

A Personalized Sentiment Model with Textual and Contextual Information

Siwen Guo

Sviatlana Höhn

Christoph Schommer

ILIAS Research Lab, CSC,
University of Luxembourg

{siwen.guo, sviatlana.hoehn, christoph.schommer}@uni.lu

Abstract

In this paper, we look beyond the traditional population-level sentiment modeling and consider the individuality in a person’s expressions by discovering both textual and contextual information. In particular, we construct a hierarchical neural network that leverages valuable information from a person’s past expressions, and offer a better understanding of the sentiment from the expresser’s perspective. Additionally, we investigate how a person’s sentiment changes over time so that recent incidents or opinions may have more effect on the person’s current sentiment than the old ones. Psychological studies have also shown that individual variation exists in how easily people change their sentiments. In order to model such traits, we develop a modified attention mechanism with Hawkes process applied on top of a recurrent network for a user-specific design. Implemented with automatically labeled Twitter data, the proposed model has shown positive results employing different input formulations for representing the concerned information.

1 Introduction

Sentiment is one of the key factors affecting human behavior. Studying the way in which sentiment is perceived, evolved and expressed is an essential part in artificial intelligence. To analyze sentiment in text, researchers have made different assumptions on linguistic behaviors that are leveraged with approaches developed based on the nature of the text, the representation of the related information and the objectives. However, majority of the studies are conducted at the population-level which assumes that people follow a common understanding with regard to the use of language. Such approaches can be inaccurate in the cases where people use the same lexical choices to convey different messages or vice versa.

Harris (2006) stated that ‘no two alike’ showing the inherent difference in human that motivates the research of personalized sentiment analysis. Grounded in the psychological works, we argue that it is significant to study the effect of individuality in the expressions and to investigate the possibility of providing a deeper understanding of the expressions from the writers’ own perspectives. In this work, we concern the use of preferred lexical choices when expressing sentiment (Reiter and Sripada, 2002) and the level of consistency in retaining a sentiment (Janis and Field, 1956).

Besides the targeted text message itself, we exploit two types of contextual information for the purpose of realizing the psychological aspects: a person’s expressions in the past and the time when the expressions were made. With the goal of discovering the effect of the contextual information, distinct formulation methods are proposed to integrate the information in the personalized sentiment model. The backbone model is a hierarchical neural network which follows a conventional embedding – recurrent – attention structure with each part rectified for the task. The embedding block is used to generate representations for the used information; the recurrent network fulfills the task of relating to the information from the past; the attention model is shaped with Hawkes process (Laub et al., 2015) in order to model the information decay for each expresser. Generally, recurrent networks consider the order of the elements in a sequence but omit the different gaps between them. Hawkes process is utilized to compensate this issue. Furthermore, a novel approach with a user – factor transformation is employed to merge the Hawkes process within the attention model and to construct user-specific processes. For evaluation, we take the data from a number of frequent users on social platforms where Twitter is used as an example. The data is domain-independent, and

it is possible to obtain self-labeled texts that aligns with our goal of understanding the expressers’ perspectives. Significant improvements are seen with certain input formulations, and different Hawkes processes applied for the users are visualized. In the end, we conclude that it is effective to introduce the contextual information to the model.

2 Related Work

Individualities are mostly considered in sentiment analysis when analyzing product-review texts (Gong et al., 2016; Chen et al., 2016b; Wu et al., 2018). A common issue that challenges the research of this area is data sparsity. It is infeasible to build or train an effective model for each user. Gong et al. (2016) address this issue by relating to a global model that captures ‘social norms’, and individualities are included by adapting from the global model via a series of linear transformations. While in the works that apply neural networks, the user information is embedded separately (Chen et al., 2016b) or added at the attention layer (Chen et al., 2016a; Wu et al., 2018) in order to make user-specific predictions at the output. In other works, the aspect of personalization is relaxed to provide user-group based predictions (Gong et al., 2017). The aforementioned approaches are modeled with domain-dependent text, while our concentration on the text associated with various topics makes the task more challenging. Wu and Huang (2016) focus on microblog posts as well and apply the same concept of using a global model and an individual model via multi-task learning as in Gong et al. (2016). In addition, users’ social relations are leveraged to enhance the individual models. Similarly, followers’ and followees’ information is also used in Song et al. (2015) while a variant of latent factor model is utilized. Most of the studies leverage earlier posts from users in order to better understand the individuality; however the evolvement of the sentiment is largely neglected — the preferences of users are considered constant. In this work, we take the user dynamics into consideration, and incorporate user information both in the input and in the Hawkes process to deal with the data sparsity and to offer personalized analysis.

