# Incorporating Message Embedding into Co-Factor Matrix Factorization for Retweeting Prediction

Can Wang[12], Qiudan Li[1], Lei Wang[1]
[1]The State Key Laboratory of Management and Control for Complex Systems
Institute of Automation, Chinese Academy of Sciences
Beijing 100190, China
[2]University of Chinese Academy of Sciences, Beijing, China
{wangcan2015, qiudan.li, l.wang}@ia.ac.cn

Daniel Dajun Zeng[123]
[3] Department of Management Information Systems
University of Arizona
Tucson, Arizona, USA
dajun.zeng@ia.ac.cn

**Abstract—With the rapid growth of Web 2.0, social media has become a prevalent information sharing and spreading platform, where users can retweet interesting messages. To better understand the propagation mechanism for information diffusion, it is necessary to model the user retweeting behavior and predict future retweets. Some existing work in retweeting prediction based on matrix factorization focuses on using user-message interaction information, user information and social influence information, etc. The challenge of improving prediction performance is how to jointly perform deep representation of these information to solve the sparsity problem and then learn a more comprehensive retweeting behavior model. Inspired by word2vec and co-factor matrix factorization model, this paper proposes a hybrid model, called HCFMF, for learning users' retweeting behavior, it first computes the message content similarity by considering the message co-occurrence, the author information and word2vec based low-dimensional representation of content, then, jointly decomposes the user-message matrix and message-message similarity matrix based on a co-factorization model. We empirically evaluate the performance of the proposed model on real world weibo datasets. Experimental results show that taking the dense representation of author and content information into consideration could allow us make more accurate analysis of users' retweeting patterns. The mined patterns could serve as a feedback channel for both consumers and management departments.**

***Keywords—retweeting prediction; co-factor matrix factorization; word2vec; low-dimensional representation***

## I. INTRODUCTION

The rapid growth of Web 2.0 has made social media a significant platform for information sharing and spreading. Take Sina-weibo, one of the most popular Twitter-like micro-blogging in china, as an example, on the platform, users can retweet interesting messages, which allows other users to track the flow of information. By making good use of these retweeting data, one can gain better insights into the propagation mechanism for information diffusion, leading to better-informed decisions and more effective policy implication. For examples, corporations can better understand users' interests and then formulate targeted marketing plans, management departments can learn who and what they need to spread their messages as soon as possible. Therefore, to provide better information service for corporations and management departments, it is necessary to analyze the retweeting data and predict whether a message can be retweeted or not by a user.

Existing studies are classified into matrix factorization based and classification/regression based methods, which investigates the problem from different points of view. Matrix factorization based methods use users' historical retweeting data to perform prediction. A probabilistic collaborative filtering model was adopted to predict future tweets[1]. [2] focused on simultaneously predicting user decisions and modeling content in Twitter based on Co-Factorization Machines (CoFM). A one-class collaborative filtering method was used to predict user retweeting behavior [3]. [4] proposed an attention-based deep neural network, which solves the feature engineering problem of classification /regression based methods, it combined the user embedding, the user's attention interests embedding, the similarity score, the tweet embedding and the author embedding into a fixed feature vector to predict retweet behavior. The behavior of retweeting is influenced by some factors such as author information, content information, and user interests, etc[5], [6]. However, these information is often very sparse, causing the performance of retweeting prediction methods to degrade significantly. Hence, the challenge of improving prediction performance is how to take all of these information into consideration and learn a dense representation to solve the sparsity problem.

Recently, [7] proposed a co-factorization model, CoFactor, which jointly decomposes the user-item interaction matrix and the item-item co-occurrence matrix with shared item latent factors, the method can be interpreted as factorizing the word co-occurrence matrix. It can be seen from the above analysis that in retweeting scenarios, users tend to retweet messages with similar content and authors, thus, it is important to take these information into account when modeling retweeting behavior.

In this paper, we focus on learning users' retweeting behavior representations from message content and author information. To capture the interactive effect between user-message interaction and deep semantics of message content, a

hybrid co-factor matrix factorization approach, HCFMF, is proposed to generate more accurate representation of user behaviors, it first computes the message content similarity by considering the author information and learned low-dimensional representation of content, then, simultaneously decomposes the user-message matrix and message-message similarity matrix based on CoFactor. The proposed model provides an efficient way for predicting future retweets using deep semantic mining and collaborative filtering.

We empirically evaluate the performance of the proposed model on real world weibo datasets. Experimental results show that taking information on author and content into consideration could allow us make more accurate analysis of users' retweeting patterns. The mined patterns could serve as a feedback channel for both consumers and management departments. A consumer could rely on the mined knowledge when looking for latest potentially interested information, the management departments could better understand the public opinions such as hot events, influential users, etc., then make better policy decisions.

Our contributions are summarized as follows:

- This work is a first step towards utilizing word2vec and collaborative filtering to analyze user's retweeting data and perform future retweets prediction.
- The model provides a unified framework that enables accurate retweets prediction by seamlessly integrating message embedding based on author information, co-occurrence similarity, semantic similarity and social similarity into co-factorization model, solving the sparsity problem and leading to a more comprehensive retweeting behavior model.
- We demonstrate the efficacy of the model on real-world datasets with quantitative and qualitative results, which help gain better insights into how consumers and management departments can make use of the mined knowledge in real scenarios.

