Hashtag Recommendation Using End-To-End Memory Networks with Hierarchical Attention

Haoran Huang, Qi Zhang, Yeyun Gong, Xuanjing Huang
Shanghai Key Laboratory of Intelligent Information Processing,
School of Computer Science, Fudan University
{huanghr15, qz, yygong12, xjhuang}@fudan.edu.cn

Abstract

On microblogging services, people usually use hashtags to mark microblogs, which have a specific theme or content, making them easier for users to find. Hence, how to automatically recommend hashtags for microblogs has received much attention in recent years. Previous deep neural network-based hashtag recommendation approaches converted the task into a multi-class classification problem. However, most of these methods only took the microblog itself into consideration. Motivated by the intuition that the history of users should impact the recommendation procedure, in this work, we extend end-to-end memory networks to perform this task. We incorporate the histories of users into the external memory and introduce a hierarchical attention mechanism to select more appropriate histories. To train and evaluate the proposed method, we also construct a dataset based on microblogs collected from Twitter. Experimental results demonstrate that the proposed methods can significantly outperform state-of-the-art methods. By incorporating the hierarchical attention mechanism, the relative improvement in the proposed method over the state-of-the-art method is around 67.9% in the F1-score.

1 Introduction

Along with the rapid development of social media, many people write brief text updates about their life on the go. Among these thousands of millions of microblogs posted every day, some contain # in front of words or unspaced phrases. The # symbol, called a *hashtag*, is usually used to mark keywords or topics in a microblog. Social media users originally created it to categorize messages. Now, hashtags have been widely used in a variety of circumstances. Hashtagged words that become very popular are often trending topics. Various works have also shown that hashtags can provide valuable information about different problems such as twitter spammer detection (Benevenuto et al., 2010), popularity prediction (Tsur and Rappoport, 2012), and sentiment analysis (Wang et al., 2011).

With the increasing requirements, the hashtag recommendation task has received considerable attention in recent years. Discriminative models have been proposed from different aspects using various kinds of features and models (Heymann et al., 2008; Liu et al., 2011), collaborative filtering (Kywe et al., 2012), generative models (Ding et al., 2013; Godin et al., 2013; She and Chen, 2014), and convolutional neural networks (CNN) (Gong and Zhang, 2016). Some of the previous works treated this task as a multi-class classification problem and used word-level features and exquisitely designed patterns to perform the task. Numerous existing studies utilized the word trigger assumption (Liu et al., 2011; Ding et al., 2013) and introduced topical machine translation models to achieve the task.

Due to the advantages of deep neural networks and the effectiveness of these methods in various NLP tasks, convolutional neural networks have also been applied to the hashtag recommendation task (Gong and Zhang, 2016). Some have also treated the hashtag recommendation task as a multi-class classification problem and incorporated an attention mechanism to handle the trigger words. This method only used a microblog as input. It did not take the history information of the user into account. However,

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 943–952, Osaka, Japan, December 11-17 2016.

the microblogs a user posted in recent history can represent their interests to some degree. Previous works (Zhang et al., 2014) also studied this issue using generative models, and the experimental results demonstrated the usefulness of the histories of users.

In this work, to incorporate the histories of users, we propose a novel end-to-end memory network architecture to combine the microblog textual information and corresponding user history to perform the task. We regard the user history as an external memory for the microblog. The memory networks can help to extract the useful features from the external memory, which are relevant to the microblog and hashtag, to construct user interest representations. With the underlying intuition that not all microblogs in the user history are equally relevant for recommending hashtags, and not all of the words in a microblog are equally important, we introduce a novel hierarchical attention mechanism and integrate it with a memory network to capture two insights about the posting histories of users. Experimental results demonstrate that the performance of the method incorporating the hierarchical attention mechanism is also better than the method without it.

The main contributions of this work can be summarized as follows:

- To incorporate the user history information, we propose to extend the end-to-end memory networks to perform the hashtag recommendation task.
- Since not all of the microblogs and words in a microblog are equivalent in importance, we introduce a novel hierarchical attention mechanism and integrate it with the end-to-end memory networks.
- Experimental results using a dataset collected from a real microblogging service demonstrated that the proposed method can achieve significantly better performance than the state-of-the-art methods.

