

Document downloaded from:

<http://hdl.handle.net/10251/180304>

This paper must be cited as:

Naskar, D.; Onaindia De La Rivaherrera, E.; Rebollo Pedruelo, M.; Singh, SR. (2020). Predicting Emotion Dynamics Sequence on Twitter via Deep Learning Approach. Association for Computing Machinery (ACM). 20-24.
<https://doi.org/10.1145/3428690.3429156>



The final publication is available at

<https://doi.org/10.1145/3428690.3429156>

Copyright Association for Computing Machinery (ACM)

Additional Information

Predicting Emotion Dynamics Sequence on Twitter via Deep Learning Approach

Debashis Naskar

Computer Technology Group Artificial Intelligence (GTI-IA), Universitat Politècnica de València, Spain
debashis@drtc.isibang.ac.in

Miguel Rebollo

Valencian Research Institute for Artificial Intelligence (VRAIN), Universitat Politècnica de València, Spain
mrebollo@upv.es

Eva Onaindia

Valencian Research Institute for Artificial Intelligence (VRAIN), Universitat Politècnica de València, Spain
onaindia@dsic.upv.es

Sanasam Ranbir Singh

Indian Institute of Technology Guwahati, India
ranbir@iitg.ernet.in

ABSTRACT

Exploring the mechanism about users' emotion dynamics towards social events and further predicting their future emotions have attracted great attention to the researchers. Despite the concreteness of the online expressions in written form, it remains unpredictable which kinds of emotions will be expressed in individual messages of Twitter users influenced by his/her friends. To investigate this, we perform an investigation on observing emotions unfolding in a consecutive sequence of tweets for a particular user based on his/her past history. In this paper, we propose an Emotion-based User Sequential Influence Model (E-USIM) on given a set of tweets related with some events (identified by the usage of a hashtag), determines how those sentiments will be distributed on behalf of a person within a conversation. We then apply the developed model to predict users' future emotions by combing of personal and interpersonal influence.

CCS CONCEPTS

• **Information systems** → **Sentiment analysis**; *Emotion dynamics*; • **Computing methodologies** → Model verification and validation;

KEYWORDS

TESC, EUSIM, RNN, GRU, Emotion dynamics

1 INTRODUCTION

In the age of social network, a famous and sophisticated microblog service Twitter is profoundly used in various tasks. Users through the use of limited vocabulary, are able to express their opinion or views about a particular topic or product. Specially, a famous and sophisticated microblog service Twitter is profoundly used in various tasks. This online expression is applicable for the businessman, stock broker and product seller for predicting customers' future opinion. With the help of these, they can appropriately modify future marketing strategies. The opinion dynamic in social networks has spread in different areas ranging from professional use to product use in everyday life such as movies, serials and politics [11].

The last few years have witnessed a massive growth in research on measuring user influence [13], dynamical influence prediction

[12] and opinion dynamics measure [3] in Twitter. In social platform, users generally follow their neighbors and if any user changes an opinion on a specific topic or product, s(he) is most likely to get influenced by his/her neighbors. Therefore, uncovering user's dynamics opinion from their neighbors and represent the modified one is very important. Based on this study, researchers have proposed several influence models by capturing their long-term historical information and current opinion status [2, 4, 11]. Therefore, predicting users' emotion from their long-term past history is relevant to better guesstimate the actual opinion on the topic or event. It is so important to explore deep analysis for capturing hidden emotion state of the user from their conversation sequence.

In this paper, we are especially interested in modeling emotion influence and predicting evolution of emotion dynamics in Twitter. The purpose of this study is to identify users' emotions which are expressed in their written tweets and attempt to understand the change in their emotions, over time. More specifically, given an incoming sequence of sentiment-labeled observation tweets before/after posting by a user, can we determine dynamics updated emotion of the user? The ultimate purpose is to predict person's emotion and to recognize how a person change his/her emotion under the influence of neighbors' opinions.

The existing works [4, 11] mostly concern about influence models based on neighbors' opinions and their studies only focus on discrete categories of sentiment polarities. But it is also important to know which significant emotion plays a major role to influence the users. Therefore, we focus on the influential role of neighbour emotions' and predict how emotions are dynamically transmitting from one state to another across different topics. To address the gap, we proposed a prediction model called *Emotion-based User Sequential Influence Model* (E-USIM) which capture neighbors' particular emotional information from their long past history. In this model, we adopt Recurrent Neural Network (RNN) architecture and its variant Gated Recurrent Units (GRUs) [5] which has an able to integrate the historical information with the new coming information for prediction. Our proposed model contains a set of primary emotions which have 90% accuracy to classify the emotions [8]. To exhibit the efficacy of the proposed model, we perform the experiments on the collected Twitter datasets on different topics.

