Building a Semantic Graph based on Sequential Language Model for Topic-Sensitive Content Extraction

Yan Liang
Department of Industrial and Systems Engineering
The Hong Kong Polytechnic University
Hong Kong SAR, CHINA
Liangyan.lynn@polyu.edu.hk

Ying Liu
Department of Mechanical Engineering
National University of Singapore
Singapore 117576
mpeliuy@nus.edu.sg

ABSTRACT
Graph-based models have been explored to extract information of interest from a text collection. They can potentially incorporate related information to rank important contents. In this paper, we design a semantic graph model for topic-sensitive contents extraction. The topic-sensitive contents refer to segments of a document with respect to a certain aspect of a topic. For example, in online product reviews, customers tend to know the sentiment aspects associated with various product features, while designers intend to understand the reasons behind, i.e. why customers like or dislike a product, and how further improvement can be made. The contents biased to a certain topic aspect can help users to locate the information of interest and gain more insights of a topic. In order to extract the topic-sensitive contents, our semantic graph first learns from the training data and models the initial scores of sentence nodes based on word weights. The word weightings are estimated in terms of how strongly a word is associated with a specific aspect through a sequential language modeling based on hidden Markov model. The link weights are represented using the estimated word weights which reveal how closely the sentences are linked with each other with respect to a certain aspect of a topic. Based on the semantic graph formed, the ranking process aims to prioritize sentences for topic-sensitive content extraction. In the experimental study, we test and validate our approach on extracting motivational reasons from patents for design analysis. The results demonstrate the merits of the proposed approach.

Categories and Subject Descriptors
H.3.3 [Information Search and Retrieval]: Text Mining

General Terms: Algorithms

Keywords
Semantic sentence graph, Topic-sensitive content, sequential language model

1. INTRODUCTION
As the increasing amount of information is digitalized in different text collections, discovering and extracting information of interest is a common task in many applications. Although a search engine can retrieve relevant documents with respect to a specify topic, it often returns many documents, so users have to spend much time to locate and extract information pieces of interest. In addition, different users have different aspects of concerns when exploring information of the same topic. For example, when searching for reviews of a product, e.g. camera, customers usually would like to know the general sentiment aspect (e.g. positive, negative and neutral) towards product features (e.g. lens body and performance). Recently many studies have focused on opinion mining to analyze people's sentiments toward different topics [1, 2]. In addition, from designers' point of view, designers may possibly intend to understand the opinion reason aspect, i.e. why customers like or dislike particular features of a product. The reasons behind the opinions can provide valuable sources for designers to analyze and understand customers' concerns and needs towards different products. Therefore, it is appealing to extract contents with respect to a certain aspect of the topic in a document to satisfy users' information needs. We refer this kind of contents as topic-sensitive contents.

From another perspective, such topic-sensitive contents server as a summary biased to a certain aspect of a topic in a text document. This summary can help users to gain more insights compared with a standard single document summarization that usually gives an overview of the topic in a document. For example, in the prior art search of engineering design, the contents of a document from different aspects, such as the motivational reason aspect, design solution aspect and solution argument aspect, can help designers to analyze and understand the design reasons for design analysis. The contents of the motivational reason aspect can help designers to understand why the design issues are received much attention. The contents from the solution argument aspect can help designers to understand the pros and cons of the relevant design solutions. Another example is in biomedical domain. Experts not only intend to know the finding aspect of the experiment, but also are interested in the aspect of causes and effects related to the findings [3]. Therefore, an interesting research question is how to extract contents that is not only related to the main topic in a document but also biased to a certain aspect specified by a user.

Graph-based models have been studied for extracting information of interest from text collections. They have been used in automatic text summarization, information extraction, and information retrieval and so on. The graph structure has nature features to represent, model and incorporate the related information in the documents for extracting information of interest. For example, in automatic summarization, a sentence graph with sentences as nodes and sentence relationships as links is used to rank the importance of sentence to generate a summary. In making use the graph model, how the nodes and links in the graph can be defined are two basic elements. In this paper, we

MLG '11 San Diego, CA, USA

propose a different way of building a semantic graph by learning the word usage patterns in representing a certain aspect of a topic for topic-sensitive contents extraction.

In our study, we would allow a user to flexibly describe their desired aspect of a topic by simply selecting some sample sentences in the documents. Based on the specific aspect, our basic idea of building a semantic graph for topic-sensitive contents extraction is to learn the word strength (weight) for each word in expressing messages with respect to a certain aspect at the sentence level. It is based on the assumption that the usages of words are sensitive to the topic aspect. We also consider that even the same words in a document they may have different abilities to indicate semantic strengths with respect to a certain aspect of a topic. In addition, based on such term weights, it would be better to model the relationships in a sentence graph for ranking sentences with respect to the aspect. Specifically, given a document, to build the semantic graph, we first estimate the term weight of each word in a sentence to indicate how strongly a word can convey the semantic information with respect to a specific aspect by exploiting a sequential language model based on hidden Markov model (HMM). Using the word weights to estimate the sentence similarity, we then build up a semantic sentence graph for a graph-based ranking method to rank the sentences in the document.

The rest of this paper is organized as follows. Section 2 reviews the relevant topics on graph-based methods and sequential models in extracting information of interest. In Section 3, we define the key concepts used in our study. In Section 4, we introduce a semantic graph model to rank sentence for topic-sensitive content extraction by using sequential language model based on HMM. Section 5 then shows the experiments and discussions about the proposed method. Section 6 concludes this paper.

2. RELATED WORK

In order to help users to organize relevant information, how useful information from a text collection can be extracted has attracted a lot of attention. In this Section, we review some relevant topics on graph-based methods and sequential models in extracting information of interest.

2.1 Graph-based Methods

Graph-based methods have been studied to extract information of interest in many applications [4]. In making use of the graph-based structure, the definition of the nodes and the measurement of the relations between nodes are two basic elements to prioritize the information.

