Collaborative Exploratory Search in Real-World Context

Naoki Tani  
Danushka Bollegala  
Shuhei Iitsuka  
Yutaka Matsuo  
The University of Tokyo,  
Hongo 7-3-1, Tokyo  
113-8565, Japan  
tani@mi.ci.i.u-tokyo.ac.jp  
danushka@iba.t.u-tokyo.ac.jp  
(iitsuka|matsuo)@biz-model.t.u-tokyo.ac.jp

Naiwala Chandrasiri,  
Kazunari Nawa  
Toyota InfoTechnology Center  
Aksasaka 6-6-20, Tokyo  
107-0052, Japan  
(np-chandrasiri|nawa)@jp.toyota-  
ict.com

Keisuke Okamoto  
Toyota Motor Corp.  
3-3-25 Higashi Ikebukuro  
Tokyo, 170-8405, Japan  
keisuke@okamoto.tec.toyota.co.jp

ABSTRACT

We propose Collaborative Exploratory Search (CES), which is an integration of dialog analysis and web search that involves multi-party collaboration to accomplish an exploratory information retrieval goal. Given a real-time dialog between users on a single topic; we define CES as the task of automatically detecting the topic of the dialog and retrieving task-relevant web pages to support the dialog. To recognize the task of the dialog, we apply the Author–Topic model as a topic model. Then, attribute extraction is applied to the dialog to obtain the attributes of the tasks. Finally, a specific search query is generated to identify the task-relevant information. We implement and evaluate the CES system for a commercial in-vehicle conversation. We also develop an iPad application that listens to conversations among users and continuously retrieves relevant web pages. Our experimental results reveal that the proposed method outperforms existing methods, which demonstrates the potential usefulness of collaborative exploratory search with practically usable accuracy levels.

Categories and Subject Descriptors

H.3.3 [Information Systems]: Information Search and Retrieval

General Terms

Algorithms

Keywords

Personalized and Collaborative Search, Information Extraction

1. INTRODUCTION

During the last decade, web searching has become commonplace in our everyday lives. Currently, most search engines supply query-based searching that is designed to navigate a user to those web pages that are most relevant to the user’s query. The use of web searching has become very common in our lives. Therefore, many new scenarios of searching arise.

For example, as cars become smarter, a driver might want to search web pages while driving a car to obtain information related to destination or nearby events. Although current navigation systems cannot usually access the web, some car manufacturers, such as Toyota, already provide a premium service for customers to search information from the web assisted by a dedicated telephone operator. The following conversation illustrates such a situation.

Driver: “Hello. I want to eat something special in Kyoto this evening.”
Operator: “OK. What kind of cuisine would you like?”
Driver: “Well... traditional Japanese.”
Operator: “OK, sir. Let me search for traditional Japanese restaurants in Kyoto.”
(Operator searches the web.)
Operator: “There is a good restaurant named Kitcho in Kyoto near your location.”
Driver: “Is it good?”
(Operator searches the web.)
Operator: “I can not guarantee the taste, but it’s been awarded three stars by Michelin.”
Driver: “OK. Could you set the location into the GPS?”
Operator: “Certainly. I sent that information to your car.”

In this scenario, the driver has an information need and the operator makes a search, the search is conducted during a dialog, keywords are extracted from the dialog and fed into the search engine, and the search results are provided promptly and help the task in the dialog. In this paper, we specifically address such a situation: multiple persons collaboratively search the web during the conversation. Our system automatically analyzes the discourse, makes a customized query, and shows the results promptly.

As described in this paper, we propose Collaborative Exploratory Search (CES). CES can be considered as the combination of multi-domain exploratory search and collaborative search. It is an integration of dialog analysis and web search that involves multiparty collaboration to accomplish an exploratory information retrieval goal. The goal of CES is to support users’ collaboration in searching. Given a real-time dialog between users on a single topic, CES extracts topic and attribute–value pairs of a dialog and seeks relevant web pages automatically. Using real-time dialog as input data, it enables unconscious search and does not block collaboration.

The outline of the proposed method is shown in Figure 1. As shown in Figure 1, we break down the CES into three parts: (i)
topic extraction, (ii) attribute extraction, and (iii) web search. For topic extraction, we apply the Author–Topic (AT) model, an unsupervised topic detection algorithm to detect the topic of a dialogue between users. For attribute extraction, we apply a two-stage method to obtain attribute–value pairs. Then the customized query is generated and the search results are shown in part (iii). We evaluate our CES approach using an actual dialog dataset by drivers and operators in Toyota. The results show that we can detect the task in a conversation with 50% accuracy in unsupervised topic extraction, and attain a 0.67 F-measure in attribute extraction.