The rest of this paper is organized as follows: In section II, we discuss relevant studies in the literature. The detailed procedure of our model is presented in Section III. We empirically evaluate our algorithm in Section IV. Section V sums up our study and discusses future research directions.

## II. Literature Review

Our work is related to retweeting behavior modelling, matrix factorization, deep learning and recommender systems. In this section, we review the related works.

### A. Retweeting Behavior Modelling

[1] firstly developed a probabilistic collaborative filtering model to predict future retweets. The results showed that the most important features for prediction are the identity of the source of the tweet and retweeter. [2] proposed Co-Factorization Machines (CoFM) to address the problem of simultaneously predicting user decisions and modeling content in social media by analyzing rich information gathered from Twitter. Since only the retweeted message can be observed, it is a one-class setting, where only positive examples or implicit feedbacks are available, [3] adopted one-class collaborative filtering method by considering the user personal preference and social influence between users and messages to predict user's retweeting behavior. To solve the feature engineering problem of classification/regression based methods, [4] proposed an attention-based deep neural network, which combines the user embedding, the user's attention interests embedding, the similarity score, the tweet embedding and the author embedding into a fixed feature vector to predict retweet behavior.

This paper aims to gain a systematic understanding of users' retweeting behaviors by learning the factors of users and messages based on co-factorization model and word2vec.

### B. Matrix Factorization

Matrix factorization is used to learn user and item factors in recommender systems, which regularizes the factors with side information such as user, item, and/or user-item covariates including item metadata and user demographics [7]. Collective matrix factorization [8] explored co-factorizing multiple matrices such as user-movie rating matrix and movie-genres matrix, which jointly factorizes these two matrices with shared movie factors to leverage the additional genre information for better preference prediction. [7] proposed a co-factorization model, CoFactor, which jointly decomposes the user-item interaction matrix and the item-item co-occurrence matrix with shared item latent factors. [9] used a word embedding model[10] to learn item embedding. User embedding are then inferred as to predict the next item in the trajectory. [11] proposed a temporal and social probabilistic matrix factorization model to predict users' potential interests in micro-blogging, which provides a unified way to fuse the time information and the social network structure to predict user interest. [12] proposed a method integrating social network structure and the user-item rating matrix. [13] proposed a SocialMF model by incorporating trust propagation into a matrix factorization for recommendation in social network.

Different from these methods, CoFactor based retweeting prediction model regularizes the message factors using the learned message similarity, which takes dense vector of content and author information into consideration.

### C. Deep Learning and Recommender Systems

Deep-learning methods are representation-learning methods, which can automatically discover multiple levels of representations from raw data. They are considered as promising methods for various tasks including topic classification, sentiment analysis, question answering and language translation [14]. Among many deep-learning methods, recursive neural network (RNN) and convolutional network (CNN) are very popular. Combining the benefits of collaborative filtering and deep learning for recommender systems is one of the hottest research topic. [15] presented the Wide & Deep learning framework to combine the strengths of wide linear models and deep neural networks. The authors productionized and evaluated the framework on the recommender system of Google Play. To solve the sparse problem of auxiliary information , [16] proposed a hierarchical Bayesian model called CDL, which jointly performs deep representation learning for the content information and collaborative filtering for the ratings matrix, the model was evaluated on CiteULike and Netflix datasets. [17] proposed a

novel deep neural network based architecture that models the combination of long-term static and short-term temporal user preferences to improve the recommendation performance, the model was applied to a commercial News recommendation system. To overcome the inherent limitation of the bag-of-words model, [18] proposed a novel context-aware recommendation model, convolutional matrix factorization (ConvMF) that integrates convolutional neural network (CNN) into probabilistic matrix factorization (PMF). MovieLens and Amazon datasets were used to test the efficacy of the proposed model. [19] introduced a Neural Network architecture, aka CFN, to perform Collaborative Filtering with side information, and tested it on Movielens data sets. [20] proposed infusing traditional user-based and item-based collaborative recommender systems with representations learned by applying the Continuous Bag of Words and Skip-gram models to a dataset of users, movies, directors, actors and tags. [21] developed two new models (BoWLF and LMLF) which exploit text reviews to regularize rating prediction on the Amazon Reviews datasets. [22] proposed to learn a semantic space in which substitutable items are positioned in close proximity. The authors evaluated the method on MovieLens data and showed that these spaces can be learned from item reviews as well as user-item ratings, using the same deep learning architecture. [23] proposed implicit CF-NADE, a neural autoregressive model for collaborative filtering tasks using implicit feedback (e.g. click/watch/browse behaviors). The training objective is to maximize a weighted negative log-likelihood. Dataset extracted from a digital TV streaming service was used to evaluate the model. [24] employed deep recurrent neural networks to provide vector representations for the text content associated with items in collaborative filtering and performed scientific paper recommendation. Word2vec is a deep-learning-inspired method that attempts to understand meaning and semantic relationships among words. It learns vector representations of words using continuous bag-of-words (CBOW) and Skip-gram [25]. Due to the good performance of capturing syntactic and semantic information, we adopt word2vec to learn dense vectors for content information.