Technically, there are very few works that investigate the different gaps between the input nodes in a recurrent neural network. Neil et al. (2016) have proposed a phased LSTM that utilizes an addi-

tional time gate to control the passing of the information. However, the time gate is triggered by periodic oscillations while modeling sensory events, which makes such a design less flexible when the time gaps are highly various. As an alternative, we explicitly add the time gaps in the Hawkes process to offer a time-sensitive modeling.

In our previous works, we have evaluated the effectiveness of considering individual differences in sentiment analysis by employing a concept-based representation and the static or universal Hawkes process (Guo et al., 2018, 2019a). In this paper, we advance the development by adopting five input formulations with different combinations of granular levels, and propose a refined model with user-specific Hawkes process to constitute a step forward in capturing the nuances of user dynamics and providing insights in the personalized modeling.

3 Personalized Sentiment Model

Motivated by the diversity in individuality, we introduce a model that considers both textual and contextual information and applies hierarchical neural networks to facilitate the aspect of personalization in sentiment analysis. Particularly, we focus on analyzing the effect of contextual information and discover ways to embed such information in the prediction process.

3.1 Model Structure

The personalized sentiment model follows a conventional embedding – recurrent – attention structure as shown in Figure 1 with modifications applied at each block. First, a formulation method is applied in order to represent the current and a number of earlier posts of a user. The embedding

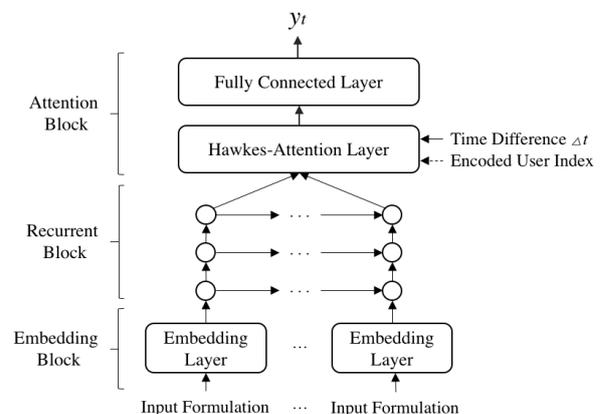


Figure 1: The structure of the sentiment model.

layer takes the formulated inputs and produces a vector for each time step at the recurrent layer. After that, the outputs of the recurrent layer together with the auxiliary time differences are fed to the attention block. Encoded user index is used in the Hawkes-attention layer when different settings of Hawkes process are considered for different users. A fully connected layer is applied afterwards to regularize the output of the Hawkes-attention layer. Finally, the output of the model y_t is generated which is the predicted sentiment label of the target text.

3.2 Textual and Contextual Information

In linguistic studies, the notion of ‘context’ varies from theory to theory that some in monologism see it as ‘secondary complications’, whereas in dialogist theory, it considers the reflexive relation between an expression and its setting or occasion essential (Linell, 2009). In this work, we target social text and regard context as an important factor in the setting of social platforms. Based on the characteristics of such text and the applicability in modeling, we categorize the information used in the sentiment model into two genres: **Textual Information** — the information that can be extracted directly from the *target* text, and **Contextual Information** — the information that is *not* present in the target text but is *associated* with the text according to the user index.

Textual Information

Text is the central part in sentiment analysis. Researchers in this area have proposed various approaches aiming to provide a deep text understanding given the complex nature of how people express their sentiments in text. Moreover, methods designed at the population-level for many other natural language processing tasks can also be used for sentiment analysis. Generally, such methods start at a pre-defined granular level and generate a representation for the text by capturing related information in each granule and the ones surrounding it. In the end, the text is represented explicitly (e.g., concepts as in Poria et al., 2014) and / or implicitly (e.g., embeddings as in Pennington et al., 2014 and Peters et al., 2018).

Contextual Information

Besides the *target* text itself, other types of information have been used to support the understanding of the sentiment as well.

