applied sciences

MDPI

Article

An Improved BiLSTM Approach for User Stance Detection Based on External Commonsense Knowledge and Environment Information

Peng Jia [1], Yajun Du [1,*], Jingrong Hu [2], Hui Li [1], Xianyong Li [1] and Xiaoliang Chen [1]

[1] School of Computer and Software Engineering, Xihua University, Chengdu 610065, China
[2] School of Computers, Chengdu University of Information Technology, Chengdu 610025, China
* Correspondence: duyajun@mail.xhu.edu.cn

Abstract: In the age of social networks, the number of tweets sent by users has led to a sharp rise in public opinion. Public opinions are closely related to user stances. User stance detection has become an important task in the field of public opinion. However, previous studies have not distinguished between user viewpoints and stances. These studies usually detected stance from the perspective of the tweet level but rarely the user level. Therefore, in this paper, we defined user stance, which is the user viewpoint (support, oppose, and neutral) toward the entire target event process. On this basis, we put forward a user stance detection method based on external commonsense knowledge (such as SenticNet) and environment information (such as a user's historical tweets, topic information, and neighbor tweets) and denote this method as ECKEI. First, in order to better integrate external commonsense knowledge into the neural network, we improved BiLSTM and called it CK-BiLSTM for complementary commonsense information to the memory cell. Secondly, we used LDA to extract the topic of user tweets and designed a topic-driven module to capture the users' neighbors' information. Finally, we used the attention mechanism to integrate information from users' historical tweets and neighbors' tweets obtained through topic information; then, we used the softmax layer to classify user stances into the support, neutral and oppose classes. In this paper, we conducted experiments and assessments on datasets containing information on Brexit and the elections to verify the practicability and effectiveness of our proposed method. Extensive experimental results on the Brexit and elections datasets show that our approach outperforms six baseline methods (SVM-ngram, NB, MTTRE (RNN), Pkudblab (CNN), TAAT, and Aff-feature). We use the average micro-F1 and average accuracy to measure performance on the detection of a user's stance. The ECKEI model makes improvements of 4.30–16.89% and 1.22–16.58% on the Brexit and election datasets, respectively, in terms of average micro-F1. Meanwhile, this model makes improvements of 4.24–17.46% and 0.48–14.64% on the Brexit and election datasets, respectively, in terms of average accuracy. Our model makes improvements of 5.34–17.30% and 2.65–19.73%, respectively, on the Brexit and election datasets in terms of average recall.

Keywords: social network; Bi-LSTM; stance detection; external commonsense knowledge; neighborhood information

check for updates

Citation: Jia, P.; Du, Y.; Hu, J.; Li, H.; Li, X.; Chen, X. An Improved BiLSTM Approach for User Stance Detection Based on External Commonsense Knowledge and Environment Information. *Appl. Sci.* **2022**, *12*, 10968. https://doi.org/10.3390/app122110968

Academic Editor: José Salvador Sánchez Garreta

Received: 5 September 2022
Accepted: 24 October 2022
Published: 29 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With the advancement of information technology and the rapid development of the Internet, the emergence of social media platforms has brought unprecedented changes to people's lifestyles and ways of thinking. Social media platforms have a large number of users. Additionally, social media has become an important source for people to understand current affairs, connect with the world, and obtain real-time information [1]. Many users love to describe their daily lives and participate in discussions on hot topics on social media platforms [2]. Hot topics refer to a major social event, a public policy that has attracted the attention of the whole people, or a country's elections and other hot events, and so on. It

often refer to hot issues that the people are most concerned about within a certain time and within a certain range. At the same time, with outbreaks of incidents, different users have different perspectives on hot events, and their stances are also different. Different stances often lead to an explosion of public opinion. Therefore, research on stance detection tasks is very necessary.

Stance detection is a sub-branch of sentiment analysis. Sentiment analysis aims to automatically determine the sentiment tendency of a user's tweet toward a specific object. The significant difference is that in the stance detection task, the purpose is to judge whether the user viewpoint towards a given target is positive (support, pro), negative (oppose, con), or neutral [3]. In real-life applications, stance detection plays a vital role in many specific tasks, such as analyzing political debates on online forums [4], fake news and false rumors [5], political stance trends [6], etc.

Due to the critical influence of stance detection in online social networks, an increasing number of people have studied stance detection in recent years. For example, Veyseh et al. [7] proposed a deep attention CNN-LSTM model that uses user useful relationships features available in additional social media such as friendship to improve performance. Pamungkas [8] proposed a stance detection model. The model uses and explores the features based on the sequence of event and emotional information.

With the rapid development of external knowledge engineering, we have discovered that external knowledge plays a vital role in the application of natural language processing (NLP). For example, short text classification [9] and sentiment analysis [10] combined with external knowledge can achieve better performance. Moreover, Mohammad et al. [11] found that people's emotions are conducive to the stance classification of tweets. Therefore, we extract emotional words by using external knowledge to improve the performance of stance detection.

Although in the previous works, the research on user stance detection achieved great success, we find that all have a common problem: ignoring the difference between stance and viewpoint. Regarding our daily life and the information, we find that stance is an affirmation of viewpoint and that their viewpoints are only one of the reasons for the users to confirm their stance. To better understand stance detection, we assume that user tweets are sent in chronological order, the user's viewpoint is related to the topic, and the user's viewpoint may change with the spread of time and events. Therefore, through the summary of the previous work, we first define the user's stance and propose a method based on integrating external commonsense knowledge and environmental information (ECKEI) to detect the stance of user-level tweets in social networks. Here, we collectively refer to the impact of topic information and neighborhood information as environmental information. User-level stance detection aims to use all tweets of a user in a period of time to analyse user stance, while tweet-level stance detection only uses one tweet to analyse user stance. The model captures the key factors in stance detection: the user's historical viewpoints, the topic information of tweets, the views of the user's neighbors driven by the topic information, and the emotion embedded in external commonsense knowledge. In our proposed framework, first, we obtain emotional information by embedding the external commonsense knowledge base SenticNet. We encode the sequence as a vector to better integrate commonsense knowledge into BiLSTM (CK-BiLSTM). Then, we have a topic-driven module to capture users' neighbors' information. Subsequently, we use the attention mechanism to integrate the information of users' historical tweets and neighbors' tweets obtained through topic information. Finally, we use a softmax layer to classify user stance. We verify through many experiments that our proposed ECKEI model is better than other baseline models.

In conclusion, our main contribution can be summarized as follows:

- We define user stance based on the user-level social network dataset and divide user stance into three categories (i.e., support, neutral, and oppose).
- To detect user stance, we propose a stance detection method based on external commonsense knowledge and environmental information (ECKEI) to detect user stance in social networks, which provides useful insights into the importance of user stance in social networks. We use external commonsense knowledge to obtain emotional information to extend BiLSTM (CK-BiLSTM) to complement ordinary BiLSTM to obtain more information.
- We conduct extensive experiments to validate our model. Through experiments on two social network datasets, we present that our method outperforms several baseline methods in the user stance detection task. Our method achieves 68.65%, 70.07%, 78.48% in average micro-F1, average acuracy and average recall on the Brexit dataset and 72.86%, 73.44%, 83.27% on the election dataset, respectively.

The rest of the paper is organized as follows. First of all, we apply for the related work in the stance detection model and external knowledge embedding in Section 2. Secondly, in Section 3, we define the stance and define the task. Again, in Section 4, we introduce our proposed method in detail. Then, in Section 5, the experimental results of the two datasets of Brexit and the US election and some baseline models are compared. Finally, Section 6 gives the conclusion of this article and its prospects.

2. Related Work

The main purpose of this section is to review the related work on stance detection based on the emotional information obtained from external commonsense knowledge and the influence of the users' surrounding environment (neighbors, topics, etc.). Therefore, it is mainly divided into two modules: stance detection and the application of external knowledge in tasks.

2.1. Stance Detection

Mohammad et al. [2] first proposed the task of detecting stance from a tweet in SemEval-2016 Task6. The stance detection task is defined as: given a tweet and a target entity (individual, organization, etc.), the system determines whether the author of the tweet is for or against the target entity. With the proposal of stance detection tasks, an increasing number of researchers are interested in stance detection. We will introduce some of the previous related works.

2.1.1. Machine Learning Methods for Stance Detection

Elfardy et al. [12] proposed an SVM-based supervision system that uses vocabulary, emotion, semantics, and potential frame semantic features to perform stance detection on specific targets in the SemEval-2016 dataset. Dey et al. [13] used a two-stage method based on SVM to detect stances in tweets. First, they determined whether the tweet was neutral or non-neutral. If it is a neutral tweet, it was deleted. If it was non-neutral, it was passed to the next stage, which classified the tweet as positive or negative.

Wojatzki et al. [14] used stacked classification and syntactic features to perform stance detection on SemEval 2016 Task6.The model demonstrated a slight performance improvement. Augenstein et al. [15] proposed training logistic regression (LR) models and auto-encoding bag-of-words to be labeled for stance detection on the Hillary Clinton dataset. Dias et al. [16] used Hillary Clinton's labeled data and Trump's unlabeled data and used a rule-based algorithm to help detect the SemVEval 2016 stance task.

Trabelsi et al. [17] introduced a purely unsupervised author interaction topic viewpoint model (AITV), which is more inclined to "heterogeneity" rather than "homogeneity" when identifying the nature of the author's online debate interaction. Furthermore, it was conducted in six corpora involving four different controversial issues extracted from two online debate forums. The results show that AITV's stance detection ability is better than

other models, even under the unsupervised model. Darwish et al. [18] proposed a stance model for detecting controversial topics. First, they created clusters among users, and then no domain or topic-level knowledge was required to specify related stances (tags) or perform actual tasks Mark. Their research results show that when implementing a clustering algorithm, using forwarding as a feature provides the best performance score. Supervised methods are exceeded when using short text and SVM models. Their findings are considered to be a big motivation for the future use of unsupervised methods for stance detection.

