Singapore Management University

Institutional Knowledge at Singapore Management University

Research Collection School Of Computing and Information Systems | School of Computing and Information Systems

8-2014

Generating Supplementary Travel Guides from Social Media

Liu YANG
Peking University

Jing JIANG
Singapore Management University, jingjiang@smu.edu.sg

Lifu HUANG
Peking University

Minghui QIU
Singapore Management University, minghui.qiu.2010@smu.edu.sg

Lizi LIAO
Singapore Management University, lzliao@smu.edu.sg

Follow this and additional works at: https://ink.library.smu.edu.sg/sis_research

Part of the Databases and Information Systems Commons, and the Social Media Commons

Citation

YANG, Liu; JIANG, Jing; HUANG, Lifu; QIU, Minghui; and LIAO, Lizi. Generating Supplementary Travel Guides from Social Media. (2014). *Proceedings of COLING 2014: The 25th International Conference on Computational Linguistics: Technical Papers, Dublin, Ireland, August 23-29 2014*. 1670-1681.
Available at: https://ink.library.smu.edu.sg/sis_research/2413

This Conference Proceeding Article is brought to you for free and open access by the School of Computing and Information Systems at Institutional Knowledge at Singapore Management University. It has been accepted for inclusion in Research Collection School Of Computing and Information Systems by an authorized administrator of Institutional Knowledge at Singapore Management University. For more information, please email cherylds@smu.edu.sg.

Generating Supplementary Travel Guides from Social Media

Liu Yang[1,2], Jing Jiang[2,*], Lifu Huang[1,2], Minghui Qiu[2], Lizi Liao[2,3]
[1]Peking University / Beijing, China, 100871
[2]Singapore Management University / Singapore, Singapore, 178902
[3]Beijing Institute of Technology / Beijing, China, 100081
yang.liu@pku.edu.cn, jingjiang@smu.edu.sg
{warrior.fu, minghuiqiu, liaolizi.llz}@gmail.com

Abstract

In this paper we study how to summarize travel-related information in forum threads to generate supplementary travel guides. Such summaries presumably can provide additional and more up-to-date information to tourists. Existing multi-document summarization methods have limitations for this task because (1) they do not generate structured summaries but travel guides usually follow a certain template, and (2) they do not put emphasis on named entities but travel guides often recommend points of interest to travelers. To overcome these limitations, we propose to use a latent variable model to align forum threads with the section structure of well-written travel guides. The model also assigns section labels to named entities in forum threads. We then propose to modify an ILP-based summarization method to generate section-specific summaries. Evaluation on threads from Yahoo! Answers shows that our proposed method is able to generate better summaries compared with a number of baselines based on ROUGE scores and coverage of named entities.

1 Introduction

Online forums and community question answering (CQA) sites contain much useful information from ordinary users, such as their personal experience, opinions, suggestions and recommendations. Extracting and summarizing information from these rich information sources has a wide range of applications. In this work, we study how to tap into user-generated content in forums such as Yahoo! Answers to generate supplementary city travel guides. Travel guides published by well-known publishers such as Lonely Planet are written by a small number of authors based on their travel experience. Presumably if we could summarize the large amount of information given by ordinary users about a city, such a summary could supplement the official travel guide and cover more up-to-date information.

However, social media content is diverse and noisy because it is contributed by many different authors. Directly applying existing multi-document summarization methods to forum and CQA threads may not produce good travel guides for the following reasons: (1) Summaries produced by standard summarization methods are not structured, but travel guides usually follow a template structure. (2) Travel guides put much emphasis on points of interest, which are usually location entities, but standard text summarization methods are not entity-oriented.

To illustrate our points, in Table 1 we show (i) the overall structure of a travel guide for Sydney from Lonely Planet, (ii) an excerpt from a summary generated by a state-of-the-art ILP-based summarization method (Gillick and Favre, 2009) from a set of threads related to Sydney, and (iii) excerpts of a structured summary generated by our proposed method. The comparison shows that the summary generated by the standard ILP method mixes information on different topics together and does not mention many

* Corresponding author.
This work is licenced under a Creative Commons Attribution 4.0 International License. Page numbers and proceedings footer are added by the organizers. License details: http://creativecommons.org/licenses/by/4.0/

Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 1670–1681, Dublin, Ireland, August 23-29 2014.

Travel Guide from Lonely Planet (`http://www.lonelyplanet.com/australia/sydney/`)
Restaurants: **Sepia**:There's nothing washed out or brown-tinged about Sepia's food: Martin Benn's picture-perfect creations are presented in . . . **Icebergs Dining Room**: Poised above the famous Icebergs swimming pool, Icebergs views sweep across the Bondi Beach arc to . . . *Shopping*: **Strand Arcade**: Constructed in 1891, the Strand rivals the QVB in the ornateness stakes. Three floors of designer fashions . . . **Westfield Sydney**: The city's newest shopping mall is a bafflingly large complex gobbling up Sydney Tower and a fair chunk of . . . *Transport*: **Sydney Airport**: Sydneys Kingsford Smith Airport , 10km south of the city centre, is Australias busiest airport, handling flights . . . **Water Taxis Combined**:Fares based on up to four passengers; add $10 per person for additional passengers. Sample fares . . .

Yahoo! Answers Summary Generated by Standard ILP Method	Yahoo! Answers Summary Generated by Our Method
It 's not too far from **Sydney** . **Sydney** is the most expensive place in **Australia** . They are a little lame ... Then you can go to **Darling Harbour**, a beautiful habour which is a 10-minute walk from town hall station . Make sure , if you are up to it to do the bridge climb , this is a real treat . There are lots of interesting things to see and do in and around **Sydney** . The suburbs-much cheaper than the **CBD**. It was in the basement of a big shopping mall . The only way to do that is to drive . Got to walk on top of the **Sydney** harbour bridge and go up centre point tower ! Walk around the street and see the beach . I would like to stay at a nice hotel . My friend and I are wanting to take a trip to **Sydney** for the summer . But you 'll need to get there by taxi . **Sydney** is so pretty, so you should be able to find stuff to do . And they have many facilities . Good luck and have fun . Public transport is not very good . Depending on what you 're in **Sydney** to do it 's hard to say . . .	*Restaurants*: Go to the two major restaurant areas close to the city **Darlinghurst** , along **Oxford Street** , and **Newtown** , along **King Street** . **Chinatown** which is off **George St.** in the city look up **Dixon st.** is a great place to get a cheap **Chinese** meal . . . *Shopping*: **Queen Victoria Building** and **Pitt St Mall** , **World Square** and the **Strand** are good ideas to check out . Hair driers you can get in many places , but the main places would be the department stores such as **Target** , **Big W** , **K-Mart** , **Myer**, **David Jones** . . . *Transport*: The **CBD** is about 15 minutes by train from the airport and there is a station at **Circular Quay** , right on the **Harbour** with access to the bridge and the **Opera House** . You can catch an intercity train with **Cityrail** from just about anywhere in **Sydney** . . .