Table 1: Size of the real datasets

Event/Topic	Categories	# of Tweets	# of Users	Avg # of Sent Tweets
#BlackMoneyDebate	Policy	616343	1260	4
#Brexit	Policy	686434	2688	6
#GrenfellTower	Accident	136821	2297	3
#SyriaGasAttack	Terrorism	10823	557	3

2 DATA PREPARATION AND EMOTION STATE LABELING

For our purpose of analyzing real-time events, we chronologically retrieved tweets through the Twitter *Search-API* and created our own datasets, as shown in Table 1. For each participating user, we extract and arrange the tweets posted by the user in the order of posting time. To identify their emotions, we employed a psychological model i.e., Russell’s model of affect [7] which presented his categorization in a wheel of emotions. According to this model, every affective experience is defined by *valence* and *arousal* coordinate in the 2D circumplex. A numerical value for valence ranges from 1 (*unpleasant*) to 9 (*pleasant*) and arousal ranges from 1 (*deactivation*) to 9 (*activation*). The core affect map identified sixteen states ($\mathcal{R}_u(i) \in \{1, 2, 3, \dots, 16\}$) and named them as *Alert, Excited, Elated, Happy, Contented, Serene, Relaxed, Calm, Fatigued, Bored, Depressed, Sad, Upset, Stressed, Nervous* and *Tense* [7], as shown in top of the Figure 1.

To determine the emotional state of a tweet using Russell’s circumplex model, we first need to estimate the valence and arousal score of the tweet. To estimate valence and arousal score of a tweet, we use ANEW dictionary of affect [1]. It provides mean and standard deviation scores of valence and arousal for unique words in English. For example, the word *love* has mean and standard deviation value for valence as ($v = [\mu : 8.72, \sigma : 0.7]$) and that of arousal ($a = [\mu : 6.44, \sigma : 3.35]$). We then use the mean points to determine the emotional state in Russell’s circumplex model, i.e., the coordinate (8.72, 6.44) falls in the *Happy* state. In a given tweet, more than one emotional word may be present. We use the formula defined in the equation 1 to determine the overall emotional state of a tweet.

$$X = \frac{\sum_{i=1}^N \frac{\mu_i}{\sigma_i}}{\sum_{i=1}^N \frac{1}{\sigma_i}} \quad (X, \mu, \sigma) \quad (1)$$

where, X is the mean value of valence (similarly, mean value of arousal), μ and σ are the word’s mean and standard deviation value of valence (equivalently for arousal) and the total number of emotional words is N .

3 EMOTION-BASED USER SEQUENTIAL INFLUENCE MODEL

In this section, we describe a sequential influence model by integrating with the temporal emotion state. This model has an ability to solve the dynamics emotion prediction problem by considering the

long history of information. By taking advantage of RNN, we propose our E-USIM architecture which able to predict new upcoming information by amalgamating the long history of opinions.

3.1 Temporal Emotional State Chain

This section describes the formation of a *Temporal Emotional State Chain* (TESC). Every participating user’s data is defined by the sequence of *outgoing* and *incoming* tweets arranged in the order of posting time. Outgoing tweets are those tweets posted by the target user about the topic/event under consideration. Whereas, the incoming tweets are those tweets posted by neighbours about target topic, and are received by the user through one of the following: hashtag-tweet (H), mention (M), reply (R), retweet (RT), following-list (FL), member-list (ML).

In E-USIM, we have formulated our problem on the basis of temporal conversation and different emotion states of the user. Given a direct graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where each vertex $u \in \mathcal{V}$ represents a user and each edge $(u, v) \in \mathcal{E}$ represents a user u following to another user v . The each user $u \in \mathcal{V}$ post a tweet message m with sentiment S at time t . Taken into account \mathcal{G} , we considered a collection of all external set for each user $u \in \mathcal{V}$, which denoted as a $\mathcal{Z}_u = \{v | (u, v) \in \mathcal{E}\}$. The total number of users is \mathcal{N} and size of the neighbor set \mathcal{Z}_u is $n(u)$.