Ammar et al. [19] proposed an unsupervised method for specific target stance detection in a polarized context, especially in Turkish politics. The model is mainly based on a convolutional neural network-based multilingual universal sentence encoder. Then, they used the uniform manifold approximation and projection (UMAP) algorithm and hierarchical density-based clustering (HDB-SCAN) to cluster the projected user vectors.

2.1.2. Deep Learning Methods for Stance Detection

With the development of deep learning, an increasing number of people are using deep learning to research stance detection. For example, in the proposed SemEval-16 stance detection, Zarrella et al. [20] proposed using an RNN to initialize feature learning and perform remote supervision on two large-scale labeled datasets. Wan et al. [21] designed a voting scheme to predict the label of the test set through a specific convolutional neural network.

Du et al. [22] proposed a neural network model that combines RNN and attention mechanism, called the target-specific attention neural network (TAN) model, which merges specific target information into the attention mechanism of stance classification and achieves the most advanced performance. Zhou et al. [23] embedded a new attention mechanism in the two-way GRU-CNN structure at the semantic level. This novel attention mechanism allows for the model to automatically pay attention to the semantic features of the information mark when the stance is specified with the target to achieve stance detection of the goal.

Dey et al. [24] proposed a two-stage LSTM model using an attention mechanism (T-PAN). In both stages, they proposed a deep neural network based on LSTM, and the attention mechanism was embedded in each step. In the first stage, they classified subjective tweets (subjective or neutral tweets) according to the topic. In the second stage, they ranked the sentiment of emotional tweets to determine the stance of a given individual tweet.

Sun et al. [25] proposed a joint neural network model to predict the stance and sentiment of tweets at the same time. They used the LSTM model to complete these two tasks, where sentiment analysis is an auxiliary task, and stance detection is the main task. In the model, mutual sentiments are obtained through parameter sharing, and the obtained sentiment information is used in the research on stance detection. Aldayel et al. [26] explored user interactions (posts, online interactions) in Twitter and compared various functions, including topic content, network interactions, user preferences, and online network connections. Cignarells et al. [27] introduced a stance detection task and provided a stance detection dataset, which includes contextual information, such as the number of reposts, support times, and release date. The author's contextual information includes the number of followers, location, and user profile. From the user's friends and followers, additional knowledge is extracted from the author, forwarding, quoting, and replying. Siddiqua et al. [28] proposed a neural ensemble model that uses the advantages of Bi-LSTM variants to better learn long-term dependencies, in which each module with an attention mechanism is combined to amplify the contribution of essential elements in the final representation.

2.2. Incorporating External Knowledge

With the advancement of knowledge engineering, many studies incorporate external knowledge into the framework to solve NLP tasks and perform well. For example,

Zhang et al. [29] proposed a model (C-GCN) that introduces the knowledge of the dependency tree into the graph convolutional network. The author adopts a syntactic tree structure to maximize the deletion of irrelevant content while merging related information. They constructed an adjacency matrix and finally provided the classification result through the softmax layer. Tian et al. [30] introduced the emotional knowledge enhancement pre-training (SKEP) as a unified emotional representation for multiple dynamic analysis tasks. Three emotion loss prediction targets use automatic mining knowledge construction for dynamic masking. The emotional information combining words and polarity and aspect levels is embedded in the pre-trained emotional representation. Xu et al. [31] proposed a deep neural network that combines background knowledge with a dialogue system model. Through the special design of a recall-gate mechanism, the model can stimulate the background knowledge as a global memory to work with the local cellular memory of LSTM, thereby enriching the ability of LSTM to capture implicit semantic cues in conversation. Zhang et al. [32] proposed a semantic-emotion knowledge transferring (SEKT) model for stance detection, using external knowledge (semantic and emotional dictionaries) as a bridge to achieve cross-target knowledge transfer. First, they constructed a semantic-emotion heterogeneous graph from external semantic and emotional dictionaries, and it was input into the graph convolutional network to learn multi-hop semantic links between words and emotional tags. Finally, by adding a new knowledge perception memory unit, the learned semantic emotion graph representation was integrated into Bi-LSTM as prior knowledge.

3. Stance Definition and Task Definition

In this section, we briefly describe the redefinition of the stance and task definition to facilitate the understanding of the following.

3.1. User Stance Definition

In the previous work, the researchers did not distinguish between user stance and user viewpoint but express both as a person's evaluative reaction to things. Here, we need to distinguish between them. The user stance affirms one's own viewpoint, and the user viewpoint is only one of the reasons for establishing a user stance. To put it simply, a user viewpoint represents a person's attitude towards a goal at that point in time, while a stance represents a person's viewpoint on the goal over a period of time, which is a "set of viewpoints". For example, in the US election, we are given a stance target "Donald Trump". During the election process, people sent a tweet in Twitter, such as "If you're not watching #HillaryClinton's speech right now you are missing her drop tons of wisdom #SemSt". Obviously, this is an attitude of viewpoint regarding Donald Trump, but as time goes on and the user receives some interference and influence from external information, at other points or over a period of time after the obtainment of a new perspective and attitude, statements such as "#Trump never breaks your heart", "#Trump will build America strong", "#Trump's good" and "#Trump gives you more than you can get" are all supporting Donald Trump. Here, it is clear that users were supporting Donald Trump for some time afterwards; we thus regard the user's stance as support for Donald Trump. Therefore, we need to formally define user stance.

User Stance: For the stance, specifically, we need to give a clear goal entity (people, event, etc.), T, and evaluate the overall viewpoint of the user U to the destination by judging the views of the user tweets during the period of the target entity T. The result is usually one of three tags.

$$stance = \{support, neutral, oppose\} \tag{1}$$

- SUPPORT: We can judge from the history tweet that the tweeter supports the target.
- NEUTRAL: We can judge from the history tweet that the tweeter is neutral or has no clue.
- OPPOSE: We can judge from the history tweet that the tweeter is against the target.

3.2. Task Definition

We formulate the user stance detection problem as: for a social media platform, for a given user i, assuming that the user's tweets are arranged in chronological order, we denote their historical tweets as $T_i = \{t_{i,1}, t_{i,2}, \cdots, t_{i,N}\}$, where $t_{i,j}(j = 1, ..., n)$ is the j-th tweet sent by user i, and each tweet $t_{i,j}$ consists of a series of words, namely $t_{i,j} = \{w_1, w_2, \cdots, w_s\}$, where $w_m(m = 1, ..., s)$ is the m-th word in tweet $t_{i,j}$. We also assume that when user i publishes a tweet, their views will be affected by the H tweets recently posted, denoting it as $N_i = \{N_{i,1}, N_{i,2}, \cdots, N_{i,L}\}$. At the same time, we obtain the user's representation, we embed the corresponding topic information, and finally, we assign different weights to these modules, send them to the attention mechanism module, and classify the stance of their hidden states through the softmax function.

4. The Proposed Method

In this section, we will introduce the proposed ECKEI model in detail. We first describe the overall framework of ECKEI and then discuss each module in detail. The specific content is as follows.

4.1. ECKEI Framework

The framework of ECKEI is shown in Figure 1. We introduce the ECKEI architecture from the bottom up. As shown in Figure 1, ECKEI takes user tweets as input. For each tweet, a specially designed CK-BiLSTM is used to process and generate embedding vectors. CK-BiLSTM is an extension of ordinary BiLSTM, which can be flexible. We obtain emotional information by embedding the external commonsense knowledge base SenticNet. When we want to encode the tweet sequence as a vector to better integrate commonsense knowledge into neural networks, we extend BiLSTM and call it CK-BiLSTM. Then we have a topic-driven module to capture the users' neighbors' information. Subsequently, we use the attention mechanism to integrate the information from users' historical tweets and neighbors' tweets obtained through topic information. Finally, we use a softmax layer to classify user stance.

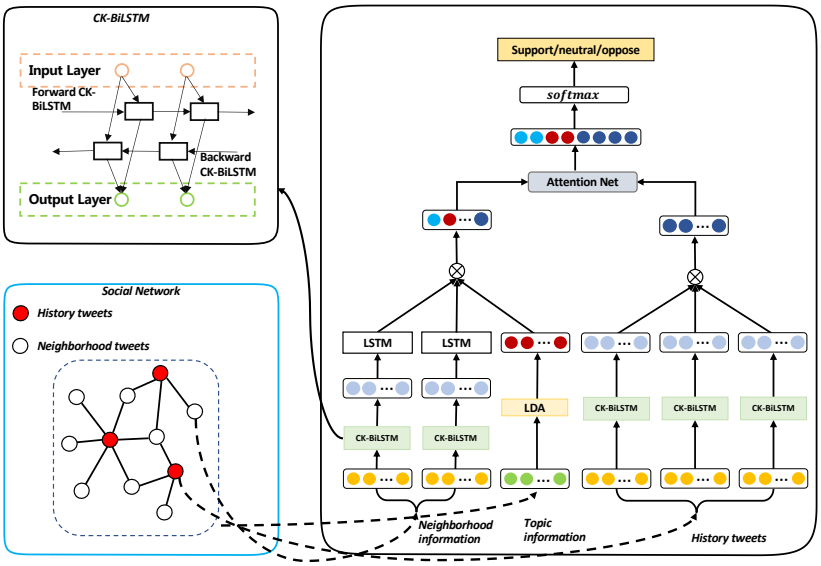


Figure 1. The overall architecture of the proposed ECKEI model.