In the context of web information retrieval, the PageRank [5] and HITS algorithms [6] are the most notable approaches to model the web as a big graph with pages as nodes and the links between pages as edges to prioritize the set of pages within the graph. In the context of information extraction, graph-based methods have been proposed for relation extraction, event extraction and so on. Hassan et al. [7] presented an unsupervised approach for some information extraction tasks like relation extraction and characterization task based on the graph mutual reinforcement. Chen et al. [8] investigated a graph based semi-supervised learning approach for relation extraction. They represented the labeled and unlabeled examples as nodes and distances of examples as edge weights. The graph was then used to rank the unlabeled examples by using the nearby labeled examples as well as by the nearby unlabeled examples. Bjorne et al. attempted to extract complex events among genes and proteins from biomedical literature using dependency parse graphs based on an array of features such as token features, the number of named entities and counts of tokens in the sentences.

Graph-based methods have also been studied in automatic text summarization to identify most salient sentences as a summary [9]. For example, Mihalcea and Tarau [10] applied their TextRank model to rank sentences in a document based on sentence connection which can be simply determined as the number of common tokens between sentences. Similar to TextRank model using the concept of PageRank algorithm as the ranking function, Eran and Radev [11] built a weighted undirected graph to represent documents by taking sentences as nodes and cosine similarity between sentences as edge weights. In addition, not only relying on the local sentence-specific information in the specific document, Wan et al. [12] attempted to use the sentences of the neighbor documents to rank sentences for single document summarization. They assumed that neighbor documents could provide additional knowledge and more clues. Zha [13] proposed a mutual reinforcement principle to generate summary and key phrases at the same time. A weighted bipartite document graph was built with sentences and terms as nodes. The scores of sentences and terms were estimated based on the principle that a term should have a high salience score if it appears in many sentences with high salience scores, while a sentence should have high salience scores if it contains many terms with high salience scores.

2.2 Sequential Models

Since textual data is a kind of sequential data, some sequential learning approaches, such as Hidden Markov models (HMMs) and conditional random fields (CRFs), [14] have been applied to extract information of interest. HMMs have been applied for relevant passage extraction from search results. He et al. [15] used passages as building blocks and they built a two-state HMM to estimate the likelihoods of passages for the relevant state and the non-relevant state respectively. The output probabilities in each state were assumed to be a Gaussian-distribution scalar. Jiang et al. [16] applied HMM models to extract coherent relevant passages. They tried to estimate the passage boundaries by labeling the relevancy and non-relevancy of single words in the document based on unigram language model. Their method can only extract a single relevant passage from a given document. For text summarization, Conroy and O'leary [17] used HMM to judge the likelihood that a sentence should be contained in the summary. They defined two kinds of states for sentences, i.e. the summary states and non-summary states and defined three sentence features, i.e. position, term number and likelihood of terms in the sentence, for sentence states estimation.

Different from HMM, CRF can integrate other features besides sentence features. Shen et al. [18] treated the document summarization task as a sequence labeling problem and used CRF to label each sentence with either summary sentence or non-summary sentence by combining features like sentence position, length and likelihood of a sentence generated by a document. Sha and Pereira [19] used CRF for noun phrase segmentation using shallow parsing features like the word, position and Part-of-Speech (POS).

Our task bears some similarities to the relevant topics described above. However, there are several major differences. Firstly, different from the techniques like text summarization to organize search results, we introduce a different way to extract contents

from a document with respect to a certain aspect described by a user. For example, the topic of a product review mainly refers to the customers' opinions about the product. The example aspect of a topic in product reviews is opinion reasons or arguments that support the opinions. A summarization of product reviews is to extract sentences to represent the overview opinion in the reviews. It may not include the contents of opinion reason aspect. Secondly, in order to select the sentences, most graph-based approaches use cosine similarity to model the relations between two sentences. However, the cosine similarity measure may not sufficiently reflect relations biased to a certain aspect. Here, we intend to learn the word strength (weight) for each word in expressing messages with respect to a certain aspect at the sentence level. In addition, based on the word weights, we present our study on building up a semantic graph to rank contents sensitive to the aspect of a topic.

3. DEFINITION

In our study, we would allow users to select some sample sentences to indicate the aspect of a topic that they are interested in in a text collection. Based on the aspect, our goal is to automatically extract the contents that are not only related to the topic in the document but also biased to a specified aspect. In this Section, we formally define the key concepts used in our study.

Definition 1 (Context): A context *C* in our study refers to a set of sentences representing the kind of contents that a user is interested in. Figure 1 shows an example of motivational aspect contents selected by a user from a patent. It indicates that the user tends to know the motivations, i.e. the questions or the limitations in the previous design.

[Patent: 5,685,074] Method of forming an inkjet printhead with trench and backward peninsulas
(A) However, due to the bending of the nozzle member, the resulting TAB head assembly has nozzles which are skewed with respect to the substrate causing ink trajectory errors.
(B) When the TAB head assembly is scanned across a recording medium the ink trajectory errors will affect the location of printed dots and thus affect the quality of printing.
(C) Additionally, a problem which occasionally manifests itself in inkjet printheads is that of a blockage occurring in an ink feed channel.

Figure 1. An example of motivational aspect in a patent.

Definition 2 (Topic): A topic *T* is a set of terms charactering a theme in a given document. It can be a document title or any object discussed in the document. For example, a patent with the title "high density nozzle array for inkjet printhead" indicates that its main topic is to introduce a high density nozzle to improve the performance of inkjet printhead.

Definition 3 (Aspect): An aspect *A* refers to the angle from which the topic is discussed. It can be the motivational reasons, purposes or cause-effect in academic articles or opinion reasons of product reviews. For example, a patent with the topic "high density nozzle array for inkjet printhead" can introduce several aspects of the topic, like the motivational aspect of why the authors focused on the design of density nozzle, the aspect of methods used to design such a nozzle and the aspect of effects in applying the methods for such a nozzle. In our study, the aspect is implicit and is embedded in the context given by a user. It allows a user to use examples of contexts to indicate what kind of information they desire. In our approach, we try to learn the patterns of the context and extract the contents with respect to the corresponding aspect from relevant documents in a text collection.