2 Related Work

2.1 Hashtag Recommendation

In recent years, various studies have been conducted on this task(Kywe et al., 2012; Ding et al., 2013; Godin et al., 2013; Sedhai and Sun, 2014; Wang et al., 2014; Gong and Zhang, 2016; Shi et al., 2016).

Kywe et al.(2012) used a similarity based method to solve this problem. They recommends hashtag by combining hashtags from the similar tweets as well as hashtags from the similar users. Many other approaches focus on the topic modelling(Ding et al., 2013; Godin et al., 2013; She and Chen, 2014). Based on the assumption that the hashtag and the trigger words of the tweets are two different language and have the same meaning, Ding et al.(2013) proposed to use translation process to model this task. In contrast to the methods that focuses on topic modelling, Shi et al.(2016) proposed a learning-to-rank approach for modelling hashtag relevance. Due to advantages of deep neural netowrks, CNN has been applied on this task(Gong and Zhang, 2016). Most of the works are based on textual information of tweets. However, some other works found there were different types of information which are helpful. Zhang et al.(2014) proposed a topical model based method to incorporate the temporal and personal information. Sedhai and Sun(2014) combined the textual information and hyperlinked information to recommend hashtags.

In this work, we propose a novel networks architecture to combine the textual information and the corresponding user history to perform the task.

2.2 Attention and Memory

The second relevant line of work is the research on attention mechanisms and memory networks. Attention mechanisms have been widely used in many studies and have shown to achieve promising result on several tasks, such as generating handwriting (Graves, 2013), machine translation (Bahdanau et al., 2014), speech recognition (Chorowski et al., 2014), action recognition (Sharma et al., 2015), caption generation (Xu et al., 2015) and so on. In recent months, memory networks(Weston et al., 2014) have been proposed and applied on natural language question answering, which have four component: input (I), generalization (G), output (O) and response (R) component. After then, Sukhbaatar et al.(2015) proposed a end-to-end memory networks and applied it on question answering and language modeling.

