User Reactions Prediction Using Embedding Features

Samin Mohammadi, Reza Farahbakhsh, Noel Crespi
Institut Mines-Telecom, Telecom Sud-Paris, CNRS UMR 5157, France.
{samin.mohammadi, reza.farahbakhsh, noel.crespi}@it-sudparis.eu

***Abstract*—By the massive available people data in social media, many digital service providers exploit widely this information to improve their services by predicting future requirements of their customers. This prediction mainly needs to study users' previous behavior and interactions and identify their preferences to provide rigorous recommendations that fulfill their requirements more favorably. Meanwhile, experiments show the prediction methods which exploit representation learning instead of traditional hand-crafted features accomplish better results and more precise predictions. In this study, we take advantage of representation learning method to predict user's future interactions by extracting users embeddings from their reactions history and exploit them in predicting future reactions. In this approach, users embeddings are used in a neural network designed with one-hidden layer and a softmax function in the end layer in order to predict users reactions. The proposed method is evaluated when user embeddings come from two different sources; users reactions history and random walks on the user network. The performance of the method has been evaluated by using a large Flickr dataset including more than 2M users and 11M users reactions sequences. The results show outperforming of the prediction method when it uses the history of user reactions to derive user embeddings.**

Index Terms Social Media, Learning, Prediction, User embedding

I. Introduction

Nowadays, every aspect of human life has been widely affected by social media. An enormous amount of data is uploaded to Online Social Networks (OSNs) every day which is analyzed and employed to improve the user services provided by those networks. Among the different research directions through analyzing social media data, predicting user's future behavior in order to serve efficient user services is one of the most attractive studies. User behavior prediction plays a key role in a wide range of applications such as recommender systems, content delivery networks, advertising campaign, election results prediction and the list goes on. User behavior comprises her preferences, her interaction [1] type such as post, comment, share, like, and so on. In this study we will focus on predicting user behavior in terms of her interaction on social media.

Once a post is published on a social network, depending to its interestingness for other users, it could attract a particular amount of user interactions. Predicting the amount of user interactions and more interestingly the users who will react to that particular post are two main trending research tracks. Some studies have focused on predicting the final size of the popularity of a content to provide a vision of trending content [1] [2]. But some others have targeted more details and tried to predict the users who will make a content popular in addition to its final popularity size [3] [4]. This study will focus on predicting the users who will interact with a newly published content in near future.

Usually prediction tasks on social networks are based on learning methods which need features to be used in the model. Finding the most efficient features that provide more accurate prediction is always one of the main challenges on using conventional learning methods such as classification, clustering, regression, etc [5][2]. Recent success stories of deep learning in extracting embedding features have led to exploit this method in different data mining tasks by skipping the manually feature extraction phase [6][7]. Aligned to this research direction, this study aims to take advantage of representation learning in order to learn user features without require to hand-crafted features. It will use users interaction history to extract their embedding features and exploit those features in prediction whether they will react to a post published by one of them.

As users react to posts that they are more interested or have friendship with the posts' publishers, a sequence of users interactions and their co-occurrence in that sequence can represent their common preferences and interests. The proposed model exploits users' reaction log as the input of the word2vec model to derive user embedding features. Depending on the type of social network, the reaction of users can be in the form of re-share, like, or comment on the post. Since we use Flikr data in this study, *marking a photo as favorite* is considered as user reaction. Previously, some models such as node2vec also have extracted user embeddings to exploit in prediction tasks such as multi-label classification [8] [9]. However, the input of node2vec is random walks over the users network graph which are not applicable for our following purpose. Because our goal is to find such features that can represent users tendency to react to a post but not their neighborhood in the graph emerging in random walks can provide this tendency. Therefore, the hypothesis of deriving features from feeding interaction logs to the word2vec model will be followed in this study which provides better features to take their benefit in interaction prediction task. Users embeddings are exploited in a one-hidden layer neural network with a softmax function in the end layer to predict users reactions. We compared the

[1]In this study, the word interaction is used interchangeably with reaction.

results when users embeddings come from the node2vec model and from the reaction sequences.

