Social Context Summarization

Zi Yang*, Keke Cai†, Jie Tang*, Li Zhang†, Zhong Su† and Juanzi Li*

*Department of Computer Science and Technology, Tsinghua University, China
† IBM, China Research Lab
{yangzi, tangjie, ljz}@keg.cs.tsinghua.edu.cn, {caikeke, lizhang, suzhong}@cn.ibm.com

ABSTRACT

We study a novel problem of *social context summarization* for Web documents. Traditional summarization research has focused on extracting informative sentences from standard documents. With the rapid growth of online social networks, abundant user generated content (e.g., comments) associated with the standard documents is available. Which parts in a document are social users really caring about? How can we generate summaries for standard documents by considering both the informativeness of sentences and interests of social users? This paper explores such an approach by modeling Web documents and social contexts into a unified framework. We propose a dual wing factor graph (DWFG) model, which utilizes the mutual reinforcement between Web documents and their associated social contexts to generate summaries. An efficient algorithm is designed to learn the proposed factor graph model. Experimental results on a Twitter data set validate the effectiveness of the proposed model. By leveraging the social context information, our approach obtains significant improvement (averagely +5.0%-17.3%) over several alternative methods (CRF, SVM, LR, PR, and DocLead) on the performance of summarization.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing; H.2.8 [**Database Management**]: Data Mining; J.4 [**Computer Applications**]: Social and Behavioral Sciences

General Terms

Algorithms, Experimentation

Keywords

Document summarization, Social context, Factor graph, Twitter

1. INTRODUCTION

Web document summarization has been widely studied for many years. Existing methods mainly use statistical or linguistic information such as term distribution, sentence position, and topics to extract the most informative sentences from standard (Web) documents. However, these methods only consider the document's content information, but ignore how users (readers) think about the document. With the rapid growth of online social networks, users can freely express what they are thinking about any Web document. For example, many news websites allow the users to directly add comments to each news document. On Twitter[1], many users post the URL address of a news document onto their microblogs, followed by some personal comments. The comments imply the importance of different sentences and can be used to help improve the quality of document summarization. More importantly, the users' comments essentially reflect which part of the document that they are interested in.

In this work, we present a novel problem of *social context summarization*. The question we intend to answer is: how to generate a summary for Web documents by considering both the informativeness of sentences and interests of social users? The concept of "context" for summarization has been previously studied and various approaches have been proposed based on different kinds of context, such as hyperlinks [1, 9], click-through data [29], comments [13, 19], or opinionated text [26, 11, 15]. Most of these methods directly integrate the context information into the target Web page to help estimate the informativeness of sentences. However, in this way, the context information is only considered as textual information. Many important information has been ignored. For example, if a user is an opinion leader, his comments should be more important than others. From a comment's perspective, if a comment has been forwarded or replied by many other users, the comment should be more important than others. One goal of this work is to consider the social influence and the information propagation for document summarization. The problem is referred to as social context summarization. The problem is clearly different from existing research and poses a set of unique challenges:

- First, social context is becoming more and more complicated. There are users, user generated contents, social networks, and implicit networks (such as the forward/reply network). How to formally define the social context is a nontrivial problem.

- Second, in the social environment, the quality of summaries strongly depends on the social context information (such as the social influence between users). How to formalize the problem in a principled framework is a challenging problem.

- Third, the social context contains inevitable noise. How to qualitatively analyze the problem and quantitatively validate

[1]http://www.twitter.com, a microblogging system.

SIGIR'11, July 24–28, 2011, Beijing, China.

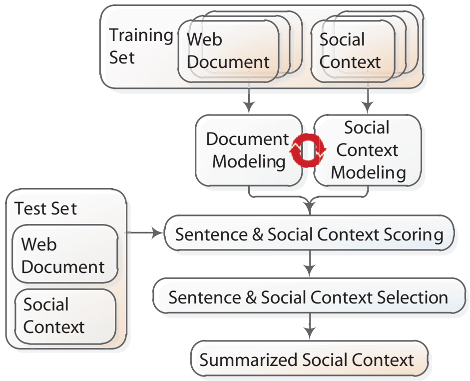

Figure 1: Overview of the proposed approach

the proposed approach on a real-world data set is also a challenging problem.

In this paper, we try to systematically investigate the problem of social context summarization for Web documents. We formulate the problem of social context summarization and propose a dual-wing factor graph (DWFG) model. The DWFG model incorporates the summarization task and the social network analysis into a unified framework. In this way, the two tasks can be mutually reinforced. We employ the Microblogging as an example to quantitatively study the social context summarization problem. In particular, we crawl a data set from Twitter. The user generated content is the tweet posted by the user. The retweeting (forwarding) and replying relationships between tweets form an implicit information network, and the following relationships between users form the user network. Some tweets have the links pointing to standard Web documents (such as news documents). The problem then becomes how to leverage both the information network and the user network to generate high-quality summaries for standard Web documents.

The overview of the proposed method is a supervised framework (as shown in Figure 1). In training, we estimate the importance of defined features and strength of dependencies for identifying key sentences and microblogs in the social contexts. In test, given a new Web document with its social context, we perform collective inference for the importance of each sentence and microblog and select a subset of sentences as the summary according to the trained models. We validate the proposed approach on the Twitter data set. The experiment results show that by leveraging the social context information, our approach can significantly improve (on average +10.8%) the performance of summarization. We also compare with a set of alternative methods (i.e., CRF, SVM, LR, PR, and DocLead), and our approach clearly outperforms (averagely +5.0%-17.3%) the baseline methods.

The paper is organized as follows: in Section 2, we introduce some notations and formally define the problem. In Section 3, we propose the factor graph model to address the problem, and in Section 4, we conduct the experiments on the Twitter dataset. Finally, in Section 5 and 6, we summarize related works and conclude.

2. PROBLEM DEFINITION

In this section, we introduce some notations and representations necessary for the problem, and then define the problem of social context summarization.

Definition 1 (Social context). Given a Web document d, its social context C_d is defined as $\langle M_d, U_d \rangle$, where M_d is a set of comments on d written by users U_d in a social network.

In the context of Web 2.0, Web documents, e.g. news or blogs, are freely discussed and commented by users. These comments again spread (e.g., by forwarding between friends) in the social network. The users' activities implicitly reflect the importance of different parts (e.g., sentences) in the document from the user's perspective. We believe that the social context, Thus, integrating the document content information and the social context information can disclose a more thorough view of the document. In this paper, we employ Twitter as the basis for our study. Specifically, given a Web document d[2] and its associated social context (tweets M_d containing the URL address of document d and users U_d who posted those tweets), we give the following definition of social context augmented network.

Definition 2 (Social Context Augmented Network, SCAN). Social Context Augmented Network $G_d = (S_d, C_d, E_d)$ is defined as a network that is built upon the sentence set S_d of document d and its social context C_d, where the edge set E_d contains three types of edges: E_d^s, E_d^m, and E_d^u. $E_d^s = \{(s_i, s_j)|s_i, s_j \in S_d\}$ represents the relationships between document sentences, $E_d^m = \{(m_i, m_j)|m_i, m_j \in M_d\}$ represents the relationships between messages, and $E_d^u = \{(u_i, u_j)|u_i, u_j \in U_d\}$ represents the relationships between users.