Most of the existing approaches discussed above performed recommendation on data sets such as CiteULike, Netflix data, Movielens, Amazon reviews, TV, News, etc. Little work has been done on integrating word2vec and collaborative filtering for retweeting prediction. Based on the above interesting research work, our work develops a hybrid co-factor matrix factorization retweets prediction model (HCFMF) based on CoFactor and word2vec, it firstly constructs user-message retweeting interaction matrix, message-word matrix and message-user matrix, secondly, learns the latent representation of content by word2vec, thirdly, computing the message similarity using the dense vector of message and author information, fourthly, user-message matrix and message-message similarity matrix are simultaneously decomposed to learn the latent user factors and message factors, finally, the proposed model is applied to predict whether a message can be retweeted or not by a user.

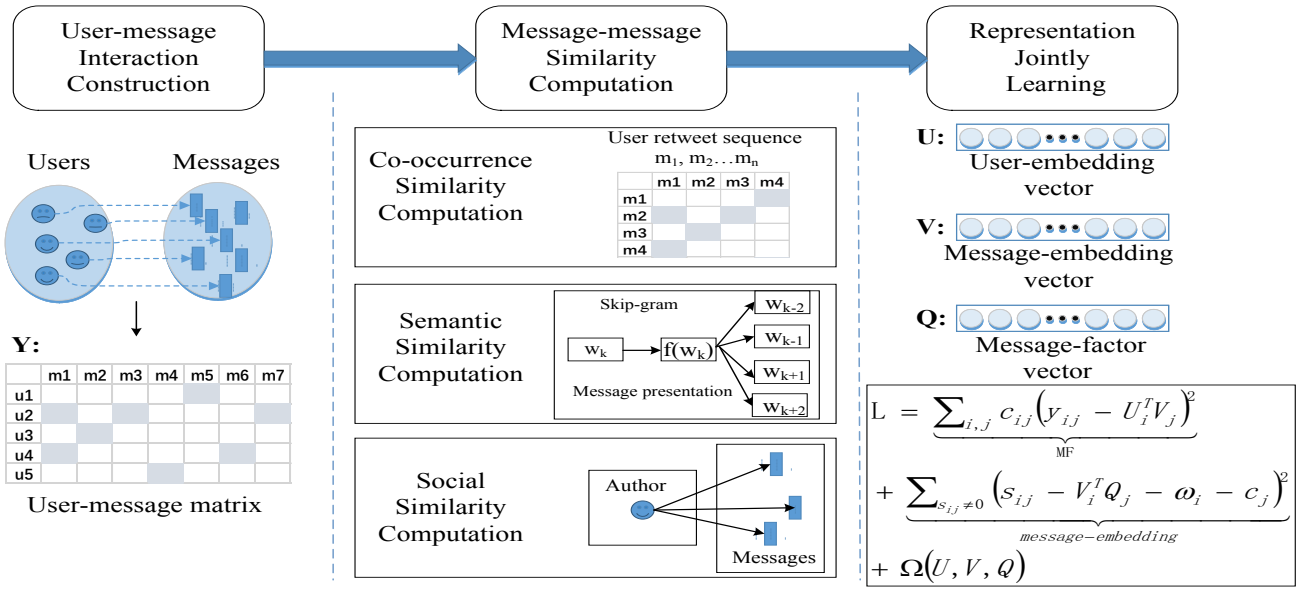


Fig. 1. System architecture of the HCFMF mode

## III. HCFMF Model for Retweeting Prediction

### A. Problem Definition

The retweeting behavior can reflect users' interests. If a user is interested in a message, she/he may retweet the message, therefore, other users can further keep track of it. Collaborative filtering-based retweeting prediction is to make automatic predictions about the interests of a user by collecting preferences from many users. The underlying assumption is that if a user A has the same interest as a user B on a message, A is more likely to have B's interest on a different message $x$ than to have the interest on $x$ of a randomly chosen user.

Suppose that there are $M$ users and $N$ messages. The user-message retweeting interaction can be represented as a $M \times N$ sparse matrix $Y \in \mathbb{R}^{M \times N}$. The $i^{th}$ row in the matrix corresponds to user $u_i$, and the $j^{th}$ column corresponds to

message $m_j$. Each entry $y_{ij}$ in $Y$ can be 0 or 1. $y_{ij} = 1$ means that user $u_i$ has retweeted the message $m_j$, and $y_{ij} = 0$ otherwise. Our goal is to predict which message that each user will retweet based on the observed entities in user-message matrix $Y$. Message content information describes the topic of the message, which further indicates a user's topical interest. However, in microblogging systems, the content information is high-dimensional and sparse, how to learn the low-dimensional representation and capture the deep semantics of the message is a challenging problem. The recent success in word2vec provides an efficient way to solve the problem. This paper aims to perform future retweets prediction by effectively integrating dense representation of message content and user-message retweeting interaction.

### B. System Architecture of the Proposed Model

The overview of our model is shown in Fig. 1, which consists of three functional modules, namely, user-message interaction matrix construction, message-message similarity computation and representation jointly learning based on co-factorization. In user-message interaction construction module, we obtain user retweeting interaction information from past retweeting data, and construct user-message interaction graph. In message-message similarity computation module, we propose a hybrid computation method based on co-occurrence similarity, semantic similarity and social similarity, which takes author information and dense vector representation into account, thus help mine similar messages more exactly. The representation jointly learning module aims to infer latent message representation by both matrix factorization and message embedding, the former encodes users' interest in messages, while the latter explains the message co-occurrence, semantics, social patterns. Considering all the above mentioned patterns will lead to a more comprehensive retweeting behavior model. We describe the message-message similarity computation module and representation jointly learning module in detail below.