4.2. Commonsense Knowledge

Mohammad et al. [11] found that sentiment helps to improve the performance of user stance detection. In this paper, we mainly use commonsense knowledge to obtain emotion as a knowledge source and embed it into the sequence encoder. Generally speaking, commonsense refers explicitly to the daily consensus that prevails among people in a social

environment. In daily life, the communication between people or the study of academic issues is a further exploration of commonsense knowledge. Take, for example, "China's capital is Beijing." This sentence is commonsense recognized by everyone. It can be said that commonsense does not need to be defined. It is recognized by people.

In recent years, there has be an increasing number of studies based on commonsense knowledge, and many knowledge bases have been published, such as DBpedia, FreeBase, ConceptNet, and SenticNet. In our model, we use the SenticNet database published by Cambria et al. [33], which contains 50,000 commonsense knowledge and rich emotional attributes (as shown in Table 1), for which it not only provides a concept-level representation but also semantic connection and emotional representation. We will use Figure 2 to explain in detail, where the arrow in the given word in the figure points to our commonsense knowledge. For example, given a word "cake", when we see the word "cake", according to our knowledge and our common sense of life, we know that cakes need to be baked, cakes can be eaten, cakes are sweet, cakes need to pass through the oven to be made, cakes can solve hunger, cakes are desserts, and so on. We are happy when we eat cakes and taste their sweetness, and we are in anticipation when we bake a cake. Thus, when we meet the word cake, through our commonsense knowledge, we can give such emotions as "joy" and "anticipation".

Table 1. Example of new knowledge concepts inferred from SenticNet.

SenticNet	IsA Event	Part of Celebration	Causes Joy	...
wedding	0.86	0.88	0.94	...
broom	0.83	0	0	...
birthday	0.85	0.98	0.97	...
sweep_floor	0	0	0	...

However, the high dimensionality of SenticNet hinders its development and application in deep neural network models. Mohammad et al. [34] established a concept-level sentiment analysis model. By reducing the dimensionality of emotional commonsense, the model allows semantic features related to concepts to be generalized. This enables intuitive clustering based on concepts of semantic and sentimental relevance. Based on the model's excellent performance, we embed commonsense knowledge into the deep neural sequence model to better perform stance detection in natural language text.

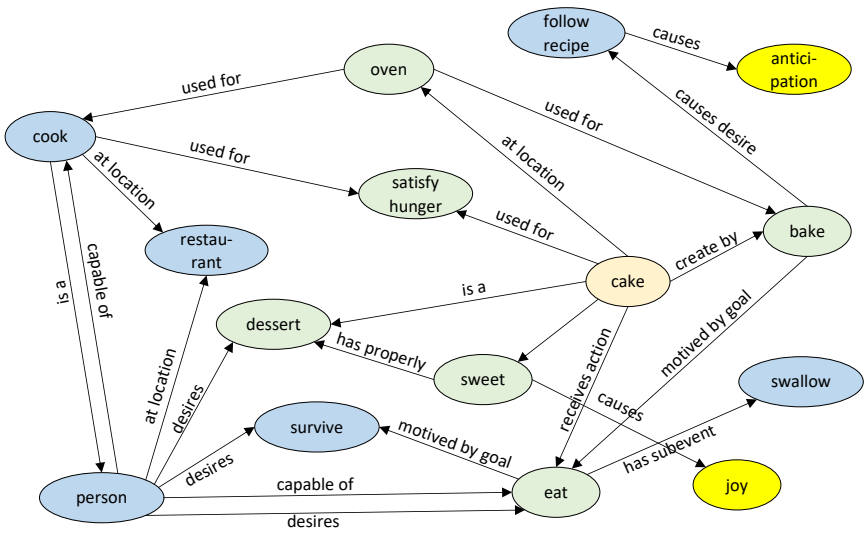


Figure 2. A sketch of the SenticNet graph showing part of the semantic network and sentiment for cake.

4.3. BiLSTM Network

After Hinton et al. [35] proposed deep belief network (DBN) in 2006, deep learning has entered a stage of rapid development. Currently, recurrent neural networks (RNN) are the most commonly used neural network model. An RNN is a neural network used to process sequence data. However, RNNs cannot solve the long-term dependency problem. Therefore, in order to enhance the performance of RNNs, researchers have proposed many different methods.

Long short-term memory (LSTM) [36] is the most widely used method to improve the RNN model. Compared with ordinary RNN, the most obvious advantage is that it can perform longer sequences. LSTM includes three gates (i.e., an input gate, a forget gate, and an output gate) and a memory unit to save the state of each neuron. A standard maintains a hidden state vector h and a cell state vector c, which control states regarding updates and output at each step. Let w_t be the input element of the *t-th* step; then, the architecture of the LSTM cell can be defined as follows:

$$f_t = \sigma\Big(W_f[w_t, h_{t-1}] + b_f\Big) \tag{2}$$

$$i_t = \sigma(W_i[w_t, h_{t-1}] + b_i) \tag{3}$$

$$\widehat{C}_t = \tanh(W_C[w_t, h_{t-1}] + b_C) \tag{4}$$

$$C_t = f_t * C_{t-1} + i_t * \widehat{C}_t \tag{5}$$

$$o_t = \sigma(W_o[w_t, h_{t-1}] + b_o) \tag{6}$$

$$h_t = o_t * \tanh(C_t) \tag{7}$$

where f_t, i_t, o_t indicate the activation vectors of the forget gate, input gate and output gate, respectively. h_{t-1} represents the hidden state at time $t-1$. W_f, W_i, W_C, W_o are the recurrent weight matrices. The biases are depicted using the notations b_f, b_i, b_c, b_o.

The standard LSTM can only capture the historical information of the sequence. Due to the complexity of the natural language sentence structure, future information may be taken into consideration when labeling the sequence. On this basis, Graves et al. [37] proposed bidirectional long short-term Memory, which uses two LSTMs to work in the forward and backward time directions, respectively. This facilitates learning of bidirectional long-term dependencies across two time steps. We can splice BiLSTM into $h_t = \left[\overrightarrow{h}_t \oplus \overleftarrow{h}_t\right]$.

4.4. CK-BiLSTM

In recent years, in order to enable the LSTM network to transmit information to obtain more comprehensive information, more innovative LSTM-based solutions have been proposed to achieve a better understanding of missing data and the subsequent patterns and relationships in the data [38]. In our lives, we often use bad to describe something, for example, a bad person, which may indicate that "bad" is a word with emotional information. It is a modifier of the next word, "person". Therefore, when there is less information, it should be deleted at the next time step.

Based on the questions raised above, we add the emotional information from commonsense knowledge to BiLSTM. We build the extension of emotional semantic knowledge by extending the method proposed by Ma et al. [39] and others. In order to help filter the information of the previous time step to the next time step, we supplement the original information with commonsense knowledge. We extend BiLSTM and fuse the external commonsense to BiLSTM (shortened for CK-BiLSTM). The CK-BiLSTM architecture is given in Figure 3. Red lines represent commonsense emotional knowledge. At each time step t, we assume that each word can trigger and map the emotional knowledge to an O

dimensional space. We use O to represent the number of emotional knowledge, such as $[\beta_{t,1}, \beta_{t,2}, \ldots\ldots, \beta_{t,m}, \ldots\ldots, \beta_{t,O}]$, where we will use the average vector to embed them.

$$\beta_t = \frac{1}{O}\sum_m \beta_{t,m} \tag{8}$$

The formula of BiLSTM with commonsense knowledge is illustrated below:

$$f_t = \sigma\Big(W_f[w_t, h_{t-1}, \beta_t] + b_f\Big) \tag{9}$$

$$i_t = \sigma(W_i[w_t, h_{t-1}, \beta_t] + b_i) \tag{10}$$

$$\widehat{C}_t = \tanh(W_C[w_t, h_{t-1}] + b_C) \tag{11}$$

$$C_t = f_t * C_{t-1} + i_t * \widehat{C}_t \tag{12}$$

$$o_t = \sigma(W_o[w_t, h_{t-1}, \beta_t] + b_o) \tag{13}$$

$$o_t^c = \sigma(W_o^c[w_t, h_{t-1}, \beta_t] + b_o^c) \tag{14}$$

$$h_t = o_t * \tanh(C_t) + o_t^c * \tanh(W_c\beta_i) \tag{15}$$

where f_t is the forget gate, i_t is the input gate, C_t is the cell state and o_t is the output gate. W_f, W_i, W_C and W_o are known as weight matrices. w_t and h_t is the input and the hidden state at time t. h_{t-1} represents the hidden state at time $t-1$. b_f, b_i, b_C, b_0 refer to biased values of different gates. The sigmoid activation function is specified by σ and *tanh*, and $*$ is the product of Hadamard. Furthermore, we have an important information supplement activation vector here, that is, o_t^c, which is a BiLSTM output gate extended with commonsense knowledge. This gate will complement the normal gate o_t^c of BiLSTM and send it to the hidden state, as shown in Figure 3, where W_o^c is the weight matrices and b_o^c is the biased values. The CK-BiLSTM layer outputs a sequence of hidden states $\{h_1, \cdots, h_m\}$, which are conveyed to an average pooling layer to obtain a tweet representation c_t.