Table 1: Comparison of different travel guides about Sydney. Top: excerpts from Lonely Planet. Bottom left: excerpt from a summary generated by standard ILP. Bottom right: excerpts from summary generated by our method. Named entities are highlighted in bold font.

interesting places to visit. The summary by our proposed method, in contrast, organizes the information into sections and has a high coverage of places a tourist can visit.

To generate the kind of summaries as shown in the bottom right of Table 1, we propose to first leverage the section structure of well-written travel guides and use a latent variable model to align forum threads with the different sections from these travel guides. Moreover, observing that points of interest are organized by sections in these travel guides, we also identify location names from user-generated content and try to uncover their underlying section labels. We then treat the remaining problem as a multi-document summarization task. We modify an Integer Linear Programming (ILP)-based extractive summarization framework (Gillick and Favre, 2009) to select sentences from forum threads to generate section-specific summaries, where we specifically emphasize the inclusion of potential points of interest for each section. Experiments using threads from Yahoo! Answers show that our proposed method generates better summaries than a number of baselines in terms of ROUGE scores and coverage of named entities.

Our work makes the following contributions. First, we study a new problem of summarizing multiple forum threads to generate city travel guides based on known template structure from well-written travel guides. Second, we propose a principled approach based on latent variable models and Integer Linear Programming. Third, we evaluate our method using real forum threads and human generated model summaries, and the results are positive.

2 Overview of Our Method

Our task is to summarize travel-related information from forum threads for potential tourists. In order to inject some structure into the generated summaries, we assume that we have a set of I well-written travel guides that correspond to I different cities and have the same structure. We refer to these travel guides as official travel guides. Each official travel guide consists of a fixed set of S sections such as *restaurants* and *shopping*, and this section structure will be used to organize our generated summaries. We further assume that each section of an official travel guide consists of a list of points of interest, each with a name and a short description, as illustrated in Figure 1. We believe that this is a fairly common structure followed by many if not all travel guides.

Given a target city, we assume that we can collect a set of threads about this city from travel-related forums. In this paper we use threads from Yahoo! Answers, but our solution does not use any CQA properties of the threads, so threads from other general forums can also be used. Our goal is to generate a text summary with S sections from these threads, where each section has a length limit.

As we have mentioned, we treat the problem as a multi-document summarization task. However, different from standard text summarization, our generated summaries should contain S sections. To achieve this goal, we first select a set of relevant threads for each section and then perform section-specific summarization from the selected threads.

Thread selection: To select relevant threads given a section, a naive solution is to rank the threads based on their relevance to the section, where relevance can be measured by, for example, cosine similarity between a thread and all the text in the given travel guides belonging to the section. But we observe that the language used in forum threads could be very different from that in the official travel guides, making it hard to measure relevance purely based on lexical overlap. For example, in the *entertainment* section, forum threads may contain words such as "djs," "Xmas," "b'day" and "anni.", but these words do not occur in the official travel guides. To overcome this difficulty, we propose to use a latent variable model that jointly models official travel guides and forum threads. We treat the S sections as S latent factors that govern the generation of the forum threads. With the latent factors observed in the official travel guides, we receive some supervision; and yet by jointly modeling both the official travel guides and the forum threads, we allow the latent factors to adapt to the lexical variations in user-generated content. In the end, the learned latent factors can help us align forum threads with the sections and subsequently select the most relevant ones for each section.

Section-specific summarization: Given the selected relevant threads for a section, we adopt an ILP-based extractive summarization framework that has been shown to be effective (Gillick and Favre, 2009). We modify the objective function in this framework to consider two factors: (1) Since not every sentence in the selected threads is highly relevant to the section, we want to give preference to those more relevant sentences in the objective function, where relevance can be measured using word distributions learned by the latent variable model. (2) Since travel guides are expected to recommend points of interest to readers, we try to maximize the coverage of section-specific location entities in the objective function.

3 Joint City Section Model

3.1 Model

In this section we present our Joint City Section Model (JCSM), which links official travel guides and forum threads. The model is a typical extension of LDA, where a number of latent topics (i.e. latent factors) are assumed to have generated the observed text. First of all, for each pre-defined section there is a latent topic. These explain words such as "food" and "menu" for *restaurants* and "store" and "mall" for *shopping*. In addition, in both travel guides and forum threads, some words are more related to the city being discussed than any specific section. For example, when New York City is being discussed, words such as "NYC" and "Manhattan" may frequently show up in any section. We therefore further assume that for each city there is a city-specific topic. A switch variable is used to determine whether a word comes from a city-specific or section-specific topic.

A special design of our model that differs from many existing LDA extensions is the treatment of named entities. We first use a named entity recognizer to identify potential names of locations from forum threads. We assume that each of these entities belongs to a section, which is indicated by a latent variable. We then assume that the section labels of the non-entity words in forum threads are dependent on the section labels of these entities. By doing so, we emphasize the importance of associating potential points of interest with sections, which will be useful when we generate summaries.

We now formally present JCSM. To simplify the model description, we assume that we work with I cities, each of which has a given, well-written travel guide and a set of forum threads. Note that in practice this model can be easily extended such that a target city with forum threads does not need to have a given travel guide to begin with. Let ϕ_i denote the word distribution for the city-specific latent topic associated with city i. Let ψ_s denote the word distribution for the section-specific latent topic for section s. Let $d_{i,s,n}$ denote the n-th word in the s-th section of the i-th city's travel guide. Here $1 \leq d_{i,s,n} \leq V$ is an index into the vocabulary with size V. Let $x_{i,s,n}$ be a switch variable associated with $d_{i,s,n}$ to indicate whether this word is city-specific or section-specific. For the j-th forum thread related to the i-th city, we assume there is a distribution over sections, denoted as $\theta_{i,j}$. For the l-th location entity in the k-th post of this thread, we assume a latent variable $c_{i,j,k,l}$ ($1 \leq c_{i,j,k,l} \leq S$) that indicates the section label of this entity. Then for the m-th word in this post, we first use a switch variable $y_{i,j,k,m}$ to determine whether the word is city-specific or section-specific. If it is section-specific, we then choose one of the entities in the same post, denoted as $z_{i,j,k,m}$, and its corresponding section label as the section for this word.

