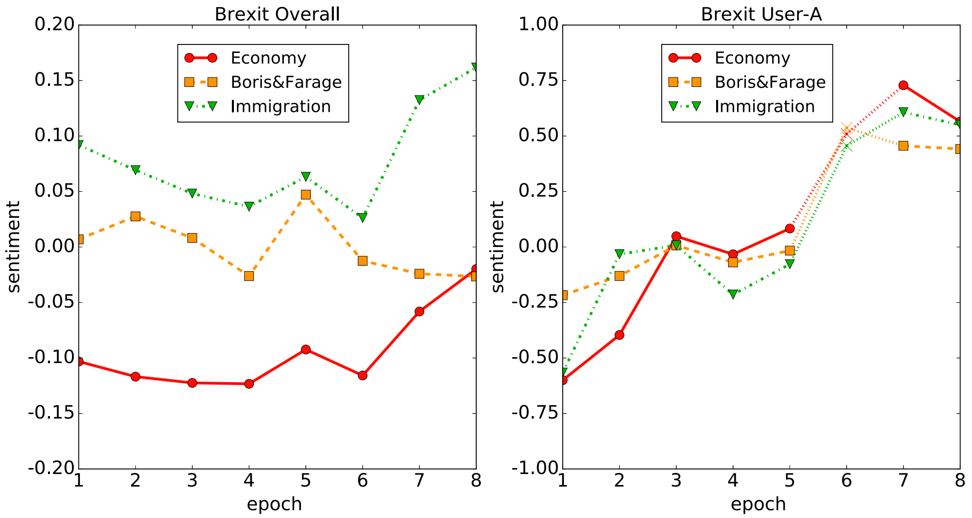

Fig. 6. Global topic-stance and user-level topic-stance on BREXIT.

4.2.6. *Tracking stance dynamics*

To demonstrate that our model is able to track stance dynamically in a sensible way, we carried out detailed studies on three example topics, '*Economy*', '*Boris&Farage*' and '*Immigration*', from BREXIT and displayed their stance transition over epochs. To obtain the overall topic-stance, we first aggregated by all users the stance distributions of a particular topic, normalized the aggregated scores and finally calculated the weighed sum as the overall stance score towards the topic. Here, we assigned weights -1, 0 and 1 to opposed, neutral and supportive stance respectively. For the user-level topic-stance, we calculated the final stance score in a similar manner.

Fig. 6 shows the overall stance and example user stance towards these three topics. We can observe that the overall stance for '*Economy*' is more towards opposing, indicating that people generally felt that Brexit would affect economy negatively. The overall stance for '*Immigration*' is however positive with more people switched to the positive view towards the end of Epoch 8. This shows that more people thought Brexit would be good to control the immigration. As for '*Boris&Farage*', the overall stance slightly fluctuates around neutral. For the user-level topic stance, User-A changed her views more drastically. She started with negatively views on all these three topics, but eventually switched to positive views at Epoch 8.

Another ability of our model is to infer a users' stance even if they posted no tweets in the current epoch. This is achieved by artificially arranging NULL tweets for them while their neighborhood contexts are designed as contexts of the user at several time points in the current epoch. As illustrated in Fig. 6, User-A did not post any tweets in Epoch 6. Nevertheless, our model was still able to predict her stance (marked with crosses). The prediction results were actually consistent with the stances revealed in User-A's tweets in Epoch 7. One of her tweets in Epoch 7 is "*The eu immigration policy will definitely crash the UK economy, taking us all with it. How many from turkey?#Brexit*", indicating that she was worried about the immigration and wanted a Brexit.

4.2.7. *Comparison with a two-class problem setting*

Some previous studies found that determining neutrality is difficult on social media data (Hu et al., 2013; Tan et al., 2011; Zhu, He, & Zhou, 2019). They thus confined their models to a two-polarity setup. To further confirm the effectiveness of the proposed model, we also delve into the two-class scenario, where the neutral tweets have been discarded. Since HLDA is an unsupervised non-parametric model, we retain the same topic embeddings that were used in the three-class setting for NOD. Other configurations remain the same. Fig. 7 plots the two-class classification accuracy of different methods on both datasets. Compared with the results in Fig. 4, we can observe improved results for the supervised methods and worse results for the LDA-based approaches, though the overall trend across epochs did not change much, especially on the ELECTION dataset, owing to its relatively small number of neutral tweets. Our proposed NOD and NOD_Coarse still outperform all the other models consistently across all the epochs on both datasets.