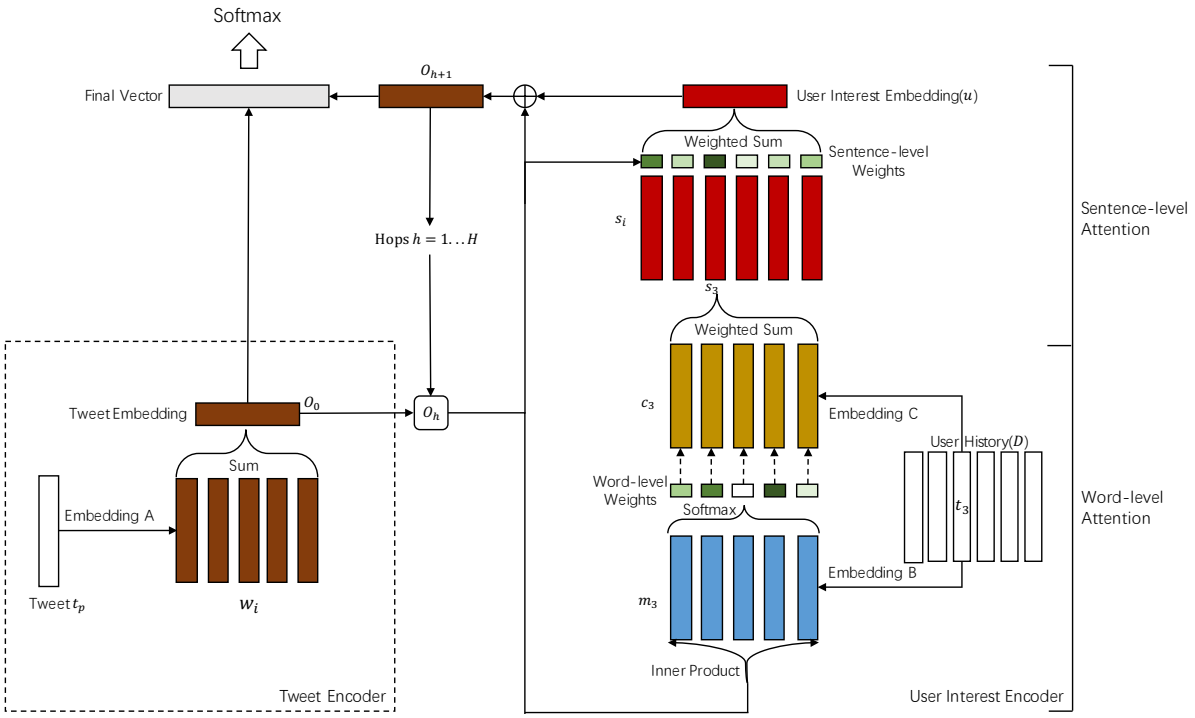

Figure 1: End-To-End Memory Networks with Hierarchical Attention

The memory networks architecture has been adopted in dialog systems(Dodge et al., 2015; Bordes and Weston, 2016; Weston, 2016) and query answering(Kumar et al., 2015; Weston et al., 2015).

In this work, we treat user history as the external memory and use a memory networks architecture with hierarchical attention to encode the user interest.

3 Approach

3.1 Preliminary

Given a tweet t_p, our task is to recommend a hashtag that is most relevant to the tweet. Based on this definition, we formulate the hashtag recommendation task as a multi-class classification problem. From the above description, we can see that the user information is also very important because each user always focuses on several aspects that may relate to the hashtag. Meanwhile, the interests of users can in most cases be represented by the tweets that they post. Hence, we use the tweet set D to represent the interests of users. Each tweet is a word sequence denoted by $t = \{w_1, w_2, ..., w_N\}$, where N is the length of the tweet. The user history contains many tweets denoted by $D = \{t_1, t_2, ..., t_M\}$, where M is the size of the history document. Let $H_p = \{H_{p1}, H_{p2}, ..., H_{p|H_p|}\}$ be the set of candidate hashtags.

3.2 The Proposed Methods

To solve this classification problem, we propose a novel end-to-end memory network architecture (HMemN2N) to combine the tweet textual information and corresponding user history, which is shown in Figure 1.

In our proposed models, we first use a tweet encoder to embed the tweet t_p. Then, the user interest encoder can extract the interest information of the user from the user history D with the help of t_p. Finally, we combine the information from two encoders and use a softmax layer to score and recommend a hashtag list. In this work, we regard the user history as an external memory for the tweet t_p. The introduction of the memory networks can help to extract the useful features from the external memory to build the user interest representations. With a underlying intuition that not all tweets in the user history are equally relevant for recommending the hashtag, not all words in each tweet are equally important,

and the tweets in the history document are relatively independent and focus on different aspects, we introduce a hierarchical attention mechanism into memory networks to capture two insights about the user history document and obtain a high-quality user interest representation.

We describe the details of different parts of Figure 1 in the following sections.

3.2.1 Tweet Encoder

As shown, the first part of the original input is the tweet t_p, and it is treated as a bag-of-words (BoW) representation. Then, we embed each word w in t_p in a continuous space and sum the embedding vectors to obtain an embedding representation. Specifically, the embedding matrix A(of size $dim \times |V|$, where V is the vocabulary and dim is the embedding dimension) is used to look up the vectors for words w, and the representation can be calculated as follows: $o_0 = \sum_i^N Aw_i$. The embedding o_0 is also treated as an initial input of the user interest encoder.

3.2.2 User Interest Encoder

The second part of the input is the user history, which is stored in the memory. We use a memory with a two-tier architecture: sentence and word. Based on the previously provided notations, the document is a tweet set: $D = \{t_1, t_2, ..., t_M\}$, which is a sentence-level structure. Then, each tweet $t_i \in D$ is divided into a bag-of-words representation: $t_i = \{w_{i1}, w_{i2}, ..., w_{iN}\}$.

