



Extractive Summarization using A Latent Variable Model

Asli Celikyilmaz¹ Dilek Hakkani-Tür²

¹University of California, Berkeley, CA

²International Computer Science Institute (ICSI), Berkeley, CA

asli@eecs.berkeley.edu, dilek@icsi.berkeley.edu

Abstract

Extractive multi-document summarization is the task of choosing sentences from a set of documents to compose a summary text in response to a user query. We propose a generative approach to explicitly identify summary and non-summary topic distributions in the sentences of a given set of documents (i.e., document cluster). Using these approximate summary topic probabilities as latent output variables, we build a discriminative classifier model. The sentences in new document clusters are inferred using the trained discriminative model. In our experiments we find that the proposed summarization approach is effective in comparison to the state-of-the-art methods.

Index Terms: topic modeling, discriminative classification, summarization

1. Introduction

Multi-document summarization requires creating a short summary of documents that are about a user-formulated query. In the case of speech summarization, the text of these documents is generated using automatic speech recognition. For newswire text, this task has been evaluated in the framework of Document Understanding Conferences (DUC) [1], resulting in test beds that can be used for summarization research. Automatically created summaries can either consist of the most important information overall (generic summarization), or of the information most relevant with respect to a user's information need.

Previous studies on summarization can be investigated under two sections: one stream of research focuses on identifying the most important sentences using a language model, or a classifier model based on features such as term-frequency and sentence length [2, 3]. Although very effective, these models usually lack the hidden semantic content information in documents. Other studies have taken probabilistic approaches by ranking sentences using estimated summary likelihoods based on purely data-driven methods [4, 5, 6], especially using Latent Dirichlet Allocation (LDA)-style topic models [8]. These methods yielded promising results in correct labeling of sentences to generate coherent summaries, since representing documents as topic distributions rather than bags of words diminishes the effect of lexical variability, while keeping the overall semantic structure of document sets intact. However one issue with these models is that they can yield too general results [7]. Furthermore, there is very little interest on probabilistic topic models to predict summaries for new document sets using previously trained models.

To address these issues, we build a probabilistic topic model, namely Summary Focused Topic Model (sLDA), that can *explicitly* discover summary and non-summary topic distributions on sentence level under the assumption that sentences convey a particular content. Viewing sentences as mixtures of

two separate sets of topics, summary and non-summary topic distributions, makes it possible to formulate this task as the problem of discovering salient sentences that are likely to be included in summary text. We predict expected summary topic distributions in sentences and use them as latent output variables, along with the set of features characterizing sentences in documents to build a discriminative classifier model. The trained model is then used to infer summary latent distributions in sentences of test documents to construct a summary.

In this work, we focus on two major tasks to generate coherent and compact summaries: (i) capturing summary-focused contents from sentences in documents with a probabilistic model described in Section 2, (ii) building a classification based approach for inferring content distributions in sentences of test documents, instead of building a probabilistic model for test documents as shown in Section 3.

2. Summary Focused Topic Model - sLDA

(1) Model Description: sLDA is an extension of LDA [8], which is based on two assumptions: (i) sentences in summaries represent coherent topics which can be discovered from sentences in document sets; (ii) the rest of the content distributions in documents represent specific content that generally do not appear in a summary text.

The sentences $M = \{o_m\}_{m=1}^M$ in a document cluster D are represented by a list of tokens $o_m = (w_1, \dots, w_{N_m})$, where $w_i \in \mathbf{W} = \{w_1, \dots, w_V\}$. Here N_m is the number of words in the m th sentence. Given document cluster sentences and human generated summary text, we construct vocabulary \mathbf{W} of size V , and identify summary words in \mathbf{W} .

K topics are represented by multinomial distributions over V unique words, where $P(w|z_k) = \phi_w^{(k)}$, $k=1, \dots, K$, is the probability of a word given topic z_k . Similarly, M sentences are represented by multinomial distributions θ over K topics, such that for a word in a sentence o_m , $P(z_k|o_m) = \theta_k^{(o_m)}$, $m=1..M$, is the probability of z_k in a given sentence o_m . We identify multinomial distributions with θ by allocating the first S distributions as summary topics, which highly likely generate words that are common in the summary and document texts. The $R=K-S$ topics represent those that are only generated from sentences which do not contain any common terms with summary text. We cast the problem as a generation of a set of summary topic distributions, which closely match with summary text unigram/bigram distributions (similar to criterion proposed by [9], which uses Kullback-Leibler (KL) divergence for evaluation). The generative model of sLDA is shown in Table 1 and graphical model in Fig.1.