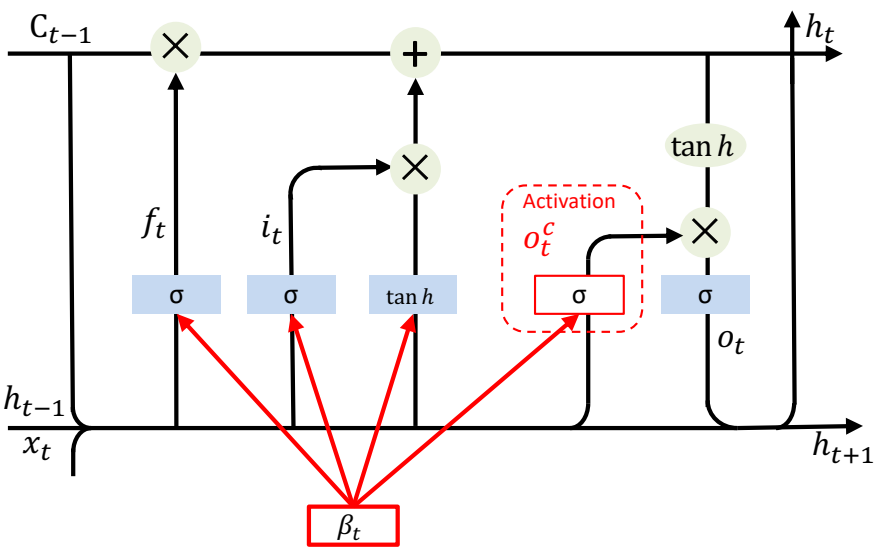


Figure 3. The structure of the commonsense knowledge BiLSTM unit.

4.5. Topic Extraction

In this section, we capture the potential topics of each tweet. Through the literature review, we found that the LDA topic extraction method in short text has very good performance as sentiment classification [40]. Thus, to capture the possible topic of each tweet, we adopt the latent Dirichlet allocation (LDA) model [41]. We extract the best topic from the historical tweets by the topic extraction method. In LDA, we convert each tweet into

bag-of-words format (so the order of words is not considered). This method mainly uses word probability to represent the topic and obtains the word probability through LDA, and the word with the highest probability in each tweet can be clearly understood. Finally, each tweet is associated with a topic represented as a vector, which is denoted as z.

Each word in the dataset is obtained through the process of "selecting a topic with a certain probability, and selecting a word from the topic with a certain probability." The specific formula is shown in Equations (16)–(20).

$$p(\theta \mid \alpha) = \frac{\Gamma\left(\sum_{i=1}^{k} \alpha_i\right)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \prod_{i=1}^{k} \theta_i^{\alpha_{i-1}} \tag{16}$$

$$p(\theta, z, w \mid \alpha, \beta) = p(\theta \mid \alpha) \prod_{n=1}^{N} p(z_n \mid \theta) p(w_n \mid z_n, \beta) \tag{17}$$

$$p(\boldsymbol{w} \mid \alpha, \beta) = \int p(\theta \mid \alpha) \left(\prod_{n=1}^{N} \sum_{z_n} p(z_n \mid \theta) p(w_n \mid z_n, \beta) \right) d\theta \tag{18}$$

$$p(D \mid \alpha, \beta) = \prod_{d=1}^{M} \int p(\theta_d \mid \alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{d_n}} p(z_{d_n} \mid \theta_d) p(w_{d_n} \mid z_{d_n}, \beta) \right) d\theta_d \tag{19}$$

$$p(\theta, z \mid \boldsymbol{w}, \alpha, \beta) = p(\theta, z, \boldsymbol{w} \mid \alpha, \beta) / p(\boldsymbol{w}, \mid \alpha, \beta) \tag{20}$$

Here, N represents the length of each tweet, parameter θ is a random variable of the k-dimensional Direchlet, parameter α represents a k-dimensional vector with components, and parameter Γ represents the gamma function. D represents the entire dataset. Equation (16) here represents the probability density function, Equation (17) represents the generation process of the topic. The topic mixture, θ, the joint distribution of N topics z, and N words w can be obtained. Then, Equation (18) represents the likelihood function for each tweet, which is obtained by integrating over θ and summing over z; Equation (19) represents the product of the likelihood functions of all tweets in the entire dataset D. Equation (20) represents that the topic distribution is derived by computing the posterior probability of the latent variable z.

4.6. Neighborhood Context

In the era of social network media, the information dissemination among people is rapid, and the tweet status of individual neighbors also affects each other. Myers et al. [42] find that social network neighbor information disseminated 71% of information on Twitter. Therefore, what we need to consider here is the neighbor relationship information.

In this paper, we propose a topic-driven module to capture neighbor information. Our main method of obtaining neighbor information is based on the previously obtained topic. At the same time, when we obtain neighbor tweets, we obtain them in chronological order. First, we obtain their hidden vectors through CK-LSTM, merge them together, and input them into LSTM. The output of LSTM is weighted by neighbor tweets and topic information z_i from Section 4.5:

$$c_i^N = \sum_{l=1}^{L} \alpha_l \widehat{h}_{i,1}^N \tag{21}$$

$$\alpha_l \propto exp\left(\left[\widehat{h}_t, z_i \right] tanh(W_h \widehat{h}_{i,1}^N + W_z z_i) \right) \tag{22}$$

where $\left\{ \widehat{h}_{i,1}^N, \widehat{h}_{i,2}^N, ..., \widehat{h}_{i,L}^N \right\}$ denotes the hidden state output of each tweet of each $N_{i,l}$ in the neighborhood context, z_i denotes the associated topic, h_t is the representation of the user own tweet at time step i, and both W_h and W_z are weight matrices. Equation (22) indicates the selected topic-driven neighbor tweet information.

Here, z_i denotes the associated topic, h_i is the representation of the user tweet at time step i, and both W_h and W_z are weight matrices.

4.7. Attention-Based User History Tweets

As mentioned above, when a user posts a tweet, they will pay more attention to their neighbor's tweets, which also contain topics of interest. Before our work, many works have proven that the attention mechanism plays an essential role in various tasks of natural language processing, such as topic perception [43], sentiment analysis [44], and prediction tasks [45], all of which have good performance. It can be seen from Figure 1 that we stably input the user's historical tweets, topic information, and neighbor information in the attention layer. First, for each tweet, its final representation is generated by combining the historical tweet representation with the neighbor information driven by the corresponding topic.

$$g_{t,i} = \frac{\alpha_1 c_i^N \bigoplus \alpha_2 c_i}{\alpha_1 + \alpha_2} \tag{23}$$

where α_1 captures the importance of topics and neighbors, and α_2 is used to capture the importance of historical tweets. If $\alpha_1 \ll \alpha_2$, then Equation (23) degenerates to a model that completely relies on historical tweet information. In Section 5.8, we will analyze the parameters of α_1 and α_2. The combined representations are then passed to a user attention layer whose output $t_{i,n}$ is a normalized weighed sum of $\{g_{t,1}, g_{t,2}, \ldots, g_{t,i}\}$.

$$t_{i,n} = \sum_{n=0}^{N} \beta_n g_{t,n} \tag{24}$$

$$\beta_n = \frac{\exp(\mathbf{W} u_{t,n})}{\sum_{n=0}^{\mathrm{N}} \exp(\mathbf{W} u_{t,i})} \tag{25}$$

$$u_{t,i} = \tanh(\mathbf{W_u} g_{t,i} + \mathbf{b}) \tag{26}$$

where $\beta_s(\mathrm{n} > 0)$ represents the user attention *s-th* neighborhood tweet. In other words, β_s essentially measures the degree of influence from the *s-th* neighborhood tweet. W and W_u is the weight vector, and the $u_{t,i}$ function is a smoothing factor calculated from $g_{t,i}$ by a fully connected layer. Finally, we input it into the softmax layer for classification.

$$y_{i,n} = \mathrm{softmax}\left(t_{i,n} W_y\right) \tag{27}$$

where W_y denotes the weight matrix of fully connected layer. The output $y_{i,n}$ of the softmax layer is the probability distribution of the final individual stance category.

5. Experimental Analysis

In this section, we conduct experiments and evaluations on the model and divide users' stances into three categories (i.e., support, neutral and oppose) in order to demonstrate the performance of our proposed methods.

5.1. Dataset

Our experiment was conduct on two datasets: BREXIT and US General ELECTION [46]. Table 2 lists the specific information in the two datasets. The BREXIT dataset contains 38,335 users and 363,961 tweets with 115,012/106,640/142,309 support/oppose/neutral tweets, respectively, and the dataset US General ELECTION contains 108,689 users and 452,128 tweets with 335,479/92,234/24,415 support/oppose/neutral tweets. In these two datasets, we can observe the proportions of support, oppose, and neutral tweets in Figure 4. For each user, we use 90% of their tweets for training and 10% for testing.

Table 2. Statistics of the two datasets.

	User	Tweet	Support	Neutral	Oppose
BREXIT	38,335	363,961	115,012	142,309	106,640
ELECTION	108,689	452,128	335,479	24,215	92,234

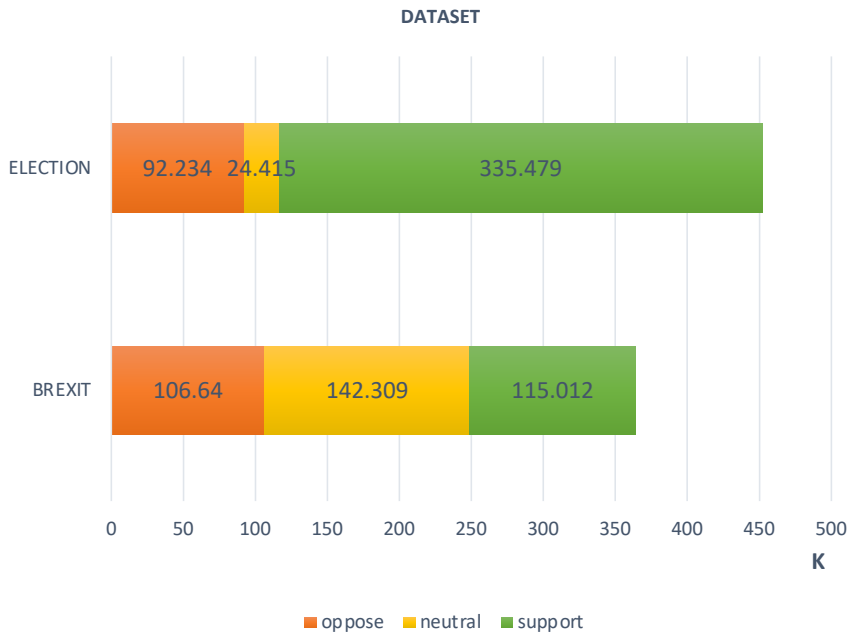


Figure 4. The number of tweets classified as support, neutral and oppose.