To obtain the user interest representation, we propose a two-level encoder architecture, which can then be stacked in what is called multiple hops, denoted as $h = 0, 1, 2, ..., H$.

Now, we first introduce a word-level attention mechanism for embedding the tweets t_i. There are two components: input memory and output memory. In the input memory component, given an input set $\{t_1, t_2, ..., t_i\}$, each word $w_{ij} \in t_i$ is embedded using a matrix B of size $dim \times |V|$ into memory vectors $\{m_{ij}\}$ of dimension d, giving $m_{ij} = Bw_{ij}$. The input memory representation m_i of tweet t_i is a matrix of size $N \times dim$, where N is the length of the tweet. However, because not all words contribute equally to the tweets meaning, and the importance degree of the word w_{ij} should be considered in our models, we propose a probability layer to achieve the goal. The match between o_h and memory vectors m_{ij} is then computed by taking the inner product followed by a softmax:

$$p_{ij} = \frac{exp(o_h^{\mathrm{T}} m_{ij})}{\sum_n^N exp(o_h^{\mathrm{T}} m_{in})}, \tag{1}$$

where o_h is the internal input state in hop h, and p_{ij} is a probability vector over the input memory.

In the output memory component, each word $w_{ij} \in t_i$ has a corresponding output vector c_{ij}, which is obtained using an embedding matrix C with the same size. Then, the output representation s_i of tweet t_i is constructed by summing the output vector c_{ij}, weighted by the probability p_{ij}, and the equation of this operation is as follows:

$$s_i = \sum_j^N p_{ij} c_{ij}, \tag{2}$$

From the above procedure, the memory is converted into a matrix s that contains M tweet embedding vectors of size dim. Next, we propose a sentence-level method to extract sentences that are important to the user and aggregate the representation of this information to form the user interest representation. The weight p_{s_i} of sentence s_i is calculated, and the user interest vector u is formed by the weighted sum of the tweet embeddings:

$$m_{s_i} = tanh(W_o o_h + W_s s_i), \tag{3}$$

$$p_{s_i} = \frac{exp(W_{ms}^{\mathrm{T}} m_{s_i})}{\sum_j^{|M|} exp(W_{ms}^{\mathrm{T}} m_{s_j})}, \tag{4}$$

$$u = \sum_i^{|M|} p_{s_i} s_i, \tag{5}$$

where $|M|$ is the number of tweets in the users history document, and the parameters in these equations are W_o, W_s, and W_{ms}.

By implementing the hops operator, the memory can be read and write iteratively using the state o_h. At the last step of each hops h, a new output state o_{h+1} is updated with $o_{h+1} = o_h + u$, where u is the user interest embedding obtained in this hop. The last output state o_H is regarded as a high-level representation of user insterest.

3.2.3 Final Prediction

Finally, the tweet embedding o_0 and the user interest output o_H can also be concatenated into the final vector f, giving $f = o_0||o_H$. Then, we can use the final vector to predict the recommended hashtag through a softmax layer:

$$p(y = h_{pi}|f; \theta_s) = \frac{exp({\theta_s^{h_{pi}}}^{\mathrm{T}}(W_f f + b_f))}{\sum_j^{|H_p|} exp({\theta_s^{h_{pj}}}^{\mathrm{T}}(W_f f + b_f))}, \tag{6}$$

where W_f, b_f and θ_s are parameters, H_p is the candidate hashtag set and h_{pi} is the i-th hashtag in H_p.

According to the scores from the last softmax layer, we can list a top-ranked recommended hashtags for each tweet.

3.3 Training

The training objective function in this work is:

$$J = \sum_{(t_p, D, h_p) \in S} -logp(h_p|t_p; D), \tag{7}$$

where h_p is the hashtag for tweet t_p and S is the training set.

The parameter list of our model is:

$$\theta = \{A, B, C, W_o, W_s, W_{ms}, W_f, b_f, \theta_s\}, \tag{8}$$

where A, B and C are three embedding matrix. W_o, W_s and W_{ms} are the parameters of attention layer in user interest encoder. W_f, b_f and θ_s are the parameters of the final predict layer.