Using users' reactions log to learn their embeddings and predict their future interactions are the main contributions of this study. Besides, the proposed model in this paper is general and can be applied to any social networks data. Our experiments show more accurate results compared to other existing approaches which can potentially be used in different recommendation scenarios.

The rest of this paper is organized as follows: Section II summarizes the relevant previous studies. The proposed methodology is presented in Section II-B1. Dataset description and the evaluation results of prediction task are discussed in Sections III-B, and Section IV-B3 concludes the study and points avenues for future research.

II. Related Work

The previous studies relevant to this paper can be summarized in two main parts: (i) popularity prediction on OSNs and (ii) representation learning on social media.

A. *Prediction of Popularity*

Prediction of popularity is an interesting research topic which can be investigated about users [10] or contents [11]. From the content perspective, once a content is published on a social network, it attracts different amount of users interactions depending on its interestingness, topic, publisher's reputation, published time and etc. [12] [13]. Meanwhile, some contents are succeeded to attract more user engagements and become popular [14]. Popularity of a content usually assesses by different cascading metrics such as number of likes, shares, views, etc.

Predicting the trend of popularity for a content (which can be a text, video, or image) and more importantly identifying the users who are going to react to that content are very valuable information for different entities such as service providers to rank the content better [15], to early discover of trending posts, to improve recommendations and even to improve their content delivery networks and user experiences [16]. This kind of prediction tasks are mainly based on the features of contents and early adapters. Depending on the social network's type, adapters can be interpreted as either likers, resharers, viewers, or so on. In [17], popularity of a content is predicted using the structural diversity of early adapters. In other studies, temporal features of early adapters are realized as the most predictive features among different features of content, user and network [2] [5] [1].

Looking the models that have been developed on different content popularity prediction tasks on OSNs shows that most of them focused on predicting the popularity size of contents in future. There are very rare researches on identifying the users who are going to react to the contents published on OSN in future [3]. Although, interactors prediction on OSNs is somehow similar to well-studied rate prediction on recommender systems (RS), but there is a main different which makes RS models improper to apply directly on interactors prediction on OSNs. Rate prediction models on RS are mainly based on interest, whereas OSNs models are primarily based the mixure of friendship and interest. Petrovic *et al.* [4] tried to predict interactors using a machine learning method based on the passive-aggressive algorithm. Authors in [18] have proposed a tree-structured long short-term memory network to learn and predict the entire diffusion path of an image in a social network. In [19], authors investigate the sequential prediction of popularity by proposing a prediction framework, by incorporating temporal context and temporal attention into account. Our study is different than the mentioned and other similar studies [20] [21] because of focusing on only users latent likelihood extracted from their reaction history.

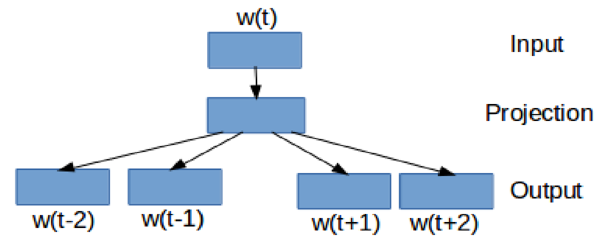

Fig. 1: The Skip-gram model architecture for predicting surrounding words given the current word *w(t)*.

B. *Latent Representation on OSNs*

In previous studies, different learning methods have been used to model social networks prediction tasks [11]. Recently, deep learning methods attract attention of researchers in variety of studies including the prediction tasks through OSNs. These methods have achieved more vivid and appreciated results in comparison to the conventional learning methods [6] [22] [9]. One of the successful deep learning architectures is word2vec model to capture word embeddings in natural language processing applications [23]. It extracts words semantic similarity using a simple architecture.

1) Word2vec: Authors in [23] have proposed two architectures as two new neural word embeddings structures. The first is Continuous Bag-of-Words (CBOW), which predicts the current word based on the context, and the second approach is Skip-gram which predicts surrounding words given the current word. Skip-gram with negative sampling (SGNS), also known as word2vec (a sample is shown in Figure 1) is an efficient method for learning high-quality word representation that captures the semantic relation of a word with its surrounding words in a corpus [6].