Compared with traditional contexts that are defined based on textual information, social context need model various dynamic social relationships, such as the *follower-followee relationships* between users, *retweeting relationships* and *replying relationships* between tweets. An example of SCAN is shown in Figure 2(a). In this figure, the upper layer includes two documents d_1 and d_2, and d_1 contains four sentences s_1, s_2, s_3, and s_4. The two documents are respectively associated with two sets of messages $M_1 = \{m_5, m_6, m_7, m_8\}$ and $M_2 = \{m_9, m_{10}\}$ in the middle layer. The lower layer refers to the user layer consisting of users u_1, u_2 and u_3, who are also associated with the messages M_1 and M_2. Besides external relationships between the objects across different layers, SCAN also describes internal relationships between objects within the same layer (as shown in Figure 2(a)).

Given this, we can formally define our problem of *Social Context Summarization*.

Definition 3 (Social Context Summarization). Given a social context augmented network G_d, the goal of social context summarization is to generate a summary which consists of two pieces of information: the most important sentences $S_d^* \subseteq S_d$ and the most representative messages $M_d^* \subseteq M_d$.

The problem of social context summarization contains two subproblems: *Key Sentence Extraction* and *Tweet Summarization*. In the former problem, we aim to identify the most important sentences from document d's content, while in the latter subproblem, we intend to find the most representative tweets from the social context C_d of document d. Social context C_d contains rich information about the document d, which is helpful for the Key Sentence Extraction problem, while the important sentences in a document can equally help Tweet Summarization in the social context. The mutual reinforcement between the two subproblems can facilitate generating a high-quality summary. Moreover, social context summarization could also answer a number of related questions, e.g., who are the most experienced users of a specific topic or a fact mentioned in a document.

[2]On Twitter, a Web document (e.g., news document) is often pointed out by a URL address, which might be in some forms of encoded shortened URLs such as by tinyurl.com and bit.ly.

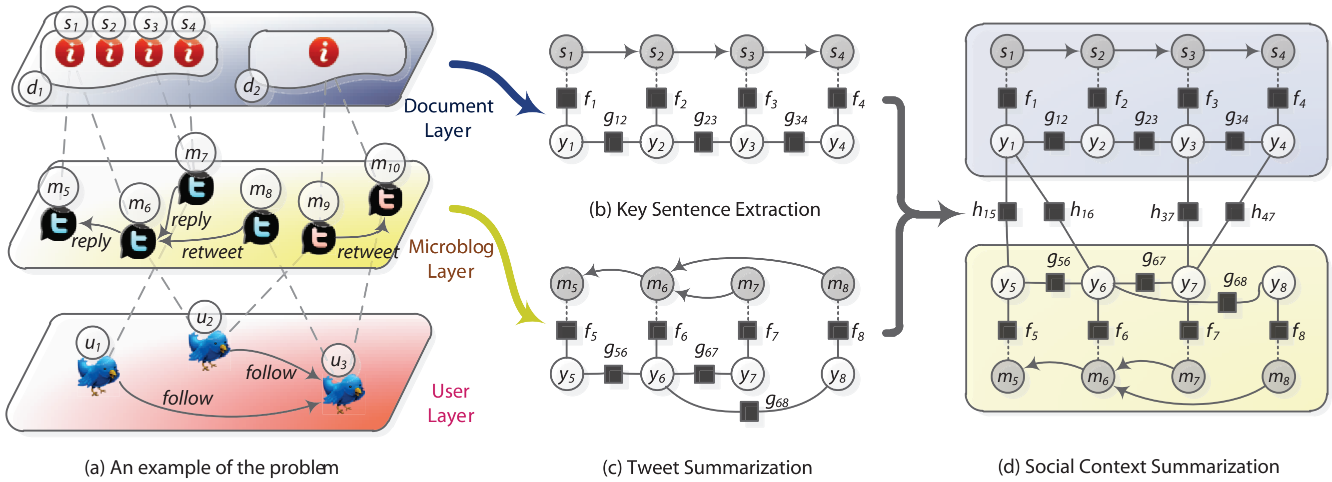

Figure 2: An example of the problem and factor graph representations for summarization tasks. In (b), (c), and (d), each gray circle with s_i indicates a sentence in the Web document; its associated white circle with y_i denotes whether the sentence should be included in the document summary. Each gray circle with m_i indicates a tweet and its associated white circle denotes whether the tweet would be included in the tweet summary.

3. PROBLEM SOLVING

In this section, we propose a dual wing factor graph (DWFG) model, which formulates the social context summarization problem in a unified learning framework. The DWFG model simultaneously incorporates all resources in social context to generate high-quality summaries for Web documents.

3.1 Basic Idea

In our Twitter data set, each Web document is associated with a social context. To generate summaries for Web documents, a straightforward method is to define a set of features to characterize the importance of each sentence, and then use a classification model to identify which sentences should be included into the summary [16, 25, 36]. To further consider the correlation between sentences, we can consider a sequential labeling approach such as conditional random field. Such a method has been studied by [8, 28]. Both of them consider the sentence local features and similarities (correlations) between sentences, and model the sentence extraction task with a linear-chain conditional random field. An example of the graphical representation is shown in Figure 2(b). The method only considers the correlation between sentences (the document layer in Figure 2(a)), but ignores the social context information resided in the microblog and user layers.

To model the tweet network, we design another similar graphical model with structures reflecting the information propagation. Figure 2(c) presents an example. Each gray circle indicates a tweet, the arrow represents the replying/retweeting relationship between two tweets. Based on such a formulation, we can define local features (content-based features) for each tweet, as well as edge features for each replying/retweeting relationship. By learning such a graphical model, we can classify which tweets are important (or informative). Obviously, this model only considers the information from the tweet side and does not consider the Web documents. An ideal way is to incorporate the two tasks together so that they can reinforce each other.

Based on these considerations, we propose a novel dual wing factor graph (DWFG) model. The graphical representation is shown in Figure 2(d). In the DWFG model, the upper layer is used to model the key sentence extraction task and the bottom layer is designed to model the tweet summarization problem. In the middle layer, we design a set of correlation factor functions h to bridge the two tasks. By carefully designing the correlation factor function h, we can elegantly combine the two tasks of key sentence extraction and tweet summarization into a unified framework. In the rest of this section, we will explain in details how we design and learn the dual wing factor graph model.

3.2 Modeling Summarization via Dual Wing Factor Graphs

We model the social context summarization problem in the dual wing factor graph (DWFG) model. Each sentence $s_i \in S_d$ or tweet $m_i \in M_d$ is associated with a binary value y_i indicating the importance of the sentence or tweet (1 representing important, and 0 representing unimportant).

We first collect a set of labeled SCANs (training set) $T = \{G_d\}_{d=1}^{n_T}$, i.e., each sentence $s_i \in S_d$ and tweet $m_j \in M_d$ in each social context C_d are associated with known binary labels y_i and y_j, moreover, we also collect the test set S of unlabeled instances, which consists of all the sentences and tweets not yet judged. Our goal is then to learn a DWFG model from the training set and apply it to predict which sentences and tweets are important in the test set S, i.e., to infer the value (label) of y, and then generate a summary for the social context.

We define three types of factor functions associated with individual instances or instance groups: *local attribute factor*, *intra-domain dependency factor*, and *inter-domain dependency factor*.

Local attribute factor. The probability that a sentence or tweet is important could be estimated by some local attributes (represented as $\mathbf{x}$), which refer to features that are inherent to the sentence or tweet itself. In general, we define similar features for sentences and tweets. The features include the average TF-IDF score over words and the log likelihood generated by the context, the position of the sentence in the document, the author's authoritativeness. Details of the defined local features for sentences and tweets are given in Section 4.