All the binary switch variables follow a global Bernoulli distribution parameterized by π. There are hyperparameters α, β, β' and γ that define the prior distributions. The complete model is depicted in Figure 1. The generative process of JCSM is also described as follows.

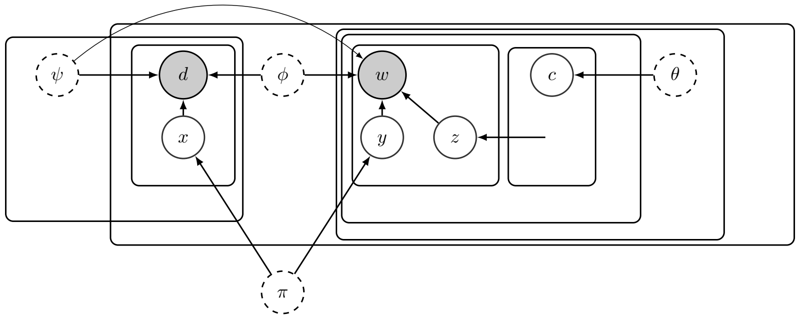


Figure 1: The plate notation of the Joint City Section Model (JCSM). Dashed variables will be integrated out in Gibbs sampling. For clarity, the Dirichlet and Beta priors are omitted. The arrow pointing to z indicates that z is drawn from a uniform distribution over the integers from 1 to L.

- For each city i, ($i = 1, 2, \cdots, I$), draw a city-specific word distribution $\phi_i \sim$ Dir(β')
- For each section s, ($s = 1, 2, \cdots, S$), draw a section-specific word distribution $\psi_s \sim$ Dir(β)
- Draw a switch distribution $\pi \sim$ Beta(γ)
- For each city i ($i = 1, 2, \cdots, I$)
  - For each section s ($s = 1, 2, \cdots, S$)
    - For the n-th word in the given travel guide
      - Draw $x_{i,s,n} \sim$ Bernoulli(π)
      - If $x_{i,s,n} = 1$, draw $d_{i,s,n} \sim$ Multi(ψ_s); otherwise, draw $d_{i,s,n} \sim$ Multi(ϕ_i).
  - For the j-th thread
    - Draw a thread specific section distribution $\theta_j \sim$ Dir(α)
    - For the k-th post
      - For the l-th entity, draw $c_{i,j,k,l} \sim$ Multi(θ_j)
      - For the m-th word, draw $y_{i,j,k,m} \sim$ Bernoulli(π). If $y_{i,j,k,m} = 1$, draw $z_{i,j,k,m} \sim$ Uniform($1, \cdots, L_{i,j,k}$) and then draw $w_{i,j,k,m} \sim$ Multi($\psi_{c_{i,j,k,z_{i,j,k,m}}}$); otherwise, draw $w_{i,j,k,m} \sim$ Multi(ϕ_i).

3.2 Inference

We use collapsed Gibbs sampling to estimate the parameters in the model. The problem is to compute the Gibbs update rules for sampling $x_{i,s,n}$, $c_{i,j,k,l}$, $z_{i,j,k,m}$, $y_{i,j,k,m}$.

Sample entity topic $c_{i,j,k,l}$

Let b denote $\{i, j, k, l\}$ and u denote $\{i, j, k\}$. We can derive the Gibbs update rule for sampling entity topic $c_{i,j,k,l}$ as follows:

$$p(c_b = s|\mathbf{C}_{\neg_b}, \mathbf{W}, \mathbf{D}, \mathbf{X}, \mathbf{Y}, \mathbf{Z}) = \frac{n^s_{i,j,\neg_b} + \alpha}{\sum_{s'=1}^{S} n^{s'}_{i,j,\neg_b} + S\alpha} \cdot \frac{\prod_{w=1}^{V} \prod_{i'=1}^{n^w_{u,y=1,z=l}} (n^w_{y=1,z=l,\neg_u} + \beta + i' - 1)}{\prod_{j'=1}^{n^{\mathbf{w}}_{y=1,z=l,u}} (\sum_{w=1}^{V} n^w_{y=1,z=l,\neg_u} + V\beta + j' - 1)},$$

where $n^s_{i,j,\neg_b}$ denotes the number of entities whose topic assignments are s in thread $\{i, j\}$ without consideration of entity $\{i, j, k, l\}$. $n^w_{u,y=1,z=l}$ denotes the number of times term w occurs in the post $\{i, j, k\}$ with the constraint that $y = 1$ and $z = l$. $n^w_{y=1,z=l,\neg_u}$ is the number of times term w occurs in all posts except the post $\{i, j, k\}$ with the constraint that $y = 1$ and $z = l$.

Sample switch label $x_{i,s,n}$

We can derive the Gibbs update rule for sampling $x_{i,s,n}$ in a similar way. Note that the sampling of $x_{i,s,n}$ is in travel guide word level. Let g denote$\{i, s, n\}$, the Gibbs update rule for sampling $x_{i,s,n}$ is as follows:

$$p(x_g = 0|\mathbf{C}, \mathbf{W}, \mathbf{D}_{\neg g}, \mathbf{X}_{\neg g}, \mathbf{Y}, \mathbf{Z}) = \frac{n^{x=0}_{\neg g} + \gamma}{\sum_{x=0}^{1} n^{x}_{\neg g} + 2\gamma} \cdot \frac{n^{w_g}_{x=0,i,\neg g} + \beta'}{\sum_{w=1}^{V} n^{w}_{x=0,i,\neg g} + V\beta'}$$

$$p(x_g = 1|\mathbf{C}, \mathbf{W}, \mathbf{D}_{\neg g}, \mathbf{X}_{\neg g}, \mathbf{Y}, \mathbf{Z}) = \frac{n^{x=1}_{\neg g} + \gamma}{\sum_{x=0}^{1} n^{x}_{\neg g} + 2\gamma} \cdot \frac{n^{w_g}_{x=1,s,\neg g} + \beta}{\sum_{w=1}^{V} n^{w}_{x=1,s,\neg g} + V\beta}$$