We restricted our model to only three parameters, α , α^s and β (Fig.1) using symmetric Dirichlet distributions. Note that the

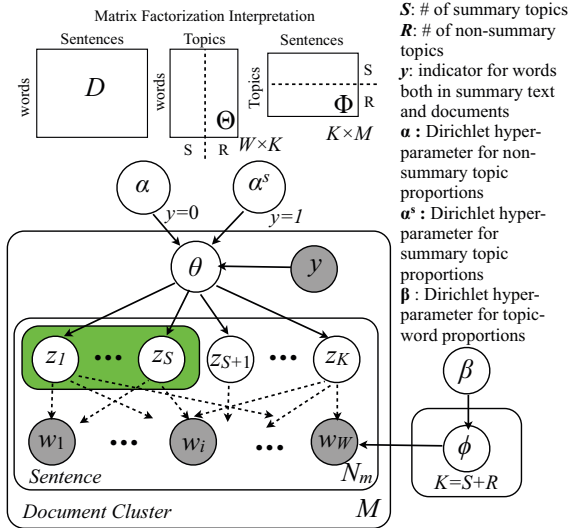


Figure 1: Graphical Representation of sLDA Model.

<ul style="list-style-type: none"> – For each topic $k=1..K$, draw distribution of words $\phi^{(k)} \sim Dir(\beta)$ – For each sentence $m=1..M$ <ul style="list-style-type: none"> * draw S # of summary topic distributions, $\theta_{1..S} \sim Dir(\alpha_s)$ * draw R # of non-summary topic distributions, $\theta_{(S+1)..K} \sim Dir(\alpha)$ * for each word i in sentence o_m; <ul style="list-style-type: none"> o if $y_i=1$; draw $z_k \sim Mult(\theta_k^{(o_m)})$, $k \in \{1..S\}$ o if $y_i=0$; draw $z_k \sim Mult(\theta_k^{(o_m)})$, $k \in \{S+1..K\}$ o draw $w_i \sim Mult(\phi^{(k)})$
--

Table 1: Generative process of sLDA.

parameter y , known *a priori*, determines from which distribution the topics should be sampled. Specifically, if word w_i in o_m exists in a summary ($y_i = 1$), then summary topic mixing variables $\theta_k^{(o_m)}$ for topics $k = 1, \dots, S$ are sampled, otherwise non-summary topic mixing variables for topics $k=(S+1), \dots, K$ are sampled ($y_i = 0$). Hence, we dedicate the first S topics in $K \times M$ (topic by sentence) count matrix to summary topics to generate the summary topic distributions, and the rest of the $R=K-S$ to non-summary topics. (In the experiments we used $S=R$).

(2) Inference: Using Gibbs Sampler [10] the equations to sample z_i variable for each word token w_i for $y_i=0$, $p(z_i = 0, z_i|w, z_{-i}, \alpha, \beta)$ can be derived as:

$$\frac{n_{wk,-i}^{VR} + \beta}{\sum_{w'} n_{w'k,-i}^{VK} + V\beta} * \frac{n_{mk,-i}^{MR} + \alpha}{\sum_{k'} n_{mk',-i}^{MK} + K\alpha} \quad (1)$$

and for $y_i=1$, $p(y_i = 1, z_i|w, z_{-i}, \alpha^s, \beta)$:

$$\frac{n_{wk,-i}^{VS} + \beta}{\sum_{w'} n_{w'k,-i}^{VK} + V\beta} * \frac{n_{mk,-i}^{MS} + \alpha^s}{\sum_{k'} n_{mk',-i}^{MK} + K\alpha^s} \quad (2)$$

where $n_{wk,-i}^{VS}$ is the count of w_i in summary topic k and $n_{mk,-i}^{MS}$ is the count of summary topic k in sentence o_m excluding current instances. Similarly $n_{wk,-i}^{VR}$ is the count of w_i in non-summary topic k and $n_{mk,-i}^{MR}$ is the count of non-summary topic

k in sentence o_m . Each sentence is considered as a mixture of summary and non-summary topics, which can be approximated from the model.

(3) Latent Sentence Labels : After topic distributions are approximated for each sentence, we compile a training dataset $D = \{(x_m, l_m)\}$, $m = 1..M$, where x_m is a representation of a sentence as a feature vector. Initially, l_m is a K dimensional vector $l_m \in (0, 1)^K = \{\hat{\theta}_{k=1..K}^{(o_m)}\}$ of the expected topic probabilities of a sentence o_m over topics $K=S+R$. We pose the multi output problem with K outputs as a binary-classification using the following voting criteria: We set binary class label $l_m=1$ for a given sentence o_m , if the sum of the summary topic probabilities are greater than the rest of the topic probabilities, and construct binary latent output variable $l = \{l_m\}_{m=1}^M$, where:

$$l_m = \begin{cases} 1, & \text{if } (\sum_{k=1}^S \hat{\theta}_k^{(o_m)}) - (\sum_{k=S+1}^K \hat{\theta}_k^{(o_m)}) > 0 \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

A binary output $l_m = 1$ indicates that the total summary topics have higher probability of being generated compared to non-summary topics, and hence the sentence is better suited for inclusion in a summary.