To estimate the significance of each feature, we introduce a weight variable λ_c for each feature c, and we define a local attribute factor $f_{i,c}$ for the feature c of each sentence s_i or tweet m_i. Formally, a factor could be defined as the local entropy:

$$f_{i,c}(\lambda_c, y_i) = \exp(\lambda_c x_{i,c} y_i) \quad (1)$$

where $x_{i,c}$ is the value of the c-th feature extracted from sentence s_i or tweet m_i.

Intra-domain dependency factor. As described in Section 3.1, we introduce factors that are capable of handling multiple instances in either sentence level or tweet level, to characterize the dependencies among sentences and tweets respectively. Intra-domain interaction may promote some sentences to become more important while inhibit others from becoming important. We associate each type of interaction with a weight μ_c indicating the confidence of the corresponding interaction. The interaction has a positive influence only if the weight μ_c is greater than 0. We introduce factor $g_{ij,c}$ to capture the dependency among sentence pair s_i and s_j or tweet pair m_i and m_j.

$$g_{ij,c}(\mu_c, y_i, y_j) = \begin{cases} \exp \mu_c & \text{if some condition holds} \\ 1 & \text{otherwise} \end{cases} \quad (2)$$

A document can be regarded as a sequence of sentences, and thus key sentence extraction could be viewed as a sequence labeling process [28], i.e., the judgment on a certain sentence is affected by the nearby sentences to avoid both sentences of high similarity are chosen simultaneously. Hence, the dependency conditions in Eq. 2 for a sentence pair s_i and s_j can be formalized as follow: the factor takes value $\exp \mu_c$ if $y_i \neq 1$ or $y_j \neq 1$. To avoid high computational complexity, we only constrain consecutive and similar sentences, i.e., establish sentence relation for sentence s_i and s_{i+1} whose mutual similarity (e.g., cosine similarity) exceeds the threshold θ_g.

Moreover, we consider the two interactions among tweets: replying and retweeting. If tweet m_i replies or retweets tweet m_j, then m_j successfully excites and attracts attentions from others, and it is reasonable that m_j is more important than its succeeding tweets in the thread. Formally, for such a tweet pair m_i and m_j, the factor takes value $\exp \mu_c$ if $y_i \leq y_j$.

Inter-domain dependency factor. By leveraging knowledge from both domains, the inter-domain relationships may benefit to the identification of social context summarization. We introduce a set of factors defined on variables across domains, which are able to coordinate the labels of sentences and tweets simultaneously. Specifically, if tweet m_j is considered as a representative tweet, i.e., $y_j = 1$, then a sentence s_i highly similar to m_j (with similarity more than a threshold θ_h) should be biased towards the same label, i.e., $y_i = 1$. Formally, for each sentence-tweet pair (s_i, m_j) of high similarity, we define

$$h_{ij}(\nu, y_i, y_j) = \begin{cases} \exp \nu & \text{if } y_i = y_j \\ 1 & \text{otherwise} \end{cases} \quad (3)$$

where ν is the weight variable that represents the significance of inter-domain dependency factor.

Objective function. Finally, the objective function can be defined as the normalized product of Eqs. 1 - 3 for all the instances. We denote Z as the normalization factor, which sums up the conditional likelihood $P(Y|X,\Theta)$ over all the possible labels of all the instances, where Y contains all the undetermined labels for sentences and tweets, i.e., $Y = \{y_i\}_i$, and Θ is the collection of weights, i.e., $\Theta = \{\lambda_c\}_c \cup \{\mu_c\}_c \cup \{\nu\}$.

We first estimate the parameters Θ with a maximum likelihood procedure on the training instances, e.g.,

$$\max_{\Theta} \frac{1}{Z} \prod_{i,j \in T} \prod_{c \in C} f_{i,c}(\lambda_c, y_i) \cdot g_{ij,c}(\mu_c, y_i, y_j) \cdot h_{ij}(\nu, y_i, y_j) \quad (4)$$

We use L-BFGS, a quasi-Newton method for solving the nonlinear optimization problem (i.e., Eq. 4). To avoid overfitting, we add a penalty term $-\frac{1}{2}||\Theta||^2/\sigma^2$, a spherical Gaussian prior, into the objective function, which is a regularization method commonly used in maximum entropy and conditional random fields [6, 27, 28].

Calculating the marginal distribution for each factor (in deriving the log-gradient of the objective function) requires a loopy sum-product inference algorithm. With the learned parameter Θ, we may summarize an unlabeled social context for a Web document in the test set by extracting important sentences, which are also identified by a similar max-sum inference procedure. The inference algorithm is introduced in the next section.

Connection with existing models. We note that the proposed DWFG model can also be viewed as a model generalized from existing models. In Eq. 4, if parameter ν is fixed as 0, i.e., all factors $\{h_{ij}\}_{i,j}$ take constant values of 1, and factors $\{f_{i,c}\}_{i,c}$ and $\{g_{ij,c}\}_{i,j,c}$ are only defined for sentences, then the simplified model only incorporates sentence local factors and sentence relation factors, and DWFG model is degenerated to a special case: the summarization approach based on linear-chain CRF [28]. Moreover, if all parameters $\{\mu_c\}_c$ are also set as 0, i.e., only the local factors $\{_{i,c}\}_{i,c}$ are non-trivial, then DWFG is turned into the logistic regression classifier [25].

3.3 Inference Algorithm

Since the graphical model DWFG proposed for summarization (cf. Figure 2(d)) contains cycles, we cannot directly employ a forward-backward algorithm like in [28] for exactly inferring the optimal labeling for a test instance. We then propose an approximate inference approach based on the loopy sum-product or max-sum algorithm.

To achieve an approximate inference for predicting labels, the algorithm contains multiple iterations for updating the beliefs, and each iteration is comprised of two phases. Here, we denote the update variables for delivering beliefs between variables and factors by p_{ij} and q_{ij}. $\{p_{ij}\}_{i,j}$ represent the messages propagating from variable (e.g., y_i) to factor (e.g., $g_{ij,c}$ or $h_{ij,c}$), and $\{q_{ij}\}_{i,j}$ represents the messages factor to variable respectively. The messages can be formulated as follows:

$$p_{ij} = r_i + \sum_{k \in N(i) \setminus \{j\}} q_{ik} \quad (5)$$

$$\begin{aligned} q_{ij} = & \max\{t_{ij}(1,1) + p_{ji}, t_{ij}(1,0)\} \\ & - \max\{t_{ij}(0,1) + p_{ji}, t_{ij}(0,0)\} \end{aligned} \quad (6)$$

where r_i corresponds to the logarithmic value of the local factor, i.e., $r_i = \sum_{c \in C} (\log f_{i,c}(\lambda_c, y_i = 1) - \log f_{i,c}(\lambda_c, y_i = 0))$. Analogously, $t_{ij}(y_i, y_j)$ is the logarithmic value of the dependency factor, i.e., $t_{ij}(y_i, y_j) = \log g_{ij,c}(\mu_c, y_i, y_j)$ or $\log h_{ij}(\nu, y_i, y_j)$. Specific to a particular dependency factors, $f_{i,j}$, $g_{ij,c}$, or h_{ij} (Eq. 1 to 3), the message q_{ij} has a more succinct expression, e.g., the sentence dependency factor $q_{ij} = \max\{p_{ji} - \mu_c, 0\} - \max\{p_{ji}, 0\}$.