Earlier Posts correspond to the texts produced by the same user in the past. The use of earlier posts leverages the assumption that a person may have similar lexical choices when expressing opinions about related topics while different individuals share different preferences in this regard. By analyzing the lexical choices and the topics or entities associated with them, the tendency of repeating such patterns in a user’s text in the future can be beneficial to the prediction.

Timestamp corresponds to the creation time of the text. It has been shown that there exists a certain level of consistency in an individual’s opinions and such consistency varies from one individual to another (Janis and Field, 1956). We study such a trait by taking the timestamp of each earlier post and applying Hawkes process to observe how the effect of the information on a user’s behaviors or opinions decays over time. Note that here, we do not distinguish the inconsistency between the time when the expression was made and the time when the sentiment was felt.

3.3 Input Formulations

We employ different formulations for the input sequence based on the representation method. For all the formulations, timestamps are apart from other information and are used as an auxiliary input directly at the attention layer with Hawkes process. Additionally, user index is used as a feature in the input in order to handle the issue of data sparsity, and by doing that, the model is able to analyze textual and contextual relations targeting a specific user. The encoded user index is also used at the Hawkes-attention layer when considering individual differences in information decay.

Atomic Representation (AR)

In this formulation, four types of components are extracted: concepts, entities, negations and user index. Concepts are extracted based on Cambria et al. (2018) which contain conceptual and affective information, and can be seen as the ‘signal terms’ regarding lexical choices. Entities are extracted based on grammatical rules as the ‘targets’ of a user’s lexical choice. Negations are extracted based on the lexicon by Reitan et al. (2015) for their ability to invert the orientation of a sentiment. User index is extracted for its role in personalization. After extracting the components, an embedding layer is applied to generate a representation vector for the text.

Representation with Pre-trained Word Embeddings (WE)

Pre-trained word vectors such as GloVe (Pennington et al., 2014) and Word2Vec (Mikolov et al., 2013), generate embeddings according to the co-occurrences of the words. The word embeddings are aggregated dimension-wise to produce a vector for each post. The user index is encoded by itself and then combined with the representation of the post at each time point of an input sequence.

Representation with Concepts and Words (CW)

Since the pre-trained word vectors do not consider the contexts of the target words, we combine the representations of both words and concepts in this formulation. The word embeddings are taken as the same as the one in the WE formulation. The concepts appeared in the text are encoded together with the associated user index so that the relation between the user and the use of concepts can be learned. Afterwards, the two types of representation of the same post are concatenated to generate an input sequence for the recurrent layer.

Representation with Deep Contextualization (DC)

Peters et al. (2018) proposed a deep contextualized representation (ELMo) that takes a finer granular level (characters) to generate embeddings for the text by leveraging a deep bidirectional language model. Prominent results are shown across a number of linguistic tasks using this representation. We apply ELMo for each post, and then combine the representation of the post and the encoded user index at each time point.

Representation with Combined Granular Levels (Combi)

This formulation combines three granular levels, namely character-level (DC), word-level and concept-level (CW). The representations are embedded separately and are concatenated afterwards as mentioned in Peters et al. (2018).

3.4 Recurrent Neural Network with Input Selection from Post History

We apply a deep recurrent neural network with long short-term memory (LSTM, Hochreiter and Schmidhuber, 1997) on the input sequences constructed with one of the input formulations. Each

input sequence X_i consists of an entry of the current post at the end of the sequence (which contains textual information and the encoded user index) and a number of earlier posts by the same user (contextual information), i.e.,

$$X_i = [H_{i-n}, \dots, H_{i-2}, H_{i-1}, F_a(x_i)] \quad (1)$$

where

$$H_j = \begin{cases} F_a(x_j) & \text{if } u(x_j) = u(x_i) \\ 0 & \text{else} \end{cases} ,$$

n is the number of earlier posts considered, F_a is the formulation chosen beforehand, and u is the user index of the post.

Additionally, a selection procedure followed Guo et al. (2019b), is performed for choosing the earlier post x_j of a target text x_i from user u . The use of this procedure is motivated by the large difference in user frequency (the number of posts of a user in a given corpus), as well as the observation of the case where the recent posts are unrelated to the current one while the related posts have appeared long before. The selection is done by calculating the similarity between the topics of each earlier post and the target text. Given a fixed number of time steps T in a recurrent network and a similarity threshold δ , the recent T earlier posts that have a similarity score larger or equal to δ are chosen. For the case where the number of chosen posts is smaller than T , other earlier posts are added in the sequence as complements prioritizing on the recent ones. After the selection, the posts in each sequence are ordered by time.

