Co-attention Memory Network for Multimodal Microblog's Hashtag Recommendation

Renfeng Ma, Xipeng Qiu, Qi Zhang, Xiangkun Hu, Yu-Gang Jiang, and Xuanjing Huang

Abstract—Hashtags are keywords describing a topic or a theme and are usually chosen by microblogging users. Hence, the hashtags can be used to categorize microblog posts. With the fast development of the social network, the task of recommending suitable hashtags has received considerable attention in recent years. Recently, most neural network methods have treated the task as a multi-class classification problem. In fact, users are constantly introducing new hashtags in a highly dynamic way. Treating the task as a multi-class classification problem with a fixed number of target categories does not allow the method to deal with the new hashtags. To address this problem, the task is reinterpreted as a matching problem and a novel co-attention memory network is proposed to represent the multimodal microblogs and hashtags. We utilize a co-attention mechanism to model the multimodal mircroblogs, and utilize the post history to represent the hashtags. Experimental results on a Twitter-based dataset demonstrated that the proposed method can achieve better performance than the current state-of-the-art methods that treat the task as a multi-class classification problem.

Index Terms—Social media, Hashtag recommendation, Microblog

1 Introduction

With the rapid development of the Internet, social media has experienced rapid growth. Twitter-like microblogging, is one of the most popular social media platforms for information generation and diffusion. Moreover, Twitter has 330 million active users per month according to its quarterly report [1]. Users can also interact with various social media outlets. Hence, microblogs have been widely used as sources for public opinion analysis [1], reputation management [2], and many other applications [3], [4], [5], [6]. According to Twitter, a specific form of metadata tag, called a hashtag, can be used to mark keywords or topics within the text of a microblog. A hashtag is a string of characters prefixed with the symbol (#). Moreover, hashtags have been used to perform various tasks, like microblog retrieval [7], query expansion [8], and sentiment analysis [9].

Even though the value of hashtags is proven by the above works, relatively few microblogs include hashtags labeled by their users. Therefore, the task of hashtag recommendation has received considerable attention in recent years. Various methods have been proposed to perform the research, including various supervised methods with a manually constructed features [10], [11]. Moreover, the collaborative filtering-based method [12] has also been used to perform this task. Rather than considering both user preferences and tweet content in recommending hashtags, generative methods [13], [14] modeled the hashtag recommendation task as a translation process from content to hashtags. With the rapid development of deep neural networks, some deep neural network models [15], [16] also have been proposed to make hashtag recommendations.

Particularly, most early methods only take textual information into consideration. However, according to the statistics, more than 42% of tweets include more than one type of information[2]. Hence, [16] proposed a novel deep neural network to recommend hashtags for multimodal tweets by using a co-attention mechanism, which proved that image information was useful for hashtag recommendation. After taking the visual information into consideration, [16] the proposed model achieved better performance for recommending hashtags. However, most of recent neural network methods treated the hashtag recommendation tasks as a multi-class classification problem. Hence, these methods can only handle fixed amounts of hashtags, and they are not flexible for emerging hashtags not seen by the model during training.

To deal with this problem, we propose a novel model that not only combines textual and visual information, but also incorporates the post history. Since hashtags are usually highly related to the tweet, measuring the similarities between the interests of the candidate hashtag and the tweet is an important factor. As the memory network was proven to be beneficial to recommendation task performance in social media [17], [18], in this paper, a novel co-attention memory network architecture is porposed to perform this task. The proposed network architecture adopted the neural memory network [19] to incorporate the content of a tweet with corresponding images and post histories that contain the candidate hashtag. It consists of two main components to model the tweet and interests of a candidate hashtag, which are incorporated into the external memory parts.

Moreover, the related entities are often only related to a small part of the image or text. Be lenient for the irrelevant or unimportant parts of the image or text, it would incorporate noise to make a prediction. Hence, using a global vector to present the image or text may not be a good choice. Motivated by the work on image question

1. https://investor.twitterinc.com/results.cfm

2. https://thenextweb.com/socialmedia/2015/11/03/what-analyzing-1-million-tweets-taught-us/

answering [20], [21] and image captioning [22], we introduce an attention mechanism to improve hashtag recommendations. The model is allowed to focus on specific parts of the input with the help of the attention mechanism. With the help of the co-attention mechanism, our model can extract important parts of the visual and textual information of the tweet, which are necessary to construct the complete meaning of the multimodal tweet. More specifically, the proposed network is an end-to-end neural memory network combined with a co-attention mechanism. This model can simultaneously take into consideration both the content of a tweet with corresponding images, and post history interests of candidate hashtag. Particularly, the model can tackle new hashtags, and does not need to re-build the model when new hashtags are added into the corpora. Finally, predictions are calculated based on the similarity features extracted from the multimodal information of tweets and the post histories that contain the candidate hashtag.

To demonstrate the effectiveness of our model, we performed experiments on a large data set collected from Twitter. The experimental results showed that the proposed method could achieve better performance than state-of-the-art methods, and also tackle new hashtags which are added into the corpora. The main contributions of our work can be summarized as follows.

- We introduce a novel matching-based framework for hashtag recommendation tasks. The problem of new hashtags that are added into the corpora can be addressed.
- We propose a novel network architecture combined with the attention mechanism to incorporate the content of a multimodal tweet and interest of the candidate hashtag.
- Experimental results demonstrate that the proposed method can achieve significantly better performance than current state-of-the-art methods.

2 Related Work

2.1 Hashtag Recommendation

Because of the increasing requirements, various studies recently have been performed for different recommendation tasks on social media, such as content recommendation [23], [24], [25], community recommendation [26], [27], [28], [29], music recommendation [30], [31], [32], news recommendation [33], [34], topic recommendation [18], [35], [36], mention recommendation [17], [37], [38] and hashtag recommendation [14], [16], [39], [40].

For the hashtag recommendation task, various methods have been proposed in the past few years. [41], [42] proposed similarity-based methods. [41] tried to extract hashtags from similar tweets as candidate hashtags, then selected hashtags from these candidates by using heuristics ranking methods. [42] made use of not only similar tweets, but also similar users to choose candidate hashtags. [13], [14], [43], [44] recommended hashtags by using topic modeling. [13] elicited latent topics with Latent Dirichlet Allocation (LDA) to recommend tags of resources in order to improve the search. [43] used LDA to model the topic distribution, using this distribution to recommend general hashtags. [14] assumed that the tweet and hashtag are talking about the same theme but in different languages, so they proposed a topic-translation model to extract the specific topic. To explore the possibility of predicting hashtags for un-annotated status updates, [39] proposed a graph-based prediction framework. [44] treated hashtags as labels of topics and proposed a model named TOMOHA to discover the relationship between words, hashtags, and tweet topics. They also combined the user following relationship into the model. [45] modeled hashtag relevance by using their proposed learning-to-rank method, which could extract time-aware features from highly dynamic content. [15] incorporated the trigger words into the model by using convolutional neural networks (CNNs). [46] tried to learn users' perceptions of topics to recommend hashtags, using topic-term relationships that were extracted by discriminative term weights.

Most of these methods only used textual information, despite the fact that Twitter allows users to post various kinds of messages such as images, videos, hyperlinks and so on. Some other works have shown that it is useful to combine different types of information into the model. [47] proposed a topical model to integrate the temporal and personal information into consideration. [48] integrated hyperlinked information with textual information and found it also useful for hashtag recommendation. [49] proposed a hierarchical attention network architecture to combine the textual information and the corresponding user history. [16] found that images posted by users in the tweets can provide valuable information for hashtag selection. Hence, they proposed a co-attention network to incorporate images and textual information of multimodal tweets.

Based on the descriptions above, we can arrive at the conclusion that making use of various valuable information can significantly improve the performance of hashtag recommendation. Inspired by this, in this work, textual information, visual information of multimodal tweets and the posting histories of hashtag are incorporated into our model to convert the hashtag recommendation task into a matching-based problem.