We can obtain the label for each sentence s_i and tweet m_i using the variables calculated in the two phases for the last iteration as follows:

$$y_i = \begin{cases} 1 & \text{if } p_{ij} + q_{ij} > 0 \text{ for some } j \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

Algorithm 1: Social context summarization with DWFG

input : A document d with social context C_d and SCAN G_d of d, weight variables Θ, and number of iterations I

output: A summary for social context C_d: important sentences S_d^* and messages M_d^*

`// initialization`

1 initialize variables $\{r_i\}_i \leftarrow 0$;

`// update message values`

2 **for** $i \leftarrow 1$ *to* I **do**

3 update variables p_{ij} according to Eq. 5;

4 update variables q_{ij} according to Eq. 6;

`// output result`

5 **foreach** $s_i \in S_d$ *and* $m_i \in M_d$ **do**

6 calculate y_i according to Eq. 7;

7 $S_d^* \leftarrow \{s_i \in S_d | y_i = 1\}$;

8 $M_d^* \leftarrow \{m_i \in M_d | y_i = 1\}$;

The learning algorithm is depicted in Algorithm 1. Initially, we calculate all local variables $\{r_i\}_i$, and initialize all update variables $\{q_{ij}\}_{i,j}$ as 0 (Line 1). Next, we compute new values for all the update variables $\{p_{ij}\}_{i,j}$ according to Eq. 5. Then we estimate the new values for all $\{q_{ij}\}_{i,j}$ according to Eq. 6. We continue to update the variables for a number of iterations until some termination condition is satisfied. Finally, the summary of the social context is generated according the update variables (Line 5 - 10).

Complexity analysis. If we denote the number of iterations for the inference algorithm as I, then the computational complexity of the algorithm is proportional to $I \cdot (|E^s| + |E^m| + |E^c|)$, where $|E^s|, |E^m|, |E^c|$ correspond to the number of sentence relationships, tweet relationships, and inter-domain relationships respectively. They can be varied from zero to many when we tune the thresholds θ_g and θ_h, which is further discussed in Section 3.2. In fact, the inference algorithm can be easily parallelized or distributed onto clusters to handle large-scale dataset, and the design of distributed algorithm will be reported elsewhere.

4. EXPERIMENT

In this section, we evaluate the proposed summarization method DWFG with manually labeled documents. We firstly introduce the data set, baseline methods that do not incorporate the relationship between the Web document domain and tweet thread domain, the evaluation metrics, and then we give the detailed discussion of the experiment results with the comparison of other approaches. More supplied materials of this work can be found at `http://arnetminer.org/socialcontext/`.

4.1 Settings and Observations

4.1.1 Data Preparation

Since there is rarely previous work study the summarization task from social perspective, to the best of our knowledge no existing benchmark dataset can be utilized for our experiments. We collected data from the most popular microblogging website, Twitter.

From 4,874,389 Twitter users, we collected 404,544,462 tweets within a period from January 1st 2010 to July 17th 2010, and then recognized all the tweets accompanied with explicit URLs (containing "http://" or "https://"). Since users might use different URL shortening services, such as tinyurl.com, bit.ly, etc., we simply implemented a general shorten URL decoder based on a HTTP client to obtain the decoded URLs. Finally, the top 200,000 high frequent URLs were finally extracted and 12,964,166 tweets talking about the same URL were grouped together. In our preliminary experiments, we observed the distribution of frequencies of URLs carried by the tweets, which is plotted in Figure 3 in log-logarithmic scale.

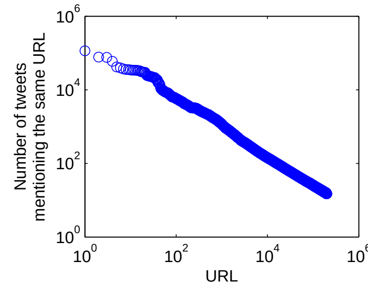

Figure 3: The distribution of URLs carried by the tweets

We see that the highest frequent URL is mentioned by 114,911 tweets in our experiment data, and the distribution of frequencies of URLs follows the power law. According to the selected URLs, we crawled the associated Web pages, and then constructed two kinds of data sets (Web pages and their corresponding social tweets). The Web documents were then segmented into a set of sentences with the jTokeniser Toolkit[3]. Our summarization algorithm was then performed on both domains.

We found that most of the Top 50 URLs correspond to advertisement pages. We therefore predefined a series of high-quality websites, such as CNN, BBC, Mashable etc., and selected a subset of URLs related to these websites for manual annotation. We note that a Web document might be referred by different URLs even if the URLs are decoded, e.g., URLs "http://news.bbc.co.uk/1/hi/england/8604663.stm", "http://news.bbc.co.uk/2/hi/uk_news/england/8604663.stm", and "http://news.bbc.co.uk/2/hi/8604663.stm" correspond to the same Web document. We further group such Web documents according to the unique document ID indicated in the URL (e.g., 8604663). Details on the five selected domains are given in Table 1.

4.1.2 Evaluation Methods

To guarantee the low noise of the manual annotation data, we further manually validated the informativeness of all the selected Web documents by posting both the Web documents and tweets on Amazon Mechanical Turk[4].

We totally issued 1145 HITs on Mechanical Turk, and for each HIT we asked at least two different workers to read both the Web documents and its corresponding tweets. All the HITs were divided into 12 batches with each assignment entitled "Key sentences and tweets extraction from news and related tweets". We gave a detailed description on how to label the sentences and tweets, and also emphazied that the workers should "extract serveral sentences from news that attract them mostly", and "after reading the news, extract the most interesting tweets that appeal you mostly". We required the workers to label no less than 5 tweets and 10 Web document sentences. Finally, 158 different users have participated in annotating the benchmark for social context summarization task. The labeled sets of sentences and tweets formed the benchmark for evaluation.[5]

[3]http://code.google.com/p/jtokeniser/

[4]http://mturk.com, an Internet marketplace to use human intelligence to solve various kinds of problems

[5]We plan to gradually publish the annotated dataset for academic use of social context summarization.

Table 1: Description on employed domains

Domain	Description	Data Size	
		Doc	Tweet
cnn.com	one of the most popular news websites	1,303	62,225
bbc.co.uk	the most popular news website in the UK	336	10,264
mtv.com	one of the most popular music television networks	176	9,848
espn.go.com	one of the world's leading sports media	171	4,320
mashable.com	the world's largest tech blog	2,940	114,441

In this paper, two performance metrics applied in [29] were adopted to evaluate the proposed approach DWFG. The first is Precision, Recall and F-measure. In the following section, we will report the evaluation on F_1 measure, which is defined as:

$$P = \frac{S_{\text{ref}} \cap S_{\text{cand}}}{S_{\text{cand}}}; R = \frac{S_{\text{ref}} \cap S_{\text{cand}}}{S_{\text{ref}}}; F_1 = \frac{2PR}{P+R}$$

where S_{cand} and S_{ref} denote the sentences contained in the candidate summary and the reference summary respectively.

Another performance metric is ROUGE [18], which measures summarization quality according to the overlap between the units, such as n-gram (referred to as ROUGE-N) etc, of machine generated summary and human generated summary. ROUGE-N is defined as follows:

$$\text{ROUGE-N} = \frac{\sum_{s\in S_{\text{ref}}}\sum_{gram_n\in s}\text{Count}_{\text{match}}(gram_n)}{\sum_{s\in S_{\text{ref}}}\sum_{gram_n\in s}\text{Count}(gram_n)}$$

where n is the length of the n-gram, $\text{Count}_{\text{match}}(gram_n)$ is the maximum number of n-grams co-occurring in a candidate summary and the reference summaries, $\text{Count}(gram_n)$ is the number of n-grams in the reference summaries.