5. Conclusions and future work

In this paper, we have proposed the Neural Opinion Dynamics (NOD) model which can effectively take into account tweet content, topical and neighborhood context for user-level stance prediction. In particular we model a user with a sequence of posts, each of which embodies the user's own tweet. On this basis, we propose an attention mechanism to better integrate the self content and social context information. Finally, we employ an RNN on the time series to model user behaviors. Compared with static models which perform one-time prediction, our model can dynamically track topic-dependent stances. Experiment results on the two Twitter datasets verified the feasibility of NOD, showing that both the context attention and the dynamic setup help improve stance classification results.

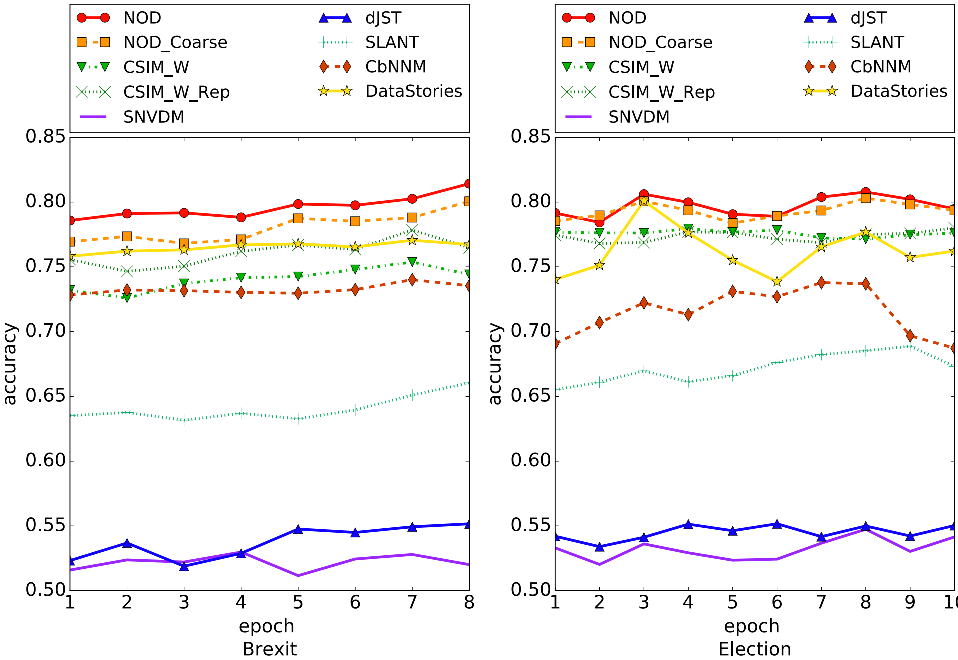

Fig. 7. Two-class classification accuracy on BREXIT and ELECTION.

In the future, we would like to explore following three directions. First, besides the neighborhood information derived from the following-follower relations, it is possible to construct the social networks using the *re-tweeting* or *mentioning* relations. However, such social networks might be more sparse and methods need to be investigated to tackle the sparsity problem. Second, in our current approach, only a fixed number of neighborhood tweets are used to form the neighborhood context. It is possible to also take into account each neighbor's social influence score since opinions from more influential users should carry higher weight. Last, NOD assumed that topic information is obtained beforehand. In future work we could investigate a unified model for joint topic-stance detection over streaming data.

Acknowledgements

We would like to thank the reviewers for their valuable comments and helpful suggestions. This work was funded by the National Key Research and Development Program of China (2016YFC1306704), the National Natural Science Foundation of China (61772132), the Natural Science Foundation of Jiangsu Province of China (BK20161430) and Innovate UK (grant no. 103652).

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.ipm.2019.03.010.

References

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv:1409.0473.

Bandhakavi, A., Wiratunga, N., Massie, S., & Padmanabhan, D. (2017). Lexicon generation for emotion detection from text. *IEEE Intelligent Systems, 32*(1), 102–108.

Barbieri, F., Kruszewski, G., Ronzano, F., & Saggion, H. (2016). *How cosmopolitan are emojis?: Exploring emojis usage and meaning over different languages with distributional semantics. Proceedings of the 2016 ACM on multimedia conference*. ACM531–535.

