A Novel Contextual Topic Model for Query-focused Multi-document Summarization

Guangbing Yang
School of Computing, University of Eastern Finland
P.O. Box 111 80101 Joensuu, Finland
Email: yguang@cs.joensuu.fi

Abstract—The problem of the oft-decried information overload negatively impacts comprehension of useful information. How to solve this problem has given rise to increase of interest in research on multi-document summarization. With the aim of seeking a new method to help justify the importance and similarity of sentences in multi-document summarization, this study proposes a novel approach based on well-known hierarchical Bayesian topic models. By investigating hierarchical topics and their correlations with respect to the lexical co-occurrences of words, the proposed contextual topic model can determine the relevance of sentences more effectively, and recognize latent topics and arrange them hierarchically as well. The quantitative evaluation results show that this model has outperformed hLDA and LDA in document modeling. In addition, a practical application demonstrates that a summarization system implementing this model can significantly improve the performance of summarization and make it comparable to state-of-the-art summarization systems.

Keywords—Machine learning, hierarchical topic model, text summarization.

I. INTRODUCTION

The information overload problem, negatively impacts the comprehension of useful information. It becomes more serious in the digital age. How to alleviate these problems is of concern to the research on automatic text summarization, especially the extractive-based multi-document summarization. The main task of this type of summarization is to find and extract the most important information from multiple documents and generate a summary. Therefore, finding an appropriate method to justify the importance (or relevance) of a string of text (e.g. a sentence) dominates this research area. Many proposed approaches use either statistical methods or Bayesian language models to produce summaries. Some approaches specify the importance of a sentence in a document by counting term frequency or inverse document frequency (TF-IDF) exclusive of stop words [5], [19]. Others identify the relevance of a sentence by using graph-based approaches [20], or Bayesian topic models in machine learning and language modeling [1], [2], [3], [13], [17]. Particularly, the Bayesian topic models have been widely adapted into the tasks of multi-document summarization due to their continual performance in document clustering and classifications. Nonetheless, most of these topic models construct topics in a flat structure and ignore the hierarchical correlations between topics in many text applications [11]. Only few of them consider the topic correlations for the problem of multi-document summarization. One is the hybrid hierarchical model for multi-document summarization in [1]; another one is the two-tiered topic model (TTM) in [2]. Both approaches use supervised learning algorithms to train the model. Although these approaches demonstrate good performance in the multi-document summarization, they do not consider the lexical co-occurrences that can play a vital role in analysis of topic correlations [11], [14] and word associations in many realistic applications, such as text summarization, speech recognition, syntactic parsing, text mining, and so on [15]. Hence, this research is motivated by a desire to deal with this limitation using a fully unsupervised learning algorithm. A model that can take account of both the lexical co-occurrences and topic hierarchies is supposed to improve the performance of summarization significantly.

In this paper, a novel model using unsupervised learning approach in machine learning is proposed for the problem of multi-document summarization. The idea came from the previous researches [1], [2] that demonstrated the correlations between general and specific topics were important features for evaluating the relevance of sentences. With the benefits of the successful Bayesian based topic modeling in natural language processing and machine learning, this model is established upon the hierarchical Latent Dirichlet Allocation model (hLDA) [11]. In addition to the hierarchical structures, the model integrates the concepts of latent topics into n-gram language models [14], [22], [25] to indicate the lexical co-occurrences. Since both the topic hierarchies and the lexical co-occurrences are related to the context of words in documents, the topics captured with this contextual information are described as contextual topics in this study, and the corresponding model is called contextual topic model. By investigating hierarchical topics and their correlations with respect to the lexical co-occurrences of words, this model is able to determine the relevance of sentences more effectively.

The rest of the paper is organized as follows: Section 2 describes the related Bayesian topic models and applications in multi-document summarization. Section 3 discusses the proposed approach in details. Section 4 describes the experiments conducted in this research to validate the approach and discuss the findings. Section 5 presents concluding remarks.