We employ the ROUGE evaluation methods implemented in the Dragon Toolkit Project[6], and report the experimental results in terms of ROUGE-1 and ROUGE-2 with stop words filtered out. Since ROUGE is a recall-oriented metric, we keep the number of sentences extracted be equal with that of the human summary for fair comparison. Specifically, we select the sentences and tweets with the greatest positive beliefs by $p(y_i = 1|\mathbf{X})$ (cf. Eq. 7).

4.1.3 Feature Description

Many features have been designed for document summarization in prior literatures. In this paper, we only extract 11 basic and straight-forward features from both domains. Besides of some features that are widely used in traditional summarization methods, we also utilize several features extracted from users' online social behaviors, e.g., the number of users following the tweet's author and the PageRank score of the author. Table 2 gives the brief definition of these features applied in this paper, where some features were represented by nominal values, e.g., Feature 1 will take value 4 if the sentence was extracted from the title of the document, 3 if it was extracted from the subtitle, 2 if the sentence was located in the first paragraph of the original document, 1 if the sentence was located in the last paragraph, and 0 otherwise.

The feature values extracted from sentence domain and tweet domain are summarized in Figure 4 and 5. Since different features take values in diverse ranges, e.g., the maximum value of Feature 4 is 15, while the maximum value of Feature 6 is 1.495, we normalize the feature values by the mean value of corresponding feature.

[6]http://dragon.ischool.drexel.edu/

Table 2: Feature list

	N°	description
Web documents	1	sentence position in document
	2	sentence position in paragraph
	3	average TF-IDF score of words in sentence
	4	the number of common words to the title
	5	sentence length
	6	the log likelihood generated by the document
Tweets	7	average TF-IDF score of words in post
	8	tweet length
	9	the log likelihood generated by the tweet thread
	10	the number of users following the author
	11	the PageRank score of the post's author

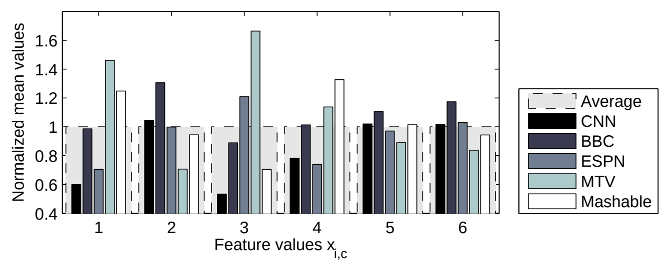

Figure 4: Comparison of feature values for sentence domain on five domains

From Figure 4, we can see that Web documents from different domains exhibit differently. For example, articles in CNN, BBC, and ESPN have smaller values of Feature 1 but greater values of Feature 2 than MTV, which indicates that news Websites CNN, BBC, and ESPN have longer articles consisting of a greater number of shorter paragraphs. Therefore, we trained an individual model on each domain respectively to capture the distinctiveness.

4.1.4 Baseline Methods

We compare DWFG with six supervised baselines methods. SVM classifiers (SVM) and logistic regression classifiers (LR) are performed for each sentence and tweet only with its local features. Linear-chain and tree-structured CRF models (LC-/TS-CRF) are respectively trained and tested on documents and tweet threads, i.e., inter-domain relationships are considered as a supplement to the basic local features. The linear-chain CRF baseline model employed in the sentence summarizaiton is equivalent to the method proposed in [28].

We also extend the feature list for each sentence and tweet by considering the features of related sentences or tweets extracted from both domains (denoted as SVM+, LR+). Specifically, for each sentence s_i in a document, we append 11 features $(x^s_{i,7}, \ldots, x^s_{i,17})$, where each of $x^s_{i,7}, \ldots, x^s_{i,12}$ adds up the corresponding feature values of its similar sentences, and each of $x^s_{i,13}, \ldots, x^s_{i,17}$ adds up the corresponding feature values of its related tweets. Similarly, for each tweet in the thread, we append 11 features, which are the sums of feature values of its relevant sentences or tweets.

In addition, we also compare DWFG with commonly applied unsupervised summarization algorithms, i.e., the importance sentences and tweets are selected according to a metric or score. First, we randomly select sentences or tweets (Random) as the basic unsupervised method. Another baseline method for summarization is to select the sentences according to their positions in the document or paragraph (DocLead and ParaLead). Finally, we apply PageRank algorithm for summarization on the whole graph consisting of three types of relationships (PR) [24].

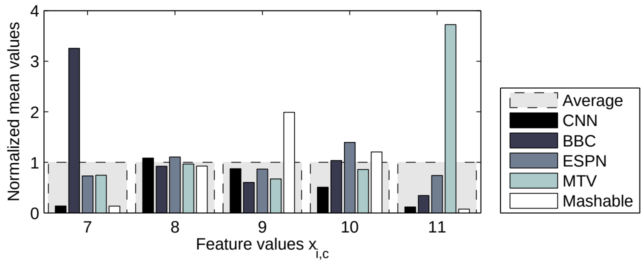

Figure 5: Comparison of feature values for tweet domain on five domains

Table 3: Experimental results for Web documents

		CNN	BBC	MTV	ESPN	Mash	All
F_1	SVM	0.288	0.322	0.490	0.337	0.321	0.351
	LR	0.284	0.340	0.531	0.352	0.297	0.361
	LC-CRF	0.307	0.349	0.596	0.364	0.340	0.391
	SVM+	0.283	0.341	0.476	0.359	0.324	0.357
	LR+	0.277	0.332	0.482	0.366	0.305	0.352
	Random	0.314	0.321	0.455	0.351	0.305	0.349
	DocLead	0.334	0.356	0.441	0.317	**0.415**	0.373
	ParaLead	0.298	0.316	0.508	0.338	0.323	0.356
	PR	**0.354**	0.338	0.453	0.351	0.399	0.379
	DWFG	0.341	**0.450**	**0.642**	**0.518**	0.330	**0.456**
R-1	SVM	0.224	**0.612**	0.392	0.520	0.511	0.452
	LR	0.197	0.599	0.585	0.583	0.599	0.513
	LC-CRF	0.281	0.551	0.667	0.583	**0.618**	0.540
	SVM+	0.176	0.563	0.400	0.635	0.546	0.464
	LR+	0.171	0.610	0.362	0.620	0.605	0.473
	Random	0.429	0.426	0.455	0.470	0.405	0.437
	DocLead	0.410	0.473	0.542	0.372	0.576	0.475
	ParaLead	0.414	0.337	0.629	0.432	0.414	0.445
	PR	**0.433**	0.325	0.563	0.426	0.482	0.446
	DWFG	0.389	0.594	**0.777**	**0.701**	0.613	**0.615**
R-2	SVM	0.151	**0.500**	0.336	0.412	0.412	0.362
	LR	0.131	0.491	0.522	0.481	0.496	0.424
	LC-CRF	0.197	0.496	0.542	0.515	0.516	0.453
	SVM+	0.115	0.463	0.351	0.539	0.449	0.383
	LR+	0.110	0.498	0.310	0.528	0.501	0.390
	Random	0.323	0.325	0.387	0.359	0.301	0.339
	DocLead	0.371	0.424	0.519	0.350	0.525	0.438
	ParaLead	0.363	0.320	0.569	0.370	0.354	0.395
	PR	**0.389**	0.307	0.533	0.387	0.441	0.411
	DWFG	0.228	0.417	**0.687**	**0.612**	**0.557**	**0.500**

4.2 Results and Analysis

4.2.1 Comparison Results

The experiments were conducted in the 10-fold cross validation procedure, where one fold is for test and the other nine folds for training. The performance results are shown in Table 3 and 4, and the best performances in the comparisons are highlighted in bold. In the following results, we set the similarity threshold for sentence dependency $\theta_g = 0.1$, and the similarity threshold for inter-domain dependency $\theta_h = 0.8$. We will further discuss the variation of performance with different assignment of thresholds in Section 4.2.2.