Sample post word topic $z_{i,j,k,m}$ and switch label $y_{i,j,k,m}$

For words in the thread posts, We can derive the Gibbs update rule for sampling post word topic $z_{i,j,k,m}$ and switch label $y_{i,j,k,m}$. Note that the sampling of $z_{i,j,k,m}$ and $y_{i,j,k,m}$ is in post word level. Let f denote$\{i, j, k, m\}$. The Gibbs update rule for sampling $z_{i,j,k,m}$ and $y_{i,j,k,m}$ is as follows:

$$p(z_f = s|\mathbf{C}, \mathbf{W}_{\neg f}, \mathbf{D}, \mathbf{X}, \mathbf{Y}_{\neg f}, \mathbf{Z}_{\neg f}) = \frac{n^{w_f}_{y=1,s',\neg f} + \beta}{\sum_{w=1}^{V} n^{w}_{y=1,s',\neg f} + V\beta} \cdot \frac{1}{L_{i,j,k}}$$

$$p(y_f = 0|\mathbf{C}, \mathbf{W}_{\neg f}, \mathbf{D}, \mathbf{X}, \mathbf{Y}_{\neg f}, \mathbf{Z}_{\neg f}) = \frac{n^{y=0}_{\neg f} + \gamma}{\sum_{y=0}^{1} n^{y}_{\neg f} + 2\gamma} \cdot \frac{n^{w_f}_{y=0,i,\neg f} + \beta'}{\sum_{w=1}^{V} n^{w}_{y=0,i,\neg f} + V\beta'}$$

$$p(y_f = 1|\mathbf{C}, \mathbf{W}_{\neg f}, \mathbf{D}, \mathbf{X}, \mathbf{Y}_{\neg f}, \mathbf{Z}_{\neg f}) = \frac{n^{y=1}_{\neg f} + \gamma}{\sum_{y=0}^{1} n^{y}_{\neg f} + 2\gamma} \cdot \frac{n^{w_f}_{y=1,s',\neg f} + \beta}{\sum_{w=1}^{V} n^{w}_{y=1,s',\neg f} + V\beta}$$

where $s' = c_{i,j,k,l}$ which is the topic index of the associated entity of this word.

Parameter estimation

After Gibbs Sampling, we can make the following parameter estimation:

$$\theta_{i,j,s} = \frac{n^{s}_{i,j} + \alpha}{\sum_{s'=1}^{S} n^{s'}_{i,j} + S\alpha}. \qquad \text{thread-section distribution.}$$

$$\psi_{s,w} = \frac{n^{w}_{s,y=1} + \beta}{\sum_{w'=1}^{V} n^{w'}_{s,y=1} + V\beta}. \qquad \text{section-word distribution.}$$

$$\phi_{i,w} = \frac{n^{w}_{i,y=0} + \beta'}{\sum_{w'=1}^{V} n^{w'}_{i,y=0} + V\beta'}. \qquad \text{city-word distribution.}$$

$$\pi_y = \frac{n^{y}_{(.)} + \gamma}{\sum_{y'=0}^{1} n^{y'}_{(.)} + 2\gamma}. \qquad \text{switch distribution.}$$

4 Generating Section-specific Summaries

With the JCSM model presented in the last section, we can learn a word distribution for each section, which can help us find more relevant content for the section. For each section, we rank the forum threads by how likely the words inside a thread is generated from the corresponding section-specific word distribution. We select the top-K threads for each section to perform section-specific summarization.

Extractive summarization has been well studied and many algorithms have been proposed. We choose to build our solution on top of an ILP-based framework proposed by Gillick and Favre (2009), partly because our experiments comparing this ILP framework and other existing methods show its advantage on our data sets (see Section 5). Below we first briefly review this ILP-based summarization framework and then present our proposed improvements.

The idea behind the ILP framework by Gillick and Favre (2009) is to maximize the coverage of so-called "concepts" from the original corpus in the generated summary. In practice, bigrams are used as concepts. Specifically, let us use i to index all the concepts from the original corpus. Let w_i denote the weight of the i-th concept computed based on its frequency and $b_i \in \{0, 1\}$ denote the absence or presence of the concept. The framework aims to maximize $\sum_i w_i b_i$, i.e. the total weighted coverage of the concepts, subject to the following constraints:

$$\begin{array}{rll} & \sum_j l_j s_j \leq L, & (l_j \text{ is the length of the } j\text{-th sentence in terms of words, and } L \text{ is the length limit of the summary.}) \\ \forall i,j: & s_j o_{i,j} \leq b_i, & (s_j \in \{0,1\} \text{ denotes the absence or presence of the } j\text{-th sentence.}) \\ \forall i: & \sum_j s_j o_{i,j} \geq b_i. & (o_{i,j} \in \{0,1\} \text{ denotes whether concept } i \text{ occurs in sentence } j.) \end{array}$$

Although this framework works well for standard summarization, our task is different. We propose the following changes to this framework:

Favoring relevant sentences: Recall that although we select presumably the most relevant threads for each section, we cannot guarantee that each sentence in these threads is related to the section. For example, we observe that the *things-to-do* section is often mixed with content from *restaurants*, *sights*, *transport* and *entertainment* sections. Also, some sentences are less relevant to the target city than others. In order to select the more relevant sentences in the summary, we propose to add the second term in Eqn. 1 below. Here j is used to index all the candidate sentences and u_j is a weight for sentence j based on its relevance.

We measure relevance with respect to both the city and the section. Let $\mathrm{LL}(j,\psi)$ denote the log likelihood of generating sentence j from the section-specific topic ψ and $\mathrm{LL}(j,\phi)$ denote the log likelihood of generating sentence j from the city-specific topic ϕ. We define u_j as follows:

$$u_j \quad \propto \quad \exp\left(\rho \mathrm{LL}(j,\psi) + (1-\rho)\mathrm{LL}(j,\phi)\right).$$

u_j are then normalized to be between 0 and 1. Note that here ρ is a manually defined parameter used to control the tradeoff between city-specific relevance and section-specific relevance. As we will show in Section 5, both relevance factors turn out to be useful.

Covering section-specific points of interest: We hypothesize that a good summary travel guide should mention potential points of interest to the reader. To this end, the last term in Eqn. 1 is added. Specifically, k is an index for unique location names we find that have been labeled as belonging to section s according to the JCSM model. $e_k \in \{0,1\}$ denotes whether the k-th entity is present in the selected sentences, and v_k denotes the weight for this entity based on its frequency.