2.2 Multimodal Tasks

Multimodal tasks have attracted a lot of attention with the information getting more diverse, and have been studied in various aspects. Futhermore, different tasks take various view of the visual information. For example, the image caption task is focus on generating a textual description of images and visual questions answering is answering the question according to a given image. [22] used long short-term memory networks (LSTM) to make use of high-level image features to generate captions. [50] proposed a scheme to detect the copy-move forgery in an image by extracting the key points for comparison. In their method, the scheme first segments the test image into semantically independent patches prior to keypoint extraction. [51] represented the image regions by CNNs, and represented the sentences by bidirectional Recurrent Neural Networks (BRNNs), then combined the two representations to generate descriptions of image regions. The major related work of multimodal tasks, the visual question answering task, has been studied by various methods as well. Based on CNNs and RNNs,

most early works tried to turn the image caption task [22], [52] into a visual question answering task [53], [54]. Recent works [55], [56], [57] showed that the attention mechanism can be significantly useful for aligning text and image information.

As mentioned above, the attention mechanism has been proved successful in various multimodal tasks. In this work, the hashtag recommendation is treated as a multimodal task and integrate visual information into our model. Furthermore, we make use of users' multimodal post histories to do a cross attention examination with features extracted by the co-attention mechanism.

2.3 Matching Problem

In natural language processing, the matching problem is an important task and has been explored with various methods. [58] surveyed recent advances in solving matching problems.

With the development of deep neural networks, the matching problem has been studied with deep neural networks and has achieved remarkable results. [59] developed a series of latent semantic models with a deep structure that projects queries and documents into a common low-dimensional space, then computes the relevance between the document and the given query with respect to their distance. To model the complicated relationships between two objects from heterogeneous domains, [60] proposed a model to combine the localness and hierarchy intrinsic for short text matching. [61] approached short text matching with a method named Deep Match Tree to make use of the texts' syntax information. It first discovers patterns to match two short portions of text, which is defined in the product space of dependency trees, then it matches the patterns by using deep neural networks. [62] used CNNs to rerank pairs of short texts, which can learn the optimal representation of text pairs and a similarity function to relate them in a supervised way from the available training data. For sentence matching, [63] proposed novel deep convolutional network architectures to represent the hierarchical structures of sentences and capture the rich matching patterns at different levels. [64] addressed sentence embedding by using LSTM.

Most of recent works have treated hashtag recommendation tasks as a multi-class classification problem, which can only handle fixed numbers of hashtags. However, because of the development of hashtags and new trends in using them, multi-class classification is not suitable for hashtag recommendation anymore. In this study, we addressed the hashtag recommendation task as a matching problem, which can be used to handle the new hashtag issue.

2.4 Attention mechanism and Memory

The attention mechanism in neural networks is based on the visual attention mechanism found in humans. Human visual attention suggests that our brains usually focus on selective parts of the whole perception regions according to demand. So the attention mechanism provides the possibility to avoid noise in the input, focusing on the useful parts that can contribute to improving the model performance, and proved to be powerful in various tasks, such as machine translation [65], speech recognition [66], action recognition [67], image classification [68], image caption generation [69] and so on. The attention mechanism gives the model the ability to select the most pertinent piece of information, rather than using all available information.

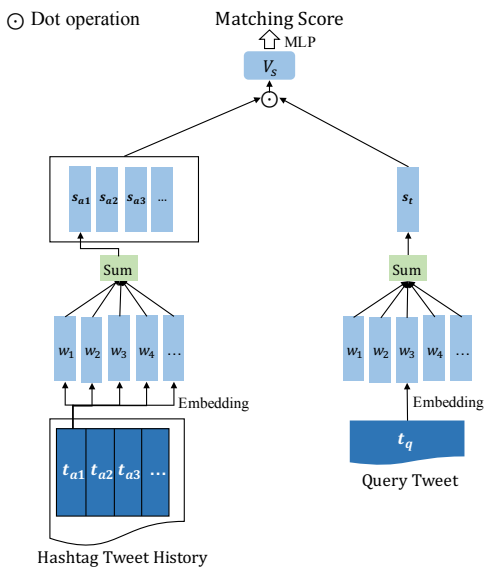


Fig. 1. Matching score for generating candidate hashtag set. Here, the representation of each tweet is formulated by summing the pre-trained word vectors of all words in each tweet. Then we generate a dot result between these representation vectors. Finally, a multi-layer perceptron is applied to the dot result to formulate matching score.

Recent works have showed that models can achieve significant improvements by using memory networks [70] in various NLP tasks, for example language modeling [19], reading comprehension [71], question answering [19], [72], [73], dialog systems [74], [75], etc.

In this work, we combine the attention mechanism with memory networks to do hashtag recommendation on multimodal tweets. We first get tweet content-based new visual representations and visual-based new tweet representations using the co-attention mechanism. Then we combine the two representations with a cross attention memory network to extract the important information from users' multimodal post histories as the users' interests. Our model can achieve remarkable recommendation results because the significant parts have been extracted.

3 Approach

In this work, we reinterpret the hashtag automatic recommendation task as a matching problem. More specifically, our model can choose whether a hashtag $h \in H$ should be recommended for the query multimodal tweet t. Moreover, the query multimodal tweet t contains both a textual part t_x and corresponding image t_i. In addition, the list of candidate hashtags H for each query tweet t is composed of the top L hashtags based on the matching score between the tweet and the post history of each hashtag. The model for generating the candidate hashtag set is illustrated in Figure 1.

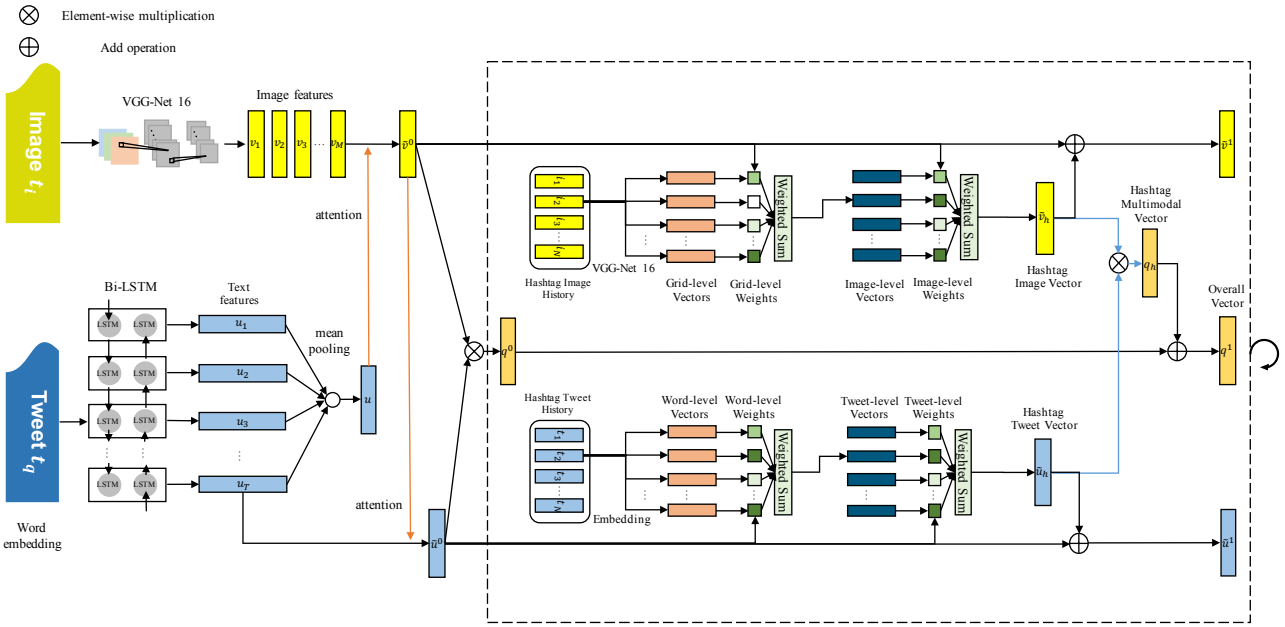


Fig. 2. This is a example of one-layer CoA-MN, which consists of two components: (1) Query Tweet Modelling, (2) Hashtag History Interests Modelling. Here, we denote $\widetilde{u}^0$ as the representation of query tweet and $\widetilde{v}^0$ as the representation of corresponding image, and use $\widetilde{u}^0, \widetilde{v}^0$ query over the candidate hashtag's tweet histories and image histories, respectively. q^k is the final representation of the overall architecture which modeling the interests similarity among query tweet and a candidate hashtag.