2. RELATED WORK

Exploratory search is defined as the situation in which the user starts from a not-so-well-defined information need and progressively discovers more on his need and on the information available to address it [6]. Kotov et al. [4] proposed question-guided search, in which a retrieval system would generate potentially interesting questions to the users automatically. The question-guided search system returns “Who is John Kennedy?” or “When was Kennedy sworn as the President of the United States?” if users entered query “john kennedy”. Rajaraman [10] produced Kosmix web service\(^1\): Kosmix is a general-purpose topic discovery engine, which responds to keyword search using a topic page that summarizes the most relevant information related to the subject associated to the search. Bozzon et al. [3] proposed Liquid Query, a paradigm that exploits the power of underlying search services and provides the user with a multi-domain exploratory search environment (e.g., travels, music, shows, food, movies, and so on). Recent studies have revealed that most people want to search with other people [5, 12]. A survey [7] of 204 people revealed that 53.4% of participants do some collaborative search tasks: travel planning, shopping, literature search, technical information, and so on. For example, TeamSearch [9] is a tabletop application that enables small, co-located groups to search for digital photographs from a metadata-tagged repository. S3 [8] allows users to share useful sites found during a web search asynchronously by representing search results in a persistent file format that can be sent to and augmented by several people. Amershi et al. introduced CoSearch [1], a system that is intended to improve the experience of co-located collaborative web search by leveraging readily available devices such as mobile phones and extra mice.

3. PROBLEM DEFINITION

Let us consider the situation where two agents, \(A\) and \(B\), are having a dialogue \(D\) about a particular topic \(T\). Dialogue \(D\) includes some attributes \(V\). \(D\) can be represented as a vector (hereinafter designated as a dialog vector)

\[
d = (a_1, b_1, ..., a_T, b_T)
\]

where \(a_i\) and \(b_i\) are sequences of words (may or may not be complete sentences) uttered respectively by \(A\) and \(B\). Elements in \(d\)

\(^1\)http://www.kosmix.com/

Figure 1: Outline of the proposed system.

Figure 2: Topic models: LDA and AT.

4. TOPIC EXTRACTION

We briefly review the Latent Dirichlet Allocation (LDA) model and Author–Topic (AT) model, which we use in our system. LDA is a Bayesian network that generates a document using a mixture of topics [2]. In its generative process, for each document \(d\), a multinomial distribution \(\theta_d\) over topics is randomly sampled from a Dirichlet distribution with parameter \(\alpha\); then, to generate each word, a topic \(z_{di}\) is chosen from this topic distribution, and a word, \(w_{di}\), is generated by random sampling from a topic-specific multinomial distribution \(\phi_{z_{di}}\). Figure 2 (a) shows the graphical model for LDA.

Despite its simplicity and numerous applications, LDA cannot capture the authorship of documents. For example, multiple authors might contribute to a particular document and we might want to extract topics considering both word distribution as well the authors who wrote those words in a document. The Author–Topic (AT) model [11] extends LDA by incorporating author information for topic extraction. Figure 2 (b) shows an AT model. Specifically, each author is associated with a multinomial distribution over topics, represented by \(\theta\). Each topic is associated with a multinomial distribution over words, represented by \(\phi\). The multinomial distributions \(\theta\) and \(\phi\) have a symmetric Dirichlet prior with hyperpa-
parameters $\alpha$ and $\beta$. For each word in a document, we sample an author $x$ uniformly from $a_d$, then sample a topic $z$ from the multinomial distribution $\theta$ associated with author $x$ and sample a word $w$ from a multinomial topic distribution $\phi$ associated with topic $z$. This sampling process is repeated $N_d$ times to generate a document $d$ (here, $N_d$ is the total number of words in document $d$). The AT model has two sets of unknown parameters $- A$ author-topic distributions $\theta$, and $T$ topic distributions $\phi$ – as well as the latent variables corresponding to the assignments of individual words to topics $z$ and authors $x$. It has been shown that parameter estimation via Expectation-Maximization (EM) algorithm engenders local maxima and is computationally inefficient in topic models. Consequently, Gibbs sampling has been used to estimate the parameters in AT models. Once we have estimated the parameters $\theta$ and $\phi$, we can use those values to compute the posterior probability of $p(t|D)$, to assign a topic to a given document $D$. We model a multi-party dialog using AT model as follows. Each author in the AT model is replaced by a speaker who participates in a dialog. A document in the AT model is replaced by a single dialog. Moreover, the words that comprise a document are replaced by the utterances in the corresponding dialog. By modeling a multi-party dialog as an AT model, we leverage on existing work on topic models to solve the collaborative explorative search problem.