Definition 4 (Topic-sensitive contents): The topic-sensitive contents are a set of sentences *S* from a document *d* that are aligned to the aspect of topic described in the context.

Our task is different from the query-based text summarization. For example, given a keyword query like "inkjet printer", the query-based summarization may give an overview about the inkjet printer design in the document, which may not be biased to the motivational aspect of the design. Our task can help to supplement the general summarization to provide deeper information sensitive to a certain aspect of the topic and provide valuable context-dependent information for different applications, such as question-answering system and opinion reason summarization.

In this study, we assume that the desired contents are included in the documents and the documents in the text collection refer to relevant information, such as the search results or texts in the same domain from the same source type such as product reviews, news and patent documents. It is because that the writing styles, such as the document length and words used, can be various from source to source.

4. EXTRACTING TOPIC-SENSITIVE CONTENTS

4.1 Overview

In this paper, we intend to utilize the structure information in a document to rank the sentences with respect to the aspect in the context. We attempt a different way to build up the sentence graph based on the word weights in expressing messages towards the specific aspect of topic.

Our first assumption is that in a particular source data type, such as product reviews and patents, there are some patterns of word usages in elaborating a certain aspect of a topic. For example, as shown in Figure 2, to describe the motivational reason aspect of an innovation, disadvantages of the previous studies are often discussed. In such a case, the terms like "limitation", "lack" and "unable" are often used to describe the motivational aspect of the topic. It suggests that the capabilities of words to indicate the aspect of a topic are sensitive to the contents. Another consideration is that even in the same domain the same words in the document may still have different abilities to indicate the relevancy of a sentence to the aspect of the topic. For example, although sentences (E) and (F) in Figure 2 contain the term "disadvantage", sentence (E) servers as a transitional sentence and it is not related to the motivational aspect of the topic. It suggests that it would be better to tread each word as a part in the sequence other than to consider the word individually.

(D) Unfortunately, however, this type of technology is expensive and often is unable to detect the extremely small drops of ink used in inkjet printing systems with photographic image quality.
(E) The prior art inkjet print cartridges include a number of disadvantages.
(F) A primary disadvantage of this technique, however, is that increasing printhead size increases the cost of the printing system.
(G) Three ink filters 36, 38 and 40 are mounted within the compartments 12, 14 and 16, respectively.
(H) The printhead temperature is determined by several means.

Figure 2. Sentence examples in patent documents.

Based on the considerations above, the basic idea of our approach is to learn the language patterns using the contexts specified by a user and to make use of the word sequential information and sentence structure information. In our semantic graph model, we first estimate the initial sentence scores to measure how strongly a sentence can deliver the sense with respect to the aspect. We transform estimating the sentence scores to estimate the word probabilities of expressing the semantic information with respect to a specific aspect of a topic. We exploit a sequential language model based on HMM. Based on the term weighting scheme, the sentence relationships in the semantic graph is estimated. Then using the semantic sentence graph, we modify the graph-based ranking process to prioritize the sentences sensitive to the aspect of context specified by the user.

4.2 Semantic Graph Model for Topic-Sensitive Contents Extraction

In order to model a semantic graph for topic sensitive contents extraction, we make two assumptions. They are: (1) if a sentence contains more words that can convey the messages with respect to the specific aspect, it is likely to have higher scores; (2) if the sentences share more similarities in delivering the message of the aspect, they tend to have similar ranking scores. Given an input document, we first measure words in each sentence on how strongly they can convey the messages with respect to the specific aspect. Then we model the sentence similarity based on the word weights to rank sentences for topic-sensitive contents extraction.

4.2.1 Sequential language model for word sense weighting based on HMM

In our case, we assume that the same words in different sentences may have different abilities to represent a certain aspect of the topic. We assume that the terms in the topic-sensitive contents are likely to be produced by language models different from the language model to generate other fragments of the document. Based on this observation, we define four groups (states) to represent words in the documents. They are Q_S, Q_U, Q_{T1} and Q_{T2}, i.e. topic-sensitive related group Q_S, unrelated group Q_U and two transitional groups Q_{T1} and Q_{T2} (as shown in Figure 3). It means that even in the same document, the same words may have different probabilities (or weights) belonging to different states.

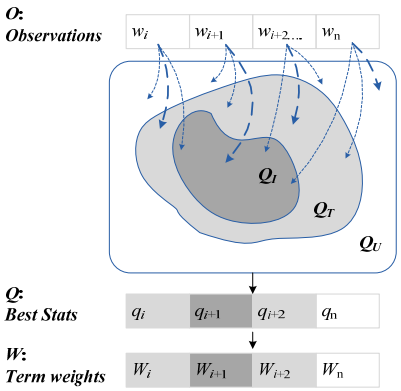

Figure 3. The word states and weights using HMM.

We model a given document $d=(w_1, w_2, \ldots, w_n)$ in the text collection as a word sequence that can be generated from four language models θ_S, θ_U, θ_{T1} and θ_{T2}, i.e. the topic-sensitive language model θ_S, the irrelevant language model θ_U, and two transitional language models θ_{T1} and θ_{T2}. The topic-sensitive language model θ_S generates the topic-sensitive text fragment. The irrelevant language model θ_U produces the text fragments that are not related to the aspect of the topic. One example is sentence (G) in Figure 2. It is a description of the ink filters location, and it contains no term to indicate that this sentence is relevant to the motivational aspect of the topic. The two transitional language models θ_{T1} and θ_{T2} generate the transitional fragments that have high uncertainty to be included in either Q_S or Q_U. One example sentence (H) in Figure 2 is a sentence of transition, which may be followed by a sentence that introduces the limitations of those means to determine the printhead temperature, or a sentence that just mentions the name of those means.