Baziotis, C., Pelekis, N., & Doulkeridis, C. (2017). *Datastories at semeval-2017 task 4: Deep LSTM with attention for message-level and topic-based sentiment analysis. Proceedings of the 11th international workshop on semantic evaluation (SEMEVAL-2017)*747–754.

Blei, D. M., Griffiths, T. L., & Jordan, M. I. (2010). The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *Journal of the ACM (JACM), 57*(2), 7.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research, 3*(Jan), 993–1022.

Cha, M., Haddadi, H., Benevenuto, F., & Gummadi, P. K. (2010). Measuring user influence in Twitter: The million follower fallacy. *ICWSM, 10*(10–17), 30.

Chen, C., Wang, Z., Lei, Y., & Li, W. (2016). *Content-based influence modeling for opinion behavior prediction. Proceedings of coling 2016, the 26th international conference on computational linguistics: Technical papers*2207–2216.

Chen, C., Wang, Z., & Li, W. (2018). Tracking dynamics of opinion behaviors with a content-based sequential opinion influence model. *IEEE Transactions on Affective Computing*.

Chen, H., Sun, M., Tu, C., Lin, Y., & Liu, Z. (2016). *Neural sentiment classification with user and product attention. Proceedings of the 2016 conference on empirical methods in natural language processing*1650–1659.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv:1406.1078.

De, A., Valera, I., Ganguly, N., Bhattacharya, S., & Rodriguez, M. G. (2016). *Learning and forecasting opinion dynamics in social networks. Advances in neural information*