From Table 3, we can see that DWFG outperforms the baseline methods in most cases in terms of both F_1 and ROUGE-N for document summarization. Moreover, we discover that the performances are statistically significantly improved on the MTV and ESPN domains by conducting sign test on the results, where the p values are much smaller than 0.01. In fact, we collect relatively fewer documents and corresponding tweets from MTV and ESPN compared with other domains, and thus, additional dependencies, especially

Table 4: Experimental results for tweet thread

		CNN	BBC	MTV	ESPN	Mash	All
F_1	SVM	0.323	0.542	0.640	0.610	0.379	0.499
	LR	0.370	0.531	0.606	0.616	0.408	0.506
	LC-CRF	0.378	0.547	0.637	0.603	**0.417**	0.516
	SVM+	0.378	0.537	0.641	0.607	0.405	0.514
	LR+	0.369	0.537	**0.725**	0.608	0.408	**0.529**
	Random	0.356	0.486	0.665	0.586	0.353	0.489
	PR	0.281	0.428	0.666	0.520	0.327	0.445
	DWFG	**0.380**	**0.547**	0.639	**0.633**	0.380	0.516
R-1	SVM	0.531	0.631	0.670	0.701	0.617	0.630
	LR	0.657	0.618	0.702	0.708	0.737	0.684
	LC-CRF	**0.673**	0.647	0.730	0.703	**0.748**	0.700
	SVM+	0.661	0.659	0.672	0.694	0.740	0.685
	LR+	0.655	**0.660**	0.692	0.756	0.737	0.700
	Random	0.631	0.617	**0.740**	0.704	0.622	0.663
	PR	0.167	0.382	0.522	0.439	0.229	0.348
	DWFG	0.669	0.647	0.731	**0.763**	0.700	**0.702**
R-2	SVM	0.486	0.571	0.661	0.678	0.570	0.593
	LR	0.616	0.556	0.696	0.684	0.698	0.650
	LC-CRF	0.610	0.563	0.708	0.699	0.682	0.653
	SVM+	**0.620**	0.598	0.663	0.671	**0.700**	0.651
	LR+	0.613	**0.601**	0.684	**0.731**	0.698	**0.666**
	Random	0.572	0.533	**0.724**	0.671	0.557	0.611
	PR	0.157	0.356	0.519	0.432	0.218	0.337
	DWFG	0.599	0.563	0.709	0.722	0.631	0.645

cross-domain dependencies boost the performance by leveraging additional information.

In contrast to the improvements in Web document summarization, DWFG performs comparably to the simpler CRF-based methods for tweet summarization. In fact, the ground truth data are manually annotated from the perspective of readers' interests and foci, which naturally reveals the users' motivations for writing tweets. Therefore, the identification of important sentences from the Web document domain rarely influences the results for identifying important tweets.

4.2.2 Impact of Thresholds θ_g and θ_h

In this section, we discuss the impact of thresholds θ_g and θ_h to our proposed approach. Although the proposed approach within a supervised framework can automatically learn the optimal model parameters Θ based on the training instances, we still need to predefine the thresholds θ_g and θ_h to control the number of inter-domain and intra-domain dependencies in the factor graph model. Specifically, with larger θ_g or θ_h, we obtain fewer dependencies, and if $\theta_g = 0$, each pair of consecutive sentences will be connected by a inter-domain factor, or if $\theta_h = 0$, all the sentences will be connected with all the tweets. To evaluate the impact of thresholds to DWFG and baseline methods (e.g., LC-CRF), we varied θ_g or θ_h from 0 to 1 with step length 0.1 respectively with the other threshold fixed. Due to space limitation, we only report the impact to the performance of DWFG in Figure 6(a) and (b) in terms of F_1, ROUGE-1, and ROUGE-2, and the performances of the baseline methods follow similar trends with different thresholds. We also plot the percentage of consecutive sentence pairs with similarity more than θ_g in Figure 6(a), and the percentage of sentence-tweet relation pairs with similarity more than θ_h in Figure 6(b).

From Figure 6(a), we can see that when θ_g increases from 0.0 to 0.5, the performance drops by $5\% \sim 16\%$ in terms of F_1 and ROUGE, which can be attributed to the lack of a complete view of sentence relations within the document. While with θ_g is 0.7, the performance reaches a local maximum when the retained sentence relations have a relatively higher quality. As shown in Figure 6(b), the performance of sentence identification reaches the global max-

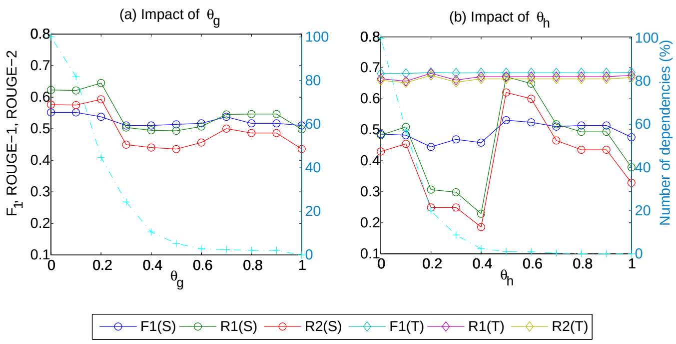

Figure 6: The impact of θ_g and θ_h to the performance of DWFG

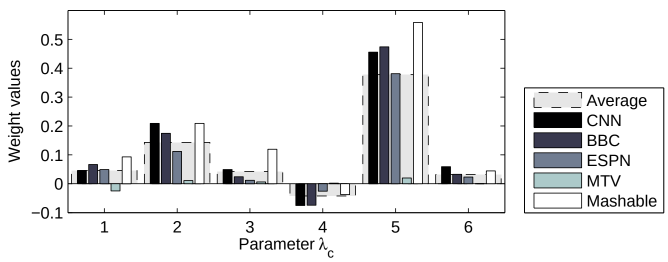

Figure 7: Parameter estimation results for sentence-level local factors on five domains

imum when θ_h is set between 0.5 and 0.6. With smaller or greater θ_h, the extracted relation pairs between sentences and tweets may contain more low-quality relations or lack of high-quality relations. Generally speaking, the performance of important tweet extraction is relatively stable.

4.2.3 *Factor Contribution Analysis*

We further analyze the contribution or significance of each factor. We show the estimated weights for sentence-level local factors $\lambda_1, \ldots, \lambda_6$ on five domains respectively and calculate their averages in Figure 7, and show the estimated weights with their averages for tweet-level local factors $\lambda_7, \ldots, \lambda_1 1$ in Figure 8.

From Figure 7, we see that most of the local factors have positive contributions to our task except for Feature 4 (the number of common words to the title). Among all the factors, we can see that Feature 5 (sentence length) and Feature 2 (sentence position in paragraph) on average are the most important local factors for identifying the important sentences. From Figure 8, we see the two most important local factors for identifying the representative tweets are Feature 8 (tweet length) and Feature 7 (average TF-IDF). In fact, we find that long tweets tend to cover both the main ideas of the Web documents and the personal comments towards them.