In addition, we treat a document as a sequence of words that are generated by different language models. We exploit hidden Markov model (HMM) to predict the state of a word in a sentence and estimate its corresponding weight of being in the state. Given a document d, we want to find the state sequences Q^* that generates each word of a sentence in d with the highest probability, and the word weights can be accordingly determined by the word probabilities.

$$Q^* = \arg\max_{Q=Q_1Q_2\ldots Q_n} p(Q_1)\,p(w_1 \mid Q_1)\prod_{i=1}^{n-1} p(Q_{i+1} \mid Q_i)\,p(w_{i+1} \mid Q_{i+1})$$

$p(Q_1)$ denotes the initial probability of state Q_1, $p(w_i|Q_i)$ denotes the output probability of word w_i at state Q_i, and $p(Q_{i+1}|Q_i)$ denotes the transition probability from state Q_i to state Q_{i+1}.

HMMs have been used for relevant passage retrieval. For example, Jiang et al. applied HMMs to estimate the boundary of a relevant passage by estimating words with either "relevant" state or "background" state in the document. The difference of our approach is that we define four states to model the word sequences, especially we include the two transitional states to model the uncertainty of tagging process. Another difference is that in the parameter estimation, we not only use the term probabilities, we also consider several ways of estimating output probabilities and sentence relationships to build the semantic graphs. We will detail these issues in Section 4.3.

4.2.2 Sentence similarity and sentence ranking

Based on the HMM to estimate the state for each word and use the corresponding output probability as word weight, we can build up a semantic graph for topic-sensitive contents ranking. We now discuss several different ways of estimating how close two sentences x_i and x_j are in terms of semantic meanings they can share. The sentence similarities are measured by combining the output probabilities generated by model θ_S on both sentences.

R1) Maximum weight strategy

The maximum weight strategy is to measure sentence similarity based on the maximum output probability of words that are labeled as Q_S in the sentences.

$$sim(x_i, x_j) = \max_{w_i \in x_i}\left\{p(w_i \mid q_i = Q_S)\right\} + \max_{w_j \in x_j}\left\{p(w_j \mid q_j = Q_S)\right\}$$

where x_i and x_j denote two sentences, and $p(w_t|q_t=Q_S)$ is the output probability of the word w_t when it is labeled as Q_S.

R2) Average weight strategy

In this strategy, we estimate the sentence similarity based on the average output probability of words that are tagged as Q_S.

$$sim(x_i, x_j) = \frac{1}{N_i} \sum_{w_i \in x_i} p(w_i | q_i = Q_S) + \frac{1}{N_j} \sum_{w_j \in x_j} p(w_j | q_j = Q_S)$$

where N_i is the number of words that are tagged as Q_S in x_i.

R3) Normalization strategy

The normalization strategy takes into account the sentence length when measuring the sentence relationship. $|x_i|$ is the length of sentence x_i.

$$sim(x_i, x_j) = \frac{1}{|x_i|} \sum_{w_i \in x_i} p(w_i | q_i = Q_S) + \frac{1}{|x_j|} \sum_{w_j \in x_j} p(w_j | q_j = Q_S)$$

R4) Different-*f* strategy

A more flexible alternative strategy is to define different functions f_α to capture proximity of sentence similarity based on the values of output probabilities.

$$sim(x_i, x_j) = \sum_{w_i \in x_i} f_\alpha\left(p(w_i | q_i = Q_S)\right) + \sum_{w_j \in x_j} f_\alpha\left(p(w_j | q_j = Q_S)\right)$$

where f_α is a score function for measuring how strongly a sentence is relevant to the specific aspect of a topic by leverage the output probabilities of words that are tagged as Q_S in sentences. In this paper, we consider to use self-information as the score function:

$$f_\alpha\left(p(w_i | q_i = Q_S)\right) = \frac{-\log\left(p(w_i | q_i = Q_S)\right)}{|x_i|}$$

Different strategies of measuring sentence relationships lead to different semantic sentence graphs. Given a document d which is segmented into sentences $X=\{x_0, x_1, \ldots, x_n\}$. We define the semantic graph $G(X, \boldsymbol{W})$ with sentences in X as nodes. The link weights $\boldsymbol{W}$ indicates the sentence similarity, where $w(x_i, x_j) = sim(x_i, x_j)$. We let $w_{ii} = 0$ to avoid loops in the graph in the later steps. The similarity matrix $\boldsymbol{W}$ is symmetrically normalized by $\boldsymbol{M} = \boldsymbol{L}^{-1/2}\boldsymbol{W}\boldsymbol{L}^{-1/2}$. $\boldsymbol{L}$ is the diagonal matrix where l_{ii} is equal to the sum of the ith row of $\boldsymbol{W}$. Based on the semantic sentence graph, we modify the manifold-ranking method [20] to rank the sentences for topic-sensitive contents extraction by integrating word weights. In our semantic sentence graph, we also introduce a vector $\boldsymbol{y} = [y_0, y_1, \ldots, y_n]^T$ where we define y_i to represent the assumption (1). For each sentence x_i, we initialize y_i as the summation of the word output probabilities if the words are labeled as Q_S state; otherwise, $y_i = 0$. We denote a vector $\boldsymbol{f} = [f_0, f_1, \ldots, f_n]^T$, where f_i is the ranking score for sentence x_i, and $\boldsymbol{f}(r)$ represents the sentence score vector $\boldsymbol{f}$ in the rth iteration. For the initial iteration, we set $f_i(0) = 1/n$.

$$\boldsymbol{f}(r+1) = \eta \boldsymbol{M} \boldsymbol{f}(r) + (1-\eta)\boldsymbol{y}$$

Then the ranking process literately updates each f_i at each iteration r based on the ranking function. η is the coefficient ($0 \le \eta \le 1$). The ranking process will not stop until the sum difference of $\boldsymbol{f}$ between two successive iterations is lower than a given threshold 0.001 in our study. Finally, the sentences are sorted based on their final scores $\boldsymbol{f}^* = [f_0^*, f_1^*, \ldots, f_n^*]^T$.