3.5 Hawkes-Attention Layer

A modified attention mechanism is applied on top of the recurrent neural network. Attention model has the ability to provide more flexibilities at the output layer (Bahdanau et al., 2015). The network can ‘attend’ to different histories based on the immediate situation. As in Yang et al. (2016), the model is defined as follows:

$$u_i = \tanh(W_t h_i + b_t) \quad (2)$$

$$\alpha_i = \frac{\exp(u_i^\top u_t)}{\sum_i \exp(u_i^\top u_t)} \quad (3)$$

$$\lambda_i = \alpha_i h_i \quad (4)$$

$$v = \sum_i \lambda_i \quad (5)$$

where h_i is the i -th output of the recurrent network, u_i is the hidden representation of h_i , and u_t is the ‘context vector’. Here, we randomly initialize the context vector which is later jointly learned with other weights during the training phase. λ_i is the representation of the information learned at time step i . Lastly, v summarizes all the information of the posts from the corresponding sequence. However, conventional recurrent networks and attention models do not differentiate the relations between time steps regarding various time intervals. To model this difference, we shape the representation of the post λ_i with Hawkes process before summarizing them at the last step (Equation 5) in the attention mechanism.

Universal Hawkes Process

Hawkes process is known for modeling the excitation and decay of information over time. When using exponential decay as the excitation function, the Hawkes conditional intensity is (Laub et al., 2015):

$$\lambda^*(t) = \lambda + \sum_{t_i < t} \alpha e^{-\beta(t-t_i)} \quad (6)$$

where λ describes the positive background intensity, t is the current time and t_i is the time when the past event happened. α and β are the most important factors in the Hawkes process that the former corresponds to the amount of excitement the past event brought to the system while the latter corresponds to the decay rate of the excitement. Taking the same concept as in Guo et al. (2019a), we see a post in the past as an ‘event’ that can influence the decision in the future and such influence decays over time. Instead of treating all the past events equally, we use λ_i in Equation 4 as the background intensity and $\alpha = \epsilon \lambda'_i$ as the amount of excitement the post at time step i contributes to the current decision. Note that $\lambda'_i = \max(\lambda_i, 0)$ for we do not consider negative effect from the past. With this modification, the information decay of a past event can also depend on the relativeness between the past and the current events, and ϵ can be seen as a scaler to balance the importance of adding the process. As a result, Equation 5 is replaced with the following:

$$v' = \sum_{i: \Delta t_i \geq 0} (\lambda_i + \epsilon \lambda'_i e^{-\beta \Delta t_i}) \quad (7)$$

where Δt_i indicates the time gap between the earlier post at time step i and the current post.

The current post is included in the summarization when $\Delta t_i = 0$. ϵ and β are learned jointly with other learnable parameters during the training phase.

User-specific Hawkes Process

In order to build user-specific Hawkes process, we compute the values of ϵ and β in Equation 7 for each user by applying learned transformation vectors on the encoded user index. In this way, different behaviors concerning the information decay can be analyzed. That is, ϵ and β are calculated as

$$\epsilon = a_\epsilon^\top E(u) \quad (8)$$

$$\beta = a_\beta^\top E(u) \quad (9)$$

where E is the user-index encoder. The transformation vectors a_ϵ and a_β are learned during the training process, and other settings remain the same with the universal Hawkes process.

Similarly, Cao et al. (2017) also integrate a Hawkes process in a neural-based system. To avoid pre-defining a time decay function, the time range in an observation is split into a number of disjoint intervals, whereas user information is embedded in the input. Although this non-parametric method can be applied in our model, the selection of the number of intervals undermines the flexibility of the process. However, as in their work, a fully connected layer is applied afterwards which takes the excited (decayed) information representation v' as input and outputs the final prediction of the sentiment y_t .

4 Experiments

We investigate the effect of textual and contextual information in personalization and evaluate the performance of the model employing different input formulations.

4.1 Dataset

The Sentiment140¹ corpus is chosen in the experiments for complying the requirements that

1. there are sufficient frequent users,
2. the text is domain-independent,
3. the desired textual and contextual information is present,

¹<http://help.sentiment140.com/for-students>, last seen on September 24, 2019

4. the corpus is annotated from the writers’ perspectives.