5.2. Pre-Processing

Since our data are not standardized, pre-processing is crucial before extracting features, and we need to normalize the data. There are many special symbols, such as URLs, emails, and other information in tweets, and we need to normalize them in the pre-processing process. We mainly dealt with this by: (1) deleting any characters that are not in the range of the English alphabet; (2) standardizing the sequence of repeated characters, such as: "Coooooooooool," which we will treat as "Cool". This post form is social. There are many in the network, and this kind is mainly to express emphasis; and (3) deleting the stop-words and applying spell correction on all words by using the stop-word list-filtering approach [47]. Looking back at our model, each user is a training instance consisting of a sequence of history tweets. In our experiment, we set a fixed length for the historical tweets posted by each user. The main method was to fill in empty tweets by users with fewer tweets or trim users with more tweets for additional tweets. We used the same rules to deal with the user neighborhood context. In our datasets, we observed that less than 11% of users posted more than 3 tweets. Therefore, we set the length of the user posting sequence to 3. At the same time, users have an average of 5 neighborhood contexts for tweets. Therefore, we set the neighborhood context size to 5 in our experiment, and we used the 300-dimensional word embedding pre-trained on the dataset.

5.3. Topic Setting

In the model module, we mentioned the use of LDA to extract the topics of user tweets. Because of the topics discussed on social networks, there may be many different topics. Therefore, during topic extraction, we only kept the most important topics. In particular, we only kept the most prominent topics and merged topics with fewer than 10,000 associated tweets as 'Others.' As shown in Table 3 and Figures 5 and 6, we used a table and a topic word cloud to display the topic information. Additionally, we introduce a "Null" topic

for those empty tweets added to the training dataset. From Figure 5 and Figure 6, in the "BREXIT" dataset, we can see that topics such as "economy," "campaign," "immigration," "vote," and "Europe" are the most important, while in the "ELECTION" dataset, topics such as "Hillary," "Trump," "vote," "candidate," and "American" are the most prominent.

Table 3. Top ten words in the discovered topics on two datasets.

Dataset	Topic Identification	Topic Words
BREXIT	Sovereignty	leave work vote mislead stay Europe country control borders independence
	Economy	EU UK economy jobs Brexit trade free NHS money tax
	Immigration	brexit UK EU EURef leaveEU voteleave England migrants refugees Muslim
	Campaign	Brexit can't remain racist attack MP JoCox murder working class
	BBCdebate	debate remain BBCdebate voteremain watching blame Boris ITVEURef argument tonight
	Boris&Farage	voteleave gove Boris Johnson Farage Brexit Michael Cameron David Geldof
	Polls	referendum EU Brexit EURef UK remain alive debate poll polls
	Vote	EURef vote referendum Thursday week today debate positive days June
ELECTION	Vote	voting pople day president candidate supporters supporter voted vote America
	Email scandal	Hillary Clinton emails FBI Comey Comey director Trump talking guy things
	Jobs	election world vote state signs jobs tax plan steel China
	Slogans	Trump Donald MAGA Clinton president vote election final Hillare IMWITHHER
	Campaign	capitol Donald campaign Trump Nugent Ted Clinton sign Reno protester
	Election	Donald indirect presideny world Clinton vote united win states campaign

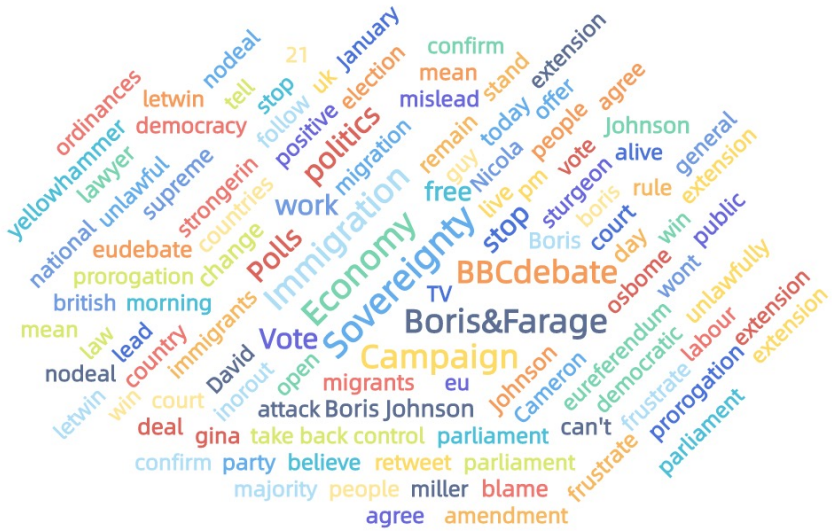


Figure 5. Topic word cloud of BREXIT dataset.

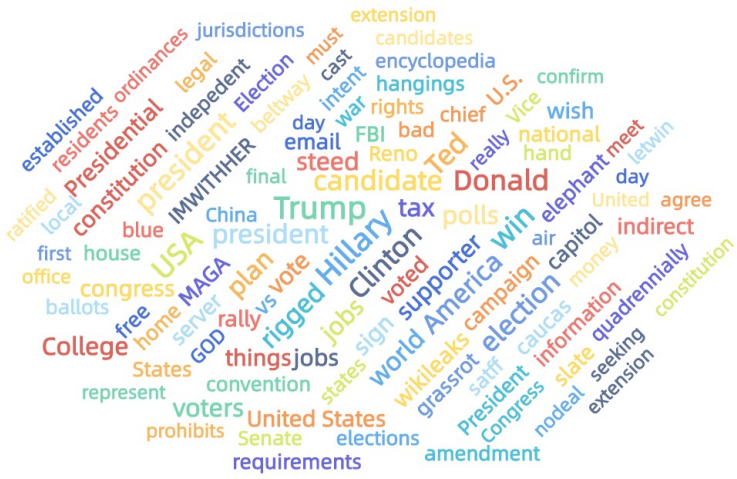


Figure 6. Topic word cloud of ELECTION dataset.

5.4. Baseline

In order to evaluate the performance of our proposed ECKEI approach, we compared our model with the classifier proposed by the organizer in task6 of the SemEval-2016 shared task and other advanced methods:

- SVM-ngram [11]: This model is proposed in SemEval-2016 task6 and mainly uses multilingual features and characters to train the SVM classifier.
- NB [32]: This model is a naive Bayesian classification model. It also uses the monolingual features and the multilingual features of characters to train the naive Bayesian model.
- MTTRE(RNN) [20]: This model uses two recurrent neural network (RNN) classifiers to identify stance.
- Pkudblab(CNN) [21]: This model designs a predictive voting scheme using a convolutional neural network (CNN). The label with the highest frequency in all iterations is used as the final classification result.
- Temporal attention(TATT) [7]: This model is a deep attention CNN-LSTM method, which takes vectors in the timeline as context to capture the temporal dynamism in users' stance evolution. It also uses useful relationship features available in additional social media such as friendship to improve performance.
- Affective-feature(Aff-Feature) [8]: The model uses and explores the features based on the sequence of events and extracts emotional features from emotional dictionaries such as EmoSenticNet (EmoSN) and Dictionary of Affect in Language (ANEW), and it uses an SVM classifier to achieve user stance classification.

5.5. Evaluation Metrics

To evaluate the performance of our proposed model and the baseline model, we used two widely used multi-label classifier scoring indicators for evaluation: accuracy, micro-F1, recall, and their average. These are defined as in Equations (28)–(33):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{28}$$

$$Avg_Acc = \frac{1}{n} \sum_{i=1}^{n} Accuracy_i \tag{29}$$

$$Micro - F1 = 2 \cdot \frac{Precision_{micro} \cdot Recall_{micro}}{Precision_{micro} + Recall_{micro}} \tag{30}$$

$$Avg_MF1 = \frac{1}{n}\sum_{i=1}^{n} Micro - F1_i \tag{31}$$

$$Recall = \frac{TP}{TP + FN} \tag{32}$$

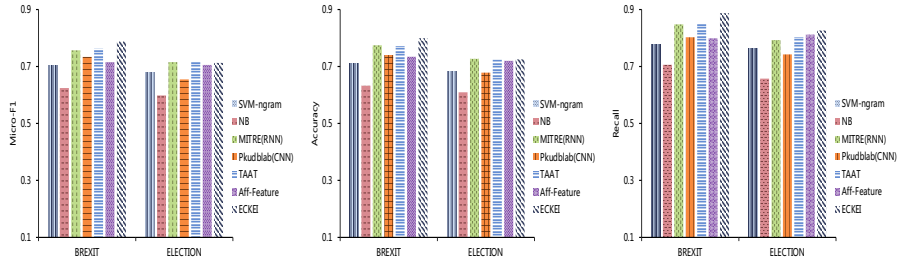
$$Avg_Recall = \frac{1}{n}\sum_{i=1}^{n} Recall_i \tag{33}$$

where n is the number of basic stance categories. TP represents the number of samples where the true label and stance label predicted by the model are both positive categories, and TN represents the number of samples where the true label predicted by the model is a positive category and the user stance label is a negative category. FP represents the number of samples where the true label is a negative category and the stance label predicted by the model is a positive category, and FN represents the number of samples where the true label and the user stance label predicted by the model are both negative categories. $Precision_{micro} = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} TP_i + \sum_{i=1}^{n} FP_i}$, and $Recall_{micro} = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} TP_i + \sum_{i=1}^{n} FN_i}$, where i stands for category.