Given a user u , we estimate the incoming tweets posted by v that the user u has received between two consecutive tweets (i.e. m^1 and m^2). The time of the two consecutive tweets sent by a user u is referred to as t_0 and t_1 . Additionally, we also include the incoming tweets posted by v before posting u ’s first tweet at time t_0 . Given a user u and a topic $\#h$, as we construct a typical temporal tweet chain or tuple chain of u ’s posted a message at each timestamp $t_u(i)$, where \downarrow denotes (incoming) tweets and \uparrow denotes the (outgoing) tweets.

$$\langle u, \#h \rangle \rightarrow \langle \dots, \downarrow m_{t_{0-1}}^{Cin}, \uparrow m_{t_0}^{Cout}, \langle \downarrow m_{t_{0+1}}^{Cin}, \dots \rangle, \uparrow m_{t_1}^{Cout}, \langle m_{t_{1+1}}^{Cin}, \dots \rangle, \uparrow m_{t_2}^{Cout}, \dots \rangle$$

When the user u posts his first tweet at t_0 on topic $\#h$, public discussion on the topic $\#h$ might have already taken place. It is denoted by the tuple $\downarrow m_{t_{0-i}}^{Cin}$, $i = 1, 2, \dots$ and $Cin/out \in \{H, RT, R, M, FL, ML\}$. Similarly, incoming tweets between the user’s tweet $\uparrow m_{t_k}^{Cout}$ and $\uparrow m_{t_{k+1}}^{Cout}$, is denoted by the tuple $\langle \downarrow m_{t_{k+0}}^{Cin}, \downarrow m_{t_{k+1}}^{Cin}, \dots \rangle$.

The emotional state of a tweet in a *temporal tweet chain* is determined using Russell’s circumplex model of affect as described in Section 2. If \mathcal{R}_i , $i \in 1, 2, 3, \dots, 16$ denotes one of the sixteen emotional regions for a given tweet, the above *temporal tweet chain* can be transformed into the following *temporal emotional state chain*.

$$\langle u, \#h \rangle \rightarrow \langle \dots, \downarrow \mathcal{R}_{i,t_{0-1}}^{Cin}, \uparrow \mathcal{R}_{i,t_0}^{Cout}, \langle \downarrow \mathcal{R}_{i,t_{0+1}}^{Cin}, \dots \rangle, \uparrow \mathcal{R}_{i,t_1}^{Cout}, \langle \mathcal{R}_{i,t_{1+1}}^{Cin}, \dots \rangle, \uparrow \mathcal{R}_{i,t_2}^{Cout}, \dots \rangle$$

All the incoming set of emotion tweets from neighbor are denoted by the tuple $\langle \downarrow \mathcal{R}_{t_{k+0}}^{Cin}, \downarrow \mathcal{R}_{t_{k+1}}^{Cin}, \dots \rangle$ and fall between the user’s personal emotion tweets $\uparrow \mathcal{R}_{t_k}^{Cout}$ and $\uparrow \mathcal{R}_{t_{k+1}}^{Cout}$.

Finally, our assumptions of user’s emotion is based on past information $s(\text{he})$ receives and his/her personal emotions. Therefore, our objective is to predict u ’s future emotion state $\mathcal{S}_u(i)$ at the next timestamp t_u based on personal emotion sequence $\uparrow \mathcal{P}\mathcal{R}_{u,i}^{Cout}(t) = \langle \uparrow \mathcal{R}_{i,t_0}^{Cout}, \dots, \uparrow \mathcal{R}_{i,t_k}^{Cout} \rangle$ sent by the user u at t and all

