Learning Sequential Correlation for User Generated Textual Content Popularity Prediction

Wen Wang[†], **Wei Zhang**[†*], **Jun Wang**[†], **Junchi Yan**[‡], **Hongyuan Zha**[♯]
[†]Shanghai Key Laboratory of Trustworthy Computing, East China Normal University
[‡]Shanghai Jiao Tong University [♯]Georgia Institute of Technology
51164500120@stu.ecnu.edu.cn, zhangwei.thu2011@gmail.com,
jwang@sei.ecnu.edu.cn, yanjunchi@sjtu.edu.cn, zha@cc.gatech.edu

Abstract

Popularity prediction of user generated textual content is critical for prioritizing information in the web, which alleviates heavy information overload for ordinary readers. Most previous studies model each content instance separately for prediction and thus overlook the sequential correlations between instances of a specific user. In this paper, we go deeper into this problem based on the two observations for each user, i.e., sequential content correlation and sequential popularity correlation. We propose a novel deep sequential model called User Memory-augmented recurrent Attention Network (UMAN). This model encodes the two correlations by updating external user memories which is further leveraged for target text representation learning and popularity prediction. The experimental results on several real-world datasets validate the benefits of considering these correlations and demonstrate UMAN achieves best performance among several strong competitors.

1 Introduction

User generated textual content (UGTC) is one of the most important types of user generated content (UGC) in the era of Web 2.0, mainly consisting of unstructured text. As natural language has complex linguistic phenomena [Manning and Schütze, 2001] and the use of words is mainly determined by the writers, UGTC could reflect thoughts and feelings of individual human beings, making it more personalized and unique from other types of UGC, such as image and video. The concrete form of UGTC is social post and it can find many specific examples in the real world, such as Twitter and Sina Weibo where users post microblogs which will be forwarded by interested users, Meetup and Douban Event where organizers post activities for others to participate, and Medium where authors post articles and others can vote on them, etc. With the booming of social posts, the issue of heavy information overload for ordinary users has arisen, and it becomes even serious when the content aggregation and distribution platforms occur, such as SmartNews and Toutiao.

In order to alleviate this issue, web hosts and content providers usually follow two technical routes, i.e., personalized recommendation and popularity-based display. While the former one [Shi *et al.*, 2014] has achieved great success by delivering content to users based on their own preference, it needs enough user historical data, which limits its application, especially for cold-start users [Schein *et al.*, 2002]. The latter one ranks content by their popularity scores, defined to measure the total interactions with users, and display them in front pages. Since it could be applied to all users, it is more general and widely used. During the past decade, studies for popularity prediction have sprung up. An important research direction in this regard is to predict the popularity score for a newly emerging target (e.g., the future total retweet count for a new tweet). It not only helps users find popular content ahead of time, but also supports marketing strategies and content distribution mechanism [Figueiredo *et al.*, 2011].

Limitations of existing studies. In the literature, [Cui *et al.*, 2011; Tsur and Rappoport, 2012; Zhao *et al.*, 2015; Martin *et al.*, 2016] have explored to predict the popularity of UGTC by mainly considering user and content information from the perspectives of feature engineering and statistical models. Unfortunately, they model each social post separately and ignore two general sequential correlations (see Figure 1(a) and 1(b)) between different social posts of a same user. The first observed correlation is the sequential content correlation meaning two posts with smaller position interval in a sequence have larger text similarity. The second is the sequential popularity correlation denoting the change of a user's content popularity is smooth. Although the study of social image popularity prediction [Wu *et al.*, 2017a] takes a step in sequential modeling for images, it only models a global user-image sequence, without differentiating users.

Inspired by these two observed correlations, we develop a novel deep sequential model named User Memory-augmented recurrent Attention Network (UMAN). The main

[*]Corresponding author is Wei Zhang. We thank the anonymous reviewers for the valuable comments. This work was supported in part by Shanghai Chenguang Program (16CG24), Shanghai Sailing Program (17YF1404500), and NSFC (61702190, U1609220, 61672231, 61672236).

idea of this model is to utilize a user's recent posts and corresponding popularity scores to enable the user memory to fuse both long-term characteristics and short-term tendencies, which will be applied to model target text and generate popularity prediction. Specifically, UMAN first adopts external user memories to produce attentive text embeddings for their recent posts. After concatenating the learned embeddings with their corresponding popularity scores, UMAN leverages recurrent neural network to generate integrated representations which are in return used to update the user memories. Ultimately, UMAN exploits the updated user memories to gain attentive text embeddings of target content and further uses both the memories and text embeddings for popularity generation. The attention mechanism [Bahdanau *et al.*, 2014] is plentifully applied in UMAN, with the intuition that good text embeddings should reflect users' characteristics and be personalized.

Contribution. We summarize the main contributions of this paper as follows:

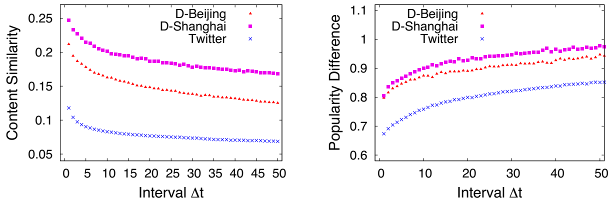
- We identify the two sequential correlations in the problem of UGTC popularity prediction, i.e., sequential content correlation and sequential popularity correlation. To our best knowledge, neither of the two correlations has been explicitly modeled in the literature. In fact, encoding the two correlations can significantly improve the popularity prediction.
- To effectively capture the two correlations, a novel deep sequential model (UMAN) is devised. Specifically, our model enables external user memory updating to fuse both long-term characteristics and short-term tendencies, which is further used for target text representation learning and popularity prediction.
- We conduct comprehensive experiments on several real-world datasets, demonstrating UMAN outperforms strong competitors, validating the benefits of its main components, and providing qualitative analysis for case studies. We make the dataset[1] publicly available for further relevant research.