The Skip-gram approach trains high quality word vectors using a simple architecture. As shown in Figure 1, the model predicts the surrounding words $(w(t-2), w(t-1), w(t+1), w(t+2))$ given the current word $w(t)$. The goal is to find word vector representations that help to predict the nearby words. More formally, given a sequence of words $w_1, w_2, ..., w_k$, where $w_i \in W(the\ vocabulary)$, the goal is to maximize:

$$1/k \sum_{i=1}^{k} \sum_{-c \leq j \leq c, j \neq 0} \log p(w_{i+j} \mid w_i) \tag{1}$$

Noticeably improved results from using word2vec can be seen not only in Natural Language Processing (NLP) [23], but also in other research domains such as social networks [8] [24]. In a method inspired by word2vec [24], authors exploit the same framework of neural word embedding and produce embeddings for items in an item-based collaborative filtering. DeepWalk [8] uses random walks like the sequences of words in word2vec to learn the latent representations of users to use on multi-label network classification tasks. Node2vec [9] is a successful method to extract user embeddings where it maximizes the likelihood of preserving network neighborhoods of nodes. As words in sentences and users in random walks correspond to users in interaction sequences, it allows us to utilize a similar method to derive users' reactions likelihood from their interaction sequences. We explain next how this adaptation has been done in this study.

III. Methodology

Given a post and its publisher, the aim of the present study is to predict users' reactions to that post. Depending on the type of social network, the reaction of users can be in the form of re-share, like, or comment on the post. We consider the prediction task as a probability function to decide whether a user will react to the post of a given publisher. To this end, we first extract user embedding features and then exploit them as input in a simple neural network with a softmax function in the last layer.

A. Reactions Sequences

Following the main objective of the study on predicting future reactions of users, we exploit users' previous reactions log to extract their embedding features at the first step of our model. In a social network with a set of N users (U), reaction of users to a given post p_i, published by u_i, will be shown as $s_i = \{u_j \mid u_j \in U, j = 1, 2, ..., m\}$ and called an interaction sequence (s_i). Where m is the number of interactors of the post p_i, and index j refers to the index of interacting users in a temporal order, shown in Figure 2. We put the publisher of the post in the first place of the sequence.

In a dataset of P published posts, interaction sequences over those posts will be presented as the set S where $S = \{s_i \mid i = 1, 2, ..., P\}$. As mentioned, each s_i includes the interactors of the corresponding post, p_i.

There are mainly two reasons behind the reaction of a user to a post. First, the relation of the user with the publisher of the post such as friendship and followership. Second, the user's interest to the content of a post which induces her reaction to that post. As our aim is to investigate the competence of users pair-wise relations in predicting their future reactions, we are supposed to achieve the second concern by extracting the likelihood of users' interests from their common reactions to the posts.

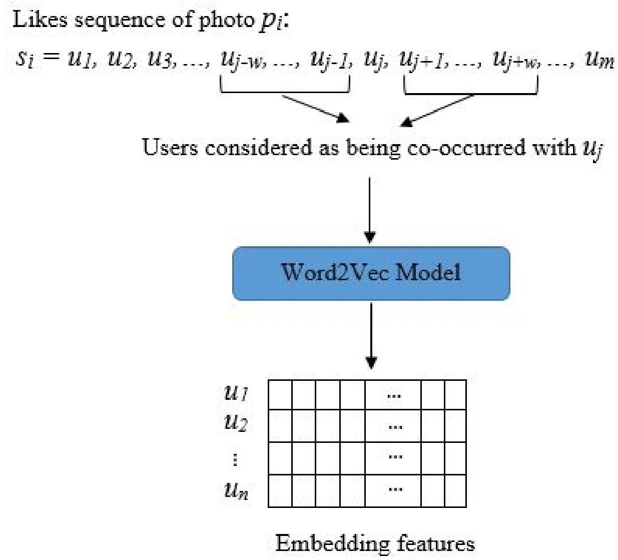

Fig. 2: Co-occurrences are computed for each user in the reaction sequences with her surrounded users, placed in a window of size w from two directions, are fed to the Word2vec model. the Word2vec Model supplies user embeddings.