5. ATTRIBUTE EXTRACTION

Once we have identified the topic of the conversation, we must extract keywords related to that topic to retrieve information to support the conversation from a Web search engine. We extract attributes and values for the main topic as keywords. For example, if the topic is about a hotel reservation, then we extract attribute values such as the location of the hotel, the number of people staying, the check-in date, the check-out date, desired price range, Extracting attributes from texts has received much attention lately as a collaboration between personalization and web search. We conduct two sets of experiments to evaluate our CES system. First, we measure the performance of topic extraction. Then we evaluate the attribute extraction method and show how many attributes are correctly chosen.

We use a dialog dataset provided by Toyota Motor Corp. The dataset includes records of conversations between a driver and an operator: A call center operator provides a concierge service to drivers and provides information services such as recommendation and reservation of restaurants, hotels, or shops, and providing a phone number or the location of landmarks and so on. The data include a speaker, utterances obtained by speech recognition plus manual fixing, and time. For privacy reasons, the names of drivers and operators are anonymized. We target Japanese real-time dialog, although our method does not depend on any specific language. The Toyota dialog dataset includes 604 dialogs and 47,322 utterances, consisting of 356,618 words. We extract attribute-value pairs and divide dialogs into six tasks as shown in Table 1. The NAVIGATION and RESTAURANT tasks are the most numerous. Drivers often want to know the locations of places, or want to search for restaurants. Although the durations of the dialogs are not so varied among tasks, the average number of turns presents some tendencies: HOTEL and RESTAURANT needs more turns than others. Drivers must clarify their preferences and a higher degree of interaction might be necessary.

Table 1: Basic statistics for each task in Toyota dialog dataset.

<table>
<thead>
<tr>
<th>Task</th>
<th>#dialogs</th>
<th>duration</th>
<th>#of turns</th>
<th>#of utterances</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOTEL</td>
<td>30</td>
<td>158 s</td>
<td>110</td>
<td>611</td>
</tr>
<tr>
<td>NAVIGATION</td>
<td>53</td>
<td>154 s</td>
<td>67</td>
<td>354</td>
</tr>
<tr>
<td>RESTAURANT</td>
<td>73</td>
<td>148 s</td>
<td>91</td>
<td>500</td>
</tr>
<tr>
<td>SHOP</td>
<td>20</td>
<td>143 s</td>
<td>71</td>
<td>381</td>
</tr>
<tr>
<td>PARKING</td>
<td>18</td>
<td>131 s</td>
<td>69</td>
<td>397</td>
</tr>
<tr>
<td>CALL</td>
<td>17</td>
<td>156 s</td>
<td>87</td>
<td>502</td>
</tr>
</tbody>
</table>

6. MULTI-DOMAIN SEARCH

After identifying the topic in a dialog and extracting appropriate keywords for that topic, we retrieve information relevant to the topic using domain-specific search engines. Here, the input is the topic and attribute-value pairs extracted from the target dialog, and the output is a list of web pages which is most relevant to the task, which satisfies the requirement of attribute-value pairs. We obtain relevant information from not only surface web but also from deep web using many web service APIs. We can thereby obtain the largest range of information and can mutually compare the results obtained using several web services. For example, the system gets HOTEL dialog among users and extracts attribute-value pairs such as (place, “Tokyo”), (heads, 2), (price, $100). In this case, the system searches for a hotel in Tokyo with a twin-bed room for $100 or less. In order for a user to compare results, web pages are selected widely from various perspectives. For instance, if users are searching for a restaurant, then we show the most popular restaurant and high-class restaurants. Users can choose from them depending on their tastes.

7. EXPERIMENTS

We evaluate the performance of CES in this section. As shown in Figure 1, CES are decomposed into three tasks: topic extraction, attribute extraction, and web search. We conduct two sets of experiments to evaluate our CES system. First, we measure the performance of topic extraction. Then we evaluate the attribute extraction method and show how many attributes are correctly chosen.

We use a dialog dataset provided by Toyota Motor Corp. The dataset includes records of conversations between a driver and an operator: A call center operator provides a concierge service to drivers and provides information services such as recommendation and reservation of restaurants, hotels, or shops, and providing a phone number or the location of landmarks and so on. The data include a speaker, utterances obtained by speech recognition plus manual fixing, and time. For privacy reasons, the names of drivers and operators are anonymized. We target Japanese real-time dialog, although our method does not depend on any specific language. The Toyota dialog dataset includes 604 dialogs and 47,322 utterances, consisting of 356,618 words. We extract attribute-value pairs and divide dialogs into six tasks as shown in Table 1. The RESTAURANT and NAVIGATION tasks are the most numerous. Drivers often want to know the locations of places, or want to search for restaurants. Although the durations of the dialogs are not so varied among tasks, the average number of turns presents some tendencies: HOTEL and RESTAURANT needs more turns than others. Drivers must clarify their preferences and a higher degree of interaction might be necessary.