2 Preliminaries

2.1 Problem Definition

Before we go into the details of the problem and method, we first provide some necessary mathematical notations used later. Throughout this paper, we use bold upper case letters to denote matrices and bold lower case letters to represent vectors. Without specification, non-bold letters mean scalars. In addition, we utilize $\mathbf{W}_{\cdot,j}$ to denote the j-th column of the matrix $\mathbf{W}$ and it is similar for other symbols.

With regard to various types of UGTC in different social medias, the two most fundamental and important elements are user and textual content which are the focus of this work. Let $\mathcal{U} = \{u_1, \ldots, u_N\}$ denote a set of users. For each user $u_n \in \mathcal{U}$, assume $\mathcal{P}_{n,T} = \{S_{n,1}, \ldots, S_{n,t}, \ldots, S_{n,c_{n,T}}\}$ includes all the posts the user posted before time T, where $c_{n,T}$ is the corresponding count of posts. Each post should have



(a) Content similarity (b) Popularity difference

Figure 1: Quantification of the two correlations on three datasets.

a corresponding popularity score, e.g., $y_{n,t}$ for $S_{n,t}$. Thus we define $\mathcal{Y}_{n,T}$ analogous to $\mathcal{P}_{n,T}$. Moreover, the text $S_{n,t}$ denotes a sequence of words as $\{w_1^{n,t}, \ldots, w_i^{n,t}, \ldots, w_{v_{n,t}}^{n,t}\}$, where $w_i^{n,t} \in \mathcal{V}$ and $v_{n,t}$ is the number of words in $S_{n,t}$.

Based on the above notations, we formally state the problem as follows:

Problem 1 (Popularity Prediction of UGTC) *Given the past social posts $\mathcal{P}_{\cdot,T}$ of users in $\mathcal{U}$ before time T and their corresponding popularity scores $\mathcal{Y}_{\cdot,T}$, the target is to learn an optimal popularity predictor that can predict the future popularity scores of users' newly posted UGTC.*

2.2 Observed Correlations

As aforementioned, we have two key observations about sequential correlations for each user. In this part, we quantify these two observations by performing data analysis on the datasets we used in the experiments and illustrated in Section 4. We first clarify the observed correlations and then provide quantitative analysis one after another.

Observation 1 *Sequential content correlation: the content similarity of two posts from a same user negatively correlates with the size of their position interval.*

To verify this, we start with a simple vector space model with each dimension being a tf-idf weight to calculate the cosine similarity of two posts. For each position interval $\Delta t \in \mathbb{Z}^+$, we assume the similarity between the t-th and $(t+\Delta t)$-th posts of user u_n is $\mathbf{D}^n_{t,t+\Delta t}$. Then we define $\Pi^1_{\Delta t}$ to be the average content similarity over all users, which is given by:

$$\Pi^1_{\Delta t} = \frac{\sum_{n=1}^{N} \sum_{t=1}^{c_{n,T}-\Delta t} \mathbf{D}^n_{t,t+\Delta t}}{\Omega^1_{\Delta t}}, \tag{1}$$

where $\Omega^1_{\Delta t}$ is the number of the summation terms.

We analyze the numerical variation with the increase of Δt and report the results in Figure 1(a). It can be easily observed that the average content similarities become smaller when Δt gets larger on all the three datasets. Particularly, an interesting phenomenon is that the similarities vary dramatically first and then the variation trends become relatively stable. Therefore, sequential content correlation indeed widely exists in UGTC. As a result, users' recent posts might be more likely related to current posts in some topics and thus it is intuitive to regard the recent posts as contextual information to complement the current posts. This might be even more beneficial when the length of a post is short.

[1] https://github.com/Autumn945/UMAN_data

Observation 2 *Sequential popularity correlation: the numerical difference of popularity scores between two posts from a same user positively correlates with the size of their position interval.*

Similar to the above procedures, we define the average popularity difference $\Pi^2_{\Delta t}$ over all users for the interval Δt as follows:

$$\Pi^2_{\Delta t} = \frac{\sum_{n=1}^{N} \sum_{t=1}^{c_{n,T}-\Delta t} |y_{n,t+\Delta t} - y_{n,t}|}{\Omega^2_{\Delta t}}, \tag{2}$$

where $\Omega^2_{\Delta t}$ corresponds to the number of summation terms in the above equation.

As Figure 1(b) shows, the average popularity differences increase with the growth of interval Δt. The variation trends of popularity difference are similar to those of content similarity, changing rapidly for the first several small Δt. These phenomena might reflect that for a specific user, the popularity of a target text is partially determined by its recent past posts' popularity.

Based on the above analysis, our model is designed to incorporate users' recent posts and their corresponding popularity to effectively explore the two correlations. Note our model also allows for more fined-grained treatment such as by further considering the correlation from the perspective of time gaps or topic categories, which we leave for future work.