Eventually, the summarization task is formulated as the following optimization problem:

$$\begin{array}{ll} \text{Maximize:} & \lambda_1 \sum_i w_i b_i + \lambda_2 \sum_j u_j s_j + (1-\lambda_1-\lambda_2)\sum_k v_k e_k \\ \text{Subject to:} & \sum_j l_j s_j \leq L, \\ & \forall i: \sum_j s_j o_{i,j} \geq b_i, \qquad \forall i,j: s_j o_{i,j} \leq b_i, \\ & \forall j: \sum_k s_j p_{j,k} \geq e_k, \qquad \forall j,k: s_j p_{j,k} \leq e_k. \end{array} \tag{1}$$

Here $o_{i,j}$ denotes whether concept i occurs in sentence j, and $p_{j,k}$ denotes whether entity k occurs in sentence j. For the weights w_i and v_k, we normalize them using the total occurrences of bigrams/entities to ensure their values are between 0 and 1. We solve the above optimization problem using the IBM ILOG CPLEX Optimizer[1].

5 Experiments

5.1 Data and Experimental Setup

We use real data from Yahoo! Answers and Lonely Planet for evaluation. We first crawl the travel guides for 10 cities from Lonely Planet, where each travel guide has 8 sections. We then crawl the top 60000 Q&A threads ranked by number of posts related to these 10 cities (6000 for each city) from Yahoo! Answers under the "travel" category where all questions have been grouped by cities. We filter out trivial factoid questions using features used by Tomasoni and Huang (2010). We then use the Stanford

[1] `http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/`

Method	*Singapore*			*Sydney*			*New York City*			*Los Angeles*			Overall Average		
	R-1	R-2	RSU4	R-1	R-2	RSU4	R-1	R-2	RSU4	R-1	R-2	RSU4	R-1	R-2	RSU4
Random	0.4091	0.1046	0.1576	0.4496	0.1100	0.1925	0.4442	0.1192	0.1858	0.4154	0.1130	0.1693	0.4309	0.1115	0.1771
Centroid	0.4029	0.0993	0.1484	0.4228	0.1100	0.1764	0.4235	0.1192	0.1722	0.3763	0.0787	0.1386	0.4133	0.1077	0.1640
LexRank	0.4396	0.1451	0.1891	0.4406	0.1296	0.1955	0.4304	0.1397	0.1859	0.4032	0.0992	0.1661	0.4350	0.1331	0.1894
DivRank	0.4534	0.1504	0.1888	0.4473	0.1161	0.1925	0.4391	0.1167	0.1804	0.4275	0.1180	0.1733	0.4487	0.1317	0.1888
GMDS	0.3918	0.0890	0.1415	0.4339	0.1066	0.1784	0.4064	0.0845	0.1576	0.3846	0.0809	0.1413	0.4045	0.0916	0.1553
ILP-BL	0.4635	0.1650	0.2000	0.4948	0.1731	0.2333	0.4691	0.1613	0.2073	0.4545	0.1445	**0.1981**	0.4755	0.1654	0.2136
Our Method	**0.4723**	**0.1655**	**0.2035**	**0.5078**	**0.1787**	**0.2397**	**0.4716**	**0.1713**	**0.2086**	**0.4543**	**0.1565**	0.1945	**0.4804**‡	**0.1715**‡	**0.2144**‡

Table 2: Comparison of the summarization results. ‡ means the result is better than others except ILP-BL in the same column at 5% significance level measured by Wilcoxon signed rank test. Note that only the average scores are tested for statistical significance based on the 32 summarization tasks in total.

NER tool to recognize named entities in these threads. Since we notice that sometimes entities tagged as PER are also possible points of interest, we include all entities of LOC, ORG and PER types. In order to use higher quality threads for evaluation, for each city we pick the top 600 threads that have the most overlapping points of interest with the Lonely Planet travel guides. On average, each thread contains 5.0 posts and 618.1 words. These 600×10 threads are used to train the JCSM model.

We need human generated model summaries for evaluation. Since it is too time consuming to ask human annotators to look through 600 threads and generate structured summaries, we instead opt to first retrieve the top 30 relevant threads per section per city based on the JCSM results and then ask human annotators to summarize these 30 threads to generate a section-specific summary. Our summarization method as well as the baselines are also applied to these 30 threads per section per city for fair comparison. We randomly select 4 cities for human annotation, giving us $8 \times 4 = 32$ section-specific summarization tasks. For each task, we ask four annotators to read all 30 threads and write a summary as model summaries in our experiments[2].

We use the following baseline algorithms for comparison: (1) **Random**, which randomly picks summary sentences. (2) **Centroid** (Radev et al., 2004), which selects sentences according to several features like tfidf, cluster centroid and position. (3) **LexRank** (Erkan and Radev, 2004b)., which applies a graph-based algorithm . (4) **DivRank** (Mei et al., 2010), which employs a time-variant random walk to enhance diversity. (5) **GMDS** (Wan, 2008), which incorporates the document-level information and the sentence-to-document relationship into the ranking process. (6) **ILP-BL**, which is the method proposed by Gillick and Favre (2009).

We empirically set Dirichlet hyperparameters $\alpha = 0.5, \beta = 0.01, \gamma = 0.01, \beta' = 0.1$. We run JCSM with 400 iterations of Gibbs sampling. For the weight parameters in the ILP model, we empirically set $\lambda_1 = 0.7, \lambda_2 = 0.1, \rho = 0.7$ after we conduct multiple experiments to determine the best values of them from 0.1 to 0.9.

5.2 Summarization Results

To compare the summaries generated by our method with those generated by the baselines, we first compute their ROUGE scores against the human generated model summaries. ROUGE scores have been widely used for evaluation of summarization systems (Lin and Hovy, 2003). We use the ROUGE toolkit[3], which provides multiple kinds of ROUGE metrics including ROUGE-N, ROUGE-L, ROUGE-W and ROUGE-SU4. In the experiment results we report three ROUGE F-measure scores, namely, ROUGE-1, ROUGE-2 and ROUGE-SU4. The higher the ROUGE scores, the better a summary is.