1082-3409/14 $31.00 © 2014 IEEE
DOI 10.1109/ICTAI.2014.92
II. RELATED WORK

The sumHLDA is a hybrid model that formulates the multi-document summarization as two steps learning problem using hLDA model for pattern discovery and a regression model for inference [1]. Human expert summaries were combined with document clusters, provided by Document Understanding Conferences (DUC), as the training corpus for model learning. Probabilities of unigram and bigram of word items in documents were used as features in a Gaussian kernel based regression model for summary sentence prediction. This approach provided a perfect example of adapting hierarchical latent topic modeling with supervised learning mechanism into the problem of extractive-based multi-document summarization. The system built based on this approach improved the summarization performance significantly. However, taking account of human expert summaries into the process of the model training may restrict the practical utility of the model.

The two-tiered topic model (TTM) proposed recently allocates topics into a hierarchical structure that only takes two levels for the topic hierarchies: the lower level represents the latent topics sampled from words, and the higher level describes the correlations between these topics [2]. To distinguish topic levels, this model classifies the topics into two groups using auxiliary criterion from human expert summaries, which is a nuisance widely required by most practical applications of language models [11]. However, restricting topic layers to only two levels and relying on expert summaries for indicating the topic levels limit the utility of the model in practical applications because latent topics captured from multiple documents may cover many ideas so as not to be easily allocated exactly into two layers. On the other hand, human summaries are not always available, especially for online applications, or of a high enough standard quality to express important information discussed in documents. Moreover, either the TTM or the sumHLDA did not consider the problem of the lexical co-occurrences of words or word associations within the context of multiple sentences or documents. Thus, to deal with these limitations, an alternative approach is necessary in multi-document summarization.

Recently developed Bayesian collection models incorporated the concept of latent topics into n-gram language models, such as the LDA-HMM model integrating topics and syntax [22], structured topic model [14], and topical n-grams [25]. These models have demonstrated excellent results on tasks related to parts-of-speech, document classifications, clustering, information retrieval, and so on. The success of these models suggests that the mechanism of incorporating the concept of latent topics into n-grams is a plausible approach that can help to solve the problems discussed in this paper. Dealing with a query-focused multi-document summarization in practice, a system needs to find a way to take account of user’s query into the process of the summarization. A common and effective approach to bring the user’s information need into the summary processing is to involve a language model as the framework to integrate the features of topic modeling into the sentence ranking. In this case, a relevance-based language model [23] has been adapted as such framework to take the sentences that are most relevant to a user query from candidate sentences, which are generated by the contextual topic model, to produce the final summary. Research in literature has also reported that relevance-based models are better suited to the tasks of summarization [4] and text segmentation [8].

III. CONTEXTUAL TOPIC MODEL AND SUMMARY GENERATION

A. Contextual Topic Model

The purpose of developing this contextual topic model (CTM) is to establish a mechanism to incorporate the hierarchical concepts of latent topics into n-grams to indicate the topic hierarchies with respect to the word associations or lexical co-occurrences. One of the main applications for this model is the multi-document summarization. As we discussed previously, the latent topics with n-gram model [14], [22], [25] evaluated word associations within latent topics, but ignored their hierarchies; the hLDA model justified topic hierarchies, but was short of the consideration for word associations. Therefore, a model that can capture both the hierarchies and word associations over latent topics is deemed to be more useful for improving performance of summarization. In this model, the idea of the original topical n-gram model [25] is extended to include the hierarchical topics for capturing topic hierarchies and word associations. In addition, the model simplifies the sampling process of the probability distributions for the lexical co-occurrence of words by integrating the level distributions with a bigram distribution from a Bernoulli distribution, rather than bring another latent topic distribution for the bigram modeling, which is the approach adapted in the topical n-grams [25].

For a particular extractive-based multi-document summarization problem, the main process can be simply described as two steps: first step is to find out the most important information to the user’s information need by analyzing the content of a set of sentences, and then extract the most relevant sentence(s) to generate a summary based on the results of this analysis. Hence, the problem focuses on determining which sentences are important to information need. The concept of ranking score for a sentence is commonly used in summarization to tackle this problem. From the perspective of machine learning or Bayesian based generative modeling, the process of obtaining the ranking score for a new sentence is a kind of prediction processing, and the process of determining how a sentence is ranked can be treated as a problem of model learning. Here the basic idea to address this problem in the proposed approach
is that the CTM is trained (also called model learning) and validated with the optimum summarization performance, and then this trained model is used to specify which sentences in new documents have potential to be a part of a summary based on their ranking scores. Therefore, the entire process of the summarization consists of two parts: model learning and summary sentence predicting (or generating).