#### 1) Message-message similarity computation

Users retweet messages based on their interests. Latest research in [7] has discovered that message-message similarity is an important indicator whether a user will like the message or not. Since many factors may affect the retweeting behavior, such as content freshness, author information and so on. Therefore, we propose a hybrid computation method taking these factors into consideration, which is based on co-occurrence similarity, semantic similarity and social similarity.

#### 2) Co-occurrence similarity computation

Potentially similar messages always frequently co-occur. We quantify the co-occurrence similarity of two messages based on SPPMI as follows [7]:

$$S_{co-occurrence}(m_i, m_j) = max\left\{log\frac{n_{pair(m_i,m_j)} \cdot P}{n_{pair(m_i)} n_{pair(m_j)}} / logP\ , 0\right\} \quad (1)$$

Where $m_i$ and $m_j$ denote message $i$ and message $j$. $n_{pair(m_i,m_j)}$ is the number of times that the two messages retweeted by the same user. $n_{pair(m_i)} = \sum_j n_{pair(m_i,m_j)}$ and $n_{pair(m_j)} = \sum_i n_{pair(m_i,m_j)}$. $P$ is the total number of message pairs. We use $logP$ to scale the similarity to [0,1].

#### 3) Semantic similarity computation

Many research has explored that the low-dimensional vector of sparse message can generate better semantic representation. We adopt SkipGram [25] to construct the latent semantic vector for word embedding, then learn latent vector of message, finally, semantic similarity between two messages are calculated as follow:

$$V(m_i) = \frac{1}{|L(i)|}\sum_{a\in L(i)} W_a \quad (2)$$

$$S_{semantic}(m_I, m_j) = \frac{V(m_i)V(m_j)}{\|V(m_i)\|\|V(m_j)\|} \quad (3)$$

Where $L(i)$ is the set of words in message $m_j$, $W_a$ is the low dimensional representation of word a, $V(m)$ is the vector representations for message $m$.

#### 4) Social similarity computation

In real scenarios, whether a user will retweet the message is largely depended on the author of the message. Users are more likely to retweet messages published by the same author. Hence, we measure the social similarity between two messages as follows:

$$S_{social}(m_i, m_j) = I(i,j) \quad (4)$$

Where $I(i,j)$ is a two-value function, the value is either 0 or 1. When messages $i$ and $j$ share the same author, $I(i,j) = 1$, otherwise, $I(i,j) = 0$.

#### 5) Message-message similarity computation

Based on the above description, we compute the message-message similarity using a linearly ensemble of co-occurrence similarity, semantic similarity and social similarity as follows:

$$S(m_i, m_j) = \frac{(S_{co-occurrence}+S_{semantic}+S_{social})}{3} \quad (5)$$

#### 6) Representation jointly learning based on Co-factorization

Given the latent semantic dimension $K$, $U \in \mathbb{R}^{K\times M}$ is the latent feature matrix of the user, and the $i^{th}$ column is donated as the feature of user $i$. Meanwhile, $V \in \mathbb{R}^{K\times N}$ is the latent feature message matrix, and the $j^{th}$ column is donated as the feature of message $j$. And $Q \in \mathbb{R}^{K\times N}$ is the item factor shared by both the MF part and the message-message similarity embedding part.

$$L_{co-factor} = \sum_{i,j} c_{ij}\left(y_{ij} - U_i^T V_j\right)^2 + \sum_{W_{ij}\neq 0}\left(W_{ij} - V_i^T Q_j - \omega_i - c_j\right)^2 + \lambda_u\|U\|_F^2 + \lambda_V\|V\|_F^2 + \lambda_Q\|Q\|_F^2 \quad (6)$$

where $\|\cdot\|_F^2$ denotes the square of the L2 norm. $\lambda$ is the hyperparameter for the regularization. $W \in \mathbb{R}^{N\times N}$ is the message-message factor matrix with nonnegative weighted value in which $W_{ij}$ reflects the better message embedding $V$. $c_{ij}$ is scale parameter used to balance the missing examples and the positive ones in the user-message matrix $Y$. $\omega$ and $c$ are bias terms. The last three regularization terms are used to avoid overfitting.

Based on the above analysis, by considering message-message similarity based message embedding, we generate the message-message factor matrix $S \in \mathbb{R}^{N\times N}$ as follows:

$$S_{ij} = S(m_i, m_j) \tag{7}$$

Similar to CoFactor, we only consider the non-negative value in $S$ when computing similarity. Then, given the user-message retweeting matrix $Y$ and the message-message factor matrix $S$, the previous optimization problem can be rewritten as follows:

$$L_{co-factor}(U,V,Q) = \sum_{i,j} c_{ij}\left(y_{ij} - U_i^T V_j\right)^2 + \sum_{S_{ij}\neq 0}\left(S_{ij} - V_i^T Q_j - \omega_i - c_j\right)^2 + \lambda_u \|U\|_F^2 + \lambda_V \|V\|_F^2 + \lambda_Q \|Q\|_F^2 \tag{8}$$