3 Model

We present the overall framework of UMAN in Figure 2. It consists of two main components corresponding to the two panels. The left involves user memory updating mechanism, aiming at encoding the mentioned two correlations by updating user memories (see Section 3.2). The right is attention-based popularity prediction which leverages the updated user memories to generate the popularity scores for the targets (see Section 3.3).

For ease of model clarification, we take the user u_n as an example. Assume we consider the user's recent M posts and aim to predict the popularity $y_{n,t}$ of $S_{n,t}$, then we have $\{S_{n,t-m}\}_{m=1}^{m=M}$ and $\{y_{n,t-m}\}_{m=1}^{m=M}$ as additional inputs. We start with the illustration of the input representations below.

3.1 Input Representation

Text representation. We represent the original one-hot vectors of words in the post $S_{n,t-m}$ as $\mathbf{X}_{n,t-m} = [\mathbf{x}_1, \ldots, \mathbf{x}_{v_{n,t-m}}]$. Through a commonly used lookup table operation, they are mapped to dense embeddings $\mathbf{E}_{n,t-m} = [\mathbf{e}_1, \ldots, \mathbf{e}_{v_{n,t-m}}]$, each of which is supposed to be a k-dimensional vector. Inspired by the idea of convolutional neural network (CNN) model for texts [Kim, 2014], we apply 1-D convolution with filter sizes (h) of 1 (unigram), 2 (bigram), and 3 (trigram) to each word embedding, which is defined as follows:

$$\mathbf{e}^h_i = \mathbf{W}^h_F \mathbf{e}_{i:i+h-1} + \mathbf{b}^h_F, \tag{3}$$

where $\mathbf{e}_{i:i+h-1}$ is the concatenation of the word embeddings $\mathbf{e}_i, \ldots, \mathbf{e}_{i+h-1}$, and $\mathbf{e}_{i:i+h-1} \in \mathbb{R}^{hk}$ with hk being the product of h and k. Without loss of generality, we let $\mathbf{W}^h_F \in \mathbb{R}^{k \times hk}$ and $\mathbf{b}^h_F \in \mathbb{R}^k$ to keep the dimensions of $\mathbf{e}^h_i$ and $\mathbf{e}_i$ being the same.

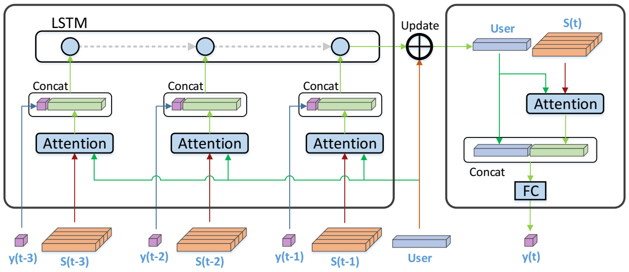


Figure 2: The architecture of the proposed UMAN. Note: take 3 recent posts and popularity scores as examples for presentation.

Different from [Kim, 2014] that adopts max pooling operation along the word dimension to get an integrated representation for each text, we apply max pooling along the dimension of filter size h, e.g., $\dot{\mathbf{e}}_{i,j} = \max\{\mathbf{e}^1_{i,j}, \mathbf{e}^2_{i,j}, \mathbf{e}^3_{i,j}\}$, to get the updated embeddings $\dot{\mathbf{E}}_{n,t-m} = [\dot{\mathbf{e}}_1, \ldots, \dot{\mathbf{e}}_{v_{n,t-m}}]$ for the post. Compared with the original word embeddings, we could capture context word information (e.g, phrase-level knowledge) and improve expressive capacity, of which the benefit is verified in later experiments.

User memory and popularity representation. We introduce a user memory $\mathbf{v}_n$ for the user u_n with the following intuition. User memory stores a global user state which could represent its long-term characteristics and the memory could be further updated by a user's recent UGTC to capture its short-term tendencies. The idea is consistent in spirit of the recently proposed memory mechanism [Sukhbaatar *et al.*, 2015; Graves *et al.*, 2016] and has successful applications such as question answering [Xiong *et al.*, 2016]. In addition, we simply take the rescaled popularity scores as popularity representation since the original scores are scalars and already continuous values.

3.2 User Memory Updating Mechanism

To encode the two sequential correlations, we utilize users' recent posts and their corresponding popularity scores to update user memory. Achieving this involves two main procedures, i.e., user-aware attentive representation learning for each of the posts and recurrent modeling for all the posts and popularity scores.

User-aware attentive representation learning. We first employ user memory to learn user-aware text representation. It reflects a user's different attentions to each word embedding, indicating personalized word importance.

