A Semantic Question/Answering System using Topic Models

Asli Celikyilmaz
Computer Science Division
University of California, Berkeley
asli@eecs.berkeley.edu

Abstract

Bayesian Topic Models have been used in different fields of natural language processing to help extract information from unstructured text. Specifically previous research on topic model based retrieval methods has shown significant performance improvements. This paper deals with more complex models of information extraction, namely Question/Answering (QA) models. For any given question posed in natural language, QA systems are designed to extract possible answer as a semantic group or a pre-defined named-entity type, i.e., person, organization, city, etc. Thus, relating query terms with existing entities in a given corpus is crucial in QA systems. Our goal is to improve performance of our QA system by utilizing information from natural groupings of words in documents, i.e., topics, in relation to named entity types in their vicinity. Our empirical analysis indicate that more accurate snippets (paragraphs) containing a true answer string for a given question can be extracted via proposed topic model in comparison to the keyword search or previous topic model approaches for information retrieval.

1 Introduction and Motivation

Question/Answering (QA) is a line of research in natural language processing, where a user poses a question in natural language, e.g., ”Who is the winner of nobel peace price in 2009?” (running example) and expects an answer as in word/phrases or a sentence. Thus, QA research attempts to deal with a wide range of question types including: fact, list, definition, semantically-constrained, cross-lingual questions, etc. In this research [1] , we deal with simple to complex factoid questions, where expected answer is a word or a phrase (e.g., Barack Obama of ’human’ named entity type).

Before we present our new cluster-based QA model, we briefly explain a typical QA process (pipeline): Initially after a given question is broken down into keywords and semantic groups, e.g., subject, object, etc., its answer type is identified via a question classifier module [3]. An answer type is typically a pre-defined named-entity such as country, number or food type (examples in Table 1). For example, our named entity recognizer (NER) module [3], trained using conditional random fields [5], can identify up to 6 coarse and 50 fine named-entity types. Thus, the answer-type (named entity) of the running question, would be a ’Human’ course entity followed by a finer sub-group, i.e., HUMAN:Individual. Later, a document retrieval module uses a search engine to extract documents/paragraphs/sentences, namely, snippets, from entire document set (corpus) that are likely to contain the answer being sought. Usually a classifier model trained on retrieved text snippets is used to extract posterior probability of an answer text being contained in each text snippet, $P(answer|snippet_i)$. Our research [3] indicate that the answer-type named entity is one of the core features that rank snippets higher when they have high likelihood of containing right answer.

[1] Supported in part by ONR N00014-02-1-0294, BT Grant CT1080028046, Omron Grant, Tekes Grant, Chevron Texaco Grant, The Ministry of Communications and Information Tech. of Azerbaijan and BISC Program of UC Berkeley.

Table 1: Samples of 6 Coarse (in **Bold**) and 50 Fine Named-Entity Types

(1) ABBR	**(2) HUMAN**	**(3) NUM**	**(4) LOC**	**(5) ENTITY**	**(6) DESC**
abbr	group	date	city	animal	defnition
exp.	individual	money	country	body	manner
..	...	...	...	...	...

A potential problem with snippet retrieval module of any QA system is that the answer string being searched is not usually found in candidate snippet, even if there is %100 word overlap between question and candidate snippets. Two most common approaches to this problem is query expansion and reformulation, which are used to extract semantically similar text from documents to improve information retrieval (IR). Such methods do not guarantee that the text being returned will contain the true answer string, which is the main goal of QA systems. Thus, in QA, IR plays a crucial role, as sentences/paragraphs with high probability of containing the right answer should be retrieved.

Hierarchical topic models have been shown to improve the retrieval performance in many different studies, e.g., [2], [7]. For instance, standard LDA models are used as Dirichlet smoothing in constructing language models for IR in [7]. One limitation of LDA is that it cannot handle the generality and specificity of IR systems which is addressed in [4]. Topic models such as LDA represents a combination of words so for IR systems it may not be precise representation. A clustering-based retrieval would be too general for usually specific representations like IR. Hence, in [4], specific and general aspects of documents are considered in a single topic modeling framework. They introduce a new topic model for IR tasks, namely, special word topic models (SWB) that can structure documents based on general topics and specific word level distributions. Specifically, for search queries that contain general and specific words, a matching document can be retrieved via their specific topic model as opposed to standard topic models for text, e.g., LDA [1].

The latter methods are crucial in QA systems because we would like to extract answer strings from text snippets which are not only semantically related to the *topic* of a given question (e.g., nobel piece prize in the running example) but also could utilize specific *focus* of the question (e.g., Barack Obama and 2009 in the running question) during retrieval process. In addition, we would like the retrieved snippets to contain named entities that we are looking for to identify candidate answer text.

The goal of this research study is to improve the performance of QA models utilizing the information of natural groupings of specific and general words in documents, i.e., topics, in relation to the named-entity types that exist in their vicinity. In other words, we would like to extend existing bayesian network models in such a way that we not only represent the documents as distribution of words, but also define named entity types in the vicinity of each word in each document.