We use stochastic gradient descent method to solve the above optimization problem. Namely, starting with zero initialization on $U$, $V$, $Q$, $\omega$, $c$, these parameters in the above formula are optimized as follows [6]:

$$U_i = \left(\sum_j c_{ij} V_j V_j^T + \lambda_u I_K\right)^{-1}\left(\sum_j c_{ij} y_{ij} V_j\right) \tag{9}$$

$$V_j = \left(\sum_i c_{ij} U_i U_i^T + \sum_{i:S_{ji}\neq 0} Q_i Q_i^T + \lambda_v I_K\right)^{-1} * \left(\sum_i c_{ij} y_{ij} U_j + \sum_{i:S_{ji}\neq 0}\left(S_{ji} - \omega_j - c_i\right) Q_i\right) \tag{10}$$

$$Q_j = \left(\sum_{i:S_{ji}\neq 0} V_i V_i^T + \lambda_Q I_K\right)^{-1} * \left(\sum_{i:S_{ij}\neq 0}\left(S_{ij} - \omega_i - c_j\right) V_i\right) \tag{11}$$

$$\omega_i = \frac{1}{|\{j:S_{ij}\neq 0\}|}\sum_{j:S_{ij}\neq 0}\left(S_{ij} - V_i^T Q_j - c_j\right) \tag{12}$$

$$c_j = \frac{1}{|\{i:S_{ij}\neq 0\}|}\sum_{i:S_{ij}\neq 0}\left(S_{ij} - V_i^T Q_j - \omega_i\right) \tag{13}$$

A weighted alternating least squares scheme is adopted to update the above parameters. Compared with the traditional matrix factorization, we update the item matrix $V$ based on two parts including user embedding part $U$ and item factor part $Q$. We learn each of the above parameters by fixing other parameters step by step until convergence.

## IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

### A. Dataset

TABLE I. THE DETAILED INFORMATION OF DATASETS

| dataset | # users | # tweets | # retweets | Sparsity |
|---|---|---|---|---|
| D100 | 8727 | 672 | 59023 | 1/100 |
| D200 | 12778 | 1211 | 83786 | 1/200 |
| D500 | 17603 | 2149 | 113382 | 1/500 |
| D1000 | 24594 | 6238 | 156794 | 1/1000 |

We use a large-scale publicly available dataset released by [6], which was crawled from Sina Weibo where users can follow other users and receive the messages from the followed users, similar to Twitter. In this paper, we construct 4 datasets D100, D200, D500, D1000 with different sparsity based on the retweets number of users and posts. The users who retweet more than 5 messages are kept in all datasets, the messages that were retweeted more than 80, 60, 40, 5 times are kept in datasets D100, D200, D500, D1000, respectively. Table I summarizes the detailed information of these datasets.

To validate the proposed method, we randomly sample 80% of the preprocessed dataset as training set, from which we randomly selected 10% as validation set. The remaining 20% is used as the test set.

### B. Baseline Methods

- CoFactor [7]: the model decomposes the user-item interaction matrix and the item-item co-occurrence matrix with shared item latent factors.
- Proposed HCFMF Model: Based on CoFactor, the model considers semantics and author information of message.

### C. Evaluation Measures

Recall@M, truncated normalized discounted cumulative gain (NDCG@M) and mean average precision (MAP@M) are main ranking-based metrics, which are used to evaluate the proposed model. For each user, all the metrics compare the predicted rank of (unobserved) messages with their true rank [7].

Recall@M considers all messages ranked within the first M to be equivalent, NDCG@M and MAP@M use a monotonically increasing discount to emphasize the importance of higher ranks versus lower ones. Formally, define $\pi$ as a permutation over all the messages, $I\{.\}$ is the indicator function, $u(\pi(i))$ returns 1 if user $u$ has retweeted message $\pi(i)$, HCFMF predicts ranking π for each user by sorting the predicted preference $U_u^T V_i$, for $i=1\cdots I$.

Recall@M for user u is defined as follows:

$$Recall@M(u,\pi) = \sum_{i=1}^{M} \frac{I\{u(\pi(i))=1\}}{min(M, \sum_{i'}^{I} I\{u(\pi(i'))=1\})} \tag{14}$$

Denominator evaluates the minimum between $M$ and the number of messages retweeted by user $u$. The maximum value of Recall@M is 1, which means ranking all relevant messages in the top M positions.

NDCG@M for user $u$ is defined as follows:

$$NDCG@M(u,\pi) = \sum_{i=1}^{M} \frac{2^{I\{u(\pi(i))=1\}}-1}{log(i+1)} \tag{15}$$

NDCG@M is the DCG@M normalized to [0,1], where one indicates a perfect ranking.

Mean average precision (MAP@M) calculates the mean of users' average precision (AP). The average precision AP@M for a user $u$ is defined as follows:

$$AP@M(u,\pi) = \sum_{i=1}^{M} \frac{Precision@i(u,\pi)}{min(i, \sum_{i'}^{I} I\{u(\pi(i'))=1\})} \tag{16}$$

### D. Experimental setting

Parameter user scaling $\lambda_U$, item scaling $\lambda_V$, factor scaling $\lambda_Q$: these hyperparameter parameters are used to adjust the weight of the regularization terms in the objective function. We choose the best value according to the performance of the model based on the validation set. We set $\lambda_Q = e^{-5}$ and $\lambda_U = \lambda_V = e^{-5} * 0.03$.

Parameter $c_{y=0}$ and $c_{y=1}$: these parameters plays the role in adjusting the strengths of positive examples and negative examples in the retweeting matrix. We set $c_{y=0} = 0.03$ and $c_{y=1} = 0.3$.