More specifically, we define a simple score function $f(\mathbf{v}_n, \dot{\mathbf{e}}_i)$ to measure the importance of word w_i as follows:

$$f(\mathbf{v}_n, \dot{\mathbf{e}}_i) = \mathbf{v}_n^\top \dot{\mathbf{e}}_i. \tag{4}$$

This calculation is commonly employed by latent factor models in recommender system [Shi *et al.*, 2014] to weigh the relevance between user and item. Based on this, we provide the formulas to calculate the attention weight $\alpha_{n,i}$ and user-aware text representation $\bar{\mathbf{e}}_{n,t-m}$, which are given by:

$$\alpha_{n,i} = \frac{\exp\Big(f(\mathbf{v}_n, \dot{\mathbf{e}}_i)\Big)}{\sum_{i'=1}^{v_{n,t-m}} \exp\Big(f(\mathbf{v}_n, \dot{\mathbf{e}}_{i'})\Big)}, \tag{5}$$

$$\bar{\mathbf{e}}_{n,t-m} = \sum_{i=1}^{v_{n,t-m}} \alpha_{n,i} \dot{\mathbf{e}}_i. \tag{6}$$

After getting $\{\bar{\mathbf{e}}_{n,t-m}\}_{m=1}^{m=M}$ through the above equations, we concatenate them with popularity representation, i.e., $\{[\bar{\mathbf{e}}_{n,t-m}; y_{n,t-m}]\}_{m=1}^{m=M}$ which will be fed into the procedure of recurrent modeling. For simplicity, we denote $[\bar{\mathbf{e}}_{n,t-m}; y_{n,t-m}]$ as $\mathbf{q}_{n,t-m}$.

Recurrent modeling. We employ long-short term memory (LSTM) [Hochreiter and Schmidhuber, 1997] to recurrently model the sequence of $\{\mathbf{q}_{n,t'-M+m}\}_{m=1}^{m=M}$ where $t' = t-1$. It can flexibly choose to forget past sequential information and remember current input representation. The hidden state of the m-th position is $\mathbf{h}_m$ which is given by:

$$\mathbf{h}_m = \mathrm{LSTM}(\mathbf{q}_{n,t'-M+m}, \mathbf{h}_{m-1}). \tag{7}$$

After recursive updating, our UMAN learns the final embedding $\mathbf{h}_M$ for the user post sequence, which is used to update the user memory through the following way:

$$\dot{\mathbf{v}}_n = \mathbf{h}_M + \mathbf{v}_n. \tag{8}$$

Consequently, the updated user memory accumulates user long-term state and short-term preference.

3.3 Attention-based Popularity Prediction

Considering the target text $S_{n,t}$ for prediction, we use the convolutional neural network based approach (see Section 3.1) to obtain the word embeddings $\dot{\mathbf{E}}_{n,t} = [\dot{\mathbf{e}}_1, \ldots, \dot{\mathbf{e}}_{v_{n,t}}]$. Afterwards, the updated user memory $\dot{\mathbf{v}}_n$ is employed to get the text embedding $\bar{\mathbf{e}}_{n,t}$ through the way shown in Equation 5 and 6.

UMAN adopts a short connection to connect the updated user memory to popularity generation, which could ensure direct information flow between users' past popularity scores and the target popularity. Specifically, we concatenate the user memory and text embedding to obtain $[\dot{\mathbf{v}}_n; \bar{\mathbf{e}}_{n,t}]$. We feed it into a fully connected layer with the parameter vector $\mathbf{w}_{FC}$ and bias b_{FC} to generate popularity prediction $\hat{y}_{n,t}$, which is given by:

$$\hat{y}_{n,t} = \mathbf{w}_{FC}[\dot{\mathbf{v}}_n; \bar{\mathbf{e}}_{n,t}] + b_{FC}. \tag{9}$$

We train UMAN in an end-to-end fashion by minimizing the mean squared error, for example, $(y_{n,t} - \hat{y}_{n,t})^2$. We generate training sequences for each user by a sliding window approach. Take the user u_n as an example again. The corresponding training sequences are from $([S_{n,1}, \ldots, S_{n,M}], [y_{n,1}, \ldots, y_{n,M}], u_n, S_{n,M+1}, y_{n,M+1})$ to $([S_{n,c_{n,T}-M}, \ldots, S_{n,c_{n,T}-1}], [y_{n,c_{n,T}-M}, \ldots, y_{n,c_{n,T}-1}], u_n, S_{n,c_{n,T}}, y_{n,c_{n,T}})$.

4 Experiments

The experiments focus on answering the important research questions below:

Q1. How does UMAN compare with the adopted baselines?

Q2. Are the main components of UMAN effective?

Data	D-Beijing	D-Shanghai	Twitter
#Users	883	1,109	8,727
#Words	85,847	104,225	38,647
#Train	27,582	35,895	223,232
#Validation	4,108	5,352	10,100
#Test	8,216	10,706	90,942

Table 1: Basic statistics of the datasets.

4.1 Experimental Setup

Datasets. We conduct experiments on three real-world datasets from two domains with different languages to ensure the generality of the proposed model. The first domain is Douban Event[2] (mostly written in Chinese), a large and popular website in China where users post activities for ordinary users to register online to attend. We use the datasets [Zhang and Wang, 2015; Yin *et al.*, 2016], aiming to predict the ultimate number of participants for each target activity. As all activities are offline and divided by cities, we choose the activities held in Beijing and Shanghai to create two datasets named D-Beijing and D-Shanghai. The second is Twitter (we only keep tweets written in English), where users publish tweets and other users interested can retweet them. We sample a dataset from the Twitter streaming archive[3], mainly consisting of the original tweets written in May of 2016.

To evaluate our model and make comparisons with other methods, we first perform text preprocessing for the three datasets, including segmenting words for Chinese text, converting all English words to lowercases, removing most punctuations, and filtering sparse words (occurring less than 5 times on the datasets). Moreover, we select users with at least 10 posts to ensure having enough sequential posts. We split the datasets in chronological order for each user. The training sets occupy about 70% in total and the validation sets are randomly sampled from the left datasets. The basic statistics of the experimental datasets are shown in Table 1.