The corpus is labeled automatically by emoticons as described in Go et al. (2009) and reflects a user-specific view in contrast to the corpus labeled by others such as the SemEval² corpus. However, the automatic labeling may also contain a certain level of noise caused by the variation in emoticon usage and the unreliability at the user end. The experimented dataset is created by taking the messages from the users who have posted at least 20 times before a pre-set timestamp. This results in 2369 users with overall 122,000 messages in which 79,009 are positive and 42,991 are negative. Furthermore, the dataset is split into a training set, a development set and a test set according to two pre-set time points to ensure that the prediction is only made based on the messages in the past. Other details of the dataset can be found in the appendix.

4.2 Experimental Settings

The experiments are conducted using Keras³ with TensorFlow⁴ backend. The concepts used in the **AR** and **CW** formulations are based on SenticNet 5⁵. In **WE** and **CW**, 100 dimensional Twitter word vectors are taken from GloVe⁶. The ELMo word representations in the **DC** input formulation are supported by TensorFlow Hub⁷, which are later re-trained with other weights in the model. The inputs with **AR**, **WE** and **DC** are encoded into different lengths based on the number of elements in each formulation, however they are suppressed at the embedding layer that generates a vector of length 100 at each time step in order to make fair comparisons. In **CW** and **Combi**, the input vectors fed to the recurrent layer are longer (164 and 264 respectively) because of the concatenation of representations. The dimension of user embeddings is set to 32. There are three recurrent layers at the recurrent block that each contains 100 units,

²<http://alt.qcri.org/semeval2017/task4/>, last seen on September 24, 2019

³<https://keras.io/>, last seen on September 24, 2019

⁴<https://www.tensorflow.org/>, last seen on September 24, 2019

⁵<https://sentic.net/downloads/>, last seen on September 24, 2019

⁶<https://nlp.stanford.edu/projects/glove/>, last seen on September 24, 2019

⁷<https://tfhub.dev/google/elmo/2>, last seen on September 24, 2019

and the number of time steps T is set to 20. For the selection procedure, the same setting is used as in Guo et al. (2019b), where Manhattan distance is used as the ground measurement for calculating topic similarities and the similarity threshold δ is set empirically at 0.8. At the attention block, the time unit is hour, and the values of ϵ and β are initialized at 0.01 and 0.001 respectively when using the universal Hawkes process; the initial values for the vectors a_ϵ and a_β when using user-specific processes are also vectors of 0.01 and 0.001 respectively, and the length of the vectors has to be the same with the dimension of user embeddings, which is 32. The dimension of the fully connected layer applied before the output is set to the same as the number of units in the recurrent layer. We report the overall accuracy of the model as well as the F_1 scores for the positive and negative classes. Detailed settings of the model, sample codes for the Hawkes process, and trained models with the **Combi** formulation can be found in the supplementary material.

4.3 Results

Table 1 shows the performance of the model in different settings. The best result is given by the **Combi** formulation with user-specific Hawkes process. Comparing to the result we have reported previously in Guo et al. (2019a) with an accuracy of 76.13, the best performance with this model has reached 80.38 using the same test set.

Results with Different Input Formulations

Across different input formulations, improvements can be seen comparing the models using the universal - and the user-specific Hawkes process. Although the increase when applying the **AR** formulation is not significant, the improvement of other formulations are significant (t test with $p < 0.05$). The lack of improvements with the **AR** formulation when learning user-specific behaviors can be caused by the sparser representation compared to the other formulations. A matching from each post to a list of concepts and negations is performed which omits information that is not present in the given list. The list of concepts provided by SenticNet 5 is more restricted than the word vectors by GloVe, and is far less flexible than the character-based representation. In addition, due to the highly unstructured nature of social texts, the preprocessing of the posts plays a significant role in the **AR** formulation, which also affects the per-

Input Formulation	Universal Hawkes Process			User-specific Hawkes Process		
	Pos. F1	Neg. F1	Accuracy	Pos. F1	Neg. F1	Accuracy
AR	75.12	76.87	76.04	74.94	77.48	76.28
WE	76.06	77.06	76.58	76.61	78.41	77.55
CW	76.43	77.71	77.10	76.90	79.10	78.06
DC	76.60	79.71	78.27	77.10	80.36	78.86
Combi	77.78	80.67	79.34	78.06	82.25	80.38

Table 1: Performance of the sentiment model when applying the universal - and the user-specific Hawkes process with different input formulations.

formance substantially.