4.3 Parameter Estimation

4.3.1 Four-state HMM structure construction

In order to discover the topic-sensitive contents, we first construct a four-state HMM structure as shown in Figure 4. Our intuition comes from the observation on how humans find contents related to a certain aspect of a topic in a document as a sequence-labeling problem. To find contents related to a certain aspect of a topic, such as motivational reasons and opinion reasons, we have some expectations of words that may be used to describe the certain aspect of a topic, such as drawbacks and negative sentiments. Then we have to scan each word from the beginning to the end and judge the probabilities of the words and the sentences matched with the expectations. Therefore, to model the difficulties of judging whether the word matches with the reader's expectation or not, we introduce two transitional states Q_{T1} and Q_{T2} to indicate the uncertainty of judgment. We design two transition states because we assume that a word in a sentence may have high probability close to Q_S group than to Q_U group.

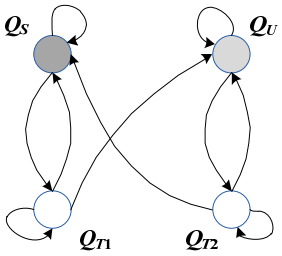

Figure 4. The proposed structure of HMM.

4.3.2 Output probabilities estimated at sentence level

Based on the HMM structure, we then estimate the parameters of the HMM in order to find the most likely state sequence of observed word sequence. The output probability of a term at each state is estimated by using the training sample data C. In the first place, the training data gives some clues to indicate words' relevance with respect to a specific aspect of a topic. The documents in C can be divided into two sentence sets, i.e. the relevant sentence set C_S which includes the sentences annotated as the desired contents and the irrelevant sentence set $C_{S'}$ which contains the sentences that are not selected by a user. Using the training data, we can get four fundamental elements at sentence level for each term as shown in Table 1. We have introduced similar elements at document level for text classification [21].

Table 1. The fundamental elements for each term obtained at sentence level

	C_S	$C_{S'}$
w_i	a_i	b_i
$\overline{w}_i$	c_i	d_i

a_i denotes the number of sentences in the relevant sentence set C_S (i.e. the sentences are annotated by a user) where the term w_i occurs at least once; b_i denotes the number of sentences in the irrelevant sentence set $C_{S'}$ (i.e. the sentences are not selected) where term w_i occurs at least once; c_i denotes the number of sentences in C_S where the term w_i does not occur; d_i denotes the number of sentences in $C_{S'}$ where the term w_i does not occur.

By using the fundamental elements of each term, the output probabilities for each term are specified by the corresponding language models. The output probabilities at Q_S are specified by the topic-sensitive language model θ_S. θ_U is used to specify the output probabilities at Q_U state. We define two strategies to represent the θ_S and θ_U.

O1) Basic strategy for θ_S and θ_U

In the basic strategy, the basic intuition is that if a term w_i occurs often in C_S, i.e. a large value of a_i, it suggests that w_i may have high relevance with respect to the specific aspect and may have high output probability at Q_S state. Similarly, a large value of b_i suggests high output probability at Q_U state.

$$p(w_i|Q_S) = p_b(w_i|\theta_S) = \frac{1}{Z_S}\frac{c(w_i, C_S)}{\sum_{j=1}^{M} c(w_j, C_S)} = \frac{1}{Z_S}\frac{a_i}{\sum_{j=1}^{M} a_j}$$

where $c(w_i, C_S)$ is the number of sentences that word w_i appears in C_S and Z_S is the normalization factor.

$$p(w_i|Q_U) = p_b(w_i|\theta_U) = \frac{1}{Z_{S'}}\frac{c(w_i, C_{S'})}{\sum_{j=1}^{M} c(w_j, C_{S'})} = \frac{1}{Z_{S'}}\frac{b_i}{\sum_{j=1}^{M} b_i}$$

where $c(w_i, C_{S'})$ is the number of sentences that word w_i appears in $C_{S'}$ and $Z_{S'}$ is the normalization factor.

O2) Term distribution strategy

In this strategy, we take the term distribution into account to estimate θ_S and θ_U. It is based on our assumption that although some topic terms have high value of a_i, they may provide limited information to indicate the desired contents. For example, in the topic of "inkjet print design", the topic terms like "inkjet" and "ink flow" may have high frequency in documents. By integrating the term distribution based on Gaussian distribution, θ_S and θ_U are estimated as follows:

$$p(w_i|Q_S) = p_b(w_i|\theta_S)\, p_S(c(w_i, d)),$$

$$p_S(c(w_i, d)) = \frac{1}{\sigma_d\sqrt{2\pi}} e^{\frac{-(c(w_i,d)-\mu_d)^2}{2\sigma_d^2}}$$

$$p(w_i|Q_U) = p_b(w_i|\theta_U)\, p_C(c(w_i, d)),$$

$$p_C(c(w_i, d)) = \Phi\left(\frac{c(w_i, d) - \mu_d}{\sigma_d}\right)$$

p_S is a function to model the term distribution based on term frequency, which is a normal distribution with mean μ_d and standard deviation σ_d of term frequency in document d. p_C is the cumulative distribution function of normal distribution on term frequency.

As for the two transitional language models θ_{T1} and θ_{T2}, we define different ways to represent the uncertainty of a word w_i containing topic-sensitive meaning or not. We estimate the output probabilities at states Q_{T1} and Q_{T2} using the elements of terms as shown in Table 1.

T1) Basic strategy for transitional models

In the basic strategy, we use the value of a_i and b_i to estimate the transitional language models

$$p(w_i|Q_{T1}) = p_b(w_i|\theta_{T1}) = \frac{1}{Z_{T1}}\frac{c(w_i, C_S)}{c(w_i, C_S) + c(w_i, C_{S'})} = \frac{1}{Z_{T1}}\frac{a_i}{a_i + b_i}$$

$$p(w_i|Q_{T2}) = p_b(w_i|\theta_{T2}) = \frac{1}{Z_{T2}}\frac{c(w_i, C_{S'})}{c(w_i, C_S) + c(w_i, C_{S'})} = \frac{1}{Z_{T2}}\frac{b_i}{a_i + b_i}$$

The value of $a_i/(a_i+b_i)$ indicates that a term with a higher value of $a_i/(a_i+b_i)$ is likely to be the topic terms since it appears often in C_S. The value of $b_i/(a_i+b_i)$ suggests that if a term is used in other aspects of the topic, then the value of $b_i/(a_i+b_i)$ tends to be higher.