In Table 2 we show the summarization results of our method (with the optimal parameter setting) and the baseline methods. For each city, the scores we show are averaged over the 8 sections. The overall average scores on the right hand side are averaged over the 4 cities. We have the following findings from the table: (1) Compared with the other baselines, the ILP-based baseline clearly shows its advantage, justifying our our design choice of adopting an ILP-based framework as the basis of our method. (2) Our method performs slightly better than ILP-BL based on the overall scores, but the difference is not statistically significant.

[2]The summary dataset can be found at `https://sites.google.com/site/liuyang198908/code-data`.
[3]`http://www.isi.edu/licensed-sw/see/rouge/`

section	*Singapore*		*Sydney*		*New York City*		*Los Angeles*	
	ILP-BL	Our Method	ILP-BL	Our Method	ILP-BL	Our Method	ILP-BL	Our Method
restaurants	0.3750	**0.5417**	0.5714	**0.7143**	0.2500	**0.3750**	0.1053	**0.2105**
hotels	0.4091	0.4091	0.0000	**0.5000**	0.3636	**0.5000**	0.4500	**0.5500**
shopping	0.1429	**0.5357**	0.3750	0.3750	0.1905	0.1905	0.0455	**0.1818**
sights	0.5000	**0.5789**	0.3846	**0.4615**	0.3636	**0.6364**	0.1143	**0.2571**
entertainment	0.1304	**0.2174**	0.2500	**0.7500**	0.0909	**0.2273**	0.2500	**0.4167**
activities	0.4167	**0.5833**	0.2500	0.2500	0.1250	**0.5000**	0.2069	**0.2759**
transport	0.3889	**0.5556**	0.7500	0.7500	0.6000	**0.8000**	0.3158	**0.7368**
things-to-do	0.2105	**0.2632**	0.2500	**0.5500**	0.4583	**0.5833**	0.0000	**0.2000**
average	0.3217	**0.4606**	0.3539	**0.5439**	0.3052	**0.4766**	0.1860	**0.3536**

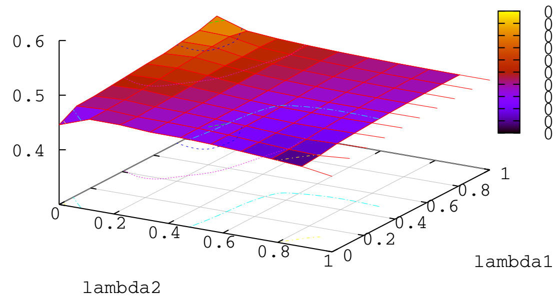
Table 3: Comparison of the recall of named entities of ILP-BL and our method.

Method	Our Complete Model	−EC	−SR	−SecRel	−CityRel
R-1	**0.4804**	0.4520	0.4657	0.4672	0.4796
R-2	**0.1715**	0.1430	0.1669	0.1652	0.1685
RSU4	**0.2144**	0.1987	0.2028	0.2039	0.2120

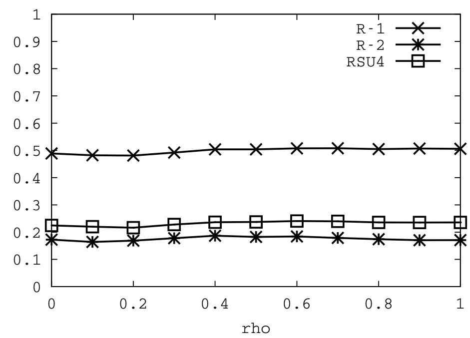
Table 4: Summarization results of the degenerate versions of our method. "−" means removing this component from our complete method. The table shows the average results over data sets of all cites.

Considering that an importance difference between our method and ILP-BL is our focus on points of interest, we further compared ILP-BL and our method using a different metric. The objective is to test the coverage of points of interest in our generated summaries versus the summaries generated by ILP-BL. To this end, we first identify all the named entities in the model summaries using the Stanford NER tool. We then check the percentage of these named entities covered in the generated summaries and report these recall scores in Table 3. We can see that for majority of the 32 section-specific summaries, our method clearly has a higher recall score than ILP-BL, showing that our method generates summaries with more potential points of interest.

To further understand whether all the components of our improved ILP method have contributed to the performance improvement, we compare our overall method with a few degenerate versions of our method. In each degenerate version, we remove a single component of the objective function. The results are shown in Table 4, where −EC removes the consideration of entity coverage (i.e. setting $\lambda_1+\lambda_2 = 1$), −SR removes the consideration of sentence relevance (i.e. setting $\lambda_2 = 0$), −SecRel removes only the section-specific relevance of the sentences (i.e. setting $\rho = 0$), and −CityRel removes only the city-specific relevance of the sentences (i.e. setting $\rho = 1$). We can see that each degenerate version of our method performs worse than the complete method, which shows that all components of the objective function are useful. In particular, entity coverage and section-specific relevance seem to be the more important components.



(a) R-1(λ_1, λ_2)



(b) ρ

Figure 2: Summarization performance of our method by varying the value of the parameters λ_1, λ_2 and ρ.

5.3 Analysis of Topic Words

We show some further analysis of our results. To begin with, we analyze the learning results of JCSM. The top words in city-specific word distributions and section-specific word distributions learnt by JCSM are presented in Table 5 and Table 6. Generally we observe clean top words for each city and each section. For each city, city-specific words are those associated with the corresponding city. For example, for Singapore, we see words such as “s$” (Singapore dollars), “sentosa” (an island resort in Singapore), “orchard” (a boulevard that is the retail and entertainment hub of Singapore) and “bugis” (a popular shopping place). For New York City, we see “square”, “times” and “manhattan”. For each section, section-specific words are those words which frequently appear when people discuss about this section, such as “menu”, “dishes” and “ seafood” for the restaurant section and “train”, “bus” and “station” for the transport section.

Topic 1 *Singapore*	Topic 2 *SFO*	Topic 3 *Chicago*	Topic 4 *Boston*	Topic 5 *LA*	Topic 6 *NYC*	Topic 7 *Seattle*	Topic 8 *Pairs*	Topic 9 *London*	Topic 10 *Sydney*
singapore	sf	chicago	boston	beach	york	downtown	paris	london	sydney
s$	san	downtown	end	hollywood	nyc	seattle	de	tube	harbour
centre	francisco	park	north	los	park	needle	metro	underground	beach
food	gate	city	downtown	angeles	central	space	eiffel	central	manly
shopping	golden	neighborhood	fenway	la	square	market	french	centre	beaches
sentosa	bay	north	bay	downtown	times	rain	la	british	house
road	bart	lake	harvard	drive	manhattan	place	du	palace	opera
orchard	union	mile	place	california	broadway	pike	tower	thames	quay
chinese	wharf	loop	city	miles	city	center	des	end	australian
mrt	muni	ave	college	hills	street	waterfront	rue	kensington	rocks
bugis	square	field	subway	long	east	area	le	station	bridge

Table 5: Top city specific words discovered by JCSM.