In this study, we use stochastic gradient descent (SGD) with the adagrad update rule to optimise our model. Dropout regularization has proved to be an effective method for reducing the overfitting in deep neural networks with millions of parameters. In this work, we use it and add $l2$-norm regularization terms for the parameters of the network to augment the cost function.

Table 1: Statistics of the evalution dataset

#Tweets	#Users	#Hashtags	#Avg.Hashtag/Tweet
288,545	36,003	3,883	1.28

4 Experiment

In this section, we first introduce the data collection. Then, we describe the experiment configurations and baseline methods. Finally, the evaluation results and analyses are given.

4.1 Dataset and Setup

We started by using Twitter's API to collect public tweets from randomly selected users. In a first step, we randomly selected 40,000 users and crawled their tweets. In this step, we obtained 77,995,265 tweets. In the second step, we selected users with more than 50 tweets and filtered out the tweets whose language was not English. Then, we filtered out the hashtags whose frequencies were very low. Third, we extracted the tweets in the original corpus that contained hashtags.

Based on the statistics, there were 281,345 tweets in our evaluation collection, which belonged to 36,003 different users. The unique number of hashtags in the corpus was 3,883, and the average number of hashtags in each tweet was 1.28. The list of hashtags annotated by their users were treated as the ground truth. The detailed statistics are listed in Table 1. All of the tweets were processed by removing stopwords and special characters. In our experiment, we split the dataset into a training set, validation set, and test set. There were 232,378 tweets in the training set and 28,092 in the validation set. The remaining 20,875 tweets were in the test set.

In this work, the poster of each tweet had a history. For the user history, we assumed that the latest tweets posted could represent the user's current interests and could be stored in the memory. In this work, the memory capacity was restricted to the most recent 5 tweets posted by the users, the maximum length of the tweets was 52, any tokens out of this range were discarded and any hashtags occuring in the history had been removed. The embedding dimension of our model was set to 300, and the number of hops was set to 2 unless noted otherwise. This configuration was also used in the other methods described in the following paragraphs. The network was used for training for 60 epochs with early stopping. The learning rate was set to l = 0.01, and the dropout rate was 0.2.

The three metrics used in this experiment to evaluate the quality of our model were the precision, recall, and F1-score (denoted as P, R, and $F1$, respectively). The number of recommended hashtags for each tweet are denoted as k, where $k = \{1, 2, 3, 4, 5\}$ and the precision, Recall, and F1-score at the k result are denoted as $P@k$, $R@k$, and $F1@k$, respectively.

4.2 Baseline

In this section, to compare with our model, we select some effective methods as a baseline and introduce a degeneration model, described as follows:

- **NB**: To achieve the task, we convert the hashtag recommendation problem into a classification problem. We apply Naive Bayes to model the posterior probability of each hashtag using only the textual information of the tweets.

- **NB+H**: To assess the usefulness of the user interest information, the textual information and user history are given to Naive Bayes to recommend hashtags.

- **SVM**: We use the pre-trained word vector [1], which was trained using a portion of a Google News dataset containing 300-dimensional vectors for 3 million words using the continuous bag-of-words model, and sum them as the feature vector of the tweet as features, which are used to implement the support vector machine for the recommendation.

- **SVM+H**: The information of the user interest history is added to the features and the support vector machine is used to achieve the task.

- **TTM**: TTM was proposed by (Ding et al., 2013) for hashtag recommendation. The authors proposed a topical translation model to recommend hashtags, which only used the tweet content.

- **CNN-Attention**: CNN-Attention was proposed by (Gong and Zhang, 2016). It was a convolutional neural network architecture with an attention mechanism, and it was the state-of- the-art method for this task. In this paper, we compare our method with it.

- **MemN2N**: The user interest encoder in our model is a memory network architecture with a hierarchical attention mechanism. Now, we replace it with the end-to-end memory network method proposed by (Sukhbaatar et al., 2015) and compare it with our proposed model.