Then, we introduce a novel co-attention memory network framework to perform the hashtag recommendation task, as illustrated in Figure 2. The "Query Tweet Modelling" utilize the co-attention mechanism to construct the multimodal representation of query tweet ($\widetilde{u}^0$ denotes the representation of query tweet and $\widetilde{v}^0$ denotes the representation of corresponding image). Further, the "Hashtag History Interests Modelling" are performed by stacked hierarchical memory networks.

The inputs of our model are the query tweet t and the candidate hashtag's posting history. The posting histories that contain the same hashtag can represent the attributes and interests of the hashtag. Hence, we utilize the tweet posting histories to model the tweet history interests of the hashtag, and the corresponding image histories to model the image history interests of the hashtag. Firstly, we utilize the pre-trained VGG-Net 16 to formulate the representation of images in our model, including the images of the query tweet and hashtag's posting histories. Meanwhile, we use a tweet encoder to represent the tweet. Secondly, we construct the representation of the query tweet by using a co-attention neural network and then encode the history interests of the candidate hashtag with the help of the representation of the query tweet. Specifically, a hierarchical attention mechanism is applied to help the encoder formulate a high-quality history interests representation. Moreover, to construct the final representation, we repeat the history interests updating procedure for k steps. The steps are denoted as $k = \{0, 1, 2, \cdots, K\}$. Finally, we use a fully connected softmax layer to make matching prediction. To be clear, we list the explanation of the key notations in Table 1.

TABLE 1
Meaning of formal notation

N	The memory capacity
T	The maximum tweet length
M	The number of image grids
e_i	The i-th image
t_j	The j-th tweet
$\widetilde{v}_h$	The candidate hashtag's image history representation
$\widetilde{u}_h$	The candidate hashtag's tweet history representation
$\widetilde{V}_M^{k-1}$	The matrix formulated by M columns of $\widetilde{v}^{k-1}$
$a_{i,M}^k$	The attention probability of each region in i-th image
q^k	The global representation after k-layer hashtag history modelling memory network

3.1 Feature Extraction

Image representation

We use a pretrained 16-layer VGGNet [76] to formulate the representation of an image. Firstly, images are rescaled into 224×224. Different from previous methods that construct a global vector as the image representation, we construct spatial features of different divisions that contain more specific information about the original image. We divide an image into $N \times N$ parts, and use VGGNet to construct a 512-dimension feature vector for each part. Hence, we can use $v_I = \{v_i | v_i \in \mathbb{R}^D, i = 1, 2, \cdots, m\}$ to represent an image, where $m = 7 \times 7$ is the number of image grids, and v_i is a 512-dimensional feature vector for grid i. For computational aspects, we use a single layer perceptron to convert each image vector into a new vector that has the same dimensions as the tweet feature vector.

Text representation

Originally, each word w_i of a query tweet t is formulated as a one-hot vector. Then, we embed each one-hot vector into a word vector x_i by utilizing a embedding layer. Hence, we can have a word-level tweet feature representation : $t = \{x_1, x_2, \cdots, x_T\}$, where T is the max length of the

tweet. Moreover, we pad zero vectors into those sentences of length less than T. Particularly, the word embedding matrix is trained end-to-end over the whole model.

The Long Short-Term Memory network (LSTM) is a kind of RNN designed to solve the issue of learning long-term dependencies, and it has been proved to be able to achieve a good performance in understanding text. However, since RNNs process sequences in time series, they tend to ignore future context information. A bidirectional RNN, which feeds each training sequence forward and backward, respectively, was proposed to address that problem. This structure provides a complete past and future context information set for each point in the input sequence of the output layer. Hence, we utilize the bidirectional LSTM to construct a sentence-level tweet features representation. At each time step, the bidirectional LSTM unit takes the word embedding vector x_t as an input vector and outputs a hidden state h_t. As shown in the following:

$$h_t{}^{(f)} = LSTM^{(f)}(x_t, h_{t-1}{}^{(f)}), \tag{1}$$

$$h_t{}^{(b)} = LSTM^{(b)}(x_t, h_{t+1}{}^{(b)}), \tag{2}$$

where $h_t{}^{(f)}$ and $h_t{}^{(b)}$ represent the hidden states at time step t from the forward and backward LSTMs, respectively. Finally, we construct a set of text representation vectors $u_T = \{u_1, u_2, \cdots, u_n\}$ by summing these two hidden state vectors at each time step [3]:

$$u_t = h_t{}^{(f)} + h_t{}^{(b)}, \tag{3}$$

where u_t is the feature vector of the t-th word in the context of the entire sentence.

3.2 Co-attention based Tweet Modelling

After the process denoted in Sec 3.1, we formulate the image representation matrix v_I and the tweet representation matrix v_T. It is clearly that texts and images contain different levels of abstraction for a tweet. Therefore, we introduce a co-attention network to construct high-level representation of the query tweet. Because the textual data is more meaningful in formulating an abstraction of the query tweet, we utilize text-based attention and image-based attention sequentially.

Text-based visual attention

Usually, a hashtag is only related to few grids of the corresponding image. And, only a few parts of the image represent the entity in the image. In other words, many grids of the image are noises according to the hashtag. Hence, instead of using a global vector to represent the image, we divide the image into 49 parts and construct the representation of each division to obtain a feature matrix v_I. Then, a text-based attention mechanism is incorporated to filter out noises and locate grids that are relevant to the corresponding hashtag.

Above all, we use an average pooling layer to summarize the sentence-level representation of a tweet to a vector: $\overline{u}$. Next, we incorporate an image attention with the help of the sentence-level representation. The operation of the text-based attention is computed by using a 2-layer feed-forward neural network (FNN) and the softmax function:

$$h_M = tanh(W_{v_M} v_M) \odot tanh(W_{v_U} \overline{U}), \tag{4}$$

$$a_M = softmax(W_h h_M), \tag{5}$$

where $\overline{U} \in \mathbb{R}^{d \times M}$ is a matrix formulated by M columns of $\overline{u}$ and $v_M \in \mathbb{R}^{d \times M}$, d is the dimension of the representation, and M is the number of divided grids of each image. $a_M \in \mathbb{R}^M$, which corresponds to the attention probability of each grid, is an m-dimensional vector. We use $\odot$ to denote element-wise multiplication of the image matrix and mean-pooling tweet matrix.

As the attention probability a_m of each image grid m is calculated in the above process, we use the weighted sum of the image grid vectors to construct the high-level representation of the image.

$$\widetilde{v}_I = \sum_m a_m v_m \tag{6}$$

Image-based textual attention

With the help of text-based visual attention, a new image feature representation $\widetilde{v}_I$ which is related to each word in the given tweet. Similar to text-based visual attention, image-based textual attention is incorporated to help the model focus on more important words when constructing the sentence-level meaning of a tweet. Specifically, we used the new image representation vector $\widetilde{v}_I$ to query the original textual feature u_T, formulating a new text representation $\widetilde{u}_T$ based on the textual attention probability distributions. The detail is as follows:

$$z_T = tanh(W_{u_T} u_T) \odot tanh(W_{u_V} \widetilde{V}_I), \tag{7}$$

$$a_T = softmax(W_z z_T), \tag{8}$$

where $\widetilde{V}_I \in \mathbb{R}^{d \times T}$ is a matrix formulated by T columns of $\widetilde{v}_I$, $u_T \in \mathbb{R}^{d \times T}$and T is the max length of tweets. And "$\odot$" denotes element-wise multiplication of the word feature matrix and new image feature matrix.

After the attention probability for each word is calculated, the new representation of the tweet is formulated by the weighted sum of each word vector:

$$\widetilde{u}_T = \sum_t a_t u_t \tag{9}$$

3.3 Hashtag History Modelling

It is clear that the hashtag history stored in the memory has a hierarchical structure. Firstly, each tweet document has many tweets: $D_T = \{t_1, t_2, \cdots, t_N\}$ and a tweet-level structure. Each tweet also has many words: $t = \{w_1, w_2, \cdots, w_T\}$, and each image document has many corresponding images:$D_I = \{e_1, e_2, \cdots, e_N\}$. Each image has been divided into grids: $e = \{v_1, v_2, \cdots, v_M\}$ and a grid-level structure. In view of the fact that not all parts of the history are equally important, we utilize a hierarchical attention mechanism to model the hashtag history. Particularly, we can stack up the history modelling layer to achieve better performance.

Grid-level modelling

3. We have tried the two methods to construct the text representation, as concatenation and summing two hidden state vectors at each time step, and the summing method is performing better than the concatenation method.