We can also observe improvements when using finer granular levels which are more sensitive and representative towards user variations. Note that using the character-based **DC** formulation alone offers better performance than using the combination of word - and concept-level representations; however the ELMo representation has a more complex structure, a higher dimensional output, and it takes longer time to re-train the weights in the network. In conclusion, the best solution for constructing representation for the inputs is to leverage the combined granular levels from character to word, and to concept (**Combi**). With such a representation, the system is able to analyze user-specific behaviors regarding the lexical usage and the consistency of sentiment.

Results for Various Lengths of History

Figure 2 shows the performance of the models while using the **CW** formulation. The models are tested for T from 1 where no earlier posts are considered, to 20 after which no significant improvement can be observed due to the number of related posts a user normally publishes.

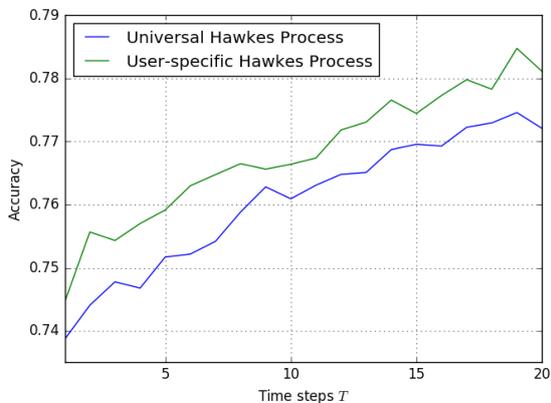


Figure 2: Comparison of the universal - and the user-specific Hawkes process-based models while using different time steps.

For the case when the user history is not incorporated in the model ($T = 1, \Delta t = 0$), we can deduce that $v' = \lambda + \epsilon\lambda'$, which leads to an accuracy of 73.87 with the universal ϵ and 74.46 with the user-specific ϵ (Equation 8).

We can observe an increase in both models when rising the number of time steps T . The increase indicates that the personalization is effective and earlier posts are valid contextual information. By using the selection procedure, the models with a smaller number of T (except for when $T = 1$) take into account more related posts in the past. The increase grows slightly faster towards smaller numbers of T , which is also caused by the limitation of user frequencies in the experimented corpus. We believe that given a sufficient number of frequent users, the performance of the proposed models can be further improved.

Results for Various User Frequencies

The performance of the models when applying for users with different frequencies can be seen in Figure 3. The x-axis corresponds to the lower bound of the user frequency. We take the lower bound for the illustration because there are different numbers of users for each frequency, and many frequencies have no users to assign to. With both models, significant growths for each input formulation can be observed while increasing the lower bound of the frequency. Note that although the **Combi** formulation gives the overall best performance, we can see from the figure that it does not give the best results in all the cases. For instance, when the user frequency is around 80, the **WE** formulation has the best accuracy in both models. However, such an observation is also restricted by the number of frequent users in general — with only 372 posts in the test set when the user frequency is at least 100, the performance is highly dependent on the remaining 3 users. Another observation is that the

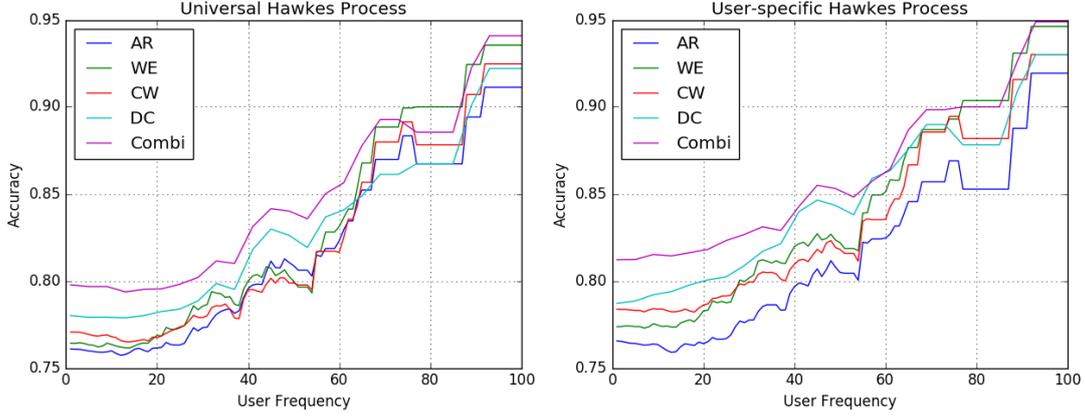


Figure 3: Performance of the universal - and the user-specific Hawkes process-based models for users with different frequencies when applying different formulations.