B. Model Learning

First, the model randomly chooses an L-level path through the topic hierarchies; second, it samples the topics distribution \( \Theta \) from an L-dimensional Dirichlet distribution; third, it draws a vector of word association proportions \( \Psi \) from an N-dimensional multiple-Bernoulli distribution (where N is the sentence length); fourth, it samples the word distribution \( \Phi \) in the sentence from a mixture of topics that are associated with nested Chinese restaurant process (nCRP) prior along this path and word association distribution that is used to determine whether the distribution of the observed word is modeled based on an n-gram or a unigram. More formally, the generative process for each sentence \( s \) in the corpus is given as follows:

1) For each topic \( k \in K \), sample \( \phi_k \sim Dirichlet(\beta) \).
2) For each sentence, \( s \in \{1, 2, ..., M\} \):
   - Draw a path \( c_s \sim nCRP(\gamma) \).
   - Draw an L-dimensional topic distribution over levels along with paths in the tree, \( \theta \sim Dir(\alpha) \).
   - Draw a word association distribution, \( \psi \sim MultiBeta(\psi|\delta_0, \delta_1) \).
3) For each word \( w \in \{1, 2, ..., N\} \) in the sentence \( s \):
   - Draw topic \( z \in \{1, 2, ..., L\} \) from \( Multinomial(\theta) \).
   - Draw word association indicator \( x \in \{0, 1\} \) from \( MultiBern(\psi) \).
4) Sample word \( w_i \) from \( p(w_i|z_i, c_s, x_j, \beta_{z_i=1}) \), a multinomial probability conditional on topic \( z_i \) within path \( c_s \) and indicated by word association \( x_j \).

The graphical model for this CTM is shown in Fig.1. In this graphical representation, the dot node denoted as \( c \) represent the scalar hyper-parameter, the circle nodes (outside the rectangle plate) denoted as \( \alpha, \beta \), and \( \delta \) are vector type hyper-parameters. Other circle nodes inside the plates represent random variables, where shaded one indicates that corresponding random variable has been set to its observed and others without shade indicate latent variables or hidden variables. The outer plate represents the identical in the corpus level, where \( M \) is the number of the sentences (or documents) in the corpus; the inner plate indicates the identical occurred in the sentence level, where \( N \) denotes the length of the sentence or the number of words in the sentence. The random variables \( w, c, z, \) and \( x \) \(^1\) are vectors that represent distributions of words, paths, topics (or levels of paths) and word associations in the joint distribution \( p(w, c, x, z; \alpha, \beta, \delta) \). To simplify this expression, one symbol \( U = \{\alpha, \beta, \delta\} \) is used to represent all those hyper-parameters in the joint probability. The generative process is given as follows:

\[
p(w, c, x, z; U) = p(w|z, c, x; U)p(z, c, x), \tag{1}
\]

Here an assumption is about the conditional independency between the distribution of path \( c \) and the distribution of word association \( x \), which is represented as \( c \perp|w \). Thus, the second term in the right side of the Eq. (1) can be expressed as follows:

\[
p(z, c, x) = p(z|c, x)p(c, x) = p(z|c)p(c, x), \tag{2}
\]

Substitute the second term of the Eq. (1) with the Eq. (2), the joint probability can be factorized as a product of two probability distributions listed as follows:

\[
p(w, c, x, z; U) = p(w|z, c, x; U)p(z, c)p(z, x) = p(z, c)p(w|z, c; U_1)p(z, x)p(w|z, x; U_2), \tag{3}
\]

where \( U_1 = \{\alpha, \beta, \gamma\} \) and \( U_2 = \{\alpha, \beta, \delta\} \) are hyper-parameters for these two distributions. The Eq. (3) mathematically verifies this joint distribution can be factorized as two models: one is the hLDA and another one is a mixture model of the topic and word association. The assumption about the independency between the distribution of path \( c \) and the distribution of word association \( x \) can be explained intuitonally from these two aspects: one is that topics assigned to each term of a phrase may be highly allocated in the same topic hierarchical level; another intuition comes from the definition of the independencies of Bayesian Network in the graphic model. In our case, these nodes, which represent variables \( c_j, x_i, \) and \( w_i \), are in a d-separate structure due to \( w_i \) is observable and the observation blocks the communication between nodes \( c_j \) and \( x_i \). This indicates that \( c_j \) and \( x_i \) are independent of each other [9].