[1] https://code.google.com/archive/p/word2vec/

Table 2: Result of different methods on the evaluation collection

Method	Precision	Recall	F1
NB	0.123	0.098	0.109
NB+H	0.198	0.156	0.175
SVM	0.232	0.181	0.203
SVM+H	0.312	0.241	0.272
TTM	0.234	0.190	0.210
CNN-Attention	0.328	0.255	0.287
MemN2N	0.501	0.403	0.446
HMemN2N	0.538	0.436	0.482

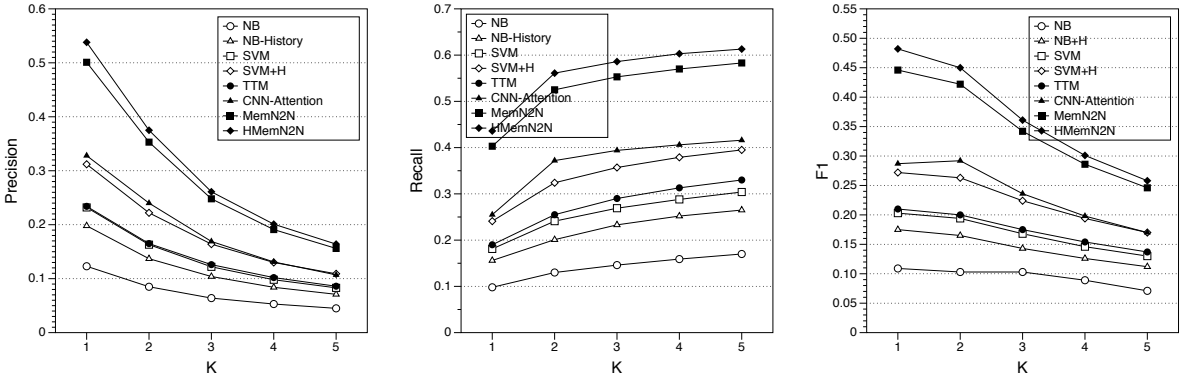

Figure 2: Precision, Recall, and F1-Score with different number of recommendation hashtags

4.3 Result and Discussion

In Table 2, we list the trend recommendation performances on our dataset using the different methods. The three metric results listed in Table 2 were obtained when we recommended the top hashtag for each tweet, i.e., $k = 1$.

In Table 2, we can see that the proposed method HMemN2N is better than all the other methods because it obtains the best result on all these metrics. Our approach provides improvements of 0.21 in precision, 0.181 in recall, and 0.195 in the F1-score over CNN-Attention, which is by far the state-of-the-art method for this task. Compared with the degeneration model MemN2N, our approach also shows a significant improvement. HMemN2N achieves a relative improvement of 7.4% in precision, 8.2% in recall, and 8.1% in the F1-score over MemN2N. The results show the practical applicability of our model, which can be used to provide users with good recommendations.

Observing the comparisons of the "NB" and the "NB+H", and the "SVM" and the "SVM+H", it is clear that the user interest history is a key ingredient in the recommendation, which is the strongest confirmation that much important information in the user history can be used to recommend hashtags in social media. This strongly suggests that we should find an effective method to extract useful information from the user history. In this paper, we provide a memory network architecture to solve this problem. The history is treated as an external memory, and a hierarchical attention mechanism is provided to help the model to extract the information, where the attention operation can be performed repeatedly and iteratively. The properties of this aspect of our proposed model have been proven to be effective by observing the results shown in Table 2.

CNN-Attention is the latest method used for this task, and it showed a good performance in this work. However, CNN-Attention only considers the content of the tweets and uses an attention mechanism to find important words in tweets. With a intuition that users will tend to repeatedly use the same hashtag, the performances of CNN-Attention and HMemN2N have a huge gap, which is mainly caused by not considering the user information. The results of the topical translation model (TTM) were obviously worse than those of our method, because it also only uses the textual information of the tweets.