Parameter $K$: In CoFactor and our model HCFMF, we set the dimension of the latent space $K$ as 50. The number of iterations is set to be 20.

TABLE II. PERFORMANCE COMPARISON ON THE WHOLE DATASET D1000 AND ALL METRICS

| | Recall@20 | Recall@50 | NDCG@100 | MAP@100 |
|---|---|---|---|---|
| **CoFactor** | 0.0998 | 0.1731 | 0.0775 | 0.0302 |
| **HCFMF** | **0.1142** | **0.1943** | **0.0874** | **0.0355** |

### *E. Experimental Analysis*

Our goal is to predict which message will be retweeted based on the user-message interaction matrix and the message-massage similarity matrix.

#### *1) Quantitative Comparison*

We compare the performance of the method with that of the baseline method on the whole data set D1000 based on the above evaluation metrics. It can be seen from the results shown in Table II that the proposed model performs better than CoFactor across all metrics, especially in terms of Recall@20 and Recall@50. The model can obtain better performance than CoFactor using joint representation of user-message matrix and message-message similarity matrix, which indicates that the semantic similarity and the social similarity have impacts on the performance of retweeting prediction. This further proves that learning joint representation of users and messages is effective for generating more accurate retweeting behavior model.

Fig. 2 shows the overall performance when we vary the number of returned posts $M = 20,40,\ldots,200$. It shows that the improvement is greater when the number of returned posts is larger. By taking into account the content and the social information of the posts, HCFMF has better performance.

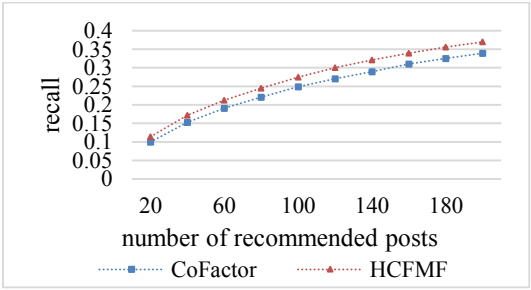


Fig. 2. Recall@M comparison on D1000

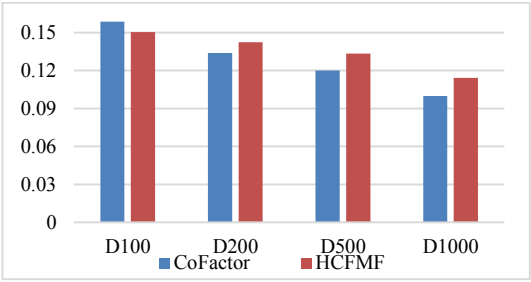


Fig. 3. Recall@20 comparison on different datasets

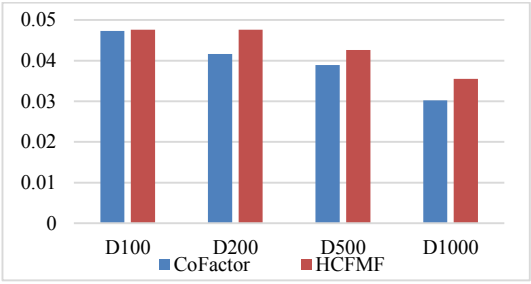


Fig. 4. MAP@100 comparison on different datasets

TABLE III. MOST RETWEETED MESSAGES WHOSE RETWEETS NUMBER IS MORE THAN 200

| ID | Message Content | The number of retweets |
|---|---|---|
| **1** | **Topic words:** Joanne Kathleen Rowling, Harry Potter film, bbc interview, Harry Potter series, hasty conception, conclusion, magical tale,episode 8, novel sequel, xian evening news | 342 |
| **2** | **Topic words:** gangster film, Crouching Tiger, actor, Eric Tsang, Michael Miu, Allan Wu, Jordan chan, cheung chi lam, Sonija Kwok, Yuan Nie, Hailu Qin, Shawn Yue, Cantonese, Chinese, wonderful, film, attention | 212 |
| **3** | **Topic words:** fill in gaps, Prometheus Bound, movie Alien, show, science fiction, heated discussion, involved in, history of film, classic, film, article, trace back, spoiler | 202 |

TABLE IV. THE TOP5 MOST RETWEETED MESSAGES PUBLISHED BY THE SAME USER

| ID | Message Content | The number of retweets |
|---|---|---|
| **1** | How does the 17-square-meter house seem to be two rooms, one hall and one bath room? What a super thing! | 180 |
| **2** | It is as beautiful as the Crystal Palace, making one feel heartbreaking. | 170 |
| **3** | Radisson blu hotel in Berlin, Germany has the world's largest cylindrical aquarium, the hotel visitors can stay in the hotel corridor and see a million liters of fish tanks and more than 1,500 tropical fish in the room. | 159 |
| **4** | Daughter's childhood. | 155 |
| **5** | Always wanted to have such a façade to do some favorite business downstairs, and live leisurely upstairs. | 133 |

To further evaluate the performance of the model in different sparse settings, Fig. 3 and Fig. 4 show Recall@20, MAP@100 comparison results of two methods on different datasets. It can be observed from Figure2 and Figure3 that compared with Cofactor, the HCFMF model achieves improvements. By further integrating semantic similarity and social similarity into Cofactor, the HCFMF model has better performance on D200, D500 and D1000 datasets in Recall@20, and has better performance on all datasets in MAP@100. The good performance indicates that the model is robust to capture the semantic representations of message content, thus solving the sparse problem of message.