2 Question Type Entity Model with Global and Local Features (Ent-LDA)

Our probabilistic model, Ent-LDA, represents each document as a distribution of words over topics, where each topic-word in a document has a relation to the 50 different named entity types. In other words, we would like the new graphical model to generate neighborhood named entities for each word generated for a topic from the multinomial distribution.

In this research, the named entity variable, i.e., ε, is a binary value, representing if there exists corresponding entity in the vicinity of corresponding word [2]. Exploiting general and specific aspects of documents [4], the new generative model also considers words specific to documents, i.e., local lexical units/words, while generating the model. Hence, for words that are local to certain documents are generated from a different distribution indicated by a binary variable x. The vicinity can be defined as a window of n sentences apart in a given document.

2.1 Generative Process Ent-LDA

Graphical model of our ”entitiy topic model Ent-LDA” for QA systems is shown in Figure 1. Ent-LDA has similar structure to LDA model [1] and SWB model [4] but we introduce additional struc-

[2] We identify named entities in each document of a given corpus.

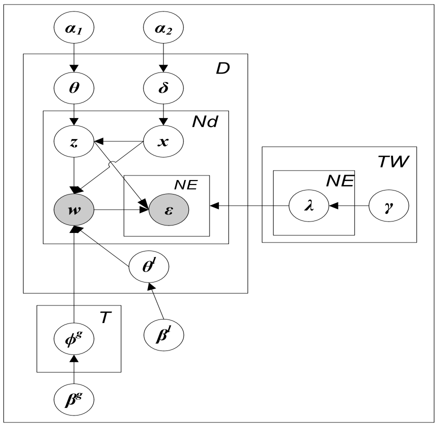


Figure 1: Graphical Model for Ent-LDA
Nd: # of words in document d
NE: # of named entity types ; T : # of topics
D : # of documents W: # of tokens
w_i : word generated from assigned topic,
z_i : topic assignment for each word,
ε_{ij} : named entity type assignment for each $word_j - topic_i$ in a document
θ_d : distrib. over topics for each document
ϕ^g : distrib. over global words for each topic
ϕ^l : distrib. over local words per document
x : sets global or specific word assignment
δ : distrib. over indicator variable x
λ : distrib. over named entity types for each topic-word relation.
$\alpha_1, \alpha_2, \gamma, \beta^g, \beta^l$: hyperparameters

tures to capture the sought named-entities in the vicinity of query words. Particularly, each word token has one latent random variable x associated with it. If $x = 0$, the generic entity topic model is used to generate the model using the corpus specific multinomial θ^g ($g : global$), whereas if $x = 1$ then words are sampled from document local multinomial θ^l ($l : local$) (with Dirichlet priors parameterized by β^g and β^l). The variable x is also sampled from a document-specific parameter δ with symmetric Dirichlet prior, α_2. Entity types ε are sampled for each word-topic sample, that is, for a given topic and a word, each entity is generated, $\varepsilon_{ne} \in \{0, 1\}$, $ne = 1, ..., NE(= 50)$. Specifically, we would like to generate binomial distribution for entity type representing the existence in the vicinity of a given word. Each term is allowed to have only one entity type, so while generating an entity type in the vicinity of a word, there will be only one entity type associated for the corresponding vicinity term. The generation process is as follows:

1. For each $t = 1, ..., T$
   - Draw discrete distribution $\phi^g \sim Dirichlet(\beta_g)$
   - For each word w_i
     - For each NE type $ne = 1, ..., NE$:
       - Draw discrete distribution $\lambda_{ti}^{ne} \sim Beta(\gamma)$
2. For each document $d = 1, ..., D$
   - Draw distributions $\theta_d \sim Dirichlet(\alpha_1)$, $\phi_d^l \sim Dirichlet(\beta_l)$, $\delta_d \sim Beta(\alpha_2)$
   - Draw a topic $z_i \sim Multinomial(\theta_d)$
   - Draw $x_i \sim Binomial(\delta_d)$
   - if $x_i = 0$, Draw a word $w_i \sim Multinomial(\phi_{z_i}^g)$
   - if $x_i = 1$, Draw a word $w_i \sim Multinomial(\phi_d^l)$
   - For each NE type $ne = 1, ...NE$: Draw an ne-type $\varepsilon_{it}^{ne} \sim Binomial(\lambda_{z_i,w_i}^{ne})$