Baselines. Since there are few well-designed sequential modeling approaches for the sequential popularity prediction of UGTC, we choose or construct comparison models based on the most related research fields.

- **HF-NMF [Cui *et al.*, 2011]**: It is an early non-sequential model for user post popularity prediction, without capturing influence from user sequential posts. We consider a simpler version without modeling social relations.
- **FeaReg**: Regression based methods are leveraged for modeling hand-crafted features such as tf-idf of text, similar to [Martin *et al.*, 2016]. We select ridge regression model for its good performance and regard all users' recent posts as features for sequential modeling.
- **DTCN-T**: Following the deep temporal context network (DTCN) for sequential image popularity prediction [Wu *et al.*, 2017a], we build a similar model for textual popularity prediction, denoted as DTCN-T. It constructs a global user-post sequence for the studied problem, without distinguishing posts for different users.
- **SRNN-P**: Inspired the session-based recurrent neural networks (SRNN) [Hidasi *et al.*, 2016] for recommendation,

[2] https://beijing.douban.com/events/week-all

[3] https://archive.org/search.php?query=twitter-stream

we adopt LSTM to model post sequences of different users. To gain post embedding, we test several methods such as LSTM and mean pooling, and select the one with better performance. We name SRNN for popularity prediction as SRNN-P.

- **ATEM-P**: The attention-based transaction embedding model (ATEM) [Wang *et al.*, 2018a] is recently proposed for the scenario of the next item recommendation, which can be adapted to our problem setting by regarding users' recent posts as contextual items. We denote ATEM for popularity prediction as ATEM-P.

To ensure fair comparisons, we make DTCN-T, SRNN-P, and ATEM-P learn textual embeddings by the same convolutional neural network used in UMAN , which is later demonstrated to achieve comparable performance with LSTM but be much more efficient. Moreover, when taking both user posts and their popularity scores as input, the above baselines except HF-NMF concatenate post embeddings and popularity scores like UMAN as well.

Evaluation protocols. We choose mean squared error (MSE) and mean absolute error (MAE), which are the two standard evaluation metrics for popularity prediction [Li *et al.*, 2017; Wu *et al.*, 2017a]. We compare UMAN with other sequential modeling approaches using a sliding window strategy on the test sets, just like the way mentioned in Section 3.3. However, we compare UMAN and HF-NMF by only evaluating on the first record of each user on the test sets since HF-NMF is a non-sequential modeling approach.

To suppress large variations of popularity, we use $y_{n,t} = \log(\frac{r_{n,t}}{d_{n,t}} + 1)$ [Wu *et al.*, 2017a] to denote the rescaled number of retweets in Twitter and $y_{n,t} = \log(p_{n,t} + 1)$ [Li *et al.*, 2017] to represent the rescaled number of participants in Douban Event, where $d_{n,t}$ means the number of days since the tweet was posted, and $r_{n,t}$ and $p_{n,t}$ correspond to the original counts of retweets and participants, respectively. The reason to penalize old tweets with $d_{n,t}$ but not activities is that each offline activity has a starting time and when the time passes, the ultimate number of participants is deterministic. However, tweets can be actually retweeted at any time.

Implementation details. We determine the hyperparameters of the adopted models on the validation datasets, and keep them fixed during comparison. The dimension of every hidden vector—including user memory, word embedding, states of RNN, factors in HF-NMF —is set to 128. The length of users' recent post sequences is set to 4 by default. We train all the deep learning based models by Adam with a learning rate 0.001, a minibatch size 64, and exponential decay rates 0.9 and 0.999. In addition, early stopping is adopted to terminate the training process based on the performance on validation datasets.

4.2 Model Comparison (Q1)

We first test the performance of UMAN and HF-NMF in a non-sequential test setting. We let UMAN do not consider users' recent popularity scores here, in accordance with HF-NMF. Table 2 shows UMAN largely outperforms HF-NMF on all the three datasets, indicating the advantage of sequential modeling, which will be further demonstrated.

Models	D-Beijing		D-Shanghai		Twitter	
	MSE	MAE	MSE	MAE	MSE	MAE
HF-NMF	1.1949	0.7987	0.9319	0.7209	0.7014	0.6256
UMAN	**0.9411**	**0.7196**	**0.8569**	**0.6884**	**0.6444**	**0.5584**

Table 2: Evaluation on each user's first record on the test sets.

Table 3 compares our model with the other sequential approaches in two settings, i.e, whether using popularity scores of users' recent posts or not. By first comparing the results belonging to setting one and those in setting two, it shows all the latter results are significantly better and more stable, indicating the recent popularity scores are indeed beneficial for the studied problem and more easily to be modeled.

Models	D-Beijing		D-Shanghai		Twitter	
	MSE	MAE	MSE	MAE	MSE	MAE
Only using users' recent posts						
FeaReg	0.9324	0.7253	0.9851	0.7477	1.1351	0.8025
DTCN-T	0.9376	0.7202	1.0347	0.7792	0.9629	0.7230
SRNN-P	1.0181	0.7644	1.0509	0.7799	1.4359	0.9218
ATEM-P	0.9872	0.7352	1.0451	0.7716	0.9914	0.7326
UMAN	**0.8806**	**0.6902**	**0.9467**	**0.7305**	**0.9310**	**0.7066**
Using both users' recent posts and corresponding popularity scores						
FeaReg	0.8131	0.6699	0.8379	0.6845	0.9316	0.7231
DTCN-T	0.9169	0.7047	0.9931	0.7517	0.9131	0.7113
SRNN-P	0.7413	0.6294	0.7610	0.6433	0.7750	0.6443
ATEM-P	0.8138	0.6547	0.8099	0.6625	0.7734	0.6382
UMAN	**0.7197**	**0.6190**	**0.7371**	**0.6345**	**0.7583**	**0.6342**

Table 3: Comparisons on the whole datasets.