T2) Ratio of *a* and *b* strategy

The second strategy to estimate the transitional language model is to use the ration between a_i and b_i.

$$p(w_i|Q_{T1}) = p(w_i|\theta_{T1}) = \frac{1}{Z_{T1}}\frac{c(w_i, C_S)}{c(w_i, C_{S'})} = \frac{1}{Z_{T1}}\frac{a_i}{b_i}$$

$$p(w_i|Q_{T2}) = p(w_i|\theta_{T2}) = \frac{1}{Z_{T2}}\frac{c(w_i, C_{S'})}{c(w_i, C_S)} = \frac{1}{Z_{T2}}\frac{b_i}{a_i}$$

T3) Ratio of *a*, *b* and *c* strategy

In this strategy, we introduce the ratio of a_i and c_i. We assume that given two terms w_i and w_j, the term with high value of a_i/c_i may have high probability to be the topic terms.

$$p(w_i|Q_{T1}) = p(w_i|\theta_{T1}) = \frac{1}{Z_{T1}}\frac{c(w_i, C_S)}{c(\overline{w}_i, C_S)} = \frac{1}{Z_{T1}}\frac{a_i}{c_i}$$

$$p(w_i|Q_{T2}) = p(w_i|\theta_{T2}) = \frac{1}{Z_{T2}}\frac{c(w_i, C_{S'})}{c(w_i, C_S)} = \frac{1}{Z_{T2}}\frac{b_i}{a_i}$$

T4) Combining term distribution strategy

In this strategy, we also consider the term distribution in the estimation of the transitional language model by introducing p_C.

$$p'(w_i|\theta_{T1}) = p_b(w_i|\theta_{T1})\, p_C(c(w_i, d))$$

$$p'(w_i|\theta_{T2}) = p_b(w_i|\theta_{T2})\, p_C(c(w_i, d))$$

4.3.3 Transition probability

When we estimate the output probabilities from the sample data, we can learn the transition probabilities of this HMM from the observed sequences using Baum-Welch algorithm. Because we focus on learning the language patterns, especially the term usages in delivering messages towards the topic-sensitive contents, the transition probabilities can be learned from the training data. We set the initial transition probabilities at each state, and then we use the training data as the observed sequences to learn the transition probabilities.

5. EXPERIMMENTS AND DISCUSSIONS

5.1 Experiment Setup

The approach we proposed can be applied to different domains. In this Section, we conducted the experiments on relevant documents of a given topic. We made this assumption because the focus here is to extract topic-sensitive contents for a relevant text collection, but not to study the retrieval performance. In this experiment, we create a scenario in engineering design domain that designers intend to extract information about motivational reasons of "inkjet printer design" from patent documents. Analyzing the contents of motivational reason aspect can help designers to understand the major issues on the topic of "inkjet printer design" and it in return help designers in design knowledge reuse, design decision-making and design innovation.

We randomly collected three hundred patent documents relevant to the topic of "inkjet printer design" from the United States patent database as our research data. The motivational reasons in our dataset were manually annotated. We investigated the profile of the dataset. On average, only 3.5% of a patent document, i.e. about 250 words per document average, is marked as contents related to the motivational reason aspect. With respect to the document length, each document has 8550 words in average and

257 sentences per document average. In terms of sentence length, the average sentence length is 33.41 words. In addition, due to the writing style of patents, 97% of the documents contain one or more sentences with more than 100 words. These sentences are relative long compared with sentences in other resources like academic journal articles.

In order to evaluate the results, we use ROUGE-1 measurement [22]. It matches the unigram co-occurrences between the systems generated results and the human annotation data in terms of *precision, recall* and *F* value. ROUGE-1 is used since it has been shown to agree with human judgment most in evaluating the machine generating text segment [22].

5.2 Experiment Results

5.2.1 Comparisons with relevant approaches

In the first experiment, since our task is a different problem related to information extraction and there is no report of performance on this problem, we implement several baseline methods for comparison. The first baseline, *BL-s*, is a simple baseline method that takes the first *k*-word long segment from a document as the result. The second method, *BL-cosine similarity*, is a graph-based approach based on manifold-ranking algorithm, in which the sentence similarity is calculated using cosine similarity based on the vector space model and the vector $\boldsymbol{y}$ is not defined. The third method, *BL-passage extraction*, is a relevant passage extraction approach using HMM to estimate the passage boundary [16]. In our approach, to estimate the word states in each sentence, we used the basic strategies, i.e. (O1+T1), to compute the output probabilities at each state. To build up our semantic graph, we applied the maximum weight strategy, i.e. R1, to model the relationships between sentences. In addition, we tuned to use paragraphs as nodes in the semantic sentence graph. It is because we intend to minimize the bias that may be generated by the large variance of sentence length in patent documents.

We first conducted the five-fold cross-validation for the extraction of motivational aspect contents from patents using the methods in Table 2. The 300 patents were evenly divided into five groups. Each time, we used four groups as the sample documents in which the user selected some sample sentences to represent the motivational aspect. One group data was used as test data. In testing, the sentences in each document were ranked by the relevant approaches. Then top sentences were selected as the topic-sensitive contents until it reached *k* words. We set $k = 250$ because it matches with the average number of words in the human annotation data. The results were the average results of the five-time experiment.

Table 2. The Rouge-1 results for topic-sensitive contents extraction with respect to motivational reason aspect

	Average recall	**Average precision**	**Average *F* value**
Semantic graph	0.9225	0.5211	0.6459
BL-passage extraction	0.4442	0.2391	0.3033
BL-cosine similarity	0.5575	0.3102	0.3899
BL-s	0.4435	0.2561	0.3170

Table 2 reports the average ROUGE-1 scores of four approaches. Our first observation is that the proposed semantic graph outperformed the other relevant approaches. The overall performance of our approach in *F* value is 0.6459. It is about more than 20% higher than the other three approaches. Secondly, we observe that the two graph-based approaches, i.e. *semantic graph* and *BL-cosine similarity*, can generate better results compared with *BL-passage extraction* using sequential model HMM. In terms of *F* value, the two graph-based approaches obtained more than 7% higher than the *BL-passage extraction*'s 0.3033. When we switch to precision performance, it shows that the graph-based approaches have more than 10% higher than the *BL-passage extraction*'s 0.2391. It reveals that using the structure information of document by a graph can help to select sentences more relevant to the topic-sensitive contents.