Topic 1 *restaurants*	Topic 2 *hotels*	Topic 3 *shopping*	Topic 4 *sights*	Topic 5 *entertainment*	Topic 6 *activities*	Topic 7 *transport*	Topic 8 *thingsToDo*
food	hotel	shop	museum	bar	visit	train	bar
restaurant	rooms	store	city	music	park	bus	place
menu	free	stores	park	club	tour	station	tour
dishes	wi-fi	shopping	art	place	fun	airport	city
place	walk	shops	building	night	city	time	food
bar	located	find	built	dance	walk	line	art
chicken	offers	clothes	world	beer	day	car	day
fish	station	wear	place	clubs	time	walk	including
fresh	features	mall	house	crowd	shopping	minutes	music
seafood	tv	place	area	bars	museum	hours	restaurant

Table 6: Top section specific words discovered by JCSM.

5.4 Parameter Sensitivity Analysis

We further give parameter sensitivity analysis for our proposed method. We show how sensitive our results are with respect to the parameters λ_1, λ_2 and ρ. We choose the Sydney data set to perform parameter sensitivity analysis. In Figure 2(a), we show how ROUGE-1 varies with respect to λ_1 and λ_2. We can see that the performance fluctuates within a limited range as we vary λ_1 and λ_2. We find the trend for ROUGE-2 and ROUGE-SU4 is similar so we leave out the figures for them. In Figure 2(b) we see that the performance is pretty stable as we vary ρ.

5.5 Sample Output and Case Study

Finally, we show a sample travel guide our method generates for Sydney in Table 7. We can see that first of all the sentences selected by our method have high relevance to the corresponding sections. Second, through observation we find that humans tend to select sentences containing more points of interest as summary. Our summary sentences contain many points of interest as highlighted, showing the advantage of our method.

Sample Summary Sentences Generated from Yahoo! Answers by Our Method for Sydney
Hotel
Sorry can not recommend you a hotels as I have no idea of pricing , but if you want a nice area , check hotels in **Bondi** and **Manly Beaches** . As for the **Acer Arena** , that is in the **Homebush Olympic Park** and you can choose to live in either **Parramatta** or the city . You need to live in one of the surrounding residential suburbs , close to a train line . Try **Alexandria** , **Newtown** , **Surry hills** for inner suburbs . . . .
Sights
You can walk around the harbor area to the **Opera House** and you can see the beautiful **Harbor Bridge** . All this is apart from the **Opera House** and the **Botanical Gardens** . Visit the **Custom House Circular Quay** and see a model of **Sydney**. You must also do a day trip to the **Blue Mountains** . **Harbour Wedding** is one of the major attraction in **Sydney** . . . .
Entertainment
George Street has a number of bars . All the bars around the harbour are really good day and night . If you want to stay in a hotel where there is entertainment at night , you could look at **Woolloomooloo** , **Darlinghurst** , **Surry Hills** , **Kings Cross** or **Potts Point** . **Newtown** is good for bars . Get them to see a theatre show or something at the **Opera House** . . . .
Things-to-do
If you are going out for the day , starting with a walk to the city will be most enjoyable . Take a public ferry from **Circular Quay** to **Darling Harbour** , about 15 minutes across the harbour and under the bridge , when you get to **Darling Harbour** go and see the **Chinese Gardens** . There are lots of interesting things to see and do in and around **Sydney** . . . .
Activities
They have good markets at the weekend and great views of the **Opera House** . The **Opera House** is free to have a look at , if you like art then walk through the **Botanical Gardens** and go and see the art gallery . If you 're feeling brave , you can do a **Harbour Bridge** walk , though I think it may be a little pricey . . . .

Table 7: Excerpts from the summary generated from Yahoo! Answers by our method for Sydney. We show summaries for the 5 sections other than the 3 sections shown in Table1. Named entities are highlighted in bold font.

6 Related Work

Multi-document summarization is a process to generate a text summary by reducing documents in size while retaining the main points of the original documents. It has been extensively studied in the NLP community, with most efforts on extractive summarization. Our work is also based on extractive summarization. Extractive summarization essentially selects a set of sentences from the original documents to form a summary.

To select sentences, different features and ranking strategies have been studied. Early work focuses on finding good features to select summary sentences. Radev et al. (2004) proposed a centroid-based summarizer which combines several pre-defined features like tfidf, cluster centroid and position to score sentences. Lin and Hovy (2002) built the NeATS multi-document summarization system using term frequency, sentence position, stigma words and simplified Maximal Marginal Relecvance (MMR). Nenkova et al. (2006) proved that high-frequency words were significant in reflecting the focus of documents. Ouyang et al. (2010) studied the influence of different word positions in summarization. Later, graph-based ranking algorithms have been successfully applied to summarization. LexPageRank (Erkan and Radev, 2004a) is a representative one based on the PageRank algorithm (Page et al., 1999). Later extensions include ToPageRank (Pei et al., 2012), which incorporates topic information into the propagation mechanism, the manifold-ranking based method for topic-focused summarization (Wan et al., 2007) and DivRank (Mei et al., 2010), which introduces a time-variant matrix into a reinforced random walk to balance prestige and diversity.

More recently, Integer Linear Programming (ILP) based framework was introduced as a global inference algorithm for multi-document summarization by McDonald (2007), which considers information and redundancy at the sentence level. Gillick and Favre (2009) studied information and redundancy at a sub-sentence, “concept” level, modeling the value of a summary as a function of the concepts it covers. In our work we also model concept level coverage of the summaries. Li et al. (2013) proposed a regression model to estimate the frequency of bigrams in the reference summary and analyzed the impact of bigram selection, weight estimation and ILP setup. Haghighi and Vanderwende (2009) constructed a sequence of generative probabilistic models for multi-document summarization, exhibiting ROUGE gains along the way. Sauper and Barzilay (2009) investigated an approach for creating a comprehensive textual overview of subject composed of information drawn from the Internet and applied ILP to optimize both local fit of information into each topic and global coherence across the entire overview. Li et al. (2011) developed an entity-aspect LDA model to cluster sentences into aspects and then extend LexRank algorithm to rank sentences. Hu and Wan (2013) proposed to use SVR model and ILP method to generate presentation slides for academic papers.