5.6. Performance Comparison

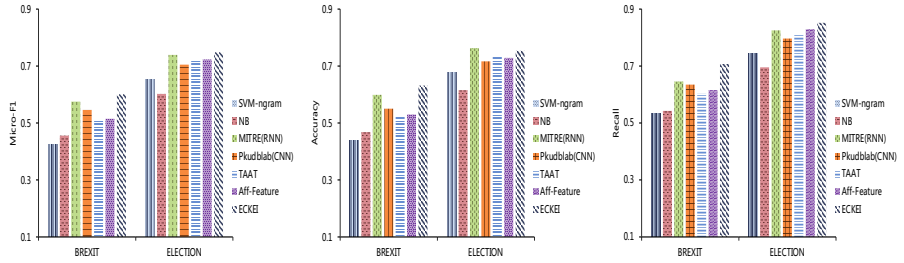
To illustrate the effectiveness of our proposed model, we first compared ECKEI to the baseline methods mentioned above. Figures 7–9 demonstrate the comparative results of different approaches when detecting stance, and the overall performances of the different approaches on the two datasets are shown in Table 4. The results on the two datasets show that the ECKEI model is more competitive. From Figures 7–9, we can observe that user stance detection methods are more inclined to large datasets and can play a important role. Regarding machine learning and neural network-based models, TAAT has better performance than other methods; this is because more features are extracted. However, the TAAT model is obviously a bit laborious when processing topic information and neighbor information.

As show in Table 4, our proposed model significantly outperforms all baseline methods by achieving the highest average micro-F1, average accuracy and average recall on two datasets. From the results, we can see the SVM-ngram model has better performance than the NB model when comparing traditional machine learning models. Regardless of the dataset, we find that the average accuracy, average micro-F1 and average recall scores of SVM n-gram are better than NB by 1.42–6.70%, which indicates that the SVM model has a great advantage in stance classification tasks. Regarding the Aff-feature model, we can find that after adding the information in the context of the tweet, it demonstrates a performance improvement of 4.88–6.41% compared to the previous deep neural network model. Then, after the historical tweets are input, they are compared with the TAAT model. The adjacent tweets of each tweet are regarded as local context. The attention mechanism is used to obtain the influence of adjacent tweets, and we can find that the model has obvious advantages when compared with other models. Finally, we find that MITRE (RNN) is the closest model to our model's average accuracy, average micro-F1 and average recall scores. The first reason to explain the better performance of the ECKEI model is that it uses the CK-BiLSTM network as an encoder to process external commonsense knowledge in order to better supplement the original information. Secondly, our proposed model considers the influence of neighbors' emotions on users' viewpoints states, which helps to explore changes in stance between user interactions.



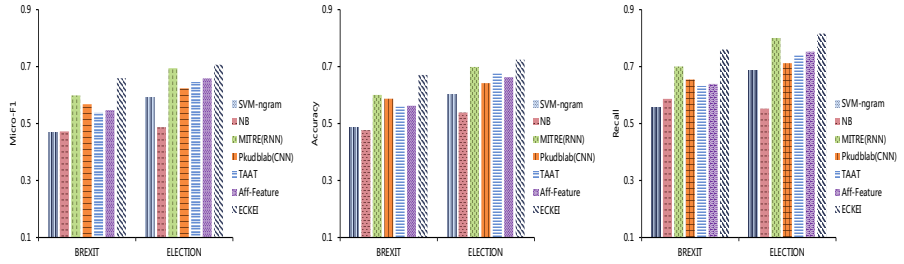
(**a**) Micro-F1 (**b**) Accuracy (**c**) Recall

Figure 7. Micro-F1, accuracy and recall results on two datasets for the support stance.



(**a**) Micro-F1 (**b**) Accuracy (**c**) Recall

Figure 8. Micro-F1, accuracy and recall results on two datasets for the neutral stance.



(**a**) Micro-F1 (**b**) Accuracy (**c**) Recall

Figure 9. Micro-F1, accuracy and recall results on two datasets for the opposing stance.

Table 4. Avg_MF1, Avg_Acc and Avg_Recall of our approaches on BREXIT and ELECTION.

Method	BREXIT			ELECTION		
	Avg_MF1	**Avg_Acc**	**Avg_Recall**	**Avg_MF1**	**Avg_Acc**	**Avg_Recall**
SVM-ngram	53.18	54.48	62.30	64.22	65.50	73.06
NB	51.76	52.59	61.18	56.28	58.80	63.54
MITRE(RNN)	64.35	65.81	73.14	71.64	72.96	80.62
Pkudblab(CNN)	61.59	62.61	69.67	66.12	67.88	75.06
TAAT	60.44	61.78	69.48	69.45	71.08	78.45
Aff-Feature	59.26	60.89	68.45	69.59	70.38	79.80
ECKEI	**68.65**	**70.05**	**78.48**	**72.86**	**73.44**	**83.27**

5.7. Ablation Experiment

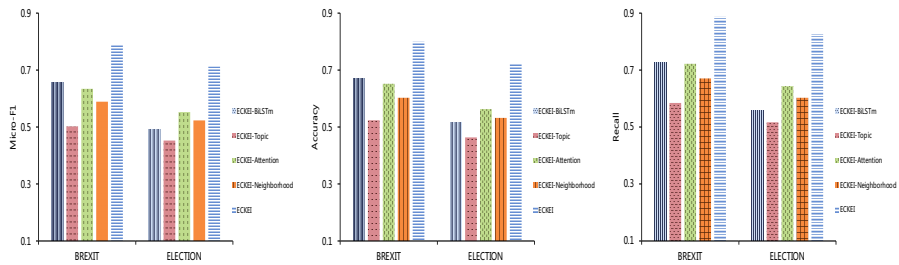
In this section, we aimed to illustrate the effectiveness of different parts of our proposed model. We designed a set of ablation experiments to further evaluate the importance of different components in the ECKEI model. First, we set up a series of methods to compare the performance of BREXIT and ELECTION. Specifically, the ECKEI model and its variant models are compared as follows.

- ECKEI-BiLSTM: In order to analyze the effect of commonsense knowledge, we used ordinary BiLSTM instead of CK-BiLSTM in ECKEI.
- ECKEI-Topic: In order to evaluate the impact of topic information on stance classification, we removed the topic part for comparison.
- ECKEI-Attention: In order to evaluate the effect of the attention mechanism, we removed the attention mechanism part for comparison.
- ECKEI-Neighborhood: In order to evaluate the influence of neighbor information, we removed the neighbor information module for comparison.

The results of the ablation experiment are shown in Figures 10–12. Further, we compared the experimental results concerning average accuracy, average micro-F1 and average recall, as shown in Table 5. From Figures 10–12, we can observe that for most stance categories, detection performance will be lower when removing each of the three key components. Specifically, in the ECKEI model, we use BiLSTM instead of CK-BiLSTM to significantly reduce the performance of the model, which shows that the introduction of commonsense knowledge can improve the quality of stance classification. In Table 5, we can observe that the ECKEI-Topic model has a significant decline in average accuracy, average micro-F1 and average recall. It indicates that the topic module in the paper is essential because our neighbor's tweets are passed. The topic information can obtain tweets with the same topic as the user, so the topic information plays a decisive role here. We find that when one clicks neighbor tweets, it has the least impact on the entire module because topic information plays a big role in neighbor tweets. We must extract neighbor tweets based on topic information to obtain user similar tweets, which can improve our stance detection performance. Finally, we observe the attention mechanism (ECKEI-Attention). When the attention mechanism module is removed, the impact on the entire model is relatively large. If there is no attention mechanism module, we cannot distinguish which is the most important information. All the information is integrated into the entire model. The attention mechanism can solve this problem very well. It can control the received information by itself. It can perform weight matching to achieve the best results.

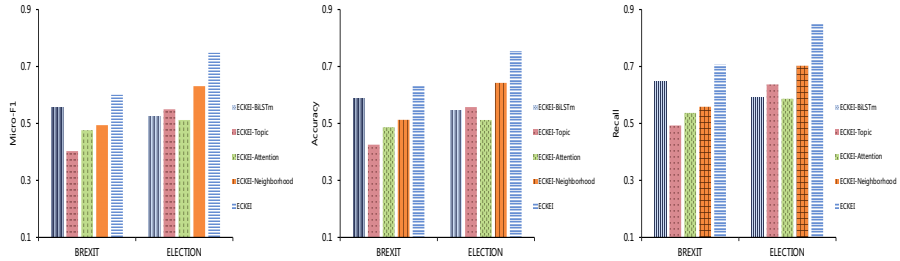
Table 5. Avg_MF1, Avg_Acc and Avg_Recall of ECKEI ablation analysis on two datasets.