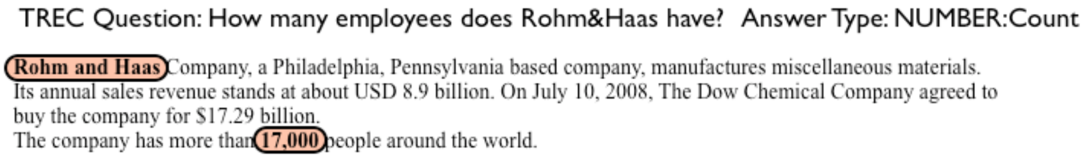
For posterior distribution approximation, Gibbs Sampling [6] is used with the following equations:

$$\left[P(z|d) = \frac{n_{td,-i}^{TD} + \lambda_1}{\sum_t n_{td,-i}^{TD} + T\lambda_1}\right], \left[P(w|x=1) = \frac{n_{d,x=1,-i} + \alpha_2}{n_{d,-i} + 2\alpha_2} \cdot \frac{n_{wd,-i}^{WD} + \beta^l}{\sum_w n_{wd,-i}^{WD} + W\beta^l}\right],$$

$$\left[P(w|x=0, z=t) = \frac{n_{d,x=0,-i} + \alpha_2}{n_{d,-i} + 2\alpha_2} \cdot \frac{n_{wt,-i}^{WT} + \beta^g}{\sum_w t_{wt,-i}^{WT} + W\beta^g}\right], \left[P(\varepsilon = ne|w, z) = \frac{n_{wt,ne,-i}^{WT,NE} + \gamma^{WT}}{n_{wt,-i}^{WT} + NE\gamma^{WT}}\right]_{ne=1}^{NE}$$

3 Discussion

We briefly presented the general architecture of QA system using probabilistic topic models, which can combine information extraction and ranking methods in one clustering schema. The algorithm exploits information about named entities in the vicinity of each word, generated from a multinomial topic distribution. Knowing the answer entity type of a question, with the new entity topic model,



TREC Question: How many employees does Rohm&Haas have? Answer Type: NUMBER:Count

Rohm and Haas Company, a Philadelphia, Pennsylvania based company, manufactures miscellaneous materials.
Its annual sales revenue stands at about USD 8.9 billion. On July 10, 2008, The Dow Chemical Company agreed to buy the company for $17.29 billion.
The company has more than 17,000 people around the world.

Figure 2: Example question & answer snippet (3 sentence limit) from TREC.

we can not only identify words that are semantically related with the question keywords/phrases but also extract information about whether the named entity we are seeking exists in the vicinity of this semantic word grouping. Although we aim at fully generative model, a partial supervision is required to learn a relation between a question and its candidate text snippets, by introducing a classifier model, e.g., textual entailment. We use the classifier to calculate the likelihood of a textual entailment relation between a question posed and candidate text snippets [3]. In the analogy of [7], here we build a hybrid approach by combining likelihood of textual entailment relation between question and answer sentence along with our Enty-LDA, which yields maximum likelihood estimates of word being generated from a given snippet with the answer type entity in their vicinity.

As an example, we used Ent-LDA to analyze previous years TREC questions [3] on AQUAINT corpus of LDC, consisting of newswire text data in English. We picked factoid type TREC questions, which expect an answer string of any of the 50 entity types. An example question from TREC and candidate text snippet from a document in the corpus containing the answer string is shown in Figure 2. When we used Ent-LDA with vicinity defined as 3 sentence window size, this text snippet was ranked high because the overlapping words (e.g., Rohm&Haas) of the question and the snippet has a ”Number:Count” entity type in its vicinity. On the other hand, standard LDA topic model did not rank this text snippet high because the probability of words being generated from this topic was not that high (since there is no semantically related words except the question topic word). With our Ent-LDA we can retrieve and rank text snippets more accurately compared to standard LDA models and our initial analysis show improvement in performance of our QA models.

Acknowledgements

We would like to thank Prof. Thomas Griffiths and Dr. Dilek Hakkani-Tur for initial discussions.

References

[1] Blei D. M. & Ng, A. Y. & Jordan, M. I. (2003) Latent Dirichlet allocation, Journal of Machine Learning Research 3: 993-1022.

[2] Buntine, W. & Lofstrom, J. & Perttu, S. & Valtonen, K. (2005) Topic specific scoring of documents for relevant retrieval, Workshop on Learning in Web Search: 22nd ICML'05, Germany.

[3] Celikyilmaz, A. & Thint, M., & Huang, Z.(2009) A Graph-based Semi-Supervised Learning for Question-Answering. ACL-2009, Main Conference, Singapore.

[4] Chemudugunta, C. & Smyth, P. & Steyvers, M. (2006) Modeling General and Specific Aspects of Documents with Probabilistic Topic Model, NIPS-2006.

[5] Lafferty, J. & McCallum, A. & Pereira, F. (2001) Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data, ICML'01.

[6] Steyvers, M. & Griffiths, T. (2006) Probabilistic topic models. In Landauer, T., McNamara, D., Dennis, S. and Kintsch, W. ed., *Latent Semantic Analysis: A road to meaning.* Laurence Erlbaum.

[7] Wei, X., & Croft, B. (2006) LDA-Based Document Models for Ad-hoc Retrieval, SIGIR'06.

[3] **T**ext **RE**trieval **C**onf.(TREC) QA track supports research on retrieving answers rather than document lists.