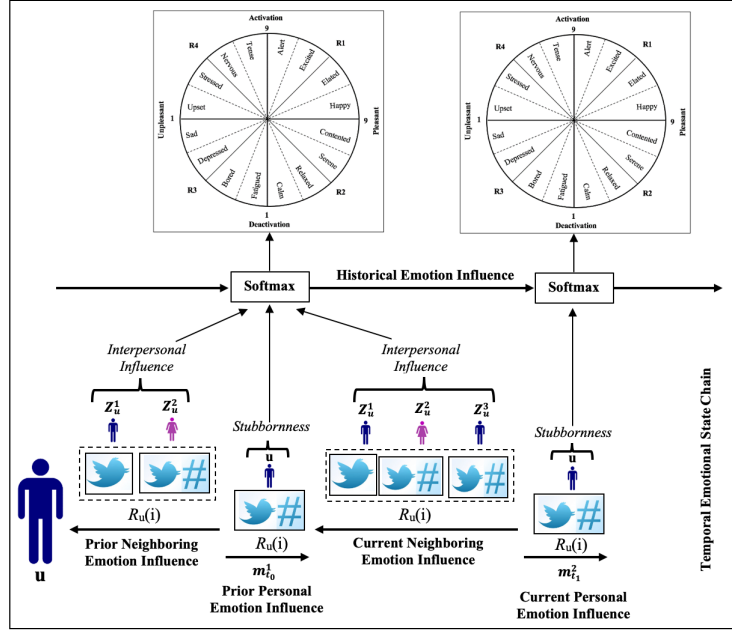


Figure 1: Emotion-based User Sequential Influence Model

the neighboring emotion of the messages $\downarrow NR_{u,i}^{in}(t) = \langle \downarrow \mathcal{R}_{i,t_0-1}^{Cin}, \dots, \downarrow \mathcal{R}_{i,t_k}^{Cin} \rangle$ received from $t-1$ to t . An example is also shown in Fig 1. In all the experimental analysis reported in the subsequent section, we use the above temporal emotion sequential influence model for each user.

3.2 Sequential Influence Model of Emotion Dynamics

In this paper, we proposed a novel sequential influence model based on recurrent neural network which has the ability to solve the emotion prediction problem by capturing four important influence factors, i.e., (1) prior neighboring emotion influence, (2) prior personal emotion influence, (3) current neighboring emotion influence, and (4) historical emotion influence.

3.2.1 Prior Neighboring Emotion Influence. The user u may update his/her emotion based on previous opinion s(he) received. When the user u posts his/her first tweet at t_0 on topic $\#h$, s(he) might have received a tweet message from the neighbor $\mathcal{Z}_u(i)$ on the similar topic at time t_{0-1} . We use $\alpha_{u,v}$ to denote as a degree influence which consider as a user's incoming emotional influence $\downarrow \mathcal{R}_{i,t_0-1}^{Cin}$ by his/her neighbor. The \tanh function helps to limit the strength of the interpersonal influence between -1 and $+1$. The prior neighboring emotion vector $\mathcal{PN}_{u,i}(t)$ is obtained before a user u sending his/her first message and the influence based on average of all prior neighbor's emotion in the set $\downarrow NR_{u,i}^{in}(t)$ at time t_{0-1} as:

$$\mathcal{PN}_{u,i}(t) = \sum_{v=1}^{n(u)} \tanh(\alpha_{u,v}) \sum \downarrow \mathcal{R}_{i,t_0-1}^{Cin} \quad (2)$$

3.2.2 Prior Personal Emotion Influence. The user u posts a personal message at each timestamp $t_u(i)$ and modify his/her emotion on the particular event. We denote $\alpha_{u,0}$ as a degree of u 's stubbornness on his/her personal post and \tanh functions helps to limit the strength of stubbornness between -1 and $+1$. The prior personal emotion vector $\mathcal{PP}_{u,i}(t)$ of u is obtained by his/her personal outgoing post $\uparrow \mathcal{PR}_{u,i}^{out}(t)$ at time t_0 . Thus, it denoted as:

$$\mathcal{PP}_{u,i}(t) = \tanh(\alpha_{u,0}) \sum \uparrow \mathcal{R}_{i,t_0}^{Cout} \quad (3)$$

3.2.3 Current Neighboring Emotion Influence. The user u updates his/her emotion based on the incoming tweet that s(he) received between two outgoing tweets. It means a user u may received a tweets at time t_{0+1} from the neighbor $\mathcal{Z}_u(i)$ after post his/her first tweet at time t_0 on topic $\#h$. The emotional information received from neighbor at $t_u(i)$ is a forms of the neighbor set $\downarrow NR_{u,i}^{in}(t)$. The current neighboring emotion vector $\mathcal{CN}_{u,i}(t)$ is obtained based on average of all neighbor's emotion in the set $\downarrow NR_{u,i}^{in}(t)$ between two time frame, i.e., t_0 and t_1 which devised as follow:

$$\mathcal{CN}_{u,i}(t) = \sum_{v=1}^{n(u)} \tanh(\alpha_{u,v}) \sum \downarrow \mathcal{R}_{i,t_{0+1}}^{Cin} \quad (4)$$