By comparing UMAN with FeaReg, we find that the performance differences are much larger in Twitter. This may be explained the fact that tweets are short and contain more spelling mistakes [Sriram *et al.*, 2010], it is not easy for the manually defined features such as TF-IDF to handle them. The better results of UMAN than those of DTCN-T reveal that considering all user-post pairs as a whole sequence is not as good as differentiating sequences for different users. We further compare UMAN with two recent deep sequential modeling approaches developed for recommendation. The better performance shows that our model is indeed suitable for the sequential popularity problem. To sum up, our model UMAN achieves the best results on both settings, which answers the question Q1.

4.3 Model Analysis of UMAN (Q2)

Ablation study. To validate the effectiveness of main components in UMAN, we conduct ablation experiments, using "w/o seq" to denote not modeling sequences of users' recent posts and popularity scores, and using "w/o mem" to mean not incorporating user memory into UMAN. Table 4 shows that the part of sequential modeling plays a major role in obtaining good performance and the incorporation of user memory can further improve the results significantly.

We then show the contributions of the two sequential correlations in UMAN by comparing the results of "w/o seq" from

Models	D-Beijing	D-Shanghai	Twitter
UMAN (w/o seq)	0.8931	0.9635	0.9593
UMAN (w/o mem)	0.7421	0.7448	0.7727
UMAN	**0.7197**	**0.7371**	**0.7583**

Table 4: Ablation study of UMAN. Note: MSE is used.

Table 4 and UMAN (only users' recent posts) and UMAN from Table 3 in an incremental manner. We can see that considering the content correlation is indeed useful for gaining better results than not considering this, and additionally incorporating popularity correlation can largely improve the performance.

Influence of sequence length. We visualize the performance variations with the increase of the modeled sequence length in Figure 3. SRNN-P is adopted as a comparison for its second best performance when modeling both recent posts and popularity scores. As expected, the performance becomes better with larger sequence length, and the variation trends turn to be stable. Regardless of the sequence length, our model UMAN outperforms SRNN-P consistently.

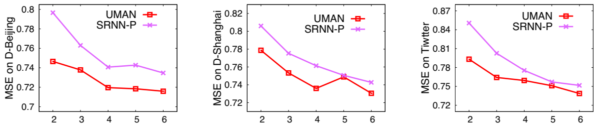


Figure 3: Results for different lengths of post sequences.

We also test UMAN considering corresponding users' all past posts. The results are 0.7219, 0.7284, and 0.7342 on the three datasets, show that considering all posts does not improve the performance much or even degrades the performance. Moreover, the computational cost is much heavier than only considering recent 4 posts.

Effectiveness of CNN for text modeling. Table 5 presents the results of CNN and other two alternatives for text modeling. "w/o cnn" means directly feeding original word embeddings to user-aware attentive representation learning and the corresponding results are not so good, especially on the Douban Event datasets. The performance of CNN is comparable with LSTM, but the efficiency of CNN is much better (e.g., 5-10 times faster in Twitter). Besides, there is a smaller performance gap in Twitter, reflecting the hardness of modeling tweets due to their noise and shortness.

Models	D-Beijing	D-Shanghai	Twitter
UMAN (w/o cnn)	0.7602	0.7670	0.7638
UMAN (lstm)	0.7233	**0.7322**	0.7613
UMAN	**0.7197**	0.7371	**0.7583**

Table 5: Performance of word embedding. Note: MSE is used.

4.4 Qualitative Analysis

We visualize the attentions of UMAN and UMAN (w/o seq) to the examples in Figure 4, where each value is the product of the attention weight and the text length. This operation makes the mean attention value to be 1, regardless of what the text length is. Thus we can easily compare the attention weights of texts with different lengths.

From a whole perspective, these attention maps show that our model more attends to head words than UMAN (w/o seq). Taking the second as an example, UMAN is able to attend the words "Sachs" and "father", while UMAN (w/o seq) gives more attentions to "Beijing station" which is not a distinguishable word on the D-Beijing dataset.