*2) Qualitative Analysis*

To gain better insights into the retweeting behaviors of messages with similar content and same author, we analyzed the top most retweeted messages with similar topic and published by the same author.

*a) Retweeting behavior analysis of messages with similar content*

The top3 most retweeted messages whose retweets number is more than 200 are listed in Table III, it can be seen from the results that the topics of the messages are all about movies, indicating that similar messages have similar retweeting behavior.

*b) Retweeting behavior analysis of the same author*

The top5 most retweeted messages published by the same author are shown in Table IV, we can see that the author is interested in house decoration and renting. The author is very popular, whose messages have been retweeted many times.

*F. Case Study*

*1) Topic analysis of past retweets and predicted retweets*

To verify whether the proposed model is able to capture the semantics of the message content, we randomly select 2 users and analyze the topics of past retweets and predicted retweets mined by Cofactor and the proposed model. From Table V, we can see that user1 may be interested in entertainment and event news in the past, on the whole, the topics of the predicted retweets mined by the two models belong to these two kinds. Because the proposed model further takes the detailed content into consideration, therefore, compared with Cofactor, the method can capture more fine-grained topics. Take his/her interested Chris Lee, a famous singer in China, as an example, he/she has retweeted some messages about Chris Lee's songs, album, news, by mining the semantics of the past retweeted messages, the topics of retweets predicted by the proposed model are Chris Lee's other information such as her beneficence, reports, etc. The predicted information can help the user comprehensively know about his/her favorite star. As for user2, the model finds that his/her focus is on news about film, digital product, flight, hence, the proposed model predicts that the user will retweet messages about news on star, The Voice of China, Air China, opening ceremony, moreover, the message published by the same author is predicted in the proposed model compared with Cofactor, enhancing the diversity of the discovered topics.

TABLE V. TOPIC ANALYSIS OF TWO USERS

| **User1** | | |
|---|---|---|
| **The topic of past retweets** | **The topic of predicted retweets by the model** | **The topic of predicted retweets by CoFactor** |
| **Topic words:** Chris Lee, Chunchun, song, album, dream, Speech of Offering an Award,stage, magnificent, dancer, wonderful, narration, patriotic movement, recapture, fish island, participate | **Topic words:** Chris Lee, China, fish island , maritime patrol , sea, guard, refute a rumor, fans, truth, network, beneficence, gratitude,media, reader,school | **Topic words:** invitation, follow, first, music box, dreamer, Beijing, school, receive, a notice to shut down, love, like, respect, byebye, Obama, Romney, picture, network, Smile, charming, temperament, dress, figure, on time, outlook on life |
| **User 2** | | |
| **The topic of past retweets** | **The topic of predicted retweets by the model** | **The topic of predicted retweets by CoFactor** |
| **Topic words:** Iphone5, ipad,material, earphone, connector,game, national flag, rowing, Olympics, London, endeavor, achievement,flight delay,news, harmony, rumor,film,performance,artist, Qiang Chen,death,classic,film,role | **Topic words:** opening ceremony, Japanese delegation, whole, the world, flight attendant, minister, Air China, Hefei, fly to, Guangzhou, game, like, The Voice of China, Hospital, cold, honey, throat, lemon water, Ham Yu, face, scar, burn, event | **Topic words:** Harbin, bridge, construction, responsible for the accident, Obama, Romney, difference, picture, network, opening ceremony, Japanese delegation whole, the world, game, like, The Voice of China, feeling,bud,care,love, protect |

*2) Hot topics mined from predicion results*

By mining the topics from the messages with a large number of retweets, management departments can be fully aware of the public's focus and concern, thus make better decision makings. Fig. 4 shows the tag cloud drawn from the top most retweeted messages on education (in red), reform (in orange) and people's livelihood (in black), where larger font indicates more popular information. It can be seen from the figure that the topic of education has received the most attention among the three topics. The main themes of people's discussions on education involves finance, employment, quality, talent, degree, Professor, campus, country. People's livelihood topic includes aspects including autism, life, situation, family, one million, support, promote, insurance policy. The concerns on reform include National Development and Reform Commission, open, cost, ladder type, water price, tap water, residents, push forward.

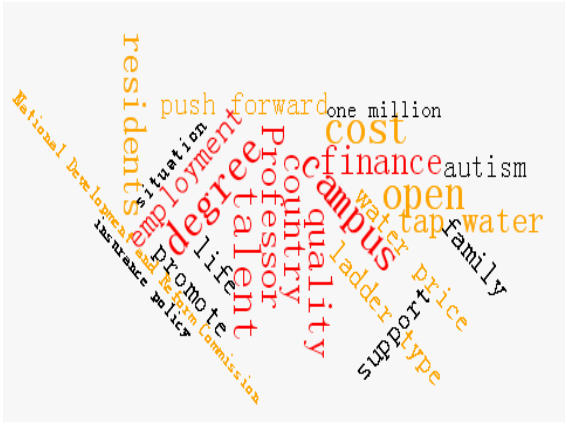


Fig. 5. The tag cloud of topics on education, reform and people's livelihood

## V. Conclusions

In this paper, we propose a novel model to predict future retweets, which models message embedding by taking the message co-occurrence, semantics, social patterns into consideration, then generate latent message representation by both matrix factorization and message embedding based on CoFactor. Experimental results on Weibo data set show the efficacy of the proposed model. This work is a first step towards utilizing word2vec and collaborative filtering to analyze user's retweeting data and perform future retweets prediction. The model can integrate other information such as user embedding, temporal information, etc. in the future.