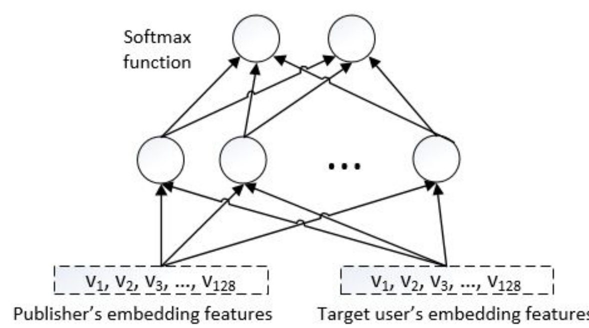

Fig. 3: One-hidden layer neural network with softmax function in the final layer to predict the target user's reaction probability.

Considering the mentioned points, we exploit users interactions sequences to derive user embeddings. In users' reaction history, the users' neighborhood illustrate their latent common tendency to react to the posts. Furthermore, since the neighborhood of users in a reaction stream can be a representation of cascading paths, user embeddings extracted from reaction streams will implicitly include cascading pattern between users as well. We use the Word2vec model in order to extract user embeddings. As Figure 2 shows, to generate user pairs required in the Word2vec model, we consider w users before and after each user in a reaction sequence to be paired with that user. Produced user pairs demonstrate the number of times that each two users are co-occurrenced in reaction streams within the window of size w. The high number of co-occurrences of user pairs indicates the more similar interests of them. Users pairs are fed to the Word2vec model and the model derives user embedding features as explained in subsection II-B1

TABLE I: The Flickr Dataset Characteristic

Attribute	Value
#Photos	11.2M
#Users	2.3M
#Photos with $\geq$ 30 likes	128K
Avg(#likes) of $\geq$ 30 likes	61
Median(#likes) of $\geq$ 30 likes	45

B. Future Reactions Prediction

We aim to use user embeddings extracted from reaction sequences to predict who will react to a given post. To reach this goal, we have designed a simple neural network with a softmax function in the final layer, as shown in Figure 3. Given a post's publisher, the network will make decision about user's reaction to that post. Inputs of the network are embedding features of the publisher and a target user, whose reaction probability is going to be predicted. The embeddings are extracted from the Word2vec model as described in previous section. The middle (hidden) layer performs a dot product with two input vectors and their weights, adds biases and applies the Rectified Linear Unit (ReLU) activation function. Output of the softmax function in the last layer will be a two-dimensional binary vector, represents the probability of the input user's reaction over the post of the given publisher. In our model, we will consider only users features to predict their future interactions without considering the content of the post.

As there is no specific method to determine the best number of layers and nodes for a neural network, we have tried different number of hidden layers as well as different number of nodes in the middle layer of the network and chose the numbers that provide outperforming results. The network is trained for different number of epochs of the data. Network uses Adam optimization method [25] implemented in Tensorflow[2] to update the weights.

IV. Evaluation

This section discusses the dataset which is used for evaluation as well as the conducted experiments and the obtained results of the proposed prediction approach compared to a baseline methods.

A. Dataset Description

The Flickr dataset from [26] is used to evaluate the proposed approach of reactions prediction. The dataset includes more than 11M photos and 2.3M users' activity log for 100 days. As mentioned previously, the reaction of users can differ from a social network to another. In this dataset, user's reaction to the photos is referred by marking the photos as user's favorite, which we refer this reaction by *like*, and the interacted users by *likers* in this study. The dataset includes the followership information between users as well. The main characteristics of the dataset are shown in Table I.

We check the distribution of users repetition in reaction sequences and observe that it follows a power-law distribution shown in Figure 4, which is very similar to the power-law distribution of words in natural language. As our technique is very similar to the Word2vec model, and the Word2vec model is very successful to capture word embeddings, this similarity verifies our method to be suitable to capture user embeddings. Our main contribution is the idea of using natural language processing techniques to model users reaction behaviour on social networks.