3.2.4 Historical Emotion Influence. The purpose of this section is to appropriately embed the entire history of emotion influence. The user u updates his/her emotion by considering all the past history. The historical emotion influence $\mathcal{HI}_{u,i}(t)$ represent as a function f at time t_u by including personal emotion sequence $\uparrow \mathcal{PR}_{u,i}^{out}(t-1)$ sent and all the neighboring emotion sequence $\downarrow NR_{u,i}^{in}(t-1)$ of the messages received by the user u . Thus, it is denoted as follows:

$$\mathcal{HI}_{u,i}(t) = f \left(\downarrow NR_{u,i}^{in}(t-1), \uparrow \mathcal{PR}_{u,i}^{out}(t-1) \right) \quad (5)$$

3.2.5 Recurrent Neural Network. For the purpose of updating user’s emotion, we proposed a sequential influence model of emotion dynamics based on *Recurrent Neural Network* (RNN). The RNN is combined with prior neighboring $\mathcal{PN}_{u,i}(t)$, prior personal $\mathcal{PP}_{u,i}(t)$, current neighboring $\mathcal{CN}_{u,i}(t)$ and historical emotion influence $\mathcal{HI}_{u,i}(t)$. To proposed the sequential influence model, we already taken place personal and neighboring emotion influence, i.e., derived by $\mathcal{X}_{u,i}(t) \in \langle \mathcal{PN}_{u,i}(t), \mathcal{PP}_{u,i}(t), \mathcal{CN}_{u,i}(t) \rangle$. To update u ’s next emotion at $t_u(i+1)$, the historical emotion influence $\mathcal{HI}_{u,i}(t-1)$ could be replaced by the internal emotion state $C_{u,i}(t-1)$. Thus, it is derived as follow:

$$C_{u,i}(t) = \tanh(\mathcal{W}_a [C_{u,i}(t-1), \mathcal{X}_{u,i}(t)] + b_a) \quad (6)$$

where, $\mathcal{W}_a \in \mathbb{R}^{d_C}$, and $b_a \in \mathbb{R}^d$, and d_C is the dimension of emotion state region.

3.2.6 Gated Recurrent Unit. In our sequential influence model, we acknowledge the GRU to prove more appropriate than the simple RNN model. This gated recurrent unit is more effective and compact to drop irrelevant information and update the relevant on with affordable computation cost. This unit is formally expressed as follows:

$$[r_{u,i}(t) = \sigma(\mathcal{W}_r [C_{u,i}(t-1), \mathcal{X}_{u,i}(t)] + b_r) \quad (7)$$

$$[g_{u,i}(t) = \sigma(\mathcal{W}_g [C_{u,i}(t-1), \mathcal{X}_{u,i}(t)] + b_g) \quad (8)$$

$$\tilde{C}_{u,i}(t) = \tanh(\mathcal{W}_c [[r_{u,i}(t) \cdot C_{u,i}(t-1), \mathcal{X}_{u,i}(t)] + b_c) \quad (9)$$

$$C_{u,i}(t) = [g_{u,i}(t) \cdot \tilde{C}_{u,i}(t) + (1 - [g_{u,i}(t)] \cdot C_{u,i}(t-1) \quad (10)$$

where, $\mathcal{W}_r, \mathcal{W}_g, \mathcal{W}_c \in \mathbb{R}^{d_C}$, and $b_r, b_g, b_c \in \mathbb{R}^d$. The reset gate $[r_{u,i}(t)$ determines how much past information forget, i.e., not relevant for the future. The update gate $[g_{u,i}(t)$ helps the model to determine how much of the past information needs to process for the future. These are two important gates which decide what information should be passed to the output and its’ control long-term and short-term memory accordingly. The current memory content gate $\tilde{C}_{u,i}(t)$ which will use the reset gate to store the relevant information from the past. The final gate used to calculate the internal emotion state $C_{u,i}(t)$ which holds information for the current unit and passes it down to the network. In order to do that the update gate is needed. It determines what to collect from the current memory content $\tilde{C}_{u,i}(t)$ and what from the previous steps $C_{u,i}(t-1)$, as mention on equation 10.