WE formulation performs better than the **CW** formulation in higher user frequencies, but it has a lower overall performance because there are more users who have published less than 30 posts than the ones who have more.

4.4 Visualization of User-specific Hawkes-Attention

In order to examine the user-specific Hawkes process, we visualize the intermediate calculations for the values of ϵ and β for 10 random users (Figure 4). Each cell in the figure corresponds to the value of $a_{\epsilon_i} * E_i(u)$ in Equation 8 (top figure) or $a_{\beta_i} * E_i(u)$ in Equation 9 (bottom figure) at dimension i for the respective user.

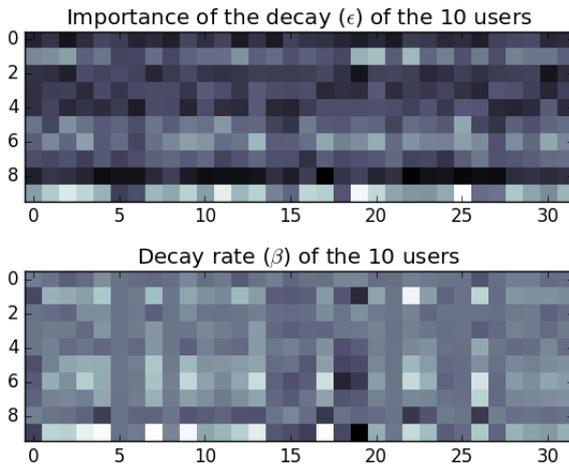


Figure 4: The vectors of ϵ and β in the user-specific Hawkes process of the random 10 users.

The effect of the learned transformation vectors on the 10 users is illustrated. It can be seen that

the last user in the figure (the one at the bottom line) has the greatest values for ϵ and β , which means that the decay factor has a great impact on the prediction for this user than the others but the influence from the past decays comparably fast — the user is affected a lot by recent events. In contrast, among the 10 users, the second last user is the least influenced by the past which is visualized in darker colors. From this figure, we can see that the different decaying processes are indeed learned for different users with the vector transformation. One may argue that the behavior of the Hawkes process also depends on the time period of the experimented dataset; however, if an earlier post (outside of the training period) is highly relevant to the current one, the large value of λ_i can still prevail regardless the value of ϵ .

5 Conclusion

This paper presents a personalized sentiment model that captures the individualities in expressing sentiment and analyzes the evolvement of sentiment over time. Particularly, we categorize the information used for the modeling into textual and contextual information, and evaluate the effectiveness of using the contextual information to boost the performance of the model. A novel attention mechanism with user-specific Hawkes process is employed for this purpose. Technically, it also provides an alternative for studying various time gaps in temporal sequences with neural networks. Different input formulations are applied in which the combined granular representation performs the best. Based on our findings, we can conclude that the individual variation indeed affects the analy-

sis, and the contextual information, as an essential part in human interactions, positively contributes to the performance.

Because the informal text we have used deviates from the language standard, the representation of input text plays a significant role in improving the performance. In the future work, we will exploit phonetic representation which can provide another source of information for such text. The posts can be transcribed into phonetic sequences, for instance, by using the International Phonetic Alphabet, in order to handle certain misspellings and to study the trend of using letters with similar pronunciations as substitutions. Moreover, other types of contextual information should be explored as well to enhance the understanding of individual behaviors on social platforms. As an example, social relations can be used to identify abnormalities in the change of sentiment, especially in the case that a user is exceptionally stimulated by other users or special events which causes untypical behaviors. The personalized model can also be helpful in other scenarios, such as to offer deep understanding for user-tailored conversations or companionship.