3. Classification for Sentence Extraction

Feature Extraction: We train a classifier model on a dataset of sentences compiled from different document clusters using binary latent variable l_m as output and frequency based *meta*-features as input variables. sLDA is trained on sentences of each document cluster using a vocabulary specific to their cluster. The classifier model, on the other hand, is built on sentences from all the document clusters. Thus, we use *meta*-n-gram features, instead of word n-gram features to represent sentences that do not necessarily have common vocabulary.

Given document cluster D , we identify the most frequent (non-stop word) unigrams, i.e., $v_{freq} = \{w_i\}_{i=1}^r \subset W$, where r is a model parameter of number of most frequent unigram features. Given sentence o_m , let d represent the document that o_m belongs to, i.e., $o_m \in d$. We measure unigram probabilities for each w_i of o_m by $p(w_i) = n_d(w_i)/n_D(w_i)$, where $n_d(w_i)$ is the number of times w_i appears in d and $n_D(w_i)$ is the number of times w_i appears in the document cluster D . For any i th feature, the value is $x_{mi} = 0$, if the given sentence does not contain w_i , otherwise $x_{mi} = p(w_i \in o_m)$. We also introduce bigram extensions. Additional features are: average term-frequency of a sentence, the rank of a sentence in a given document and sentence word count.

Classifier Model: We utilize support vector machines (SVM) [11] with the RBF kernel $e^{-\|x_i - x_j\|^2}$ as binary classifier. The regularization parameter C and kernel parameter γ are selected by cross-validation. We use Platt's posterior probabilities [12], [13] to estimate probability of summary topics being generated from a given sentence o_m ,

$$P(l = k|x) = \hat{\theta}_k^{(o_m)} = 1/(1 + e^{A\hat{f}+B})$$

where $k \in \{0,1\}$, binary latent output variable, and A and B are estimated by minimizing the negative log-likelihood, and \hat{f} are the decision values of training data learned from the SVM classifier.

Once we train the classifier, we use the model to predict the total summary topic proportions of sentences in test documents, $D = \{(x_m^{test})\}$ and obtain $\hat{P}(l = 1|x^{test})$, likelihood of a given sentence being extracted for a summary text.

Sentence Scoring: Topic-based scores obtained from sLDA might be too coarse to generate high quality summaries. Since term frequencies is shown to play an important role in determining the importance of sentences, we use SUMBASIC [3], which assigns a score to each sentence S based on the count of high frequency words it contains by

$$ScoreUni_s = \sum_{w \in S} \frac{1}{|S|} P_D(w)$$

where $P_D(w_i)$ is the observed unigram probabilities in the document collection D . We also measure bi-gram probabilities to calculate sentence scores, ScoreBi(S) and derive a combination of topic-based and the two frequency based scores,

$$Score_s = a * \hat{P}(l = 1 | x^{test}) + b * ScoreUni(s) + c * ScoreBi(s)$$

In the experiments, we predicted the optimum values of a, b, c via cross validation on training document set.

Redundancy Elimination: To eliminate redundant sentences in the generated summary, we incrementally add onto the summary the highest ranked sentence o_m and check if o_m significantly repeats the information already included in the summary until the algorithm reaches word count limit. We use a word overlap measure between sentences normalized to sentence length. A o_m is discarded if its similarity to any of the previously selected sentences is greater than a threshold identified by a greedy search on the training dataset.

4. Experiments

We used DUC2005-2006 datasets to train and validate our models, and evaluate the performance on DUC2007 sets, see Table 2(a) for a description of the data sets. The task is to create at most 250 word long summary text for a given document cluster. Performance measure is based on ROUGE [14], [15], which evaluates summaries based on the maximum number of overlapping units between generated summary text and a set of human generated summary text. We present R-1 (recall against unigrams), R-2 (recall against bigrams), and R-SU4 (recall against skip-4 bigrams) results. For SVM cross validation we use $(C, \gamma) : ([2^{-5}, 2^{-3} \dots 2^{15}], [2^{-5}, 2^{-3} \dots 2^{15}])$ parameters.