In the input image set $D_I = \{e_1, e_2, \cdots, e_N\}$, each grid's vector $r_{i,j} \in e_i$ is extracted by a 16-layer VGGNet and saved as a visual memory vector. And $r_{i,j}$ is a 512 dimension vector. To simplify the calculating process, we utilize a fully connected layer $v_{i,j} = Wr_{i,j}$ to convert each grid's original vector $r_{i,j}$ to the same dimension as the tweet feature vector.

Because not all grids in an image are equally important to model the image history interests, at the k-th hop of the history modelling network, we utilize the image representation vector $\widetilde{v}^{k-1}$ of the last hop to query the grid vectors, constructing a new representation of each history image. The detail is illustrated as follows:

$$h^k_{i,M} = tanh(W^k_M v_{i,M}) \odot tanh(W^k_{\widetilde{v}} \widetilde{V}^{k-1}_M), \tag{10}$$

$$a^k_{i,M} = softmax(W^k_h h^k_{i,M}), \tag{11}$$

$$v^*_i = \sum_j^M a^k_{i,j} v_{i,j}, \tag{12}$$

where $\widetilde{V}^{k-1}_M \in \mathbb{R}^{d \times M}$ is a matrix formulated by M columns of $\widetilde{v}^{k-1}$ and M is the grid number of each image. $a^k_{i,M} \in \mathbb{R}^M$, which corresponds to the attention probability of each region in i-th image.

Then, each image is converted into a fixed-size vector $v^*_i \in \mathbb{R}^d$, which represents the interest embedding of the i-th image.

Image-level modelling

As we formulate a new representation v^*_i for each history image based on a grid-level attention mechanism, there is no doubt that each image is unequally relevant to model a hashtag's image history interests. To model the whole image history interests of a hashtag, we query the new representation of each history image with the help of the last step of the image history interests vector $\widetilde{v}^{k-1}$:

$$h^k_N = tanh(W^k_N v^*_N) \odot tanh(W^k_{\widetilde{v}_N} \widetilde{V}^{k-1}_N), \tag{13}$$

$$a^k_N = softmax(W^k_{h_N} h^k_N), \tag{14}$$

$$\widetilde{v}_h = \sum_i^N a^k_i v^*_i, \tag{15}$$

where $\widetilde{V}^{k-1}_N \in \mathbb{R}^{d \times N}$ is a matrix formulated by N columns of $\widetilde{v}^{k-1}$ and N is the image capacity of the image history memory, $\widetilde{v}_h$ is the representation of the candidate hashtag's image history. And $a^k_N \in \mathbb{R}^N$, which corresponds to the attention probability of each image in an image history memory.

Through the above procedure, we formulate the representation $\widetilde{v} \in \mathbb{R}^d$ for a candidate hashtag's whole image history interests and d is the dimension of the vector.

Word-level modelling

Similar to the image history, the tweet part history also has a two-level architecture. Above all, in the text history set $t_1, t_2, ..., t_N$, each word $w_{i,j}$ of the corresponding tweet t_i is embedded into an d-dimension textual memory vector $c_{i,j}$ by utilizing an embedding matrix A (the shape of A is $d \times |V|$), as $c_{i,j} = Aw_{i,j}$. The memory $c_{i,j}$ in this step is similar to the image memory $v_{x,y}$, which is also named input memory. And we can project the input words of historical tweets into the same space by applying the textual memory.

In order to filter out irrelevant words, at the k-th hashtag history interests modelling layer, we formulate attention probabilities over a hashtag's word memory vector set with the help of utilizing the last hop of the textual representation $\widetilde{u}^{k-1}$.The match between input memory vector $c_{i,j}$ and $\widetilde{u}^{k-1}$ is computed by incorporating the inner product followed by a softmax layer:

$$z^k_{i,T} = (\widetilde{u}^{k-1})^{tr} c_{i,T}, \tag{16}$$

$$p^k_{i,T} = softmax(W^k_z z^k_{i,T}), \tag{17}$$

where $(\widetilde{u}^{k-1})^{tr}$ is the transpose of last step of the tweet representation vector $\widetilde{u}^{k-1}$ and T is the maximum length of each tweet. And $p^k_{i,T} \in \mathbb{R}^T$ denotes the attention probability of each word in i-th tweet.

Unlike the grid-level encoder, we apply a new embedding matrix B to project the word $w_{i,j}$ into another memory vector $u_{i,j}$ (as same as d-dimension and named output memory), as $u_{i,j} = Bw_{i,j}$. Then, the tweet-level representation is generated by summing all output memory vectors weighted by the attention probability denoted above:

$$u^*_i = \sum_j^T p^k_{i,j} u_{i,j}, \tag{18}$$

Following the above steps, we convert each tweet in the tweet history into a fixed-length vector that denotes the representation of the tweet.

Tweet-level modelling

In order to formulate the tweet-level history interests of a candidate hashtag, we introduce a tweet-level encoder to select important parts of the tweet history memory. Given the encoded set of tweets $s = \{u^*_1, u^*_2, \cdots, u^*_N\}$, the history interest representation of the candidate hashtag's textual history is constructed by a weighted sum of the new tweet representations denoted above. The weights of each tweet are described as the important level of the corresponding tweet in the tweet history. The equation of this procedure is as follows:

$$z^k_N = tanh(W^k_N u^*_N) \odot tanh(W^k_{\widetilde{u}_N} \widetilde{U}^{k-1}_N), \tag{19}$$

$$p^k_N = softmax(W^k_{z_N} z^k_N), \tag{20}$$

$$\widetilde{u}_h = \sum_i^N p^k_i u^*_i, \tag{21}$$

where $\widetilde{U}^{k-1}_N \in \mathbb{R}^{d \times N}$ is a matrix formulated by N columns of $\widetilde{u}^{k-1}$, N is the tweet capacity of the tweet history memory, and $\widetilde{u}_h$ is a representation of the candidate hashtag's tweet history. And $p^k_N \in \mathbb{R}^N$ denotes the attention probability of each tweet in a tweet history memory.

Stacked history modelling network

In order to model more complex history interests and calculating the similarity between the query tweet and a candidate hashtag, we try to repeat the tweet history modelling network iteratively by using the last generated representations. Moreover, the stacked procedure can be summarized as follows: for the k-th (where k is greater than or equal to 1) tweet history modelling layer, we formulate the image history interests and the tweet history interests representation of the candidate hashtag based on the query tweet, respectively. The new query vector is formed by

adding the new feature vector to the previous vector. The detail is as follows:

$$\widetilde{u}^k = \widetilde{u}^{k-1} + \widetilde{u}_h^k, \tag{22}$$

$$\widetilde{v}^k = \widetilde{v}^{k-1} + \widetilde{v}_h^k, \tag{23}$$

where $\widetilde{u}^0$ is initialized by tweet presentation $\widetilde{u}_T$ of the query tweet, and $\widetilde{v}^0$ is initialized by image presentation $\widetilde{v}_I$ of the query tweet.

Further, the k-th global representation vector, which is used to model the history interests similarity between the candidate hashtag and the query tweet, is updated after the k-th image history interests and tweet history interests representation of the candidate hashtag:

$$q_h^k = \widetilde{u}_h^k \odot tanh(W_h \widetilde{v}_h^k), \tag{24}$$

$$q^k = q^{k-1} + q_h^k, \tag{25}$$

where q_h^k is the whole interests representation (named final representation) of the candidate hashtag's tweet histories and the corresponding image histories at k-th hashtag history modelling memory layer. $\odot$ is element-wise multiplication. Particularly, considering the inequalities of information density of word vectors and picture grid vectors, we apply an additional layer W_h to make the image history representation have similar information densities with tweet history representation.

3.4 Final Prediction

Based on the representation generated using the above steps, we incorporate a single-layer softmax classifier to determine whether or not a candidate hashtag h should be recommended for the query tweet t. The feature vector is passed into the fully connected layer:

$$f = \sigma(W_q q^k + b_q), \tag{26}$$

where W_q is the weight parameter of the fully connected hidden layer, b_q is the bias parameter of the hidden layer, q^k is the final representation obtained after the k times hashtag history modelling layer and σ is a non-linear activation function.

The final prediction is generated by a softmax layer:

$$p(y = i|f; \theta_s) = \frac{exp(\theta_s^i f)}{\sum_j exp(\theta_s^j f)}, \tag{27}$$

where θ_s^i is a weight vector of the i-th class and $j \in \{0, 1\}$.