C. Probabilistic Inference

The purpose for performing the probabilistic inference is to find the posterior distribution of this CTM by estimating parameters of this model with the training data. The trained model then can be used to predict the outcomes when giving a new input data, which means the trained model can “generate” summaries if given new documents. Because the posterior distribution of this CTM is intractable, an

\(^1\)In this paper, all the variables, like \( w, c, z, \) and \( x \), which are printed as bold font, represent the vectors of variables; the regular font variables with subscripts, like \( w_i, x_j, \) etc., represent an individual value in the corresponding vector.
approximation must be appealed to perform the posterior inference. A Markov chain Monte Carlo (MCMC) algorithm has been used to approximate the posterior of this model. As discussed in previous section, this CTM consists of an hLDA model and a mixture model of topic and word association.

For details about the posterior inference of hLDA, expressed as $p(w, z, c; U_1)$ in Eq. (3), one can refer to the original work of hLDA model [11]. Here the discussion focuses on the second joint probability distribution, expressed as $p(w, z, x; U_2)$ in the Eq. (3). Based on the Bayes’ rule, given the joint probability distribution $p(w, z, x; U_2)$, the posterior distribution for both hidden variables $z$ and $x$ is given as:

$$p(z, x|w; U_2) = \frac{p(w|z, x; U_2)p(z, x)}{\sum_{z,x} p(w|z, x; U_2)p(z, x)} \propto p(w|z, x; U_2)p(z, x), \quad (4)$$

If evaluating the graphical structure of the CTM in Fig.1, one can find that $x$ and $z$ are d-separated given $w$ observed, and thus the $x$ and $z$ are conditional independent given $w$, denoted as $z \perp x|w$ [9]. By taking account of this conditional independency into the joint distribution expressed in Eq. (4), this joint distribution can be factorized as the product of two posterior distributions:

$$p(z, x|w; U_2) \propto p(w|z, x; U_2)p(z, x) = p(w|z; \beta)p(z; \alpha)p(w|x; \beta)p(x; \delta), \quad (5)$$

where $z$, $x$, and $w$ are the vectors of topic allocations, word association assignments, and observed word distribution in the corpus level; $\alpha$, $\beta$, and $\delta$ are hyper-parameters, which can be fixed values that are usually expected according to the data and the limitations of the analysis, or unknown quantities a priori to be inferred from some scalars namely “hyper-hyperparameters” [11]. In this study, the fixed values from experiments are used for simplification and the actual values for these hyper-parameters will be given in the discussion of the experiment conducted in this study.

The particular MCMC algorithm namely the collapsed Gibbs sampling algorithm [18] is employed in this approximation processing. For detailed implementation of the collapsed Gibbs sampling algorithm for hLDA, one can refer to the original work [11]. This article focuses on the approximation of the collapsed Gibbs sampling algorithm for the mixture model of topic and word association.

Consider the generative process discussed previously, given each observed word in a sentence of the corpus, for each of the sampled topic variable $z_{s,i}$ and word association variable $w_{s,i}$ (where the subscript $s \in \{1, 2, ..., M\}$ represents the index of the sentence in the corpus, can be treated as the row number of the matrix $z$ and $x$; the subscript $i \in \{1, 2, ..., N\}$ represents the index of a word in a sentence, which can be seen as the column number of the topic matrix; the superscript with parentheses represents a value assigned to this variable. For example, the notation $z_{s,i} = l$ is the same as $z_{s,i} = l$ represents the topic $z$ is for $i^{th}$ term in sentence $s$, and its topic assignment value is $l$. The same notation will be used for other variables in the expression of the Gibbs sampling later), so the conditional posterior distribution of the topic allocation for $z_{s,i}$ is given as:

$$p(z_{s,i} = l|z_{s,-i}, w) \propto p(w_{s,i}|z_{s,i}, z_{s,-i})p(z_{s,i} = l|z_{s,-i}), \quad (6)$$