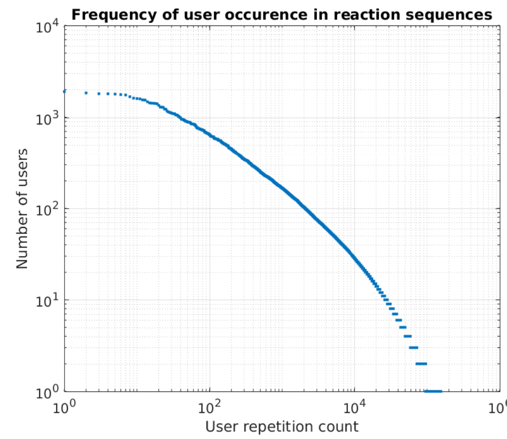

Fig. 4: The power-law distribution of users in reaction sequences.

Shown in Section III-A, reactions to a post provide us a list of users who have liked that post. Since our aim is to learn users representations from the like sequences, we need to choose those photos from the dataset that have enough like sequence length to support the window concept in the Word2vec model. Therefore in our experiments we only consider photos which have at least 30 likes. This threshold number can be different depending on the datatset size. The final dataset includes 128k photos whose number of likes are more than 30. The dataset is divided into two parts. The first part, with 50% of the data, is used to learn embeddings. And the remaining part of the data is used to train and test of the designed neural network.

B. Likers Prediction Experiments

1) Model Configuration: As mentioned, 50% of the prepared dataset is considered as the input of the Word2vec model to extract users embeddings. We set the window size to 10 and the number of features to 128. Derived embeddings are used to predict the like prediction. As shown in Figure 3, we designed a one-hidden layer neural network with embeddings of publisher and the target user, whose reaction is going to be predicted, in the input layer, 8 nodes in the middle layer and two nodes in the final layer indicating the binary like probability. As there is no specific way to determine the number of layers and nodes in a neural network, we evaluated the results when we have more or less layers and nodes in hidden layer, however the one-hidden layer network with 8 nodes in the middle layer outperforms other configurations.

We consider the remaining 50% of the dataset to train and test the designed prediction network. Since the dataset is

[2]https://www.tensorflow.org/

TABLE II: Hand-crafted user features used by SVM classifier to predict user's future reactions.

Feature	Description
#Likes_p	Number of photos has been liked by publisher.
#Likes_u	Number of photos has been liked by target user.
#Photos_p	Number of photos that publisher has published.
#Photos_u	Number of photos that target user has published.
#Reciprocal_likes(p→u)	Number of the given publisher's photos has been liked by target user.
#Reciprocal_likes(u→p)	Number of target user's photos has been liked by the given publisher.
#Mutual_likes	Number of photos from other users that both publisher and the target user have liked.

composed of photo's like sequences, we have only the users who have liked the photos. We needed to generate negative samples who have not liked the given photo. To have a fair dataset, the number of negative samples for each photo is considered to be equal to the number of users (likers) in that photo's like sequence. Negative samples for each photo are chosen from the publisher's friends and non-friends in the same portions that they are distributed in the like sequence.

2) Baseline Methods: We compared the prediction results with two base-line methods. The first baseline method is a Support Vector Machine (SVM) classifier [27] which uses some hand-crafted features. We choose this conventional learning method to compare with our method in order to provide a comparison between hand-crafted features and embedding features. Following the aim of this study which is prediction of user's future reactions using only previous reactions' history, we extract the user features listed in Table II to be used by SVM. We use our proposed prediction network when user embeddings come from a different source, as the second baseline. In this method, user embeddings are derived from random walks over the user graph (Node2vec) [9]. We aim to compare the efficiency of user embeddings when they come from two different sources. This comparison will reveal whether our idea of extracting user embeddings from user reactions can benefit the prediction task followed by this study. We chose the same feature size and window length for both like sequences and random walks. Performance of the experiments is evaluated in different epoch numbers, learning rates, and batch size, and eventually we chose the numbers that provide high performance.

3) Experimental Results: For our designed newtork, we examined different values of learning rate and batch size to find the best configuration of the network. In order to avoid presenting different numbers of each parameter and their different combinations, we show the results of the best performing examined values of the parameters. Table III represents the results for the SVM classifier, and our proposed approach when it uses like sequences embeddings and random walks embeddings. Three following metrics are considered to compare, *precision*, *recall*, and *F1_score*:

TABLE III: The performance of the proposed reaction prediction model in two different input sources.