According to the scores output from the softmax layer, we can select the top-ranked recommended hashtags for the query tweet.

3.5 Training

In our work, the training objective function is formulated as follows:

$$J = \sum_{(t_q, a, c, i) \in D} -logp(i|t_q, a, c; \theta), \tag{28}$$

where D is the training data set, $i \in \{0, 1\}$ is the label of the double tuple (t_q, h). When $i = 1$, the candidate hashtag h should be recommended to the query tweet t_q, and $i = 0$ represents the candidate hashtag h that should not be recommended. θ is the whole parameter set of our model.

To minimize the objective function, we use a stochastic gradient descent (SGD) with the Adam [77] update rule, and the learning rate $\alpha = 0.001, \beta = (0.9, 0.99)$. The batch size is 256. The model is implemented in Keras, and all parameters are initialized by Keras in default methods. Then, we utilize the dropout and add l_2-norm terms for the regularization (the parameter of the dropout is 0.2 in our training procedure).

4 Experiment

In this section, we first describe the data set collected from Twitter. Then, we introduce the experiment setting and baseline methods. Finally, analyses are given according to the performance of our experiments.

4.1 Dataset and Setup

We started by using Twitter's API[4] to collect public tweets from randomly selected users. We randomly selected 1.2 million users and crawled their post histories, including 252.6 million tweets. Then, we selected those tweets that contained both images and hashtags from the above collection. Among them, 2.05 million tweets were chosen. Moreover, we filtered out the hashtags whose frequencies were very low in our data set, and the unique number of hashtags preserved in the corpus was 3,280. Next, we randomly picked out 8 tweets for each preserved hashtag as the history set of the hashtags. Then the history set contains 26,240 tweets and 26,240 corresponding images. Finally, the collection we constructed contained 334,019 tweets with corresponding images. The average number of hashtags per tweet was 1.15 in the corpus. The detailed statistics are shown in Table 3. We split the dataset into a training set and a test set, with a ratio of 8:2, and randomly selected 20% of the training set as the valid set.

For text words, we filtered out the stop words and low-frequency words in our work. The constructed word vocabulary contained 259,410 distinct words. For images, we downloaded images from the retrieved urls and rescaled them to 224×224. Then we fed them into a pre-trained VGG-16 network. The outputs of the last pooling layer of VGGnet were extracted as the image features. For the memory portion, the capacity of the memory was restricted to 5 tweets with corresponding images, and the maximum length of each tweet was 34. In other words, we randomly extracted 5 tweets from each hashtag history set that contained 8 tweets, and used these 5 tweets to present the hashtag's history interests and stored them in the supporting memory. The size of candidate hashtag set was 10 (we also took experiments on a candidate hashtag set of 30 or 50, and our model achieved better performance on the candidate hashtag set with a size of 10). The embedding dimension in the experiment was 300 (we also transferred the image feature dimension from 512 to 300), and the depth of hashtag history modeling memory layer was set to 5. The learning rate was set to 0.01, and the dropout rate was set to 0.2.

4. https://developer.twitter.com/

TABLE 2
Comparison performance of different methods on the testing dataset.

Method	Precision	Recall	F1-Score	Hits@3	Hits@5
NB	0.078	0.067	0.072	0.123	0.147
SVM	0.187	0.189	0.188	0.303	0.366
KNN	0.150	0.121	0.134	0.283	0.342
LSTM+CNN	0.201	0.197	0.199	0.321	0.383
LSTM+CNN+H	0.411	0.401	0.405	0.520	0.637
TTM [14]	0.185	0.184	0.184	0.300	0.361
TOMOHA [44]	0.177	0.186	0.181	0.304	0.363
CNN-Attention [15]	0.217	0.216	0.216	0.310	0.368
Co-Attention [16]	0.288	0.271	0.279	0.366	0.400
CoA-MN	**0.533**	**0.518**	**0.525**	**0.654**	**0.730**

TABLE 3
Statistics of the evaluation dataset. Avg Hashtag/Tweet represents the average number of manually labelled hashtags per tweet.

#Tweets	334,019
#Images	334,019
#Hashtags	3,280
#Avg Hashtag/Tweet	1.15

In our work, we used three metrics to evaluate the performance of our model, which are the precision (P), recall (R), and the F1-score (F1). The number of recommended hashtags for each tweet is denoted as L, where $L = \{1, 2, 3, 4, 5\}$, the precision, recall, and F1-score at the L result are denoted as P_L, R_L, and $F1_L$, respectively. Moreover, we incorporated Hits@3 and Hits@5 to represent the percentage of correct results recommended from the top n results.

4.2 Baseline

To analyze the effectiveness of our model, we evaluated some effective methods including the state-of-the-art methods as baselines on the constructed corpus, described as follows:

- **NB**: To perform the hashtag automatically recommending task, we converted the problem into a multi-classification problem. We applied Naive Bayes to model the posterior probability of each hashtag by only using the textual information of the tweets.
- **SVM**: We utilized the pre-trained word vector[5] and summed them as the feature vector of the tweet. These pre-trained 100-dimensional word vectors were trained on aggregated global word-word co-occurrence statistics from a Twitter corpus. The corpus contain 1.2 million words and it using the Glove architectures for computing vector representations of words. Then, we used these tweet feature vectors to implement the support vector machine for the recommendation.
- **KNN**: We also utilized the pre-trained 100-dimensional Glove Twitter word vector and averaged them as the feature vector of the tweet, then used the cosine similarity distance between tweet feature vectors to assign the hashtag label that was the most common among its k nearest neighbors. We recorded the best results when the K was equal to 15.
- **LSTM+CNN**:LSTM+CNN also treated the hashtag automatically recommending task as a multi-classification problem. We also combined the textual feature processed by LSTM with visual feature processed by CNN to model and make a prediction.
- **LSTM+CNN+H**:To assess the usefulness of the hashtag posting history, the query tweet and hashtag posting history were given to LSTM and CNN to make a decision on whether or not a candidate hashtag should be recommended. Specifically, the difference between this method and the proposed model, is that the LSTM+CNN+H just simply adds the image feature vector and the text feature vector of the query tweet without any attention mechanism.
- **TTM**: TTM was proposed by [14] for hashtag recommendation. The authors proposed a topical translation model to recommend hashtags, which only used the textual information.
- **TOMOHA**: TOMOHA was proposed by [44] for hashtag recommendation and was a supervised topic model-based solution. The authors treated hashtags as labels for topics, and they developed a supervised topic model to discover relationship among words, hashtags and tweet topics.
- **CNN-Attention**:CNN-Attention was proposed by [15] and was a CNN architecture that used the attention mechanism to incorporate trigger words.
- **Co-Attention**: Co-Attention network was proposed by [16], and the paper incorporated textual and visual information to recommend hashtags for multimodal tweets. This was the state-of-the-art approach used for the hashtag recommendation task.

4.3 Result and Discussion

The performance of different methods on our dataset is listed in Table 2. The first three metric results (Precision, Recall and F1-score) listed in Table 2 were obtained when we

5. https://nlp.stanford.edu/projects/glove/

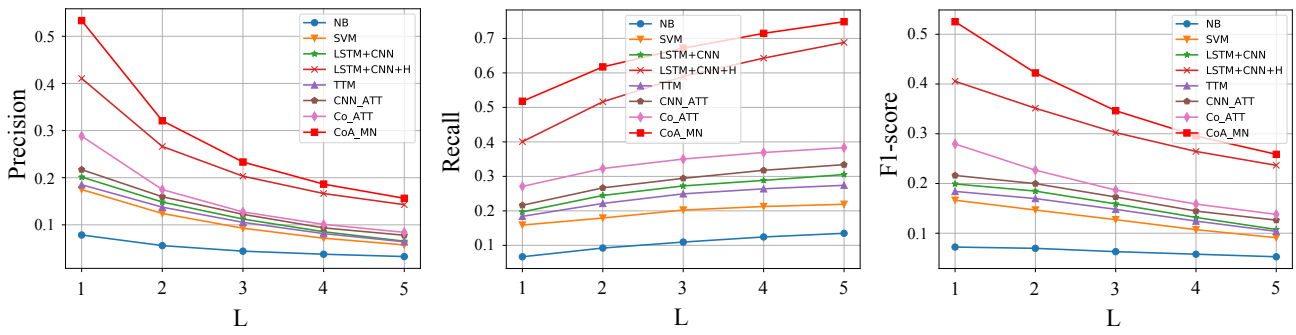


Fig. 3. Precision, Recall and F1-score with different amount of recommended users

recommended the top one hashtag for each query tweet. The last two metric results (Hits@3 and Hits@5) represented how many hits items are found within the top-n recommended items ($n = 3$ and $n = 5$, respectively). We can find that our proposed model (CoA-MN) achieves a better performance than other comparison methods of all metrics.