We evaluate the performance of our sLDA model using ROUGE in comparison to the following state-of-the-art summarizers: (1) PYTHY [16] a supervised sentence extraction summarization system, (2) HIERSUM [6] a generative summarization method based on topic models, (3) ILP (ICSI Summarizer) [17] integer programming based approach¹. We also compared results to a Baseline method, which simply replaces the scoring algorithm of our sLDA with a simple scoring function, which is measured by its lexical similarity (cosine distance) to maximum matching summary sentence. Then we build a regression model with the same variables as our sLDA to create a summary.

4.1. Model Performance Evaluations

During training of sLDA models, we find $K=8$ as the optimum topic number after empirical experiments $K = \{2^2, 2^3, 2^4, 2^5\}$ and choose the optimum feature defined in previous section based on classifier accuracy on validation data.

The sLDA ROUGE results against state-of-the-art summarization methods are shown in Table 2.b. As shown in the table, the proposed sLDA method achieves the best performance.

¹<http://code.google.com/p/icsisumm>

Criteria	ILP	sLDA	Tie
Non-redundancy	18	26	49
Coherence	23	45	25
Focus	19	53	21
Overall	26	45	22

Table 3: Frequency results of manual quality evaluations. Results are statistically significant based on t-test. Tie indicates evaluations where two summaries are rated equal.

Based on the ROUGE score when stop words are used, the sLDA achieves a raw performance gain of 4-5% with respect to all baselines, except the ILP summarizer. Intuitively, ILP picks the optimum subset of sentences, and hence eliminates redundancies and uses the space for including other word n-grams, while the other methods use greedy algorithms during sentence selection, and may still include redundancies. In addition, ILP uses word bi-grams as concepts during training, thus resulting better R-2 performance. Contrarily, sLDA neither optimizes a model based on sentence redundancy nor on bi-gram concepts. Nevertheless, sLDA R-1 and R-SU4 scores are comparable (if not better). In without stop words ROUGE scores, sLDA performance is significantly better against all models except R-2, because during training the topic model in sLDA stop words are not added to the vocabulary, contrary to HIERSUM.

4.2. Manual Quality Evaluations

Here, we manually evaluate quality of summaries, a common DUC task. Human annotators are given two sets of summary text for each document set, generated from two different summarizer models and are asked to mark the better summary according to four criteria: (1) Non-redundancy (which summary is less redundant), (2) Structure and Coherence (which summary is more coherent), (3) Focus and Readability (content and not include unnecessary details), (4) Overall performance. We compared sLDA summaries to the next best system, the ILP summarizer. We asked 4 annotators to rate DUC2007 predicted summaries (average 23 summary pairs). A total of 93 pairs are judged and evaluation results in frequencies are shown in Table 3. The participants rated sLDA generated summaries more coherent and focused compared to ILP. Except for the non-redundancy criteria, all the results in Table 3 are statistically significant (based on t-test on 95% confidence level.) indicating that sLDA summaries are rated significantly better.

4.3. Hybrid Summarization Model

Based on the results of previous experiments, we conclude that sLDA can generate coherent and focused summaries while not performing as good on recall against bi-grams (R-2) as opposed to ILP. In order to further improve summarization performance, here we combine the sLDA and ILP methods in a following way: We use the sLDA scores in ILP summarizer tools to build sLDA-ILP Hybrid models. While ILP summarizer works with concept-level weights, sLDA generates sentence scores. Hence, instead of weighing word n-grams according to their document frequencies, as in the previous work [17], in hybrid sLDA-ILP models, we estimate weights for word n-grams by using the scores of sentences in which they appear. We used the total and average scores of all sentences an n-gram appears as the weight of that n-gram.

This ensures a good concept weighting technique with a

	DUC 2005	DUC 2006	DUC 2007
# docSets	50	50	45
# docs/docSet	31	25	25
# docs/docSet	28	28	21

(a)

ROUGE	w/o stop words			w/ stop words		
	R-1	R-2	R-SU4	R-1	R-2	R-SU4
Baseline	32.4	7.4	10.6	41.0	9.3	15.2
PYTHY	35.7	8.9	12.1	42.6	11.9	16.8
HIERSUM	33.8	9.3	11.6	42.4	11.8	16.7
ILP	33.9	9.1	12.2	44.5	12.6	17.3
sLDA	36.2	9.0	12.5	45.4	11.5	17.3

(b)

Table 2: (a) Description of DUC datasets used in experiments. (b) ROUGE comparisons with the best systems on DUC2007 dataset.