Method	BREXIT			ELECTION		
	Avg_MF1	Avg_Acc	Avg_Recall	Avg_MF1	Avg_Acc	Avg_Recall
ECKEI-BiLSTM	62.08	63.55	69.50	50.67	52.05	56.78
ECKEI-Topic	45.30	46.76	52.72	49.82	50.44	56.15
ECKEI-Attention	54.10	55.06	61.52	55.80	56.48	62.67
ECKEI-Neighborhood	53.57	54.65	61.07	64.47	65.13	68.02
ECKEI	**68.65**	**70.05**	**78.48**	**72.86**	**73.44**	**83.27**



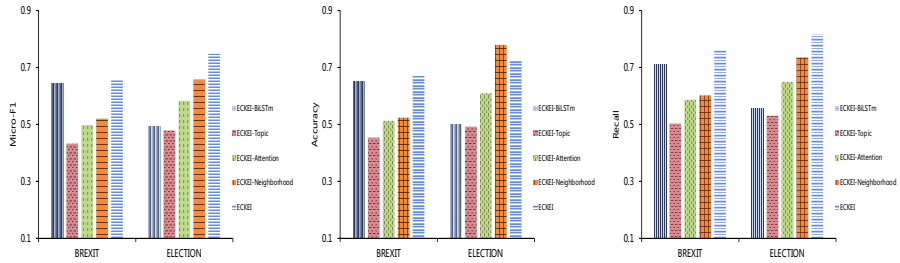
(**a**) Micro-F1 (**b**) Accuracy (**c**) Recall

Figure 10. Micro-F1, accuracy and recall ablation results on two datasets for the support stance.



(**a**) Micro-F1 (**b**) Accuracy (**c**) Recall

Figure 11. Micro-F1, accuracy and recall ablation results on two datasets for the neutral stance.

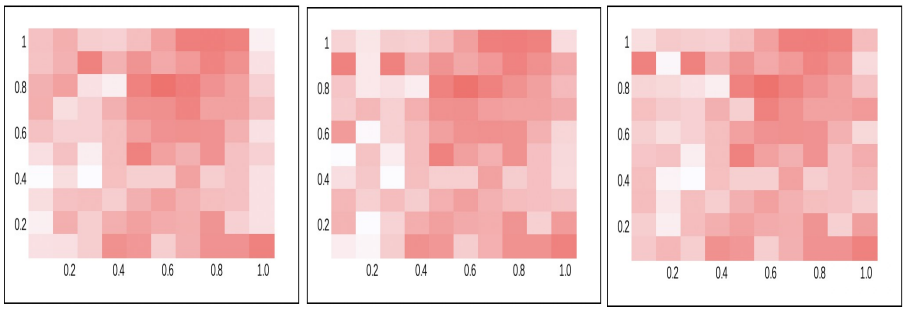


(**a**) Micro-F1 (**b**) Accuracy (**c**) Recall

Figure 12. Micro-F1, Accuracy and Recall ablation results on two datasets for the opposing stance.

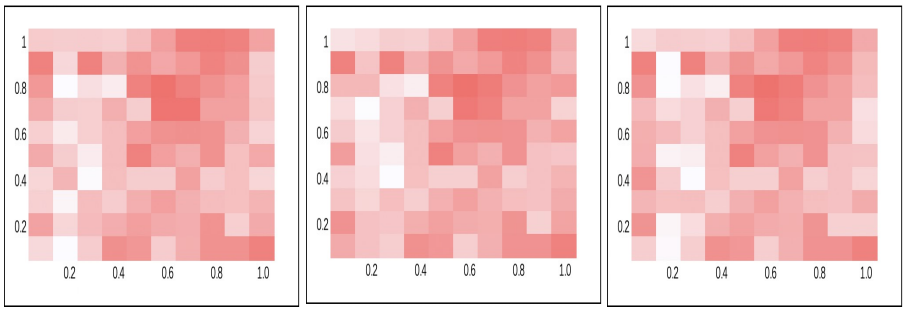
5.8. Parameter Analysis

In order to analyze the importance of the user's historical tweets and neighbor topic information, in Section 4.7, we give two parameters above, α_1 and α_2. In this subsection, we mainly analyze the influence of the two parameters α_1 and α_2 on the Avg_Acc and Avg_MF1 values. When we set α_1 to 0, we do not consider neighbor topic information. When we set α_2 to 0, we will not view the user's historical tweets. Therefore, in our research work, the value of α_2 cannot be 0, which would violate our definition of user-level stance. At the same time, to determine the influence of the parameters α_1 and α_2 on the Avg_Acc and Avg_MF1 values on the two datasets, we have made a heat map on the two datasets to analyze the influence of α_1 and α_2, as shown in Figures 13 and 14. We can see from these figures that when α_1 is 0.6 and α_2 is 0.8, our average micro-F1, average accuracy and average recall values are the best.



(**a**) Avg_Acc (**b**) Avg_MF1 (**c**) Avg_Recall

Figure 13. The influence of α_1 and α_2 on the Avg_MF1, Avg_Acc and Avg_Recall of the dataset BREXIT.



(**a**) Avg_Acc (**b**) Avg_MF1 (**c**) Avg_Recall

Figure 14. The influence of α_1 and α_2 on the Avg_MF1, Avg_Acc and Avg_Recall of the dataset ELECTION.

6. Conclusions and Future Work

In this paper, we discuss the user stance detection problem and redefine stance. Therefore, we propose a stance detection model based on commonsense knowledge and surrounding environmental information to find the tweets sent by users regarding a particular event. In the commonsense knowledge module, we have extended the ordinary BiLSTM to better and more effectively integrate emotional commonsense when the sequence is encoded into a vector. Additionally, we added the topic information and the users' neighbors' information into the model. At the same time, we propose an attention mechanism to better integrate history content and social contextual information. Finally, we use a softmax layer for the stance classification of user information. Compared with ordinary tweet-level stance detection, our model can automatically track stances related to topics. In order to verify the effectiveness of our model, we conduct numerous experiments on two Twitter datasets, BREXIT and ELECTION. Experimental verification shows that our model has a significant improvement in accuracy compared with the baseline method. To verify the effectiveness of the modules in the model, we conducted ablation experiments to show that external commonsense knowledge and neighbor information has a specific effect on improving stance detection performance.

In our research, we only discussed a small number of eigenvalues. Although our model is significantly better than several other baseline methods, this method still has much room for improvement on the model. In future work, we will consider more representative feature values and compare their performance. In addition, we will also consider adopting new deep learning methods to improve the performance of stance detection.

Author Contributions: Methodology, software, data curation and writing—original draft, P.J.; supervision, writing—review and editing, funding acquisition, and formal analysis, Y.D.; investigation, and validation, J.H.; formal analysis and software, H.L.; funding acquisition and formal analysis, X.L.; formal analysis, X.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation (Grant Nos. 61872298, 61802316, and 61902324) and the Sichuan Regional Innovation Cooperation Project (Grant No. 2021YFQ008).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon request. All source codes and experiment details are available at https://github.com/PengJ/ECKEI, (accessed on 4 September 2022).