Thirdly, when we compared the two graph-based approaches, i.e. *semantic graph* and *BL-cosine similarity* in Table 2, we observe that our semantic graph approach can generate much better results than *the BL-cosine similarity*. The *F* value of our approach, 0.6459, is about 25% higher than *the BL-cosine similarity*'s 0.3899. In terms of average recall and precision performance, our approach obtained 0.9225 and 0.5211 respectively, which are about 35% and 20% higher than the *BL-cosine similarity*'s 0.5575 and 0.3102 respectively. It shows that the our ways to learn the language patterns, especially on estimating the ability of words to deliver messages with respect to a certain aspect, can better help to leverage the information between words and sentences for topic-sensitive contents extraction. The estimated term weights can better help to reflect the relationships between sentences compared with the cosine similarity used in *BL-cosine similarity*.

When we compared our approach and *BL-passage extraction,* which both attempt to use HMM, the results of our approach is about 34% higher than *BL-passage extraction*'s 0.3033 in terms of *F* value. It indicates that our ways of introducing four states in the HMM structure, i.e. topic-sensitive state, unrelated state and two transitional states, can better estimate and model word weights with respect to a certain aspect of a topic.

5.2.2 Different strategies for sentence relationships in the semantic graph

In the second experiment, we investigated the performance of different strategies discussed in Section 4.2.2 to model the sentence relationships in the semantic graph. For comparison purposes, the four sentence relationships, i.e. R1-R4, were implemented based on using the same methods to estimate the output probabilities, i.e. the basic strategies (O1+T1).

Table 3. The Rouge-1 value for topic-sensitive contents extraction with different sentence relationship measurement

Sentence relationship measurement	Average recall	Average precision	Average *F* value
R1 Maximum weight strategy	0.9225	0.5211	0.6459
R2 Average weight strategy	0.4954	0.2624	0.3348
R3 Normalization strategy	0.4667	0.2665	0.3315
R4 Different-*f* strategy	0.5732	0.3345	0.4093

Table 3 shows the results of different strategies for sentence relationships estimation in the semantic graph for topic-sensitive contents extraction. Our first observation is that the maximum weight strategy outperformed the other three strategies. The *F* value of maximum weight strategy is 0.6459, which is about 31% better than average weight strategy's 0.3348, about 31% better than normalization strategy's 0.3315 and around 24% higher than

different-*f* strategy's 0.4093. Besides the maximum weight strategy, the other three strategies are more or less by integrating the weights of multiple words that are labeled as Q_S. Since we estimated the word states at the sentence level, if a sentence contains more words, it is possible that more words in the sentence are tagged as Q_S. However, since the state estimation process using HMM cannot always guarantee perfect results, it may have high possibility to obtain wrong states if the sentence has more words.

5.2.3 Different strategies for output probabilities

The third experiment evaluates different strategies as discussed in Section 4.3 to estimate the output probabilities for term weightings. In the experiment, we set to use different-*f* strategy as the sentence relationship, since it utilizes the output probabilities of multiple words and it would be better to compare the performance among different output probabilities.

Table 4. The Rouge-1 value for topic-sensitive contents extraction with different output probabilities

Strategies for output probability estimation	Average recall	Average precision	Average *F* value
(O1+T1)	0.5732	0.3345	0.4093
(O1+T2)	0.5345	0.2470	0.3379
(O1+T3)	1.0000	0.4640	0.6339
(O2+T4)	0.4882	0.2384	0.3039

Table 4 shows the results of different strategies for output probabilities estimation. We first observe that (O1+T3) generated the best performance compared with other three strategies to estimate the output probabilities. In *F* value, it produces 0.6339, which is about 22% higher than (O1+T1)'s 0.4093, about 29% better than (O1+T2)'s 0.3379 and about 33% higher that (O2+T4)'s 0.3039. When we compare (O1+T3) with (O1+T1) and (O1+T2), which used O1 to estimate the output probabilities in topic-sensitive state and unrelated state, it indicates that different ways of estimating the output probabilities for the transitional states can affect the results significantly. In (O1+T3), by leveraging the ratio of a_i, b_i and c_i, the ranking process can help to generate the contents that cover most of words in the annotation results. Its average recall performance is about 40% higher than the other three strategies. Although (O2+T4) combines the term distribution, it cannot help to improve the performance. It may need to analyze how other information, such as term distribution and term position information, can be integrated into the output probability estimation for sequential language modeling.

5.2.4 Performance with different lengths of contents

In the fourth experiment, we compared the performance of our approaches and other relevant methods shown in Table 2 by selecting different length of segments to form topic-sensitive contents.

Figure 5 shows the average *F* values of the four approaches in Table 2 by selecting different lengths of segments as topic-sensitive contents. Our first observation is that as the length of segments selected increases, our approach can help to extract contents that are relevant to the motivational reasons. Although the four approaches obtain comparable results at the beginning when the length is set 100 words, our approach is able to continue select sentences that are more relevant to the motivational reason aspect as the length increases.

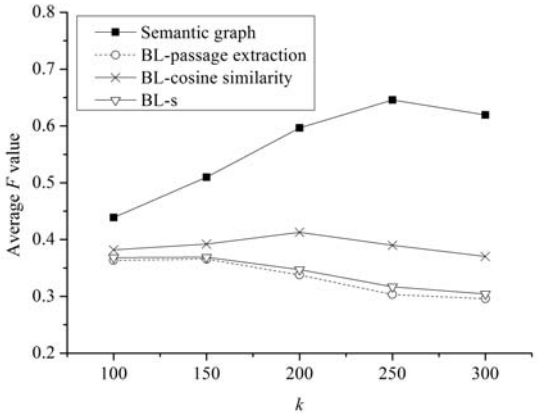

Figure 5. The average *F* value with different lengths of segments selected.