Input of the model	Precision	Recall	F1
SVM classifier	0.501	0.512	0.506
Random walks	0.584	0.684	0.630
Reaction sequences	0.609	0.756	0.673

$$Precision = \frac{tp}{tp + fp} \tag{2}$$

$$Recall = \frac{tp}{tp + fn} \tag{3}$$

$$F1\_score = 2\frac{Precision * Recall}{Precision + Recall} \tag{4}$$

SVM classifier has the lowest values for all metrics. It shows that the SVM classifier using the defined features could not be discriminative in this prediction task. The results for our approach presented in Table III come from a configuration of 0.001 for learning rate and 512 for batch size. *Random walks* shows the performance of the model using user embeddings extracted from random walks over users graph, and *Reaction sequences* represents the performance of the same model when embeddings come from reaction sequences. As we can see, *reactions sequences* approach achieves higher performance than *Random walks*, and both of them behave better than SVM classifier. *F1_score* for the prediction model using reactions sequences embeddings is 67.3%, better than *F1_score* for random walks embeddings which is 63%. Precision and recall metrics get their highest values with 60.9% and 75.6% respectively, in *reactions sequences* approach which are more accurate than the result of *random walks* method.

This implies that users' neighborhood in reactions sequences can represent their likelihood better where the likelihood concerns their tendency to react to each other's posts. The most reasonable explanation for performing reaction sequences embeddings better than random walks embeddings is that users' neighborhood in reaction sequences involves implicitly users' preferences similarity in addition to their friendship. While in random walks graph, links are the only things keep them to be neighbor. It proves our hypothesis of using users' reaction logs to discover their latent similarity in terms of reacting to each others posts.

As a representative parameter assessment, Table IV shows precision, recall, and F1_score of the model in different examined learning rates and batch sizes when the model uses reactions sequences embeddings as input. As we can observe in this table, although the performance of the model obtains very close values in some configurations but when the learning rate is set to 0.001 and batch size to 512 it achieves its highest value by 67.3% F1_score. According to our observations,

TABLE IV: The performance of the model (with reaction sequences embeddings as input) with different learning rates and batch sizes.

Parameter		Precision	Recall	F1
Learning rate	0.1	0.577	0.746	0.651
	0.01	0.669	0.550	0.604
	0.001	0.609	**0.756**	**0.673**
Batch size	40	**0.694**	0.422	0.525
	512	0.609	**0.756**	**0.673**
	1024	0.577	0.687	0.627
	2048	0.628	0.676	0.651

when learning rate is 0.1 the model's performance is not stable. By decreasing it to 0.001, the network's weights get updated slightly and it helped the network to converge after almost 9 epochs. In our dataset, we reached the best performance when the batch size is set to 512.

V. Conclusion

This study aimed to predict users future reactions (e.g. likes, comments, shares) on ONSs using their reaction history on the published posts. Toward this goal, the proposed model first extracts users embeddings from their reactions log and use them to predict future engagement of users. Reaction history of users comes from their previous engagement to the content published on social media. We took advantage of user embeddings out of reactions sequences, where users likelihood represents their close relationship or preference similarity, to predict their reaction when a post gets published by a publisher. Users embeddings are exploited in a one-hidden layer neural network with a softmax function in the end layer. We compared the results when users embeddings come from the node2vec model and from the reaction sequences. The experiments show higher precision when users embeddings are derived from reaction sequences. It means that reactions sequences present better likelihood of users than random walks through users graph, in terms of revealing their potential probability to react to a post. Although we mainly focused on Flickr dataset, the proposed model is a general approach that can be applied to different social networks. As a future direction of this research, we will take advantage of user graph and draw out the subgraph of each reactions sequence in order to find an approach that can extract users embeddings with no need to random walks over subgraphs.