Above all, compared with Co-Attention, which was the state-of-the-art method for the automatic hashtag recommendations, our proposed model (CoA-MN) shows a 24.5% absolute improvement in terms of Precision, a 24.7% absolute improvement in Recall and a 24.6% absolute improvement in F1-score. Particularly, the Hits@3 and Hits@5 results of our proposed results of our model are greater than 0.654 and 0.730, respectively. Therefore, we can find that 65.4% of correct hashtags can be found in the top 3 of the recommendation list and 73.0% of the hashtags can be recommended in the top 5.

Observing the comparisons of the "LSTM+CNN" and the "LSTM+CNN+H", it illustrates that the hashtag's posting tweet histories and corresponding images are key features to improving the performance of hashtag recommendations. From the results table, we can observe that normally combining the textual feature with visual features cannot make a perfect hashtag prediction. After incorporating the posting histories of hashtags, we can clearly find that "LSTM+CNN+H" achieves a 21.0% absolute improvement in terms of Precision, a 20.4% absolute improvement in Recall and a 20.6% absolute improvement in F1-score than "LSTM+CNN". Moreover, the Hits@3 and Hits@5 results of our proposed results from "LSTM+CNN+H" are greater than all other methods except our model. In other words, it is the strongest confirmation that much important information in the hashtag posting history can be used to recommend hashtags in social media.

From Table 2, we can observe that our proposed model (CoA-MN) consistently achieves a better performance in all evaluation results than "LSTM+CNN+H". Along with our proposed model (CoA-MN) achieves 12.2% absolute improvement in terms of Precision, 11.7% absolute improvement in Recall and 12.0% absolute improvement in F1-score than "LSTM+CNN+H". It is therefore proved that our model can extract more useful information from both textual features and visual features with the help of the co-attention mechanism. Hence, by incorporating a hashtag' tweet posting history and corresponding images and utilizing the co-attention mechanism to combine these two features, our proposed model achieved great performance on the hashtag recommendation task.

Figure 3 shows the Precision, Recall and F1-score of the models with different numbers of recommended hashtags. Each point of the curve represents the number of hashtags recommended ranging from 1 to 5. To emphasize our experimental results, we used the red line combined with a square label to draw Figure 3. It is shown that Precision decreases and Recall increases as the number of recommended hashtags increases. Particularly, our model achieves extremely better performance than other methods in all metrics. The curve that is higher on the graph indicates the better performance. From Figure 3, we can see that the performance of our proposed model is the highest of all the methods in all conditions as the number of hashtags recommended ranges from 1 to 5. Definitely, Figure 3 not only denotes that our proposed method was significantly better than the state-of-the-art methods, but it also proves that our model is beneficial for hashtag recommendation that incorporates the posting histories of hashtags.

4.4 Parameter Influence

There are several critical hyper-parameters influencing the performance of our proposed model. To evaluate the influence of the parameters used in our model, we varied one parameter and fixed the others in turn. The effects of the hashtag candidate set size are shown, the results of different depth of hashtag history modeling memory network are compared, the performance with different embedding dimensions and the existence of a dropout layer were investigated. Based on the experimental results, we can observe that the proposed model could achieve stable performance, in the condition of various parameter settings.

First of all, we evaluated the influence of the hashtag candidate set size and performed experiments on candidate sets of size 10, 30 and 50. In Figure 4, we show the histogram of the three metrics (Precision, Recall and F-score) with different candidates sets. It is clear that when the size of the hashtag candidate set is smaller, the better performance of these three metric is achieved. We also obtained the best performance on the candidate set with the size of 10, which indicated that the first step of generating candidate sets have

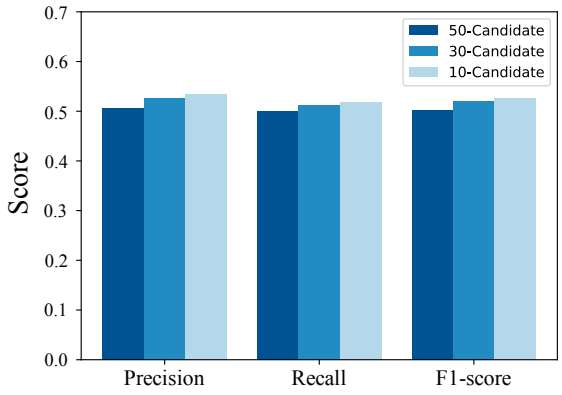


Fig. 4. Influence of Candidate size

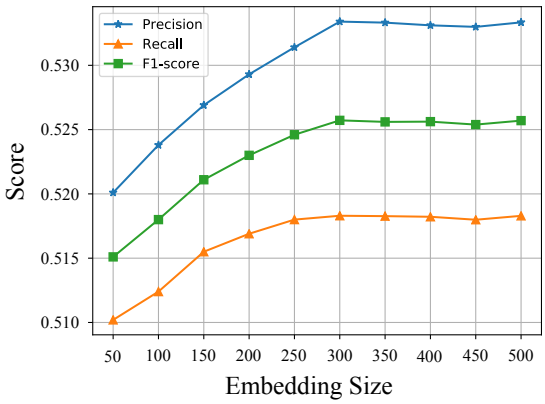


Fig. 6. Influence of Embedding size

a great effect. Hence, future work can also go on how to generate smaller but higher quality candidate sets.

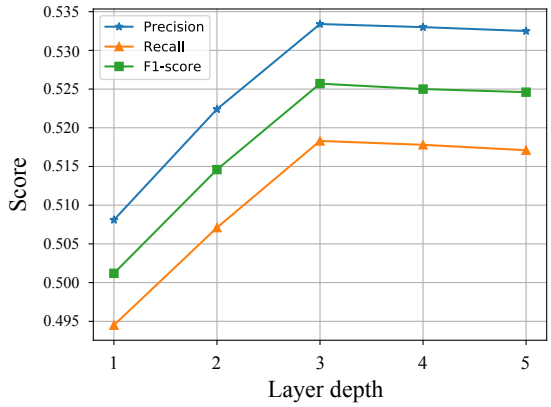


Fig. 5. Performance on a different layer of hashtag history modeling memory network

The second parameter we evaluated is the depth of the hashtag history modelling memory layer, which we varied from 1 to 5 in this experiment. In Figure 5, we draw the Precision, Recall and F-score curves to show the depth's influence on the performance. Along with the increase in the depth of the hashtag history modelling memory layer, the results are better. We also obtained the best performance with the 3-layer hashtag history modelling and the performance with more than 3-layer was slightly lower than the best one, which indicates the robustness of our model along with the deeper depth. Since increasing the number of layers of the network will make the model more complex with more parameters without a significant improvement in the results, hence, the performance of the 4-layer and 5-layer were slightly lower than the 3-layer. The figure demonstrates that the multi-depth of the hashtag history modelling memory network works better than a single-level one.

Figure 6 shows the contributions of the embedding dimension to the performance. To evaluate how it influenced the performance, we fixed the depth of the hashtag history modelling memory layer to 3, the size of a candidate set to 10 and tried different embedding dimensions. The comparison results shown in Figure 5 demonstrate that the models with a high embedding dimensions performed better than those with a low dimension. The results improved when the dimension was increased from 50 to 300, and the results from 300 to 500 embedding dimension were fluctuating around the result of 300 dimensions. In a word, the size of the embedding dimension represents the expression ability for vocabulary and images, and a higher dimension can enhance both the textual feature and visual feature expression ability. To recommend an appropriate hashtag using a more complex model, it is considerate to choose a high embedding dimension. But the Figure 6 shows that, in our work, if we want to give an effective suggestions and simply the complexity of the model in the meantime, the 300 embedding dimension would be a better choice.