Acknowledgments: The authors would like to thank all editors and anonymous reviewers for their valuable comments and suggestions, which have significantly improved the quality and presentation of this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gong, C.; Du, Y.; Li, X.; Chen, X.; Li, X.; Wang, Y.; Zhou, Q. Structural hole-based approach to control public opinion in a social network. *Eng. Appl. Artif. Intell.* **2020**, *93* , 103690. [CrossRef]
2. Du, Y.; Zhou, Q.; Luo, J.; Li, X.; Hu, J. Detection of key figures in social networks by combining harmonic modularity with community structure-regulated network embedding. *Inf. Sci.* **2021**, *570*, 724–743. [CrossRef]
3. Mohammad, S.; Kiritchenko, S.; Sobhani, P.; Zhu, X.; Cherry, C. SemEval-2016 Task 6: Detecting Stance in Tweets. In Proceedings of the 10th International Workshop on Semantic Evaluation, San Diego, CA, USA, 16–17 June 2016; pp. 31–41. The Association for Computer Linguistics: Stroudsburg, PA, USA, 2016. [CrossRef]
4. Lai, M.; Cignarella, A.T.; Farías, D.I.H.; Bosco, C.; Patti, V.; Rosso, P. Multilingual stance detection in social media political debates. *Comput. Speech Lang.* **2020**, *63*, 101075. [CrossRef]
5. Gorrell, G.; Aker, A.; Bontcheva, K.; Derczynski, L.; Kochkina, E.; Liakata, M.; Zubiaga, A. SemEval-2019 Task 7: RumourEval, Determining Rumour Veracity and Support for Rumours. In Proceedings of the 13th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2019, Minneapolis, MN, USA, 6–7 June 2019; pp. 845–854. [CrossRef]
6. Jannati, R.; Mahendra, R.; Wardhana, C.W.; Adriani, M. Stance Classification Towards Political Figures on Blog Writing. In Proceedings of the International Conference on Asian Language Processing, Bandung, Indonesia, 15–17 November 2018; pp. 96–101. [CrossRef]
7. Veyseh, A.P.B.; Ebrahimi, J.; Dou, D.; Lowd, D. A Temporal Attentional Model for Rumor Stance Classification. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, Singapore, 6–10 November 2017; pp. 2335–2338. [CrossRef]
8. Pamungkas, E.W.; Basile, V.; Patti, V. Stance Classification for Rumour Analysis in Twitter: Exploiting Affective Information and Conversation Structure. *arXiv* **2019**, arXiv:1901.01911.
9. Wang, J.; Wang, Z.; Zhang, D.; Yan, J. Combining Knowledge with Deep Convolutional Neural Networks for Short Text Classification. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, Melbourne, Australia, 19–25 August 2017; pp. 2915–2921. [CrossRef]
10. Ofek, N.; Poria, S.; Rokach, L.; Cambria, E.; Hussain, A.; Shabtai, A. Unsupervised Commonsense Knowledge Enrichment for Domain-Specific Sentiment Analysis. *Cogn. Comput.* **2016**, *8*, 467–477. [CrossRef]
11. Mohammad, S.M.; Sobhani, P.; Kiritchenko, S. Stance and Sentiment in Tweets. *ACM Trans. Internet Techn.* **2017**, *17*, 26:1–26:23. doi: [CrossRef]
12. Elfardy, H.; Diab, M.T. CU-GWU Perspective at SemEval-2016 Task 6: Ideological Stance Detection in Informal Text. In Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, 16-17 June 2016; pp. 434–439. [CrossRef]
13. Dey, K.; Shrivastava, R.; Kaushik, S. Twitter Stance Detection—A Subjectivity and Sentiment Polarity Inspired Two-Phase Approach. In Proceedings of the IEEE International Conference on Data Mining Workshops, ICDM Workshops 2017, New Orleans, LA, USA, 18–21 November 2017; pp. 365–372. [CrossRef]
14. Wojatzki, M.; Zesch, T. ltl.uni-due at SemEval-2016 Task 6: Stance Detection in Social Media Using Stacked Classifiers. In Proceedings of the 10th International Workshop on Semantic Evaluation, San Diego, CA, USA, 16–17 June 2016; pp. 428–433. [CrossRef]
15. Augenstein, I.; Vlachos, A.; Bontcheva, K. USFD at SemEval-2016 Task 6: Any-Target Stance Detection on Twitter with Autoencoders. In Proceedings of the 10th International Workshop on Semantic Evaluation, San Diego, CA, USA, 16–17 June 2016; pp. 389–393. [CrossRef]
16. Dias, M.; Becker, K. INF-UFRGS-OPINION-MINING at SemEval-2016 Task 6: Automatic Generation of a Training Corpus for Unsupervised Identification of Stance in Tweets. In Proceedings of the 10th International Workshop on Semantic Evaluation, San Diego, CA, USA, 16–17 June 2016; pp. 378–383. [CrossRef]
17. Trabelsi, A.; Zaïane, O.R. Unsupervised Model for Topic Viewpoint Discovery in Online Debates Leveraging Author Interactions. In Proceedings of the Twelfth International Conference on Web and Social Media, Stanford, CA, USA, 25–28 June 2018; pp. 425–433.
18. Darwish, K.; Stefanov, P.; Aupetit, M.J.; Nakov, P. Unsupervised User Stance Detection on Twitter. In Proceedings of the Fourteenth International AAAI Conference on Web and Social Media, Held Virtually, Original Venue, Atlanta, GA, USA, 8–11 June 2020; pp. 141–152.
19. Rashed, A.; Kutlu, M.; Darwish, K.; Elsayed, T.; Bayrak, C. Embeddings-Based Clustering for Target Specific Stances: The Case of a Polarized Turkey. In Proceedings of the Fifteenth International AAAI Conference on Web and Social Media, Held Virtually, 7–10 June 2021; pp. 537–548.
20. Zarrella, G.; Marsh, A. MITRE at SemEval-2016 Task 6: Transfer Learning for Stance Detection. In Proceedings of the 10th International Workshop on Semantic Evaluation, San Diego, CA, USA, 16–17 June 2016; pp. 458–463. [CrossRef]
21. Wei, W.; Zhang, X.; Liu, X.; Chen, W.; Wang, T. pkudblab at SemEval Task 6 : A Specific Convolutional Neural Network System for Effective Stance Detection. In Proceedings of the 10th International Workshop on Semantic Evaluation, San Diego, CA, USA, 16–17 June 2016; pp. 384–388. The Association for Computer Linguistics: Stroudsburg, PA, USA, 2016. [CrossRef]
22. Du, J.; Xu, R.; He, Y.; Gui, L. Stance Classification with Target-specific Neural Attention. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, Melbourne, Australia, 19–25 August 2017; pp. 3988–3994. doi: [CrossRef]

23. Zhou, Y.; Cristea, A.I.; Shi, L. Connecting Targets to Tweets: Semantic Attention-Based Model for Target-Specific Stance Detection. In Proceedings of the Web Information Systems Engineering—WISE—18th International Conference, Puschino, Russia, 7–11 October 2017; Volume 10569, pp. 18–32. doi: [CrossRef]
24. Dey, K.; Shrivastava, R.; Kaushik, S. Topical Stance Detection for Twitter: A Two-Phase LSTM Model Using Attention. In Proceedings of the Advances in Information Retrieval—40th European Conference on IR Research, Grenoble, France, 26–29 March 2018; Volume 10772, pp. 529–536. doi: [CrossRef]
25. Sun, Q.; Wang, Z.; Li, S.; Zhu, Q.; Zhou, G. Stance detection via sentiment information and neural network model. *Front. Comput. Sci.* **2019**, *13*, 127–138. [CrossRef]
26. Aldayel, A.; Magdy, W. Your Stance is Exposed! Analysing Possible Factors for Stance Detection on Social Media. *Proc. ACM Hum. Comput. Interact.* **2019**, *3*, 205:1–205:20. [CrossRef]
27. Cignarella, A.T.; Lai, M.; Bosco, C.; Patti, V.; Rosso, P. SardiStance @ EVALITA2020: Overview of the Task on Stance Detection in Italian Tweets. In Proceedings of the Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian, Final Workshop (EVALITA 2020), Online Event, 17 December 2020.
28. Siddiqua, U.A.; Chy, A.N.; Aono, M. Tweet Stance Detection Using an Attention based Neural Ensemble Model. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019; pp. 1868–1873. [CrossRef]
29. Zhang, Y.; Qi, P.; Manning, C.D. Graph Convolution over Pruned Dependency Trees Improves Relation Extraction. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 2205–2215. [CrossRef]
30. Tian, H.; Gao, C.; Xiao, X.; Liu, H.; He, B.; Wu, H.; Wang, H.; Wu, F. SKEP: Sentiment Knowledge Enhanced Pre-training for Sentiment Analysis. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 5–10 July 2020; pp. 4067–4076. [CrossRef]
31. Xu, Z.; Liu, B.; Wang, B.; Sun, C.; Wang, X. Incorporating loose-structured knowledge into conversation modeling via recall-gate LSTM. In Proceedings of the International Joint Conference on Neural Networks, Anchorage, AK, USA, 14–19 May 2017; pp. 3506–3513. [CrossRef]
32. Zhang, B.; Yang, M.; Li, X.; Ye, Y.; Xu, X.; Dai, K. Enhancing Cross-target Stance Detection with Transferable Semantic-Emotion Knowledge. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 3188–3197. [CrossRef]
33. Cambria, E.; Poria, S.; Bajpai, R.; Schuller, B.W. SenticNet 4: A Semantic Resource for Sentiment Analysis Based on Conceptual Primitives. In Proceedings of the 26th International Conference on Computational Linguistics, Osaka, Japan, 11–16 December 2016; pp. 2666–2677.
34. Cambria, E.; Fu, J.; Bisio, F.; Poria, S. AffectiveSpace 2: Enabling Affective Intuition for Concept-Level Sentiment Analysis. In Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, TX, USA, 25–30 January 2015; pp. 508–514.
35. Hinton, G.E.; Osindero, S.; Teh, Y.W. A Fast Learning Algorithm for Deep Belief Nets. *Neural Comput.* **2006**, *18*, 1527–1554. [CrossRef] [PubMed]
36. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. 8.1735. [CrossRef] [PubMed]
37. Graves, A.; Schmidhuber, J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Netw.* **2005**, *18*, 602–610. [CrossRef] [PubMed]
38. Weerakody, P.B.; Wong, K.W.; Wang, G.; Ela, W. A review of irregular time series data handling with gated recurrent neural networks. *Neurocomputing* **2021**, *441*, 161–178. [CrossRef]
39. Ma, Y.; Peng, H.; Cambria, E. Targeted Aspect-Based Sentiment Analysis via Embedding Commonsense Knowledge into an Attentive LSTM. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence(AAAI-18), New Orleans, LA, USA, 2–7 February 2018; pp. 5876–5883.
40. Burns, N.; Bi, Y.; Wang, H.; Anderson, T.J. Enhanced Twofold-LDA Model for Aspect Discovery and Sentiment Classification. *Int. J. Knowl. Based Organ.* **2019**, *9*, 1–20. [CrossRef]
41. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent Dirichlet Allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.
42. Myers, S.A.; Zhu, C.; Leskovec, J. Information diffusion and external influence in networks. In Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Beijing, China, 12–16 August 2012; pp. 33–41. [CrossRef]
43. Yang, Y.; Zhou, D.; He, Y. An Interpretable Neural Network with Topical Information for Relevant Emotion Ranking. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 3423–3432. [CrossRef]
44. Ren, Y.; Zhang, Y.; Zhang, M.; Ji, D. Context-Sensitive Twitter Sentiment Classification Using Neural Network. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; pp. 215–221.
45. Miura, Y.; Taniguchi, M.; Taniguchi, T.; Ohkuma, T. Unifying Text, Metadata, and User Network Representations with a Neural Network for Geolocation Prediction. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, BC, Canada, 30 July–4 August 2017; pp. 1260–1272. [CrossRef]
46. Zhu, L.; He, Y.; Zhou, D. Neural opinion dynamics model for the prediction of user-level stance dynamics. *Inf. Process. Manag.* **2020**, *57*, 102031. [CrossRef]

47. Xue, N. Steven Bird, Evan Klein and Edward Loper. Natural Language Processing with Python. *Nat. Lang. Eng.* **2011**, *17*, 419–424. [CrossRef]