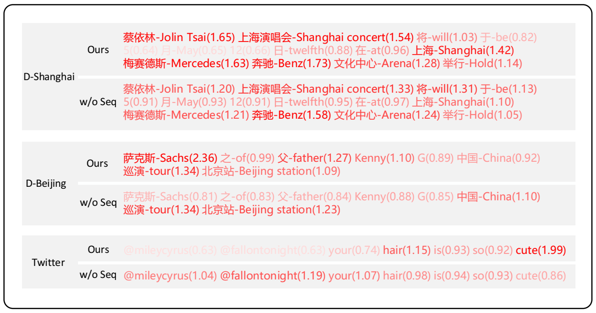

D-Shanghai
Ours
蔡依林-Jolin Tsai(1.65) 上海演唱会-Shanghai concert(1.54) 将-will(1.03) 于-be(0.82)
5(0.64) 月-May(0.65) 12(0.66) 日-twelfth(0.88) 在-at(0.96) 上海-Shanghai(1.42)
梅赛德斯-Mercedes(1.63) 奔驰-Benz(1.73) 文化中心-Arena(1.28) 举行-Hold(1.14)
w/o Seq
蔡依林-Jolin Tsai(1.20) 上海演唱会-Shanghai concert(1.33) 将-will(1.31) 于-be(1.13)
5(0.91) 月-May(0.93) 12(0.91) 日-twelfth(0.95) 在-at(0.97) 上海-Shanghai(1.10)
梅赛德斯-Mercedes(1.21) 奔驰-Benz(1.58) 文化中心-Arena(1.24) 举行-Hold(1.05)
D-Beijing
Ours
萨克斯-Sachs(2.36) 之-of(0.99) 父-father(1.27) Kenny(1.10) G(0.89) 中国-China(0.92)
巡演-tour(1.34) 北京站-Beijing station(1.09)
w/o Seq
萨克斯-Sachs(0.81) 之-of(0.83) 父-father(0.84) Kenny(0.88) G(0.85) 中国-China(1.10)
巡演-tour(1.34) 北京站-Beijing station(1.23)
Twitter
Ours
@mileycyrus(0.63) @fallontonight(0.63) your(0.74) hair(1.15) is(0.93) so(0.92) cute(1.99)
w/o Seq
@mileycyrus(1.04) @fallontonight(1.19) your(1.07) hair(0.98) is(0.94) so(0.93) cute(0.86)

Figure 4: Case study for attention visualization.

5 Related Work

Popularity prediction. It has attracted a lot of attention over the last decade and studied online video [Pinto *et al.*, 2013], offline activity [Wang *et al.*, 2018b], Wikipedia link [Dimitrov *et al.*, 2017], social text [Cui *et al.*, 2011], academic paper [Xiao *et al.*, 2016], multi-modal social image [Zhang *et al.*, 2018], and etc. The relevant studies can be roughly categorized into two groups: 1) static popularity prediction [Cui *et al.*, 2011; Dimitrov *et al.*, 2017; Zhang *et al.*, 2018; Wang *et al.*, 2018b] can predict target popularity just when it occurs; and 2) dynamic popularity prediction [Pinto *et al.*, 2013; Xiao *et al.*, 2016] requires a target's early popularity pattern and some studies [Cao *et al.*, 2017] could even bridge the gap between modeling dynamic diffusion and predicting popularity [Shulman *et al.*, 2016].

However, most related studies have not investigated sequential correlation between different targets. The most relevant one is [Wu *et al.*, 2017a] for predicting image popularity, yet it only constructs a global sequence consisting of images from different users. In contrast, we construct a textual post sequence for each user and have verified the benefit of the sequential modeling in our experiments. Noting that the focus of this paper is to apply the two sequential correlations to static popularity prediction setting. But since the sequential correlations work well for predicting targets' final popularity, it is promising to utilize the popularity dynamics of users' recent posts for inferring the targets' dynamics, which could be an interesting direction for further study.

Deep sequential modeling for recommendation. It is a related direction from the field of recommender system, aiming at mining knowledge from users' recently interacted item sequences for future item recommendation. [Hidasi *et al.*, 2016] first utilized RNN to model the sequential relations between items occurring in a user session and further recommend items which will be chosen in the same session. [Wu *et al.*, 2017b] proposed a coupled recurrent neural network model by supposing not only users having sequences composed by items but also items having user-based sequences. [Quadrana *et al.*, 2017] further developed a hierarchical RNN which can model multiple sessions for each user. Recently, [Wang *et al.*, 2018a] presented an attention based model, giving different importance weights to users' recently interacted items for the next item recommendation. To achieve more comprehensive comparison, we choose some representative sequential modeling approaches as baselines, which could be easily adapted to our problem setting without major modification of their model architectures.

6 Conclusion

In this paper, we have studied the problem of user generated textual content popularity prediction. Inspired by the two key observations about the sequential content correlation and sequential popularity correlation, we have developed a novel deep sequential model named UMAN to encode these two correlations by updating the external user memories with the users' recent textual posts and corresponding popularity scores. We conduct comprehensive experiments on several real-world datasets, demonstrating UMAN is effective for the problem and verifying the main components of UMAN are beneficial for improving the performance.

References

[Bahdanau *et al.*, 2014] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014.

[Cao *et al.*, 2017] Qi Cao, Huawei Shen, Keting Cen, Wentao Ouyang, and Xueqi Cheng. Deephawkes: Bridging the gap between prediction and understanding of information cascades. In *CIKM*, pages 1149–1158, 2017.

[Cui *et al.*, 2011] Peng Cui, Fei Wang, Shaowei Liu, Mingdong Ou, Shiqiang Yang, and Lifeng Sun. Who should share what?: item-level social influence prediction for users and posts ranking. In *SIGIR*, pages 185–194, 2011.

[Dimitrov *et al.*, 2017] Dimitar Dimitrov, Philipp Singer, Florian Lemmerich, and Markus Strohmaier. What makes a link successful on wikipedia? In *WWW*, pages 917–926, 2017.

[Figueiredo *et al.*, 2011] Flavio Figueiredo, Fabrício Benevenuto, and Jussara M. Almeida. The tube over time: characterizing popularity growth of youtube videos. In *WSDM*, pages 745–754, 2011.