4.2.4 *Case Study*

In this section, we demonstrate an example of the inference step for a specific Web document, an article entitled “Women try to take body on plane at Liverpool airport”[7], with its social context. In Figure 9, the left column lists a portion of sentences of the Web document, and the right column lists a portion of tweets containing URLs (or shortened URLs) directing to the article (the selected texts are indicated by bold font). The established inter-domain and intra-domain dependencies are shown in arrows. Furthermore, beliefs propagated from local factors and pair-wise dependency factors in the last iteration of our inference algorithm are partly shown with the associated variables taking values of 1 (in colored rounded rectangles). Beliefs taking values of 0.5 indicate that the corresponding factors have no preference on whether the sentences are regarded as part of the summary or not. Beliefs taking values greater than 0.5 convey positive attitudes, and the greater the belief values, the stronger the confidence that the associated variables should take values of 1. According to the calculated beliefs, the summary for the social context is generated based on the selected sentences and tweets (in bold).

[7]http://news.bbc.co.uk/2/hi/8604663.stm

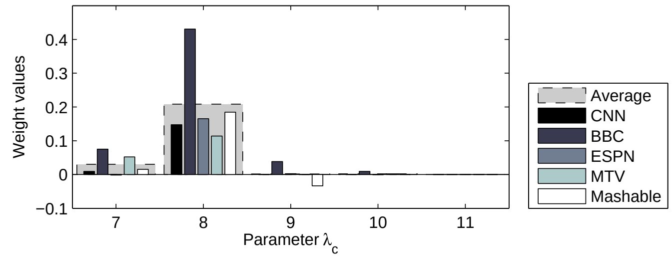

Figure 8: Parameter estimation results for tweet-level local factors on five domains

We can see that the local features, e.g., statistical features, still play a major role for social context summarization. For example, since the most common words or phrases in the Web documents include “women”, “dead person”, “body”, “Liverpool Airport”, and those in tweet threads include “Liverpool airport”, “Weekend At Bernie’s”, texts that cover these words or phrases are more likely chosen, and the probability that the relevant sentence-tweet pairs are simultaneously selected is boosted. Moreover, various types of relations also come into play. For example, since the last two tweets shown in the right column form a retweet pair, the importance of the content is evaluated more important, and thus the related sentence (the fourth sentence) in the document then receives a higher belief (0.51) of taking a positive decision. As we suggested, in the social context summarization task, the tweet thread contributes additional information (e.g., Weekend At Bernie’s[8]) to the original document content, which unveils the users’ interests from an alternative angle.

5. RELATED WORK

Web-page summarization techniques have been widely studied for many years and various approaches have been developed. These approaches can be either supervised or unsupervised, and also can be generic or query-dependent. Since this paper mainly studies context summarization, we focus our literature review for approaches with or without consideration of context.

Two kinds of approaches have been designed for web-page summarization, supervised and unsupervised. Traditional supervised summarization approaches treat the summarization task as a two-class classification problem [16, 25, 36] or a sequence labeling problem [8, 28] at the sentence level, where each sentence is represented by a vector of features. Comparably, unsupervised methods rely on a set of heuristic rules to develop the summarization. Web-page summarization can also be either generic or query-dependent. Generic summarization targets to cover the main idea of the page while query-oriented summary is to present the information that is most relevant to the given queries [4, 31].

Without consideration of context, the extracted summary is com-

[8]a 1989 American motion picture comedy, which has a similar plot as the news story.

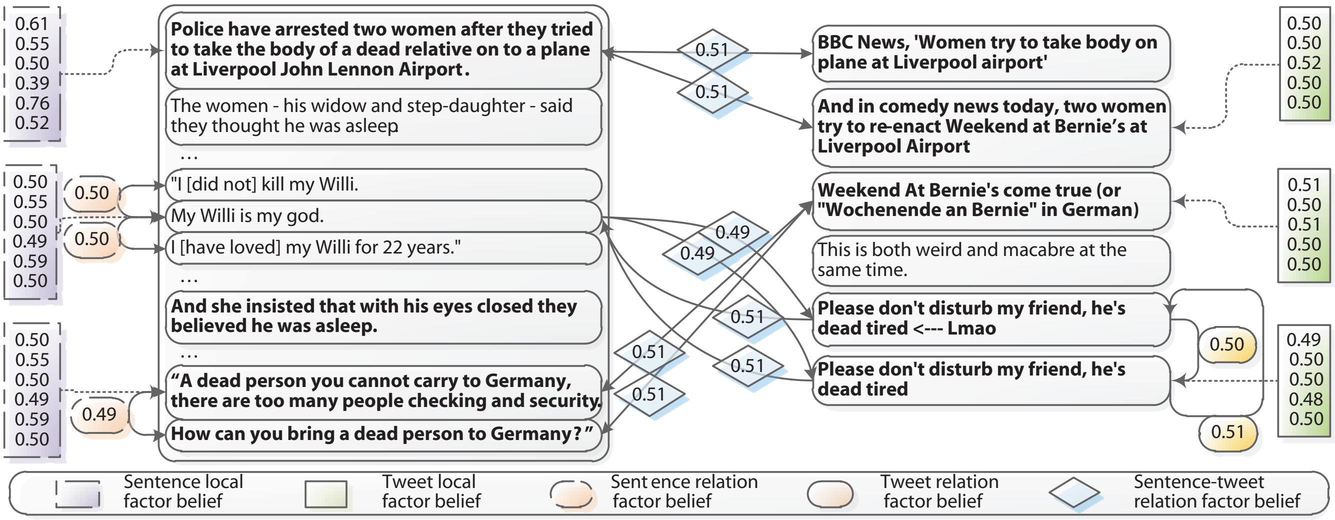

Figure 9: An example of a social context summary with propagated beliefs. Left column is a part of the Web document, and the right column is a portion of tweet thread. Bold texts correspond to the summary for the Web document; colored rounded rectangles indicate the beliefs propagated in the inference algorithm.

posed of sentences from the Web documents, and thus features from local content of a document is the key to summarization. Traditional document-oriented features can be defined either from linguistic, such as rhetorical structure [22], lexical chains [2] or statistical perspectives, such as term significance [20], sentences similarity [24] and topic detection [12]. Although document-oriented features can disclose most of the basic characteristics of summary sentences, as stated in [29], the textual information of a Web document may be scarce and diverse in topics and, moreover, contain a lot of noise.

Document-oriented features cannot fully capture the main idea of a Web document. In the past few years, some work starts to utilize various kinds of context to assist document summarization, such as external documents or cited articles [23]. User requirement is one of the most important kind of context [10, 32]. In the study of [21], user's needs come from a set of documents selected by user, where the top content words were extracted according to their G^2 score and then treated as users' interests. Hyperlinks among Web pages are another kind of context. Based on the text surrounding the hyperlink, summarization of the target Web page can be realized either by extracting the related sentences in surrounding text [1] or by extracting significant sentences from the linked Web page [9]. Similar to the hyperlink ontext, Sun et al. [29] utilize search-engine clickthrough data to construct the extra knowledge. In their work, Web page and query terms collected from the clickthrough data work together to decide the significance of each word in sentences for summarization. With the rapid growth of social websites, comments-oriented approach was studied, where the most important comments are selected and leveraged into sentence selection for summarization. Traditional feature-based methods and graph-based methods for summary sentence extraction have been applied for commented sentence selection [13, 33, 19], or opinionated text [26, 11, 15].