where $z_{s,-i}$ and $w_{s,-i}$ are the vectors of topic allocations and observed words leaving out $z_{s,i}$ and $w_{s,i}$ respectively. This equation presents Bayes’ rule, where the first term in the right hand side of the Eq. (6) is the likelihood, and the second is the prior. In this equation, the variable $z_{s,i}$ can take the value from 1 to L (which is the maximum value of the level allocated), and the observed word $w_{s,i}$ can take a value from 1 to V (which is the maximum index value of the vocabulary or the size of the vocabulary). The parameters of the topic variable $z_s$ for the sentence $s$ are $\theta_s$, which is an L-dimensional vector of random variables from a Dirichlet distribution, and the probability density of $\theta_s$ for each element in the vector is $p(\theta_s|\alpha) \sim Dir(\alpha)$ and corresponding topic sampling is given by this multinomial distribution $p(z_s|\theta_s) \sim Mult(\theta_s)$. Thus, each probabilistic value can be expressed as $p(z_{s,i} = l) = \theta_{s,l}$. Note that index of the parameters $\theta_s$ (an L-dimensional vector) is denoted as $l$, is the value of the topic assignment for topic variable $z_{s,l}$. The parameters of the word variables $w$ are $\phi$, which is a V-dimensional vector (index $v=1, \ldots, V$) and each value is a positive real number that is the probability of word $w$ in topic $l$. It is drawn from a Dirichlet distribution where the probabilistic density is given as $p(\phi_v|\beta) \sim Dir(\beta)$. The corresponding word distribution is drawn from the multinomial distribution, which is $p(\phi_v|l|w_{-i}, z_{-i})$ associated with
Because the MCMC algorithm (a particular case collapsed Gibbs sampling is used here) is applied for the inference, the conditional probabilities for \( z_{s,i} \) depend only on \( z_{s,-i} \) and \( w \) and the parameters \( \theta \) and \( \phi \) can be integrated out. Thus, the conditional posterior probabilities presented in Eq. (6) can be obtained as:

\[
p(z_{s,i}=l|z_{s,-i},w) \propto \frac{n_{s,i}^{(v)} + \beta_{vl} - 1}{\sum_{v=1}^{V}(n_{s,i}^{(v)} + \beta_{vl})} \times \frac{n_{s,i}^{(l)} + \alpha_l}{\sum_{l=1}^{K}(n_{s,i}^{(l)} + \alpha_l)} - 1,
\]

where \( n_{s,i}^{(v)} \) is the number of instances of the same word \( w_i = v \) assigned to topic \( l \) without counting the current one. The notation \( n_{s,i}^{(l)} \) is the number of words \( w_i = v \) in sentence \( s \) assigned topic \( l \), not including the current one, and \( n_{s}^{(l)} \) is the total number of words in the sentence \( s \) that were assigned to topic \( l \). \( V \) is the size of the vocabulary and \( K \) is the dimension of the topic assignments.

Similarly, the posterior distribution of the word association by integrating out \( \psi \) using the collapsed Gibbs sampling is given as:

\[
p(x_{s,i} = k|w, x_{s,-i}) = \frac{p(w|x_s) \times p(w|x_{s,-i})}{p(w|x_{s,-i})p(w_i)} \times \frac{\delta_k + n_{s,i}^{(k)}}{\sum_{k=0}^{1}(\delta_k + n_{s,i}^{(k)})}
\times \left\{ \begin{array}{ll}
\frac{n_{s,i;l=1}^{(v)} + \beta_{vl}n_{s,i}^{(v)}}{n_{s,i;l=1}^{(v)} + \beta_{vl}n_{s,i}^{(v)} + n_{s,i-l=1}^{(v)}} & \text{if } k = 1 \\
\frac{\beta_{vl}n_{s,i}^{(v)}}{\sum_{v=1}^{V}(\beta_{vl}n_{s,i}^{(v)})} & \text{if } k = 0
\end{array} \right.
\]