3.3 Emotion-based Prediction

For the purpose of predicting u ’s future emotion state $S_u(i)$, we formulate the problem as a predicting probability distribution by given internal emotion state $C_{u,i}(t)$ and the output layer for the E-USIM is a softmax function over all categories of Russell’s model of affect [7]. The graphical representation of E-USIM is shown in Figure 1.

$$\mathbb{P}(S_u(i)|C_{u,i}(t)) = \text{Softmax}(C_{u,i}(t) + b) \quad (11)$$

Where, softmax function, i.e., $\text{Softmax}(\Theta^{(i)}) = \exp(\Theta^{(i)}) / \sum_{j=1}^{\mathcal{R}_k} \exp \Theta^{(i)}$ and $b \in \mathbb{R}^{\mathcal{R}}$. \mathcal{R}_k denotes the sixteen emotion regions.

3.4 Parameter Estimation

To train our model, we utilise the gradients by using the Back Propagation Through Time (BPTT) approach [14] and maximize the log-likelihood of the sequences of emotions. To implement this algorithm, we use Tensor-Flow¹ open-source software library with a few lines of code.

$$\mathcal{L} = \sum_{u=1}^N \sum_{i=1}^{\mathcal{R}_u} \log \mathbb{P}(S_u(i) | \downarrow \mathcal{NR}_{u,i}^{in}(t), \uparrow \mathcal{PR}_{u,i}^{out}(t)) \quad (12)$$

4 EXPERIMENTAL EVALUATION

4.1 Dataset & Baselines

For the purpose of our study, we collected our datasets with the *Search-API* related to four different events through the hashtags #BlackMoneyDebate, #Brexit, #GrenfellTower and #SyriaGasAttack respectively. It consists of 6.80K users over a total of 14.50M tweets. From the raw datasets, we filtered out users who sent 0 or only one message and tweets that did not show any emotion (null values of valence and arousal). Table 1 shown the final figures after filtering. For validation of our methods, we compare it with the three baseline methods, i.e. DeGroot model [6], Flocking model [9] and Voter model [10].

4.2 Evaluation Metrics on Emotions Prediction

Given the sequence of u ’s emotion, we split the data into training dataset and test dataset according to the posting time. The training dataset is made by emotional sequence from the previous steps $C_{u,i}(t-1)$ for each user $u_i \in U$. Since the length of emotional sequences are not the equal, therefore we predict the last opinion for each user u after learning from the training set.

The emotion prediction performance for all methods are evaluated after optimizing from $\mathbb{P}(S_u(i)|C_{u,i}(t))$ to the sequence of emotion state $S_u(i)$ of the users. To evaluate the performance of our proposed method on a different topic, we have used two different measures of error. One is MSE (Mean Square Error):

$$MSE = \frac{\sum_{t=1}^N (\mathcal{R}_u(i) - S_u(i))^2}{N} \quad (13)$$

Another one is MAPE (Mean Absolute Percentage Error):

$$MAPE = \frac{\sum_{t=1}^N \left| \frac{\mathcal{R}_u(i) - S_u(i)}{\mathcal{R}_u(i)} \right|}{N} \times 100 \quad (14)$$

where for both measures, $\mathcal{R}_u(i)$ is the actual value, $S_u(i)$ is the predicted value and N is the number of users on each emotional sequence.

Table 2 represents a comparative analysis of the prediction error (MSE and MAPE) of three state-of-the-art algorithms along with our proposed method.

Compare with other baseline methods, the prediction performance of DeGroot is consistently good for each dataset. But during the training phase, this model was iterated multiple times to converge and predict the emotion. Flocking is comparatively better

¹<https://www.tensorflow.org/>

Table 2: Performance metrics of emotions prediction for each dataset. The first each column of the each dataset is forecasting error in terms of MSE and second one is MAPE with percentage.

Topic	#BlackMoneyDebate		#Brexit		#GrenfellTower		#SyriaGasAttack	
	MSE	MAPE(%)	MSE	MAPE(%)	MSE	MAPE(%)	MSE	MAPE(%)
DeGroot	1.3412	3.61%	2.5112	2.80%	1.1160	2.20%	4.0420	7.34%
Flocking	1.8551	4.58%	3.3331	3.02%	1.2751	2.84%	4.0782	7.91%
Voter	1.9321	5.37%	4.3421	3.21%	1.6342	3.92%	6.0554	8.83%
E-USIM	0.0521	1.32%	0.0415	0.94%	0.0513	1.44%	0.0811	2.42%