Our work is different from standard ILP-based multi-document summarization. We designed a latent variable model to first separate the threads to be summarized into sections based on model gravel guides.

We also emphasized the inclusion of potential points of interest in formulating the ILP optimization problem.

Our work is also closely related to previous work on answer summarization in community-based QA sites. Previous work on summarizing answers is mainly based on query focused multi-document summarization techniques to summarize multiple answer documents given a single question. Liu et al. (2008) proposed a CQA question taxonomy to classify questions in CQA and question-type oriented answer summarization for better reuse of answers. Tomasoni and Huang (2010) proposed two concept-scoring functions to combine quality, coverage, relevance and novelty measures for answer summary in response to a question and showed that their summarized answers constitute a solid complement to best answers voted by CQA users. Chan et al. (2012) presented an answer summarization method for complex multi-sentence questions. For our work, we study a new problem of summarizing multiple threads to automatically generate city travel guides based on known template structure from well-written travel guides, which is different from the setting of single Q&A thread summarization in the previous related studies.

7 Conclusion and Future Work

In this paper we proposed a summarization framework to generate well structured supplementary travel guides from social media based on a latent variable model and integer linear programming. The latent variable model could align forum threads with the section structure of well-written travel guides. Compared to standard concept based ILP methods, our method additionally tries to cover more named entities as points of interest and maximizes sentence relevance scores measured by section-specific and city-specific word distributions learnt by the latent variable model. Extensive experiments with real data from Yahoo! Answers show that our proposed method is able to generate better summaries compared with a number of multi-document summarization baselines measured by ROUGE scores.

Currently our generated summaries may have overlap with the well-written model travel guides. In the future, we plan to improve our method to emphasize the selection of *additional* information from social media compared with the model travel guides. We will also look into the problem of how to summarize information that does not fit into the template structure derived from model travel guides.

Acknowledgments

This work was done during Liu Yang's visit to Singapore Management University. The authors would like to thank the reviewers for their valuable comments on this work.

References

Wen Chan, Xiangdong Zhou, Wei Wang, and Tat-Seng Chua. 2012. Community answer summarization for multi-sentence question with group l1 regularization. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, pages 582–591, Stroudsburg, PA, USA. Association for Computational Linguistics.

Günes Erkan and Dragomir R. Radev. 2004a. Lexpagerank: Prestige in multi-document text summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, volume 4 of *EMNLP '04*.

Günes Erkan and Dragomir R. Radev. 2004b. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Int. Res.*, 22(1):457–479, December.

Dan Gillick and Benoit Favre. 2009. A scalable global model for summarization. In *Proceedings of the Workshop on Integer Linear Programming for Natural Langauge Processing*, ILP '09, pages 10–18, Stroudsburg, PA, USA. Association for Computational Linguistics.

Aria Haghighi and Lucy Vanderwende. 2009. Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, pages 362–370, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yue Hu and Xiaojun Wan. 2013. Ppsgen: Learning to generate presentation slides for academic papers. In *Proceedings of the 21st international jont conference on Artifical intelligence*, IJCAI '13, pages 2099–2105.

Peng Li, Yinglin Wang, Wei Gao, and Jing Jiang. 2011. Generating aspect-oriented multi-document summarization with event-aspect model. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1137–1146, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chen Li, Xian Qian, and Yang Liu. 2013. Using supervised bigram-based ilp for extractive summarization. In *Proceedings of the 51th Annual Meeting of the Association for Computational Linguistics*, ACL '13, pages 1004–1013, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chin-Yew Lin and Eduard Hovy. 2002. From single to multi-document summarization: A prototype system and its evaluation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 457–464, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chin-Yew Lin and Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 71–78, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yuanjie Liu, Shasha Li, Yunbo Cao, Chin-Yew Lin, Dingyi Han, and Yong Yu. 2008. Understanding and summarizing answers in community-based question answering services. In *Proceedings of the 22Nd International Conference on Computational Linguistics - Volume 1*, COLING '08, pages 497–504, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ryan McDonald. 2007. A study of global inference algorithms in multi-document summarization. In *Proceedings of the 29th European Conference on IR Research*, ECIR'07, pages 557–564, Berlin, Heidelberg. Springer-Verlag.

Qiaozhu Mei, Jian Guo, and Dragomir Radev. 2010. Divrank: The interplay of prestige and diversity in information networks. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, pages 1009–1018, New York, NY, USA. ACM.

Ani Nenkova, Lucy Vanderwende, and Kathleen McKeown. 2006. A compositional context sensitive multi-document summarizer: Exploring the factors that influence summarization. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, pages 573–580, New York, NY, USA. ACM.

You Ouyang, Wenjie Li, Qin Lu, and Renxian Zhang. 2010. A study on position information in document summarization. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 919–927, Stroudsburg, PA, USA. Association for Computational Linguistics.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November. Previous number = SIDL-WP-1999-0120.

Yulong Pei, Wenpeng Yin, and Lian'en Huang. 2012. Generic multi-document summarization using topic-oriented information. In *Proceedings of the 12th Pacific Rim International Conference on Trends in Artificial Intelligence*, PRICAI'12, pages 435–446, Berlin, Heidelberg. Springer-Verlag.

Dragomir R. Radev, Hongyan Jing, Malgorzata Styś, and Daniel Tam. 2004. Centroid-based summarization of multiple documents. *Inf. Process. Manage.*, 40(6):919–938, November.

Christina Sauper and Regina Barzilay. 2009. Automatically generating wikipedia articles: A structure-aware approach. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, ACL '09, pages 208–216, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mattia Tomasoni and Minlie Huang. 2010. Metadata-aware measures for answer summarization in community question answering. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 760–769, Stroudsburg, PA, USA. Association for Computational Linguistics.

Xiaojun Wan, Jianwu Yang, and Jianguo Xiao. 2007. Manifold-ranking based topic-focused multi-document summarization. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 2903–2908, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Xiaojun Wan. 2008. An exploration of document impact on graph-based multi-document summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 755–762, Stroudsburg, PA, USA. Association for Computational Linguistics.