Table 3: Parameter Influence on the evaluation collection

Model	embedding dim	# of hops	Precision	Recall	F1
CNN-Attention	300	-	0.328	0.255	0.287
MemN2N	300	2	0.501	0.403	0.446
	300	3	0.481	0.385	0.428
HMemN2N	300	1	0.521	0.421	0.466
	50	2	0.462	0.370	0.411
	100	2	0.508	0.410	0.454
	200	2	0.530	0.430	0.475
	300	2	0.538	0.436	0.482
	400	2	0.536	0.437	0.481
	500	2	0.538	0.437	0.482
	300	3	0.534	0.433	0.478
	300	4	0.528	0.428	0.473
	300	5	0.521	0.421	0.466

Considering the comparison between MemN2N and HMemN2N, the results show that it is necessary to introduce a hierarchical structure to store and select information. Each tweet has its own degree of importance in different recommendations, and each word in each tweet should also be given individual attention. Our model further utilizes an attention mechanism with a hierarchical structure to improve the information extraction. Compared to MemN2N, HMemN2N had superior performances across the board, which clearly demonstrated the effectiveness of the hierarchical mechanism.

In Figure 2, we list the result of models with different numbers of recommendation hashtags. The number of hashtags recommended k ranges from 1 to 5. From Figure 2, we can see that HMemN2N outperforms all of the baseline methods in all three metric curves with varying k. Clearly, the precision result decreases as k increases, and the recall result increases as k increases. The highest F1-score is obtained when we recommend the top 1 hashtag for each tweet. By analyzing the result, we can see that our model achieves the best performance in this task because all of the curves for HMemN2N are the highest in the graphs.

4.4 Parameter Influence

To evaluate the influence of the parameters used in our model, we changed the critical parameters to those in our dataset and list the results in Table 3. The effects of different hop numbers are shown, and the performances with different embedding dimensions are investigated.

From Table 3, we observe that changing the number of hops has some impact on the overall performance. However, the results disprove that more hops are better, because when the number of hops is larger than 2, the performances of HMemN2N and MemN2N are both decreasing.

The results listed in Table 3 also show the contribution of the embedding dimension to the performance. We fix the number of hops to 2 and vary the embedding dimension. A higher embedding dimension results in a better performance. When the dimension is low such as $dim = 50$ or $dim = 100$, the result is very poor. Our proposed model performs very well with a high embedding dimension. When the dimension is equal to 300, 400 or 500, we all can obtain the good performance. The size of the embedding dimension represents the expression ability of each word, and a higher dimension can enhance the text feature expression ability. To recommend a more appropriate hashtag, it is suggested to choose a high embedding dimension.

5 Conclusion

In this paper, we proposed a novel end-to-end memory network architecture that combines a tweets textual information and the corresponding user interest information for the hashtag recommendation task. The user interest history was a key ingredient of the recommendation and was adopted in this work. We

treated the user history as an external memory and proposed a novel memory network with a hierarchical attention mechanism to encode the user interest. To evaluate the proposed method, we collected data from real word twitter services. The experimental results on the evaluation dataset demonstrated that the proposed method could achieve better results than the current state-of-the-art methods for this task because they do not consider the user information.

Acknowledgments

The authors wish to thank the anonymous reviewers for their helpful comments. This work was partially funded by National Natural Science Foundation of China (No. 61532011, 61473092, and 61472088), the National High Technology Research and Development Program of China (No. 2015AA015408), and IBM Faculty Award 2016.

References

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

Fabricio Benevenuto, Gabriel Magno, Tiago Rodrigues, and Virgilio Almeida. 2010. Detecting spammers on twitter. In *Collaboration, electronic messaging, anti-abuse and spam conference (CEAS)*, volume 6, page 12.

Antoine Bordes and Jason Weston. 2016. Learning end-to-end goal-oriented dialog. *CoRR*, abs/1605.07683.

Jan Chorowski, Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. End-to-end continuous speech recognition using attention-based recurrent NN: first results. *CoRR*, abs/1412.1602.

Zhuoye Ding, Xipeng Qiu, Qi Zhang, and Xuanjing Huang. 2013. Learning topical translation model for microblog hashtag suggestion. In *Proceedings of IJCAI 2013*.