References

[1] Sarah Elsharkawy, Ghada Hassan, Tarek Nabhan, and Mohamed Roushdy. Towards feature selection for cascade growth prediction on twitter. In *Proceedings of the 10th International Conference on Informatics and Systems*, pages 166–172. ACM, 2016.

[2] Justin Cheng, Lada Adamic, P Alex Dow, Jon Michael Kleinberg, and Jure Leskovec. Can cascades be predicted? In *Proceedings of the 23rd international conference on World wide web*, pages 925–936. ACM, 2014.

[3] Qi Zhang, Yeyun Gong, Jindou Wu, Haoran Huang, and Xuanjing Huang. Retweet prediction with attention-based deep neural network. In *CIKM*, 2016.

[4] Sasa Petrovic, Miles Osborne, and Victor Lavrenko. Rt to win! predicting message propagation in twitter. In *ICWSM*, 2011.

[5] Benjamin Shulman, Amit Sharma, and Dan Cosley. Predictability of popularity: Gaps between prediction and understanding. 2016.

[6] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, 2013.

[7] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.

[8] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710. ACM, 2014.

[9] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016.

[10] Samin Mohammadi, Reza Farahbakhsh, and Noël Crespi. Popularity evolution of professional users on facebook. 2017.

[11] Alexandru Tatar, Marcelo Dias de Amorim, Serge Fdida, and Panayotis Antoniadis. A survey on predicting the popularity of web content. *Journal of Internet Services and Applications*, 5, 2014.

[12] Ioannis Arapakis, Mounia Lalmas, B Barla Cambazoglu, Mari-Carmen Marcos, and Joemon M Jose. User engagement in online news: Under the scope of sentiment, interest, affect, and gaze. *Journal of the Association for Information Science and Technology*, 65(10):1988–2005, 2014.

[13] Anjana Susarla, Jeong-Ha Oh, and Yong Tan. Social networks and the diffusion of user-generated content: Evidence from youtube. *Information Systems Research*, 23(1):23–41, 2012.

[14] Saeideh Bakhshi, David A Shamma, and Eric Gilbert. Faces engage us: Photos with faces attract more likes and comments on instagram. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*, pages 965–974. ACM, 2014.

[15] Pu Gong and HuanYu Wu. A cache partition policy of ccn based on content popularity. *International Journal of Advanced Science and Technology*, 92:9–16, 2016.

[16] Zhi Wang, Wenwu Zhu, Minghua Chen, Lifeng Sun, and Shiqiang Yang. Cpcdn: Content delivery powered by context and user intelligence. *IEEE Transactions on Multimedia*, 17(1):92–103, 2015.

[17] Peng Bao, Hua-Wei Shen, Junming Huang, and Xue-Qi Cheng. Popularity prediction in microblogging network: a case study on sina weibo. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 177–178. ACM, 2013.

[18] Wenjian Hu, Krishna Kumar Singh, Fanyi Xiao, Jinyoung Han, Chen-Nee Chuah, and Yong Jae Lee. Predicting the image propagation path in online social networks. 2018.

[19] 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. *arXiv preprint arXiv:1712.04443*, 2017.

[20] Jack Hessel, Lillian Lee, and David Mimno. Cats and captions vs. creators and the clock: Comparing multimodal content to context in predicting relative popularity. In *Proceedings of the 26th International Conference on World Wide Web*, pages 927–936. International World Wide Web Conferences Steering Committee, 2017.

[21] Yaqian Duan, Xinze Wang, Yang Yang, Zi Huang, Ning Xie, and Heng Tao Shen. Poi popularity prediction via hierarchical fusion of multiple social clues. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1001–1004. ACM, 2017.

[22] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. Collaborative deep learning for recommender systems. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1235–1244. ACM, 2015.

[23] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[24] Oren Barkan and Noam Koenigstein. Item2vec: Neural item embedding for collaborative filtering. *arXiv preprint arXiv:1603.04259*, 2016.

[25] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[26] Meeyoung Cha, Alan Mislove, and Krishna P. Gummadi. A Measurement-driven Analysis of Information Propagation in the Flickr Social Network. In *In Proceedings of the 18th International World Wide Web Conference (WWW'09)*, Madrid, Spain, April 2009.

[27] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.