We first measure the performance of topic extraction. The task is to classify the dialog into one of the six task types. We devide 604 dialogs into 404 training data and 200 test data. We annotated the task type of each dialog manually. We use pLDA2 as a baseline with the number of topics $T=10$. LDA parameters are set to $\alpha = 0.1$ and $\beta = 0.01$. Table 3 presents the performance of AT model, used in our CES system, and a baseline. Overall, the precision and recall of AT are higher than those of LDA. Table 2 portrays a confusion matrix of classification. The upper part is the result of LDA; the lower part is the result of AT model. Apparently, LDA classifies dialog mostly to NAVIGATION and RESTAURANT, whereas AT model classifies into all tasks properly. From the AT result, we can find that HOTEL and RESTAURANT tasks are confusing because both

2http://code.google.com/p/plda/
Table 2: Confusion matrix of the topic extraction. Legend: H (HOTEL), N (NAVIGATION), R (RESTAURANT), S (SHOP), P (PARKING), C (CALL).

<table>
<thead>
<tr>
<th></th>
<th>H</th>
<th>N</th>
<th>R</th>
<th>S</th>
<th>P</th>
<th>C</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>0</td>
<td>21</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>N</td>
<td>0</td>
<td>43</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>0</td>
<td>0.84</td>
</tr>
<tr>
<td>R</td>
<td>0</td>
<td>43</td>
<td>14</td>
<td>7</td>
<td>6</td>
<td>0</td>
<td>0.20</td>
</tr>
<tr>
<td>S</td>
<td>0</td>
<td>12</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0.17</td>
</tr>
<tr>
<td>P</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0.22</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Precision = 0.00 0.34 0.60 0.18 0.14 0.00 0.58 0.65 0.61 0.39 0.66 0.51 0.27 1.00 0.15

(b) AT model: Proposed method

<table>
<thead>
<tr>
<th></th>
<th>H</th>
<th>N</th>
<th>R</th>
<th>S</th>
<th>P</th>
<th>C</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>16</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>3</td>
<td>0.57</td>
</tr>
<tr>
<td>N</td>
<td>3</td>
<td>23</td>
<td>9</td>
<td>9</td>
<td>0</td>
<td>8</td>
<td>0.44</td>
</tr>
<tr>
<td>R</td>
<td>18</td>
<td>6</td>
<td>25</td>
<td>11</td>
<td>0</td>
<td>6</td>
<td>0.38</td>
</tr>
<tr>
<td>S</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>9</td>
<td>0</td>
<td>1</td>
<td>0.53</td>
</tr>
<tr>
<td>P</td>
<td>0</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>0.15</td>
</tr>
<tr>
<td>C</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>0.40</td>
</tr>
</tbody>
</table>

Precision = 0.39 0.66 0.51 0.27 1.00 0.15

Table 3: Performance of the proposed method and baselines.

<table>
<thead>
<tr>
<th>Method</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDA</td>
<td>0.21</td>
<td>0.24</td>
<td>0.22</td>
</tr>
<tr>
<td>AT (proposed)</td>
<td>0.50</td>
<td>0.41</td>
<td>0.45</td>
</tr>
</tbody>
</table>

Table 4: Performance of the proposed method and baselines in the attribute extraction task.

<table>
<thead>
<tr>
<th>Method</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>AE-clueword (proposed)</td>
<td>0.65</td>
<td>0.73</td>
<td>0.68</td>
</tr>
<tr>
<td>AE-clueword-driver</td>
<td>0.63</td>
<td>0.71</td>
<td>0.67</td>
</tr>
<tr>
<td>AE-clueword-operator</td>
<td>0.58</td>
<td>0.65</td>
<td>0.61</td>
</tr>
<tr>
<td>AE-driver</td>
<td>0.52</td>
<td>0.59</td>
<td>0.55</td>
</tr>
<tr>
<td>AE-operator</td>
<td>0.50</td>
<td>0.56</td>
<td>0.53</td>
</tr>
</tbody>
</table>

8. CONCLUSION

We proposed collaborative exploratory search (CES), which involves multiparty collaboration to accomplish an exploratory information retrieval goal. Experimental results from a topic extraction and attributes extraction task showed that the proposed method outperforms baselines.

9. REFERENCES