Finally, we also compared the performance of our model with and without dropout layers, and the results are shown in Table 4. The depth of the hashtag history modelling layer was 3 and the embedding dimensions were 300. There is no doubt that the model achieved better performance with the help of a dropout layer. Compared to the model without a dropout layer, the one with a dropout layer achieves an improvement of 1.5% in Precision, along with a 1.4% increase in Recall and 1.4% increase in F1-score. In other words, even without a dropout layer, our model achieved greater than 23% improvements in each category, compared with the state-of-art method.

TABLE 4
The performance with and without dropout on our datasets

With Dropout	Precision	Recall	F1-score
No	0.518	0.504	0.511
Yes	**0.533**	**0.518**	**0.525** ·

5 Conclusion

In this work, we proposed a CoA-MN to combine textual and visual information of hashtag's posting history by applying a hierarchical attention mechanism on external

memory. In view of the explosive growth in social media use and the diversity of information, we incorporate the textual information and visual information to perform the hashtag recommendation task. Since tweets and images are not equally important in modelling query tweet representation, we utilize the co-attention mechanism, which generates textual attention and visual attention sequentially. The most important aspect is that we converted the hashtag recommendation task to a matching-based task by incorporating the posting histories of hashtags. This allows us to address the problem that previous methods can only handle fixed amounts of hashtags and fail to deal with newly appeared hashtags. We also constructed a large data collection retrieved from live microblog services to evaluate the effectiveness of our model. Experimental results showed that the proposed method achieves better performance than state-of-the-art methods which treats the hashtag recommendation task as a multi-class classification task.

ACKNOWLEDGMENTS

The authors would like to thank...

REFERENCES

[1] A. Bermingham and A. F. Smeaton, "Classifying sentiment in microblogs: is brevity an advantage?" in *ACM International Conference on Information and Knowledge Management*, 2010, pp. 1833–1836.

[2] Pang, Bo, Lee, and Lillian, "Opinion mining and sentiment analysis," *Foundations & Trends in Information Retrieval*, vol. 2, no. 1, pp. 459–526, 2008.

[3] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *Journal of Computational Science*, vol. 2, no. 1, pp. 1–8, 2011.

[4] H. Becker, M. Naaman, and L. Gravano, "Learning similarity metrics for event identification in social media," pp. 291–300, 2010.

[5] I. Guy, U. Avraham, D. Carmel, S. Ur, M. Jacovi, and I. Ronen, "Mining expertise and interests from social media," pp. 515–526, 2013.

[6] Sakaki, Takeshi, Okazaki, Makoto, Matsuo, and Yutaka, "Earthquake shakes twitter users: real-time event detection by social sensors," 2010.

[7] M. Efron, "Hashtag retrieval in a microblogging environment," in *International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2010, pp. 787–788.

[8] A. Bandyopadhyay, K. Ghosh, P. Majumder, and M. Mitra, "Query expansion for microblog retrieval," vol. 1, no. 4, pp. 368–380, 2012.

[9] X. Wang, F. Wei, X. Liu, M. Zhou, and M. Zhang, "Topic sentiment analysis in twitter: a graph-based hashtag sentiment classification approach," in *ACM International Conference on Information and Knowledge Management*, 2011, pp. 1031–1040.

[10] T. Ohkura, Y. Kiyota, and H. Nakagawa, "Browsing system for weblog articles based on automated folksonomy," 2006.

[11] P. Heymann, D. Ramage, and H. Garcia-Molina, "Social tag prediction," in *International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2008, pp. 531–538.

[12] M. K. Su, T. A. Hoang, E. P. Lim, and F. Zhu, "On recommending hashtags in twitter networks," in *International Conference on Social Informatics*, 2012, pp. 337–350.

[13] R. Krestel, P. Fankhauser, and W. Nejdl, "Latent dirichlet allocation for tag recommendation," in *ACM Conference on Recommender Systems, Recsys 2009, New York, Ny, Usa, October*, 2009, pp. 61–68.

[14] Z. Ding, X. Qiu, Q. Zhang, and X. Huang, "Learning topical translation model for microblog hashtag suggestion." in *International Joint Conference on Artificial Intelligence*, 2013, pp. 2078–2084.

[15] Y. Gong and Q. Zhang, "Hashtag recommendation using attention-based convolutional neural network," in *International Joint Conference on Artificial Intelligence*, 2016, pp. 2782–2788.

[16] Q. Zhang, J. Wang, H. Huang, X. Huang, and Y. Gong, "Hashtag recommendation for multimodal microblog using co-attention network," in *Twenty-Sixth International Joint Conference on Artificial Intelligence*, 2017, pp. 3420–3426.

[17] H. Huang, Q. Zhang, X. Huang, H. Huang, Q. Zhang, and X. Huang, "Mention recommendation for twitter with end-to-end memory network," in *Twenty-Sixth International Joint Conference on Artificial Intelligence*, 2017, pp. 1872–1878.

[18] H. Huang, Q. Zhang, J. Wu, and X. Huang, "Predicting which topics you will join in the future on social media," in *International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2017, pp. 733–742.

[19] S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus, "End-to-end memory networks," *Computer Science*, 2015.

[20] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola, "Stacked attention networks for image question answering," pp. 21–29, 2015.

[21] L. Ma, Z. Lu, and H. Li, "Learning to answer questions from image using convolutional neural network," *Computer Science*, 2015.

[22] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Computer Vision and Pattern Recognition*, 2015, pp. 3156–3164.

[23] K. Chen, T. Chen, G. Zheng, O. Jin, E. Yao, and Y. Yu, "Collaborative personalized tweet recommendation," in *International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2012, pp. 661–670.

[24] I. Guy, N. Zwerdling, I. Ronen, D. Carmel, and E. Uziel, "Social media recommendation based on people and tags," in *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2010, pp. 194–201.

[25] I. Ronen, I. Guy, E. Kravi, and M. Barnea, "Recommending social media content to community owners," in *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. ACM, 2014, pp. 243–252.

[26] S. Lo and C. Lin, "Wmr–a graph-based algorithm for friend recommendation," in *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*. IEEE Computer Society, 2006, pp. 121–128.

[27] H. Ma, I. King, and M. R. Lyu, "Learning to recommend with social trust ensemble," in *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2009, pp. 203–210.

[28] J. D. Mcauliffe and D. M. Blei, "Supervised topic models," in *Advances in neural information processing systems*, 2008, pp. 121–128.

[29] W. Zhang, J. Wang, and W. Feng, "Combining latent factor model with location features for event-based group recommendation," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2013, pp. 910–918.

[30] J. Bu, S. Tan, C. Chen, C. Wang, H. Wu, L. Zhang, and X. He, "Music recommendation by unified hypergraph: combining social media information and music content," in *Proceedings of the 18th ACM international conference on Multimedia*. ACM, 2010, pp. 391–400.

[31] M. Kaminskas and F. Ricci, "Contextual music information retrieval and recommendation: State of the art and challenges," *Computer Science Review*, vol. 6, no. 2-3, pp. 89–119, 2012.

[32] M. Schedl and D. Schnitzer, "Hybrid retrieval approaches to geospatial music recommendation," in *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2013, pp. 793–796.

[33] Q. Li, J. Wang, Y. P. Chen, and Z. Lin, "User comments for news recommendation in forum-based social media," *Information Sciences*, vol. 180, no. 24, pp. 4929–4939, 2010.

[34] E. Shmueli, A. Kagian, Y. Koren, and R. Lempel, "Care to comment?: recommendations for commenting on news stories," in *Proceedings of the 21st international conference on World Wide Web*. ACM, 2012, pp. 429–438.

[35] H. Liang, Y. Xu, D. Tjondronegoro, and P. Christen, "Time-aware topic recommendation based on micro-blogs," in *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM, 2012, pp. 1657–1661.

[36] J. Chen, R. Nairn, L. Nelson, M. Bernstein, and E. Chi, "Short and tweet: experiments on recommending content from information streams," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2010, pp. 1185–1194.

[37] Y. Gong, Q. Zhang, X. Sun, and X. Huang, "Who will you "@"?" in *ACM International on Conference on Information and Knowledge Management*, 2015, pp. 533–542.

[38] B. Wang, C. Wang, J. Bu, C. Chen, W. V. Zhang, D. Cai, and X. He, "Whom to mention: expand the diffusion of tweets by@ recommendation on micro-blogging systems," in *Proceedings of the*

22nd international conference on World Wide Web, 2013, pp. 1331–1340.