*processing systems*397–405.
Dermouche, M., Velcin, J., Khouas, L., & Loudcher, S. (2014). *A joint model for topic-sentiment evolution over time. Data mining (ICDM), 2014 IEEE international conference on*. IEEE773–778.
Du, J., Gui, L., He, Y., Xu, R., & Wang, X. (2019). Convolution-based neural attention with applications to sentiment classification. *IEEE Access*.
Gilbert, C. H. E. (2014). *Vader: A parsimonious rule-based model for sentiment analysis of social media text. Eighth international conference on weblogs and social media (ICWSM-14)*216–225 available at (20/04/16) http://comp.social.gatech.edu/papers/icwsm14.vader.hutto.pdf.
He, Y., Lin, C., Gao, W., & Wong, K.-F. (2013). Dynamic joint sentiment-topic model. *ACM Transactions on Intelligent Systems and Technology (TIST), 5*(1), 6.
Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.
Hu, X., Tang, L., Tang, J., & Liu, H. (2013). *Exploiting social relations for sentiment analysis in microblogging. Proceedings of the sixth ACM international conference on web search and data mining*. ACM537–546.
Jia, P., Friedkin, N. E., & Bullo, F. (2017). Opinion dynamics and social power evolution over reducible influence networks. *SIAM Journal on Control and Optimization, 55*(2), 1280–1301.
Jozefowicz, R., Zaremba, W., & Sutskever, I. (2015). *An empirical exploration of recurrent network architectures. International conference on machine learning*2342–2350.
Kullback, S. (1997). *Information theory and statistics*. Courier Corporation.
Li, Y., Pan, Q., Yang, T., Wang, S., Tang, J., & Cambria, E. (2017). Learning word representations for sentiment analysis. *Cognitive Computation, 9*(6), 843–851.
Lim, K. W., & Buntine, W. (2014). *Twitter opinion topic model: Extracting product opinions from tweets by leveraging hashtags and sentiment lexicon. CIKM*. ACM1319–1328.
Lin, C., & He, Y. (2009). *Joint sentiment/topic model for sentiment analysis. CIKM*. ACM375–384.
Ma, Y., Peng, H., & Cambria, E. (2018). *Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM. AAAI*5876–5883.
Majumder, N., Poria, S., Gelbukh, A., Akhtar, M. S., Cambria, E., & Ekbal, A. (2018). *IARM: Inter-aspect relation modeling with memory networks in aspect-based sentiment analysis. Proceedings of the 2018 conference on empirical methods in natural language processing*3402–3411.
Marchetti-Bowick, M., & Chambers, N. (2012). *Learning for microblogs with distant supervision: Political forecasting with Twitter. Proceedings of the 13th conference of the European chapter of the association for computational linguistics*. Association for Computational Linguistics603–612.
McPherson, M., Smith-Lovin, L., & Cook, J. M. (2001). Birds of a feather: Homophily in social networks. *Annual Review of Sociology, 27*(1), 415–444.
Miura, Y., Taniguchi, M., Taniguchi, T., & Ohkuma, T. (2017). *Unifying text, metadata, and user network representations with a neural network for geolocation prediction. Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers)1. Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers)* 1260–1272.
Mnih, V., Heess, N., & Graves, A. (2014). *Recurrent models of visual attention. Advances in neural information processing systems*2204–2212.
Mohammad, S. M., Kiritchenko, S., & Zhu, X. (2013). NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. arXiv:1308.6242.
Mozetič, I., Torgo, L., Cerqueira, V., & Smailović, J. (2018). How to evaluate sentiment classifiers for Twitter time-ordered data? *PloS one, 13*(3), e0194317.
Oliveira, N., Cortez, P., & Areal, N. (2016). Stock market sentiment lexicon acquisition using microblogging data and statistical measures. *Decision Support Systems, 85*, 62–73.
Qiu, M., Sim, Y., Smith, N. A., & Jiang, J. (2015). *Modeling user arguments, interactions, and attributes for stance prediction in online debate forums. Proceedings of the 2015 SIAM international conference on data mining*. SIAM855–863.
Ren, Y., Zhang, Y., Zhang, M., & Ji, D. (2016). *Context-sensitive Twitter sentiment classification using neural network. AAAI*215–221.
Sasaki, K., Yoshikawa, T., & Furuhashi, T. (2014). *Online topic model for Twitter considering dynamics of user interests and topic trends. Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*1977–1985.
Sutskever, I., Vinyals, O., & Le, Q. V. (2014). *Sequence to sequence learning with neural networks. Advances in neural information processing systems*3104–3112.
Tan, C., Lee, L., Tang, J., Jiang, L., Zhou, M., & Li, P. (2011). *User-level sentiment analysis incorporating social networks. Proceedings of the 17th ACM SIGKDD international conference on knowledge discovery and data mining*. ACM1397–1405.
Tang, D., Qin, B., & Liu, T. (2015). *Document modeling with gated recurrent neural network for sentiment classification. Proceedings of the 2015 conference on empirical methods in natural language processing*1422–1432.
Tang, D., Zhang, Z., He, Y., Lin, C., & Zhou, D. (2019). Hidden topic–emotion transition model for multi-level social emotion detection. *Knowledge-Based Systems, 164*, 426–435.
Thonet, T., Cabanac, G., Boughanem, M., & Pinel-Sauvagnat, K. (2016). *Vodum: A topic model unifying viewpoint, topic and opinion discovery. ECIR*. Springer533–545.
Thonet, T., Cabanac, G., Boughanem, M., & Pinel-Sauvagnat, K. (2017). *Users are known by the company they keep: Topic models for viewpoint discovery in social networks. Proceedings of the 2017 ACM on conference on information and knowledge management*. ACM87–96.
Vo, D.-T., & Zhang, Y. (2015). *Target-dependent Twitter sentiment classification with rich automatic features. IJCAI*1347–1353.
Volkova, S., & Van Durme, B. (2015). *Online bayesian models for personal analytics in social media. AAAI*2325–2331.
Wang, J., Zhao, W. X., He, Y., & Li, X. (2014). Infer user interests via link structure regularization. *Acm Transactions on Intelligent Systems & Technology, 5*(2), 1–22.
Wang, Y., Huang, M., & Zhao, L. (2016). *Attention-based LSTM for aspect-level sentiment classification. Proceedings of the 2016 conference on empirical methods in natural language processing*606–615.
Xing, C., Wu, W., Wu, Y., Liu, J., Huang, Y., Zhou, M., et al. (2017). *Topic aware neural response generation. AAAI17. AAAI* 3351–3357.
Yang, Y., Deyu, Z., & He, Y. (2018). *An interpretable neural network with topical information for relevant emotion ranking. Proceedings of the 2018 conference on empirical methods in natural language processing*3423–3432.
Zarrella, G., & Marsh, A. (2016). Mitre at semeval-2016 task 6: Transfer learning for stance detection. arXiv:1606.03784.
Zhu, L., He, Y., & Zhou, D. (2019). Hierarchical viewpoint discovery from tweets using bayesian modelling. *Expert Systems with Applications, 116*, 430–438.