LDA Based Similarity Modeling for Question Answering

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

Dilek Hakkani-Tur
International Computer
Science Institute
Berkeley, CA
dilek@icsi.berkeley.edu

Gokhan Tur
Speech Technology and
Research Laboratory
SRI International
Menlo Park, CA, USA
gokhan@speech.sri.com

Abstract

We present an exploration of generative modeling for the question answering (QA) task to rank candidate passages. We investigate Latent Dirichlet Allocation (LDA) models to obtain ranking scores based on a novel similarity measure between a natural language question posed by the user and a candidate passage. We construct two models each one introducing deeper evaluations on latent characteristics of passages together with given question. With the new representation of topical structures on QA datasets, using a limited amount of world knowledge, we show improvements on performance of a QA ranking system.

1 Introduction

Question Answering (QA) is a task of automatic retrieval of an answer given a question. Typically the question is linguistically processed and search phrases are extracted, which are then used to retrieve the candidate documents, passages or sentences.

A typical QA system has a pipeline structure starting from extraction of candidate sentences to ranking true answers. Some approaches to QA use keyword-based techniques to locate candidate passages/sentences in the retrieved documents and then filter based on the presence of the desired answer type in candidate text. Ranking is then done using syntactic features to characterize similarity to query. In cases where simple question formulation is not satisfactory, many advanced QA systems implement more sophisticated syntactic, semantic and contextual processing such as named-entity recognition (Molla et al., 2006), coreference resolution (Vicedo and Ferrandez, 2000), logical inferences (abduction or entailment) (Harabagiu and Hickl, 2006) translation (Ma and McKeowon, 2009), etc., to improve answer ranking. For instance, *how* questions, or spatially constrained questions, etc., require such types of deeper understanding of the question and the retrieved documents/passages.

Many studies on QA have focused on discriminative models to predict a function of matching features between each question and candidate passage (set of sentences), namely **q/a pairs**, e.g., (Ng et al., 2001; Echihabi and Marcu, 2003; Harabagiu and Hickl, 2006; Shen and Klakow, 2006; Celikyilmaz et al., 2009). Despite their success, they have some room for improvement which are not usually raised, e.g., they require hand engineered features; or cascade features learnt separately from other modules in a QA pipeline, thus propagating errors. The structures to be learned can become more complex than the amount of training data, e.g., alignment, entailment, translation, etc. In such cases, other source of information, e.g., unlabeled examples, or human prior knowledge, should be used to improve performance. Generative modeling is a way of encoding this additional information, providing a natural way to use unlabeled data.

In this work, we present new similarity measures to discover deeper relationship between q/a pairs based on a probabilistic model. We investigate two methods using Latent Dirichlet Allocation (LDA) (Blei, 2003) in § 3, and hierarchical LDA (hLDA) (Blei, 2009) in § 4 to discover hidden concepts. We present ways of utilizing this information within a discriminative classifier in § 5. With empirical experiments in § 6, we analyze the effects of generative model outcome on a QA system. With the new representation of conceptual structures on QA

Proceedings of the NAACL HLT 2010 Workshop on Semantic Search, pages 1–9,
Los Angeles, California, June 2010.

datasets, using a limited amount of world knowledge, we show performance improvements.

2 Background and Motivation

Previous research have focused on improving modules of the QA pipeline such as question processing (Huang et al., 2009), information retrieval (Clarke et al., 2006), information extraction (Saggion and Gaizauskas, 2006). Recent work on textual entailment has shown improvements on QA results (Harabagiu and Hickl, 2006), (Celikyilmaz et al., 2009), when used for filtering and ranking answers. They discover similarities between q/a pairs, where the answer to a question should be entailed by the text that supports the correctness of its answer.

In this paper, we present a ranking schema focusing on a new similarity modeling approach via generative and discriminative methods to utilize best features of both approaches. Combinations of discriminative and generative methodologies have been explored by several authors, e.g. (Bouchard and Triggs, 2004; McCallum et al., 2006; Bishop and Lasserre, 2007; Schmah et al., 2009), in many fields such as natural language processing, speech recognition, etc. In particular, the recent "deep learning" approaches (Weston et al., 2008) rely heavily on a hybrid generative-discriminative approach: an unsupervised generative learning phase followed by a discriminative fine-tuning.

In an analogical way to the deep learning methods, we discover relations between the q/a pairs based on the similarities on their *latent topics* discovered via Bayesian probabilistic approach. We investigate different ways of discovering topic based similarities following the fact that it is more likely that the candidate passage entails given question and contains true answer if they share similar topics. Later we combine this information in different ways into a discriminative classifier-based QA model.

The underlying mechanism of our similarity modeling approach is Latent Dirichlet Allocation (LDA) (Blei et al., 2003b). We argue that similarities can be characterized better if we define a semantic similarity measure based on hidden concepts (topics) on top of lexico-syntactic features. We later extend our similarity model using a hierarchical LDA (hLDA) (Blei et al., 2003a) to discover latent topics that are organized into hierarchies. A hierarchical structure is particularly appealing to QA task than a flat LDA, in that one can discover abstract and specific topics. For example, discovering that *baseball* and *football* are both contained in a more abstract class *sports* can help to relate to a general topic of a question.