Jesse Dodge, Andreea Gane, Xiang Zhang, Antoine Bordes, Sumit Chopra, Alexander Miller, Arthur Szlam, and Jason Weston. 2015. Evaluating prerequisite qualities for learning end-to-end dialog systems. *CoRR*, abs/1511.06931.

Fréderic Godin, Viktor Slavkovikj, Wesley De Neve, Benjamin Schrauwen, and Rik Van de Walle. 2013. Using topic models for twitter hashtag recommendation. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 593–596. ACM.

Yeyun Gong and Qi Zhang. 2016. Hashtag recommendation using attention-based convolutional neural network. In *IJCAI 2016, Proceedings of the 26rd International Joint Conference on Artificial Intelligence, New York City, USA., July 9-15, 2016*.

Alex Graves. 2013. Generating sequences with recurrent neural networks. *CoRR*, abs/1308.0850.

Paul Heymann, Daniel Ramage, and Hector Garcia-Molina. 2008. Social tag prediction. In *SIGIR*.

Ankit Kumar, Ozan Irsoy, Jonathan Su, James Bradbury, Robert English, Brian Pierce, Peter Ondruska, Ishaan Gulrajani, and Richard Socher. 2015. Ask me anything: Dynamic memory networks for natural language processing. *CoRR*, abs/1506.07285.

Su Mon Kywe, Tuan-Anh Hoang, Ee-Peng Lim, and Feida Zhu. 2012. On recommending hashtags in twitter networks. In *Social Informatics*, pages 337–350. Springer.

Zhiyuan Liu, Xinxiong Chen, and Maosong Sun. 2011. A simple word trigger method for social tag suggestion. In *Proceedings of EMNLP*, pages 1577–1588. Association for Computational Linguistics.

Surendra Sedhai and Aixin Sun. 2014. Hashtag recommendation for hyperlinked tweets. In *The 37th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '14, Gold Coast , QLD, Australia - July 06 - 11, 2014*, pages 831–834.

Shikhar Sharma, Ryan Kiros, and Ruslan Salakhutdinov. 2015. Action recognition using visual attention. *CoRR*, abs/1511.04119.

Jieying She and Lei Chen. 2014. Tomoha: Topic model-based hashtag recommendation on twitter. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 371–372. ACM.

Bichen Shi, Georgiana Ifrim, and Neil J. Hurley. 2016. Learning-to-rank for real-time high-precision hashtag recommendation for streaming news. In *Proceedings of the 25th International Conference on World Wide Web, WWW 2016, Montreal, Canada, April 11 - 15, 2016*, pages 1191–1202.

Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, and Rob Fergus. 2015. End-to-end memory networks. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 2440–2448.

Oren Tsur and Ari Rappoport. 2012. What's in a hashtag?: content based prediction of the spread of ideas in microblogging communities. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 643–652. ACM.

Xiaolong Wang, Furu Wei, Xiaohua Liu, Ming Zhou, and Ming Zhang. 2011. Topic sentiment analysis in twitter: a graph-based hashtag sentiment classification approach. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 1031–1040. ACM.

Yuan Wang, Jishi Qu, Jie Liu, Jimeng Chen, and Yalou Huang. 2014. What to tag your microblog: Hashtag recommendation based on topic analysis and collaborative filtering. In *Web Technologies and Applications - 16th Asia-Pacific Web Conference, APWeb 2014, Changsha, China, September 5-7, 2014. Proceedings*, pages 610–618.

Jason Weston, Sumit Chopra, and Antoine Bordes. 2014. Memory networks. *CoRR*, abs/1410.3916.

Jason Weston, Antoine Bordes, Sumit Chopra, and Tomas Mikolov. 2015. Towards ai-complete question answering: A set of prerequisite toy tasks. *CoRR*, abs/1502.05698.

Jason Weston. 2016. Dialog-based language learning. *CoRR*, abs/1604.06045.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 2048–2057.

Qi Zhang, Yeyun Gong, Xuyang Sun, and Xuanjing Huang. 2014. Time-aware personalized hashtag recommendation on social media. In *COLING*, pages 203–212.