## Acknowledgment

This research is supported by the Key Research Program of the Chinese Academy of Sciences under Grant No. ZDRW-XH-2017-3; National Natural Science Foundation of China under Grant No. 71621002,61671450, 61402123.

## References

[1] T. R. Zaman, R. Herbrich, J. Van Gael, and D. Stern, "Predicting information spreading in twitter," in *Workshop on computational social science and the wisdom of crowds, nips*, 2010, pp. 17599-601.

[2] L. Hong, A. S. Doumith, and B. D. Davison, "Co-factorization machines: modeling user interests and predicting individual decisions in twitter," in *Proceedings of the sixth ACM international conference on Web search and data mining*, 2013, pp. 557-566.

[3] B. Jiang, J. Liang, Y. Sha, R. Li, W. Liu, H. Ma, *et al.*, "Retweeting Behavior Prediction Based on One-Class Collaborative Filtering in Social Networks," in *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, 2016, pp. 977-980.

[4] Q. Zhang, Y. Gong, J. Wu, H. Huang, and X. Huang, "Retweet Prediction with Attention-based Deep Neural Network," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 2016, pp. 75-84.

[5] Q. Zhang, Y. Gong, Y. Guo, and X. Huang, "Retweet Behavior Prediction Using Hierarchical Dirichlet Process," in *AAAI*, 2015, pp. 403-409.

[6] J. Zhang, J. Tang, J. Li, Y. Liu, and C. Xing, "Who influenced you? predicting retweet via social influence locality," *ACM Transactions on Knowledge Discovery from Data (TKDD),* vol. 9, p. 25, 2015.

[7] D. Liang, J. Altosaar, L. Charlin, and D. M. Blei, "Factorization Meets the Item Embedding: Regularizing Matrix Factorization with Item Co-occurrence," in *Proceedings of the 10th ACM Conference on Recommender Systems*, 2016, pp. 59-66.

[8] A. P. Singh and G. J. Gordon, "Relational learning via collective matrix factorization," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008, pp. 650-658.

[9] E. Guàrdia-Sebaoun, V. Guigue, and P. Gallinari, "Latent Trajectory Modeling: A Light and Efficient Way to Introduce Time in Recommender Systems," in *Proceedings of the 9th ACM Conference on Recommender Systems*, 2015, pp. 281-284.

[10] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111-3119.

[11] H. Bao, Q. Li, S. S. Liao, S. Song, and H. Gao, "A new temporal and social PMF-based method to predict users' interests in micro-blogging," *Decision Support Systems,* vol. 55, pp. 698-709, 2013.

[12] H. Ma, T. C. Zhou, M. R. Lyu, and I. King, "Improving recommender systems by incorporating social contextual information," *ACM Transactions on Information Systems (TOIS),* vol. 29, p. 9, 2011.

[13] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *Proceedings of the fourth ACM conference on Recommender systems*, 2010, pp. 135-142.

[14] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature,* vol. 521, pp. 436-444, 2015.

[15] H.-T. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, *et al.*, "Wide & Deep Learning for Recommender Systems," in *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, 2016, pp. 7-10.

[16] H. Wang, N. Wang, and D.-Y. Yeung, "Collaborative deep learning for recommender systems," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015, pp. 1235-1244.

[17] Y. Song, A. Elkahky, and X. He, "Multi-Rate Deep Learning for Temporal Recommendation."

[18] D. Kim, C. Park, J. Oh, S. Lee, and H. Yu, "Convolutional Matrix Factorization for Document Context-Aware Recommendation," in *Proceedings of the 10th ACM Conference on Recommender Systems*, 2016, pp. 233-240.

[19] F. Strub, R. Gaudel, and J. Mary, "Hybrid Recommender System based on Autoencoders," in *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, 2016, pp. 11-16.

[20] G. Zanotti, M. Horvath, L. N. Barbosa, V. T. K. G. Immedisetty, and J. Gemmell, "Infusing Collaborative Recommenders with Distributed Representations," in *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, 2016, pp. 35-42.

[21] A. Almahairi, K. Kastner, K. Cho, and A. Courville, "Learning distributed representations from reviews for collaborative filtering," in *Proceedings of the 9th ACM Conference on Recommender Systems*, 2015, pp. 147-154.

[22] J. B. Vuurens, M. Larson, and A. P. de Vries, "Exploring Deep Space: Learning Personalized Ranking in a Semantic Space," in *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, 2016, pp. 23-28.

[23] Y. Zheng, C. Liu, B. Tang, and H. Zhou, "Neural Autoregressive Collaborative Filtering for Implicit Feedback," in *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, 2016, pp. 2-6.

[24] T. Bansal, D. Belanger, and A. McCallum, "Ask the GRU: Multi-task Learning for Deep Text Recommendations," in *Proceedings of the 10th ACM Conference on Recommender Systems*, 2016, pp. 107-114.

[25] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781,* 2013.