References

- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.
- Erik Cambria, Soujanya Poria, Devamanyu Hazarika, and Kenneth Kwok. 2018. SenticNet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings. In *Proceedings of AAAI*.
- Qi Cao, Huawei Shen, Keting Cen, Wentao Ouyang, and Xueqi Cheng. 2017. DeepHawkes: bridging the gap between prediction and understanding of information cascades. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 1149–1158. ACM.
- Huimin Chen, Maosong Sun, Cunchao Tu, Yankai Lin, and Zhiyuan Liu. 2016a. Neural sentiment classification with user and product attention. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1650–1659.
- Tao Chen, Ruifeng Xu, Yulan He, Yunqing Xia, and Xuan Wang. 2016b. Learning user and product distributed representations using a sequence model for sentiment analysis. *IEEE Computational Intelligence Magazine*, 11(3):34–44.
- Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1(12).
- Lin Gong, Mohammad Al Boni, and Hongning Wang. 2016. Modeling social norms evolution for personalized sentiment classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 855–865.
- Lin Gong, Benjamin Haines, and Hongning Wang. 2017. Clustered model adaption for personalized sentiment analysis. In *Proceedings of the 26th International Conference on World Wide Web*, pages 937–946. International World Wide Web Conferences Steering Committee.
- Siwen Guo, Sviatlana Höhn, and Christoph Schommer. 2019a. Looking into the past: evaluating the effect of time gaps in a personalized sentiment model. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, pages 1057–1060. ACM.
- Siwen Guo, Sviatlana Höhn, and Christoph Schommer. 2019b. Topic-based historical information selection for personalized sentiment analysis. In *Proceedings of the 27th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*, pages 379–384.
- Siwen Guo, Sviatlana Höhn, Feiyu Xu, and Christoph Schommer. 2018. Personalized sentiment analysis and a framework with attention-based Hawkes process model. In *International Conference on Agents and Artificial Intelligence*, pages 202–222. Springer.
- Judith Rich Harris. 2006. *No two alike: Human nature and human individuality*. WW Norton & Company.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.
- Irving L Janis and Peter B Field. 1956. A behavioral assessment of persuasibility: Consistency of individual differences. *Sociometry*, 19(4):241–259.
- Patrick J Laub, Thomas Taimre, and Philip K Pollett. 2015. Hawkes processes. *arXiv preprint arXiv:1507.02822*.
- Per Linell. 2009. *Rethinking language, mind, and world dialogically*, chapter 3. IAP.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

- Daniel Neil, Michael Pfeiffer, and Shih-Chii Liu. 2016. Phased LSTM: Accelerating recurrent network training for long or event-based sequences. In *Advances in Neural Information Processing Systems*, pages 3882–3890.
- Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.
- Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237.
- Soujanya Poria, Erik Cambria, Grégoire Winterstein, and Guang-Bin Huang. 2014. Sentic patterns: Dependency-based rules for concept-level sentiment analysis. *Knowledge-Based Systems*, 69:45–63.
- Johan Reitan, Jørgen Faret, Björn Gambäck, and Lars Bungum. 2015. Negation scope detection for Twitter sentiment analysis. In *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 99–108.
- Ehud Reiter and Somayajulu Sripada. 2002. Human variation and lexical choice. *Computational Linguistics*, 28(4):545–553.
- Kaisong Song, Shi Feng, Wei Gao, Daling Wang, Ge Yu, and Kam-Fai Wong. 2015. Personalized sentiment classification based on latent individuality of microblog users. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.
- Fangzhao Wu and Yongfeng Huang. 2016. Personalized microblog sentiment classification via multi-task learning. In *Thirtieth AAAI Conference on Artificial Intelligence*.
- Zhen Wu, Xin-Yu Dai, Cunyan Yin, Shujian Huang, and Jiajun Chen. 2018. Improving review representations with user attention and product attention for sentiment classification. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489.

A Experimental Settings

Dataset	Starting Time	2009.04.06-22:19:57
	Ending Time	2009.06.25-10:28:28
	# Training Samples	82,361
	Timestamp 1	2009.06.03-00:00:00
	# Validation Samples	16,437
	Timestamp 2	2009.06.07-00:00:00
	# Test Samples	23,202
Recurrent Block	Recurrent Dropout	0.4
Attention Block	Fully-connected Layer	100 nodes
	Activation Function	tanh
	Output Layer	2 (Pos. & Neg.)
	Activation Function	softmax
Optimizer	Adam (learning rate: 0.001)	
Loss	Categorical Crossentropy	

Table 2: Additional details of the technical setup.