where \( n_{s,i;l=1}^{(v)} \) is the number of times the \( i \)-th word immediately follows the \((i-1)\)-th word in the sentence \( s \), \( n_{s,i}^{(v)} \) is the number of the \( i \)-th word in sentence \( s \), \( n_{s,i;l=1}^{(v)} \) is the number of the \((i-1)\)-th word, and \( N \) is the total number of words in the sentence. The notation \( n_{s,i}^{(k)} \) is the number of the variable \( x_{s,i} = k \) (where \( k = 0 \) or \( 1 \)) given the \( i \)-th word \( w_i \). The notation \( n_{s,i;l=1}^{(v)} \) denotes the number of times the word \( w_{i-1} = v' \) assigned \( x_{s,i} = 1 \) and \( n_{s,i}^{(v)} \) is the number of times the word \( w_i = v \) assigned the word association \( x_{s,i} = 0 \). Correspondingly, the notation \( \beta_{vl} \) represents the word sampling with respect to the topic assignment for word \( w_{i-1} = v' \), and the notation \( \beta_{vl} \) represents the word sampling with respect to the topic assignment for word \( w_i = v \).

With these posterior distributions, a full Gibbs sampling algorithm can be applied to obtain the stationary distribution of the corresponding Markov chain, represented as \( \{z, c, x\} \). For a detailed discussion of the Gibbs sampling algorithm for hLDA and LDA model, one can refer to the original work of hLDA [11] and [12]. The multinomial parameter sets \( \theta \) and \( \phi \) for topic and word sampling, as well as the Bernoulli parameter \( \psi \) can be obtained from this stationary distribution of the corresponding Markov chain. The model with these parameters can be seen as a trained model and it can predict (or “generate”) summary sentences by given new documents as input observations of the model.

### D. Summary Generation

1) **Query-focused summary generation:** To generate a query-focused summary in a particular summarization application, a relevance-based language model [23] is used to involve users query in the settings of the multi-document summarization. Relevance-based language model is a Bayesian generative language model that presents a theoretical justification for evaluating relevance based only on user’s query and document corpus [23]. The model is built based on the assumption that both the query and documents are samples from the same unknown relevance model, denoted as \( R \). Based on this theorem, given the query \( Q \), the probabilistic distribution of sampling a word \( w \) in the relevance-based sentence model is as follows:

\[
p(w|Q) = \frac{p(Q|w) \cdot p(w|R)}{p(Q|R)}
\]

For more details about the relevance-based language model, one can refer to the original work [23]. Here the focus is on how to integrate it with the CTM to rank the sentences over their relevance to the query.

2) **Model integration and sentence ranking:** Suppose given the CTM and the relevance sentence model, represented as \( C_l \) and \( R \), respectively, the process of measuring the relevance of sentences (or ranking the sentences) is the process of predicting likelihood of sentences with the given user query over the models: \( C_l \) and \( R \). This can be modeled as a joint distribution of the sentence and the query given the model \( C_l \) and \( R \) in Equation (10).

\[
p(w,q|C_l, R) \propto p(w|C_l)p(w|R)p(q|w)/p(q|R)
\]

where variable \( w \) represents the sentence to be ranked, and \( C_l \) and \( R \) represent the trained CTM and relevance models respectively. The above joint probability distribution can be factorized as a product of two distributions. The probabilistic values of this distribution can be approximated as the measure of the sentence relevance weighted by its predictive likelihood from the CTM.

To facilitate the sentence ranking process, each sentence is assigned a ranking score, which is defined as a log summation of the predictive likelihood of a sentence represented
by Eq. (10). The ranking score is described as:

\[
\text{score}(w, q) = \sum_{j=1}^{N_s} \log p(w_j|C_t)p(w_j|q, R) = \sum_{j=1}^{N_s} \left( \log p(w_j|C_t) + \sum_{i=1}^{N_q} \log p(w_j|q_i, R) \right), \tag{11}
\]

where \(N_s\) is the number of words in the sentence \(s\), and \(N_q\) is the number of terms in the query \(q\). For calculating the probabilistic value of \(p(w_j|q_i, R)\), one can refer the work of [24]. The probabilistic value of \(p(w_j|C_t)\) can in principle be calculated by integrating out all parameters from the joint distribution expressed in Eq. (1).