ROUGE w/ stop words	R-1	R-2	R-SU4
sLDA	45.4	11.5	17.3
sLDA-ILP Hybrid	45.2	12.8	17.6

Table 4: ROUGE results for sLDA and Hybrid Summarization Methods (sLDA and ILP) on DUC 2007 dataset.

good sentence selection algorithm. We used the same set-up as in section 5.1 to build sLDA-ILP models using with stop words. As depicted in Table 4, with the hybrid model, R-2 results are improved significantly while rest of the ROUGE scores of sLDA and sLDA-ILP are comparable.

5. Conclusions

Probabilistic topic models have been successfully used to solve natural language processing problems, however very little attention has been made to inferring topic distributions for unseen datasets. In this paper, we present a new probabilistic model for the problem of multi-document summarization. Our approach is based on learning summary content distributions from document sets using provided summary texts as supervision. We find that a simple discriminative learning method trained on word frequencies as inputs, and approximated topic probabilities given sentences as outputs, can produce comparable results to the state-of-the-art multi-document summarization methods. One prominent feature of our proposed modeling tool is that it can be used not only for summarization tasks, but also for any other problem related to information extraction from text, where hidden topics can be extracted from unlabeled text collections. In this sense, our approach is generic and forms a baseline.

One area of direction we would like to explore as a future study is capturing n-gram relations of words (lexical units) while approximating the content distributions. Longer lexical units are rather shortened in summary text (e.g., "The Vice President of USA, Joe Biden" is usually found in a summary text as "Vice President Biden"). A generative model that can capture such dependencies would be valuable and may enable sentence compression, a very important step in summarization. Another area of direction is making the approach robust to errors in automatic speech recognition and spontaneous speech.

6. References

- [1] P. Over and W. Liggett, "Introduction to duc: An intrinsic evaluation of generic news text summarization systems," in *Proc. DUC*, Philadelphia, PA, USA, July 2002.
- [2] D.R. Radev, H. Jing, M. Stys, and D. Tam, "Centroid-based summarization for multiple documents," in *In Int. Jnl. Information Processing and Management*, 2004.
- [3] A. Nenkova and L. Vanderwende, "The impact of frequency on summarization.," in *Tech. Report MSR-TR-2005-101, Microsoft Research, Redwood, Washington*, 2005.
- [4] I. Titov and R. McDonald, "A joint model of text and aspect ratings for sentiment summarization," in *ACL-08:HLT*, 2008.
- [5] J. Tang, L. Yao, and D. Chens, "Multi-topic based query-oriented summarization," in *SIAM International Conference Data Mining*, 2009.
- [6] A. Haghighi and L. Vanderwende, "Exploring content models for multi-document summarization," in *NAACL HLT-09*, 2009.
- [7] C. Chemudugunta, P. Smyth, and M. Steyvers, "Modeling general and specific aspects of documents with probabilistic topic model," in *Proc. NIPS-06*, 2006.
- [8] D. M. Blei, A. Ng, and M. Jordan, "Latent dirichlet allocation," in *Jrnl. Machine Learning Research*, 3:993-1022, 2003.
- [9] C.-Y. Lin, G. Cao, J. Gao, and J.-Y. Nie, "An information-theoretic approach to automatic evaluation of summaries," in *HLT-NAACL-06*, 2006.
- [10] T. Griffiths and M. Steyvers, "Finding scientific topics," in *PNAS*, 101(Supp. 1): 5228-5235, 2004.
- [11] B. Boser, I. Guyon, and V. Vapnik, "A training algorithm for optimal margin classifier.," in *Proc. 5th Workshop Computational Learning Theory*, 1992.
- [12] J. Platt, "Probabilistic outputs for support vector machines and comparison to regularized likelihood methods.," in *In A.J. Smola, P.L. Bartlett, B. Scholkopf and D. Schuurmans, eds. Advances in Large Margin Classifiers*, 2000.
- [13] T.F. Wu, C.J. Lin, and R.C. Weng, "Probability estimates for multi-class classification by pairwise coupling.," in *Jrnl. Machine Learning Research*, 2004, vol. 5, pp. 975-1005.
- [14] C.-Y. Lin and E.H. Hovy, "Automatic evaluation of summaries using n-gram co-occurrence statistics," in *Proc. HLT-NAACL, Edmonton, Canada*, 2003.
- [15] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *In Proc. ACL Workshop on Text Summarization Branches Out*, 2004.
- [16] K. Toutanova, C. Brockett, M. Gamon, J. Jagarlamudi, H. Suzuki, and L. Vanderwende, "The PYTHY summarization system: Microsoft Research at DUC 2007," in *Proc. DUC*, 2007.
- [17] D. Gillick, Benoit Favre, and D. Hakkani-Tür, "The icsi summarization system at tac 2008," in *Proc. NIST Text Analysis (TAC)*, 2008.