[39] E. Khabiri, J. Caverlee, and K. Y. Kamath, "Predicting semantic annotations on the real-time web," in *Proceedings of the 23rd ACM conference on Hypertext and social media*. ACM, 2012, pp. 219–228.

[40] A. Rae, B. Sigurbjörnsson, and R. van Zwol, "Improving tag recommendation using social networks," in *Adaptivity, Personalization and Fusion of Heterogeneous Information*. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE, 2010, pp. 92–99.

[41] E. Zangerle, W. Gassler, and G. Specht, "Recommending #-tags in twitter," in *Proceedings of the Workshop on Semantic Adaptive Social Web (SASWeb 2011). CEUR Workshop Proceedings*, vol. 730, 2011, pp. 67–78.

[42] S. M. Kywe, T.-A. Hoang, E.-P. Lim, and F. Zhu, "On recommending hashtags in twitter networks," in *International Conference on Social Informatics*. Springer, 2012, pp. 337–350.

[43] F. Godin, V. Slavkovikj, W. De Neve, B. Schrauwen, and R. Van de Walle, "Using topic models for twitter hashtag recommendation," in *Proceedings of the 22nd International Conference on World Wide Web*. ACM, 2013, pp. 593–596.

[44] J. She and L. Chen, "Tomoha: Topic model-based hashtag recommendation on twitter," in *Proceedings of the 23rd International Conference on World Wide Web*. ACM, 2014, pp. 371–372.

[45] B. Shi, G. Ifrim, and N. Hurley, "Learning-to-rank for real-time high-precision hashtag recommendation for streaming news," in *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2016, pp. 1191–1202.

[46] A. Tariq, A. Karim, F. Gomez, and H. Foroosh, "Exploiting topical perceptions over multi-lingual text for hashtag suggestion on twitter." in *FLAIRS Conference*, 2013.

[47] Q. Zhang, Y. Gong, X. Sun, and X. Huang, "Time-aware personalized hashtag recommendation on social media," in *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, 2014, pp. 203–212.

[48] S. Sedhai and A. Sun, "Hashtag recommendation for hyperlinked tweets," in *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. ACM, 2014, pp. 831–834.

[49] H. Huang, Q. Zhang, Y. Gong, and X. Huang, "Hashtag recommendation using end-to-end memory networks with hierarchical attention," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2016, pp. 943–952.

[50] J. Li, X. Li, B. Yang, and X. Sun, "Segmentation-based image copy-move forgery detection scheme," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 3, pp. 507–518, 2015.

[51] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3128–3137.

[52] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2625–2634.

[53] H. Gao, J. Mao, J. Zhou, Z. Huang, L. Wang, and W. Xu, "Are you talking to a machine? dataset and methods for multilingual image question," in *Advances in neural information processing systems*, 2015, pp. 2296–2304.

[54] M. Ren, R. Kiros, and R. Zemel, "Exploring models and data for image question answering," in *Advances in neural information processing systems*, 2015, pp. 2953–2961.

[55] K. Chen, J. Wang, L.-C. Chen, H. Gao, W. Xu, and R. Nevatia, "Abc-cnn: An attention based convolutional neural network for visual question answering," *arXiv preprint arXiv:1511.05960*, 2015.

[56] H. Nam, J.-W. Ha, and J. Kim, "Dual attention networks for multimodal reasoning and matching," *arXiv preprint arXiv:1611.00471*, 2016.

[57] Y. Zhu, O. Groth, M. Bernstein, and L. Fei-Fei, "Visual7w: Grounded question answering in images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4995–5004.

[58] H. Li, J. Xu *et al.*, "Semantic matching in search," *Foundations and Trends® in Information Retrieval*, vol. 7, no. 5, pp. 343–469, 2014.

[59] P.-S. Huang, X. He, J. Gao, L. Deng, A. Acero, and L. Heck, "Learning deep structured semantic models for web search using clickthrough data," in *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*. ACM, 2013, pp. 2333–2338.

[60] Z. Lu and H. Li, "A deep architecture for matching short texts," in *Advances in Neural Information Processing Systems*, 2013, pp. 1367–1375.

[61] M. Wang, Z. Lu, H. Li, and Q. Liu, "Syntax-based deep matching of short texts," *arXiv preprint arXiv:1503.02427*, 2015.

[62] A. Severyn and A. Moschitti, "Learning to rank short text pairs with convolutional deep neural networks," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2015, pp. 373–382.

[63] B. Hu, Z. Lu, H. Li, and Q. Chen, "Convolutional neural network architectures for matching natural language sentences," in *Advances in neural information processing systems*, 2014, pp. 2042–2050.

[64] H. Palangi, L. Deng, Y. Shen, J. Gao, X. He, J. Chen, X. Song, and R. Ward, "Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 24, no. 4, pp. 694–707, 2016.

[65] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[66] J. Chorowski, D. Bahdanau, K. Cho, and Y. Bengio, "End-to-end continuous speech recognition using attention-based recurrent nn: First results," *arXiv preprint arXiv:1412.1602*, 2014.

[67] S. Sharma, R. Kiros, and R. Salakhutdinov, "Action recognition using visual attention," *arXiv preprint arXiv:1511.04119*, 2015.

[68] V. Mnih, N. Heess, A. Graves *et al.*, "Recurrent models of visual attention," in *Advances in neural information processing systems*, 2014, pp. 2204–2212.

[69] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International Conference on Machine Learning*, 2015, pp. 2048–2057.

[70] J. Weston, S. Chopra, and A. Bordes, "Memory networks," *CoRR*, vol. abs/1410.3916, 2014. [Online]. Available: http://arxiv.org/abs/1410.3916

[71] F. Hill, A. Bordes, S. Chopra, and J. Weston, "The goldilocks principle: Reading children's books with explicit memory representations," *arXiv preprint arXiv:1511.02301*, 2015.

[72] A. Kumar, O. Irsoy, P. Ondruska, M. Iyyer, J. Bradbury, I. Gulrajani, V. Zhong, R. Paulus, and R. Socher, "Ask me anything: Dynamic memory networks for natural language processing," in *International Conference on Machine Learning*, 2016, pp. 1378–1387.

[73] A. Miller, A. Fisch, J. Dodge, A.-H. Karimi, A. Bordes, and J. Weston, "Key-value memory networks for directly reading documents," *arXiv preprint arXiv:1606.03126*, 2016.

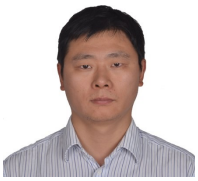
[74] J. Dodge, A. Gane, X. Zhang, A. Bordes, S. Chopra, A. Miller, A. Szlam, and J. Weston, "Evaluating prerequisite qualities for learning end-to-end dialog systems," *arXiv preprint arXiv:1511.06931*, 2015.

[75] A. Bordes, Y.-L. Boureau, and J. Weston, "Learning end-to-end goal-oriented dialog," *arXiv preprint arXiv:1605.07683*, 2016.

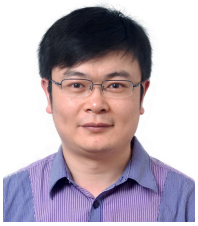
[76] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *Computer Science*, 2014.

[77] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *Computer Science*, 2014.

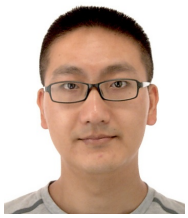
Renfeng Ma received his bachelor degree in computer science from East of China Normal University. He is a master student at Fudan University. His research interests include natural language processing and information retrieval.



Xipeng Qiu received the PhD degree in computer science from Fudan University. He is an associate professor of computer science at Fudan University, Shanghai, China. His research interests include natural language processing and deep learning.



Qi Zhang received the PhD degree in computer science from Fudan University. He is an associate professor of computer science at Fudan University, Shanghai, China. His research interests include natural language processing and information retrieval.



Xiangkun Hu received his bachelor degree in information security from Harbin Institute of Technology. He is a master student at Fudan University. His research interests include natural language processing and information retrieval.

Yu-Gang Jiang received the PhD degree in computer science from City University of Hong Kong. He is a professor of computer science at Fudan University, Shanghai, China. His research interests include multimedia content analysis and computer vision.

Xuanjing Huang received the PhD degree in computer science from Fudan University. She is a professor of computer science at Fudan University, Shanghai, China. Her research interests include natural language processing and information retrieval.