[Graves *et al.*, 2016] Alex Graves, Greg Wayne, Malcolm Reynolds, et al. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626):471, 2016.

[Hidasi *et al.*, 2016] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. In *ICLR*, 2016.

[Hochreiter and Schmidhuber, 1997] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[Kim, 2014] Yoon Kim. Convolutional neural networks for sentence classification. In *EMNLP*, pages 1746–1751, 2014.

[Li *et al.*, 2017] Cheng Li, Jiaqi Ma, Xiaoxiao Guo, and Qiaozhu Mei. Deepcas: An end-to-end predictor of information cascades. In *WWW*, pages 577–586, 2017.

[Manning and Schütze, 2001] Christopher D. Manning and Hinrich Schütze. *Foundations of statistical natural language processing*. MIT Press, 2001.

[Martin *et al.*, 2016] Travis Martin, Jake M. Hofman, Amit Sharma, Ashton Anderson, and Duncan J. Watts. Exploring limits to prediction in complex social systems. In *WWW*, pages 683–694, 2016.

[Pinto *et al.*, 2013] Henrique Pinto, Jussara M. Almeida, and Marcos André Gonçalves. Using early view patterns to predict the popularity of youtube videos. In *WSDM*, pages 365–374, 2013.

[Quadrana *et al.*, 2017] Massimo Quadrana, Alexandros Karatzoglou, Balázs Hidasi, and Paolo Cremonesi. Personalizing session-based recommendations with hierarchical recurrent neural networks. In *RecSys*, pages 130–137, 2017.

[Schein *et al.*, 2002] Andrew I. Schein, Alexandrin Popescul, Lyle H. Ungar, and David M. Pennock. Methods and metrics for cold-start recommendations. In *SIGIR*, pages 253–260, 2002.

[Shi *et al.*, 2014] Yue Shi, Martha Larson, and Alan Hanjalic. Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACM Comput. Surv.*, 47(1):3:1–3:45, 2014.

[Shulman *et al.*, 2016] Benjamin Shulman, Amit Sharma, and Dan Cosley. Icwsm. pages 348–357, 2016.

[Sriram *et al.*, 2010] Bharath Sriram, Dave Fuhry, Engin Demir, Hakan Ferhatosmanoglu, and Murat Demirbas. Short text classification in twitter to improve information filtering. In *SIGIR*, pages 841–842, 2010.

[Sukhbaatar *et al.*, 2015] Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, and Rob Fergus. End-to-end memory networks. In *NIPS*, pages 2440–2448, 2015.

[Tsur and Rappoport, 2012] Oren Tsur and Ari Rappoport. What's in a hashtag?: content based prediction of the spread of ideas in microblogging communities. In *WSDM*, pages 643–652, 2012.

[Wang *et al.*, 2018a] Shoujin Wang, Liang Hu, Longbing Cao, Xiaoshui Huang, Defu Lian, and Wei Liu. Attention-based transactional context embedding for next-item recommendation. In *AAAI*, pages 2532–2539, 2018.

[Wang *et al.*, 2018b] Wen Wang, Wei Zhang, and Jun Wang. Factorization meets memory network: Learning to predict activity popularity. In *DASFAA*, pages 509–525, 2018.

[Wu *et al.*, 2017a] Bo Wu, Wen-Huang Cheng, Yongdong Zhang, Qiushi Huang, Jintao Li, and Tao Mei. Sequential prediction of social media popularity with deep temporal context networks. In *IJCAI*, pages 3062–3068, 2017.

[Wu *et al.*, 2017b] Chao-Yuan Wu, Amr Ahmed, Alex Beutel, Alexander J. Smola, and How Jing. Recurrent recommender networks. In *WSDM*, pages 495–503, 2017.

[Xiao *et al.*, 2016] Shuai Xiao, Junchi Yan, Changsheng Li, Bo Jin, Xiangfeng Wang, Xiaokang Yang, Stephen M. Chu, and Hongyuan Zha. On modeling and predicting individual paper citation count over time. In *IJCAI*, pages 2676–2682, 2016.

[Xiong *et al.*, 2016] Caiming Xiong, Stephen Merity, and Richard Socher. Dynamic memory networks for visual and textual question answering. In *ICML*, pages 2397–2406, 2016.

[Yin *et al.*, 2016] Hongzhi Yin, Zhiting Hu, Xiaofang Zhou, Hao Wang, Kai Zheng, Quoc Viet Hung Nguyen, and Shazia Sadiq. Discovering interpretable geo-social communities for user behavior prediction. In *ICDE*, pages 942–953, 2016.

[Zhang and Wang, 2015] Wei Zhang and Jianyong Wang. A collective bayesian poisson factorization model for cold-start local event recommendation. In *SIGKDD*, pages 1455–1464, 2015.

[Zhang *et al.*, 2018] Wei Zhang, Wen Wang, Jun Wang, and Hongyuan Zha. User-guided hierarchical attention network for multi-modal social image popularity prediction. In *WWW*, pages 1277–1286, 2018.

[Zhao *et al.*, 2015] Qingyuan Zhao, Murat A. Erdogdu, Hera Y. He, Anand Rajaraman, and Jure Leskovec. Seismic: A self-exciting point process model for predicting tweet popularity. In *SIGKDD*, pages 1513–1522, 2015.