Figure 6 shows the average precision values with respect to different lengths of segments selected as topic-sensitive contents. We observe that in the same lengths of segments, our approach can achieve higher precision performance compared with other three relevant approaches. It indicates that the segments that were selected by our approaches are more relevant to the motivational reason contents compared with other three approaches. As the length increases, it shows that our approach can produce stable average precision performance compared with other three approaches. It suggests that our approach can harvest the relevant segments at the earlier stage and the topic-sensitive contents extracted by our approach match well with the human annotated contents.

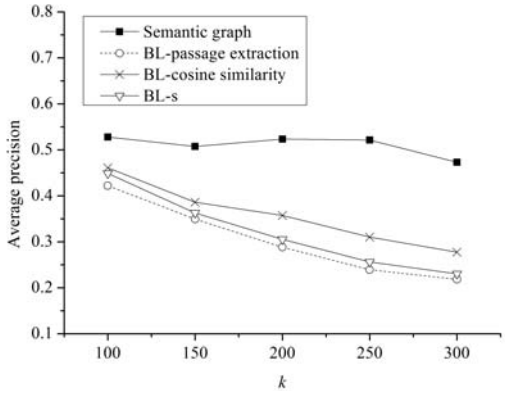

Figure 6. The average precision with different lengths of segments selected.

6. CONCLUSIONS

In this paper, we have studied a problem of extracting relevant contents with respect to certain aspects of a topic where such aspects are annotated by an end user. We name this topic-sensitive content extraction. In our work, we would allow a user to describe the desired aspects of a topic by selecting some sample sentences in the documents flexibly and in an ad hoc manner. By designing a semantic graph model to learn the language patterns from the sample sentences highlighted, we identify the contents biased with respect to certain aspects. To build this semantic graph model, we first exploit a sequential

language model based on hidden Markov model to estimate the word weights which indict their inclination towards different aspects. Then the semantic graph model is built up using such weights for sentences ranking. Our experiments of extracting contents closely associated with the motivational aspect in patent documents demonstrate that the proposed approach is able to generate better results compared to baseline approaches. In the future, we plan to explore other factors, e.g. term distribution and term position information, to better refine language modeling in graph formation so that the performance of content extraction can be further lifted.

7. ACKNOWLEDGMENTS

The work described in this paper was supported by a grant at the National University of Singapore (R265-000-362-133).

8. REFERENCES

[1] B. Pang and L. Lee. Opinion Mining and Sentiment Analysis. *Foundations and Trends in Information Retrieval*, 2 (1-2): 1-135, 2008.

[2] X. Ding, B. Liu and P.S. Yu. A holistic lexicon-based approach to opinion mining *Proceedings of the international conference on Web search and web data mining*, ACM, Palo Alto, California, USA, 2008.

[3] S.J. Athenikos and H. Han. Biomedical question answering: A survey. *Computer methods and programs in biomedicine*, 99 (1): 1-24, 2010.

[4] L. Getoor and C.P. Diehl. Link mining: a survey. *SIGKDD Explor. Newsl.*, 7 (2): 3-12, 2005.

[5] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30 (1-7): 107-117, 1998.

[6] J.M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46 (5): 604-632, 1999.

[7] H. Hassan, A. Hassan and O. Emam. Unsupervised information extraction approach using graph mutual reinforcement *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Sydney, Australia, 2006.

[8] J. Chen, D. Ji, C.L. Tan and Z. Niu. Relation extraction using label propagation based semi-supervised learning *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Sydney, Australia, 2006.

[9] S.J. Karen. Automatic summarising: The state of the art. *Information Processing & Management*, 43 (6): 1449-1481, 2007.

[10] R. Mihalcea and P. Tarau, TextRank: Bringing Order into Texts. in *Conference on Empirical Methods in Natural Language Processing*, (2004).

[11] G. Erkan and D.R. Radev. LexRank: Graph-based Lexical Centrality as Salience in Text Summarization *Journal of Artificial Intelligence Research*, 22, 2004.

[12] X. Wan and J. Xiao. Exploiting neighborhood knowledge for single document summarization and keyphrase extraction. *ACM Trans. Inf. Syst.*, 28 (2): 1-34, 2010.

[13] H. Zha. Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, Tampere, Finland, 2002.

[14] J. Lafferty, A. McCallum and F. Pereira, Conditional random fields: Probabilistic models for segmenting and labeling sequence data. in *the Eighteenth International Conference on Machine Learning*, (2001), 282-289.

[15] D. He, D. Demner-fushman, D.W. Oard, D. Karakos and S. Khudanpur, Improving passage retrieval using interactive elicitation and statistical modeling. . in *the 13th Text REtrieval Conference*, (2004).

[16] J. Jiang and C. Zhai. Extraction of coherent relevant passages using hidden Markov models. *ACM Trans. Inf. Syst.*, 24 (3): 295-319, 2006.

[17] J.M. Conroy and D.P. O'leary. Text summarization via hidden Markov models *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, New Orleans, Louisiana, United States, 2001.

[18] D. Shen, J.-T. Sun, H. Li, Q. Yang and Z. Chen. Document summarization using conditional random fields *Proceedings of the 20th international joint conference on Artifical intelligence*, Morgan Kaufmann Publishers Inc., Hyderabad, India, 2007.

[19] F. Sha and F. Pereira. Shallow parsing with conditional random fields *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, Association for Computational Linguistics, Edmonton, Canada, 2003.

[20] D. Zhou, J. Weston, A. Gretton, O. Bousquet and B. Schölkopf, Ranking on data manifolds. in *Advances in Neural Information Processing Systems 16*, (2004), MIT Press.

[21] Y. Liu, H.T. Loh and A. Sun. Imbalanced text classification: A term weighting approach. *Expert Systems with Applications*, 36 (1): 690-701, 2009.

[22] C.-Y. Lin and E. Hovy. Automatic evaluation of summaries using N-gram co-occurrence statistics *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, Edmonton, Canada, 2003, 71-78.