than the voter model and this model updates the emotion of a user by calculating the average value of his/her neighbours. If any user does not have any neighbours, then this model makes difficult to predict the right emotion of that user. The performance of voter model is not much impressive among all methods. Since this model update user’s emotion randomly, thus it cannot judge the actual emotional sequence for prediction. Thus, the performance of this model is unsatisfactory. $E - USIM$ performs consistently much better than all baseline methods over each dataset. If we look in an individual topic, $E - USIM$ method for #Brexit perform well with minimum error (i.e. 0.0415 MSE and 0.94% of MAPE). Most importantly, $E - USIM$ properly captures all the past history of emotion influence for each user. As a result, it provides better result in terms of forecasting emotion even at a distant future. Due insufficient historical information over the network, it’s difficult to learn the actual influence for emotion prediction. Therefore, prediction performance may even harm the results for the topic with small dataset.

5 CONCLUSION

In this paper, we propose a Emotion-based User Sequential Influence Model (E-USIM) to incorporate with the historical information for predicting emotion dynamics from single-hashtag Twitter conversations. To devise such a model, we adopt the standard recurrent neural network architecture which integrate the historical opinion information with the new coming information for prediction. To train our model, we compute Back Propagation Through Time (BPTT) approach and maximize the log-likelihood of the sequences of emotions. To identify specific emotion of the user instead of measure only polarity, we employed a psychological model i.e. Russell’s model of affect which presented his categorization in a wheel of emotions. Based on the current work, we examine users’ expressions by considering different emotional influence factors. As a result, our approach outperforms the existing approaches in terms of both Mean Square Error (MSE) as well as Mean Absolute Percentage Error (MAPE).

6 ACKNOWLEDGEMENTS

This work is partially supported by the Spanish MINECO project TIN2017-88476-C2-1-R.

REFERENCES

- [1] MM Bradley and PJ Lang. 2010. Affective norms for English Words (ANEW): Affective ratings of words and instruction manual. *University of Florida, Gainesville, FL Technical Report C-2* (2010).
- [2] Chengyao Chen, Zhitao Wang, Yu Lei, and Wenjie Li. 2016. Content-based influence modeling for opinion behavior prediction. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. 2207–2216.
- [3] Chengyao Chen, Zhitao Wang, and Wenjie Li. 2017. Modeling Opinion Influence with User Dual Identity. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, 2019–2022.
- [4] Chengyao Chen, Zhitao Wang, and Wenjie Li. 2018. Tracking Dynamics of Opinion Behaviors with a Content-based Sequential Opinion Influence Model. *IEEE Transactions on Affective Computing* (2018).
- [5] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).
- [6] Morris H DeGroot. 1974. Reaching a consensus. *J. Amer. Statist. Assoc.* 69, 345 (1974), 118–121.
- [7] Lisa Feldman Barrett and James A Russell. 1998. Independence and bipolarity in the structure of current affect. *Journal of personality and social psychology* 74, 4 (1998), 967.
- [8] Maryam Hasan, Emmanuel Agu, and Elke Rundensteiner. 2014. Using hashtags as labels for supervised learning of emotions in twitter messages. In *ACM SIGKDD Workshop on Health Informatics, New York, USA*.
- [9] Rainer Hegselmann, Ulrich Krause, et al. 2002. Opinion dynamics and bounded confidence models, analysis, and simulation. *Journal of artificial societies and social simulation* 5, 3 (2002).
- [10] Richard A Holley and Thomas M Liggett. 1975. Ergodic theorems for weakly interacting infinite systems and the voter model. *The annals of probability* (1975), 643–663.
- [11] Bhushan Kulkarni, Sumit Agarwal, Abir De, Sourangshu Bhattacharya, and Niloy Ganguly. 2017. SLANT+: A nonlinear model for opinion dynamics in social networks. In *Data Mining (ICDM), 2017 IEEE International Conference on*. IEEE, 931–936.
- [12] Wei Pan, Wen Dong, Manuel Cebrian, Taemie Kim, James H Fowler, and Alex Sandy Pentland. 2012. Modeling dynamical influence in human interaction: Using data to make better inferences about influence within social systems. *IEEE Signal Processing Magazine* 29, 2 (2012), 77–86.
- [13] Fabián Riquelme and Pablo González-Cantergiani. 2016. Measuring user influence on Twitter: A survey. *Information Processing & Management* 52, 5 (2016), 949–975.
- [14] Paul J Werbos. 1990. Backpropagation through time: what it does and how to do it. *Proc. IEEE* 78, 10 (1990), 1550–1560.