Different from previous works, we study to leverage multifaceted social media information for Web document summarization, especially social influence among users [30] and retweeting relations among messages [35]. However, we adopt a totally different approach to not only incorporate the extra knowledge extracted from microblogs, but also take full advantage of conventional techniques in single document summarization. In recent years, the rapid growth of microblogging services provide a more efficient way for information communication. Here, people can freely issue various comments on any topic they interested in. Compared with traditional tightly-coupled relationship between Web document and comments, messages from microblogs can provide more valuable information beneficial for summarization. Microblog has been widely studied in recent years. Some work focuses on investigating the characteristics of Twitter, e.g.,[17], [7], [14], while some work analyzed the patterns of retweets on Twitter, influential twitter and the routines of changes of hashtags, etc., e.g., [34], [5], [3], [35]. To the best of our knowledge, little work in the literature has tried to use microblog data for Web-page summarization.

6. CONCLUSION AND FUTURE WORK

In this paper, we explore a novel problem of social context summarization and aim to utilize the mutual reinforcement between Web document and its associated social data to building a high-quality summary. In our study, the importance of each document sentence is firstly predicted by considering a series of local features of a document. At the same time, the social context relating to the Web document is associated with it, in which the significant sentences are also identified by taking advantage of various social factors. We formally define the concept of social context for Web document and propose a unified summarization approach through factor graph model. Our experiments are implemented on a set of Web documents and associated microblog messages. The experiment results prove that the proposed summarization method shows significant improvement over the baseline approaches on social context summarization task.

To systematically combine the content analysis and social behaviors represents a new and interesting direction for information retrieval. There are many future directions of this work. For example, due to the fact that not only tweets are highly associated with other tweets, users are also connected by the friendship relations, we can extend this work by establishing the connection among users and adding the dependencies between users and their tweets. Intuitively, the influence among users will also affect the identification of important tweets, and subsequently influence the importance of sentences in Web documents.

7. *ACKNOWLEDGMENTS

The work is supported by the Natural Science Foundation of China (No. 61073073, No. 60703059, No. 60973102), Chinese National Key Foundation Research (No. 60933013, No.61035004), National High-tech R&D Program (No. 2009AA01Z138).

8. REFERENCES

[1] E. Amitay. Automatically summarising web sites - is there a way around it? In *CIKM'00*, pages 173–179, 2000.

[2] R. Barzilay and M. Elhadad. Using lexical chains for text summarization. In *ACL Workshop on Intelligent Scalable Text Summarization*, pages 10–17, 1997.

[3] d. boyd, S. Golder, and G. Lotan. Tweet, tweet, retweet: Conversational aspects of retweeting on twitter. In *HICSS'10*, pages 1–10, 2010.

[4] J. Carbonell and J. Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *SIGIR'98*, pages 335–336, 1998.

[5] M. Cha, H. Haddadi, F. Benevenuto, and K. P. Gummadi. Measuring user influence in twitter: The million follower fallacy. In *ICWSM'10*, pages 10–17, 2010.

[6] S. F. Chen and R. Rosenfeld. A gaussian prior for smoothing maximum entropy models. Technical Report CMU-CS-99-108, Carnegie Mellon University, 1999.

[7] M. Cheong and V. Lee. Integrating web-based intelligence retrieval and decision-making from the twitter trends knowledge base. In *SWSM'09*, pages 1–8, 2009.

[8] J. M. Conroy and D. P. O'Leary. Text summarization via hidden markov models. In *SIGIR'01*, pages 406–407, 2001.

[9] J.-Y. Delort, B. Bouchon-Meunier, and M. Rifqi. Enhanced web document summarization using hyperlinks. In *Hypertext'03*, pages 208–215, 2003.

[10] A. Díaz and P. Gervás. User-model based personalized summarization. *Information Processing & Management*, 43(6):1715–1734, 2007.

[11] K. Ganesan, C. Zhai, and J. Han. Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In *COLING'10*, pages 340–348, 2010.

[12] Y. Gong and X. Liu. Generic text summarization using relevance measure and latent semantic analysis. In *SIGIR'01*, pages 19–25, 2001.

[13] M. Hu, A. Sun, and E.-P. Lim. Comments-oriented document summarization: understanding documents with readers' feedback. In *SIGIR'08*, pages 291–298, 2008.

[14] A. Java, X. Song, T. Finin, and B. Tseng. Why we twitter: understanding microblogging usage and communities. In *WebKDD/SNA-KDD'07*, pages 56–65, 2007.

[15] H. D. Kim and C. Zhai. Generating comparative summaries of contradictory opinions in text. In *CIKM'09*, pages 385–394, 2009.

[16] J. Kupiec, J. Pedersen, and F. Chen. A trainable document summarizer. In *SIGIR'95*, pages 68–73, 1995.

[17] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? In *WWW'10*, pages 591–600, 2010.

[18] C.-Y. Lin and E. Hovy. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *NAACL'03*, pages 71–78, 2003.

[19] Y. Lu, C. Zhai, and N. Sundaresan. Rated aspect summarization of short comments. In *WWW'09*, pages 131–140, 2009.

[20] H. P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159–165, 1958.

[21] I. Mani and E. Bloedorn. Machine learning of generic and user-focused summarization. In *AAAI'98/IAAI'98*, pages 820–826, 1998.

[22] D. Marcu. From discourse structures to text summaries. In *ACL Workshop on Intelligent Scalable Text Summarization*, pages 82–88, 1997.

[23] Q. Mei and C. Zhai. Generating impact-cased summaries for scientific literature. In *ACL'08*, pages 816–824, 2008.

[24] R. Mihalcea. Language independent extractive summarization. In *ACL'05*, pages 49–52, 2005.

[25] M. Osborne. Using maximum entropy for sentence extraction. In *ACL Workshop on Automatic Summarization*, pages 1–8, 2002.

[26] M. J. Paul, C. Zhai, and R. Girju. Summarizing contrastive viewpoints in opinionated text. In *EMNLP'10*, pages 66–76, 2010.

[27] F. Sha and F. Pereira. Shallow parsing with conditional random fields. In *NAACL'03*, pages 134–141, 2003.

[28] D. Shen, J. tao Sun, H. Li, Q. Yang, and Z. Chen. Document summarization using conditional random fields. In *IJCAI'07*, pages 2862–2867, 2007.

[29] J.-T. Sun, D. Shen, H.-J. Zeng, Q. Yang, Y. Lu, and Z. Chen. Web-page summarization using clickthrough data. In *SIGIR'05*, pages 194–201, 2005.

[30] J. Tang, J. Sun, C. Wang, and Z. Yang. Social influence analysis in large-scale networks. In *SIGKDD'09*, pages 807–816, 2009.

[31] J. Tang, L. Yao, and D. Chen. Multi-topic based query-oriented summarization. In *SDM'09*, pages 1147–1158, 2009.

[32] C. Teng, N. Xiong, Y. He, L. T. Yang, and D. Liu. A behavioural mode research on user-focus summarization. *Mathematical and Computer Modelling*, 51(7-8):985–994, 2010.

[33] X. Wan and J. Yang. Multi-document summarization using cluster-based link analysis. In *SIGIR'08*, pages 299–306, 2008.

[34] J. Weng, E.-P. Lim, J. Jiang, and Q. He. Twitterrank: Finding topic-sensitive influential twitterers. In *WSDM'10*, pages 261–270, 2010.

[35] Z. Yang, J. Guo, K. Cai, J. Tang, J. Li, L. Zhang, and Z. Su. Understanding retweeting behaviors in social networks. In *CIKM'10*, pages 1633–1636, 2010.

[36] J.-Y. Yeh, H.-R. Ke, W.-P. Yang, and I.-H. Meng. Text summarization using a trainable summarizer and latent semantic analysis. *Inf. Process. Manage.*, 41(1):75–95, 2005.