### IV. Experiments and Discussions

#### A. Perplexity of Contextual Topic Model

To evaluate this CTM quantitatively, a perplexity of a held-out test set is used to measure the quality of the model. The perplexity is a monotonically decreasing value of the likelihood for the test data. It is the exponent of the cross entropy of the data, and can be defined as:

\[
\text{perplexity}(P_{\text{emp}}, q) = \exp\left(-\sum_{d=1}^{M} \frac{\log P_{\text{emp}}(w_d)}{N_d}\right), \tag{12}
\]

Here \(P_{\text{emp}}(w_d)\) is the test data likelihood that is estimated by the CTM that was generated using training data set. Based on this definition, a lower perplexity value indicates better predictive performance of the model.

In this experiment, n-fold cross validation is employed to perform this hold-out test. The data set is randomly split into \(n\) subsets, and the perplexity measure for hold-out is repeated \(n\) times. In each test, the method selects one of the \(n\) subsets as the hold-out test set and uses the other \(n-1\) subsets for model training. The mean of \(n\) perplexities is used to evaluate the model. Three different corpora were used to build the n-fold cross validation subsets. The NIPS data set contains 1,732 conference papers with 46,874 unique terms from NIPS conferences between the years 1987 and 1999. AQUINIT dataset in DUC 2006 corpus contains 750 news articles with 23,663 unique terms from the Associated Press and New York Times (1998-2000) and Xinhua News Agency (1996-2000). The Seafood\(^2\) corpus is a smaller data set, and includes 156 text documents with 13,031 unique terms. The contents of that data are about Seafood industries.

This CTM was compared to hLDA model and LDA model described in the literature [10], [11]. All models ran for 1000 iterations of the Gibbs sampler with \(\alpha, \beta, \gamma, \delta\) hyper-parameters set as 2.0, 1.0, 1.0, and 2.5 respectively. The number of hierarchy levels was set to 3 for both the CTM and hLDA model. In addition, all models were trained based on the same data and using the Gibbs EM with exactly the same stopping criteria as well. Lower perplexity values indicate better generalization performance and higher possibility to avoid overfitting. The perplexity results are shown as Fig. 2. In addition, a computational time comparison between CTM, hLDA, and LDA is given based on the experimental environment of this research. All model training and hold-out tests were performed using a computer with a configuration of 2.4GHz Intel Xeon processor and 32G memory. The average training time for CTM is about 76 minutes; for hLDA is about 8 minutes, and LDA model used around one and half minutes to train NIPS data. Although the CTM used an extremely long time for model training, its testing time is the shortest among these three models. The CTM used 0.26 seconds, hLDA used 1.5 seconds, and LDA used 0.5 seconds to finish the hold out test.

#### B. ROUGE Evaluation

1) Experimental data and evaluation metrics: Two data sets from the DUC 2005 and DUC 2006 tasks were used to perform the evaluation task. The corpus from DUC 2005 was used for training and parameter tuning and the corpus from DUC 2006 was used for testing. This experiment used ROUGE [6] to evaluate the performance of the summarization system. The evaluation metrics: ROUGE-1, ROUGE-2, and ROUGE-SU4 were investigated during the experiment, and the average recall values of these metrics were reported in this experiment.

2) Experimental results: In this experiment, the summarization system that implemented the CTM and relevance-based sentence model was compared with three other summarizers to demonstrate the effectiveness of the proposed method. Those three systems were a baseline system, which was a relevance model based summarizer without query expansion (namely RMSUM and its Run ID is as shown in Table 1). The RMSUM was built upon the relevance based language model with Dirichlet smoothing approach [7]. The second system expanded the baseline system with query expansion that used topic words and phrases generated from the CTM. This summarizer was called ERMSUM and its Run ID was ER. The third system employed the topical n-gram model [25] with the relevance based sentence model (namely TopicalN and its Run ID is as shown in Table 1). The system implemented with the CTM was named CTMSUM and its Run ID was CTM, as shown in Table I. In order to compare the performance of the multi-document summarization solution addressed in this study, the result of the summarizer with the highest ROUGE scores in DUC 2006 are also listed in Table I. Run-ID 24 is for the system IITH-Sum [16], the best performance in DUC2006. Comparing the ROUGE evaluation results of CTMSUM with results from other systems listed in Table 1, one can see the significant performance gains achieved in CTMSUM system. It has achieved better results in recall measures for

---

\(^2\)The Seafood dataset can be downloaded from http://users.iit.demokritos.gr/~izavits/datasets/Seafood_corpus.zip
Figure 2: Perplexity results of an n-fold cross validation on NIPS (left), AQUINT (middle), and Seafood (right) corpora for the three different models, CTM, hLDA, and LDA. All results demonstrate that the CTM has better generalization performance than other two models when the number of topics is more than 60.

Table I: Experimental results of ROUGE evaluation on DUC 2006 data

<table>
<thead>
<tr>
<th>Run-ID</th>
<th>System</th>
<th>ROUGE-1</th>
<th>ROUGE-2</th>
<th>ROUGE-SU4</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTM</td>
<td>CTMSUM</td>
<td>0.4157</td>
<td>0.0968</td>
<td>0.1548</td>
</tr>
<tr>
<td>24</td>
<td>IIITH-Sum</td>
<td>0.4098</td>
<td>0.0951</td>
<td>0.1546</td>
</tr>
<tr>
<td>TN</td>
<td>TopicalN</td>
<td>0.4032</td>
<td>0.0908</td>
<td>0.1502</td>
</tr>
<tr>
<td>ER</td>
<td>ERMSUM</td>
<td>0.3914</td>
<td>0.0796</td>
<td>0.1365</td>
</tr>
<tr>
<td>RM</td>
<td>RMSUM</td>
<td>0.3816</td>
<td>0.0724</td>
<td>0.1287</td>
</tr>
</tbody>
</table>

Similar results were reported in the experiments for other two data sets. All three models reported larger perplexity values in the ACQUIT data set, but the CTM outperformed hLDA and LDA when the number of topics was above 60. In the Seafood data set, the CTM outperformed other two models in the entire range of topics. In addition, all perplexity values were smaller than values evaluated in other two corpora, since the Seafood corpus is the smallest data set in size and has only 156 documents with 13,031 unique terms in total.

For the experiment results in a practical application, as shown in Table I, the system CTMSUM outperforms the top performing system (Run-ID 24, which is the one that performed the best ROUGE evaluation in DUC 2006) on all major recall metrics. These experimental results validate that the approach proposed in this article can improve summarization performance. In addition, the system CTMSUM has achieved significant improvements in ROUGE-2 evaluation when comparing with baseline system (increased 33.7%). With respect to both topic hierarchies and lexical co-occurrence of words, this result demonstrates that contextual topics produced by the CTM indeed significantly impacts the performance of summarization reported in this article.

V. CONCLUSIONS AND FUTURE WORK

The integration of hierarchical topics into n-gram models shows potential for an effective and efficient solution to the problem of multi-document summarization. In order to evaluate the proposed method thoroughly, two different evaluation scenarios have been provided in this research. Comparing against the well-known Bayesian based topic models, the CTM proposed in this research not only allows for multiple hierarchies between topics, but also considers word associations in the same hierarchical level. Moreover, the experimental results reported in this article for a real application case have demonstrated that the proposed model is suitable for the task of multi-document summarization.

C. Discussion

The comparison between the CTM and LDA have shown that for all data sets in three corpora, the CTM obtains much lower perplexity values when the number of topics is above 60. For the cases with fewer topics, the CTM obtains similar perplexity values as LDA. A possible reason is that the CTM includes a hierarchical structure for topics, and probabilistic values of topics are weighted with their level distributions. Therefore, when the number of the hierarchical levels of a sampling path (each path represents a document) increases, the number of topics allocated to each level decreases. Although the number of topics can be theoretically infinite in a Bayesian hierarchical topic model, in a realistic case, it is finite because the observed document always contains certain number of words [11]. This is also the reason that the perplexity values have less variance when the number of sampling topics is greater than 150 (see Fig.2 all three panels).

In the NIPS data set (left panel of Fig. 2), the CTM demonstrates lower perplexity values in a wide range of topics than both hLDA and LDA models. It shows similar perplexities as hLDA model in the range of 10-20 topics, but much better than LDA model in the entire range of topics.
The experimental results have also demonstrated that the proposed approach can significantly improve the summarization performance, rendering the summarizer comparable to state-of-the-art summarization systems. These findings also encouraged to extend this model to different tasks, such as the emotion recognition, image processing and event detection, and suggested a direction in the future work for more sophisticated methods in order to improve performance and reduce overall computational time.

REFERENCES