3 Similarity Modeling with LDA

We assume that for a question posed by a user, the document sets D are retrieved by a search engine based on the query expanded from the question. Our aim is to build a measure to characterize similarities between a given question and each candidate passage/sentence $s \in D$ in the retrieved documents based on similarities of their hidden topics. Thus, we built bayesian probabilistic models on passage level rather than document level to *explicitly* extract their hidden topics. Moreover, the fact that there is limited amount of retrieved documents D per question ($\sim$100 documents) makes it appealing to build probabilistic models on passages in place of documents and define semantically coherent groups in passages as *latent concepts*. Given window size n sentences, we define a passage as $s = (|D| - n) + 1$ based on a n-sliding-window, where $|D|$ is the total number of sentences in retrieved documents D. There are 25+ sentences in documents, hence we extracted around 2500 passages for each question.

3.1 LDA Model for Q/A System

We briefly describe LDA (Blei et al., 2003b) model as used in our QA system. A passage in retrieved documents (document collection) is represented as a mixture of fixed topics, with topic z getting weight $\theta_z^{(s)}$ in passage s and each topic is a distribution over a finite vocabulary of words, with word w having a probability $\phi_w^{(z)}$ in topic z. Placing symmetric Dirichlet priors on $\theta^{(s)}$ and $\phi^{(z)}$, with $\theta^{(s)} \sim Dirichlet(\alpha)$ and $\phi^{(z)} \sim Dirichlet(\beta)$, where α and β are hyper-parameters to control the sparsity of distributions, the generative model is given by:

$$\begin{array}{rcll} w_i|z_i, \phi_{w_i}^{(z_i)} & \sim & Discrete(\phi^{(z_i)}), & i = 1, ..., W \\ \phi^{(z)} & \sim & Dirichlet(\beta), & z = 1, ..., K \\ z_i|\theta^{(s_i)} & \sim & Discrete(\theta^{(s_i)}), & i = 1, ..., W \\ \theta^{(s)} & \sim & Dirichlet(\alpha), & s = 1, ..., S \end{array} \quad (1)$$

where S is the number of passages discovered from the document collection, K is the total number of topics, W is the total number of words in the document collection, and s_i and z_i are the passage and the topic of the ith word w_i, respectively. Each word in the vocabulary $w_i \in V = \{w_1, ...w_W\}$ is assigned to each latent topic variable $z_{i=1,...,W}$ of words.

After seeing the data, our goal is to calculate the expected posterior probabilities $\hat{\phi}_{w_i}^{(z_i)}$ of a word w_i in a candidate passage given a topic $z_i = k$ and expected posterior probability $\hat{\theta}^{(s)}$ of topic mixings of a given passage s, using the count matrices:

$$\hat{\phi}_{w_i}^{(z_i)} = \frac{n_{w_i k}^{WK} + \beta}{\sum_{j=1}^{W} n_{w_j k}^{WK} + W\beta} \quad \hat{\theta}^{(s)} = \frac{n_{sk}^{SK} + \alpha}{\sum_{j=1}^{K} n_{sj}^{SK} + K\alpha} \tag{2}$$

where $n_{w_i k}^{WK}$ is the count of w_i in topic k, and n_{sk}^{SK} is the count of topic k in passage s. The LDA model makes no attempt to account for the relation of topic mixtures, i.e., topics are distributed *flat*, and each passage is a distribution over all topics.

3.2 Degree of Similarity Between Q/A via Topics from LDA:

We build a LDA model on the set of retrieved passages s along with a given question q and calculate the *degree of similarity* $\text{DES}^{LDA}(q,s)$ between each q/a pair based on two measures (Algorithm 1):

(1) $\boldsymbol{sim}_1^{LDA}$: To capture the lexical similarities on hidden topics, we represent each s and q as two probability distributions at each topic $z = k$. Thus, we sample sparse unigram distributions from each $\hat{\phi}^{(z)}$ using the words in q and s. Each sparse word given topic distribution is denoted as $p_q^{(z)} = p(\mathbf{w}_q | z, \hat{\phi}^{(z)})$ with the set of words $\mathbf{w}_q = (w_1, ..., w_{|q|})$ in q and $p_s = p(\mathbf{w}_s | z, \hat{\phi}^{(z)})$ with the set of words $\mathbf{w}_s = (w_1, ..., w_{|s|})$ in s, and $z = 1...K$ represent each topic.

The sparse probability distributions per topic are represented with only the words in q and s, and the probabilities of the rest of the words in V are set to zero. The W dimensional word probabilities is the expected posteriors obtained from LDA model (Eq.(2)), $p_s^{(z)} = (\hat{\phi}_{w_1}^{(z)}, ..., \hat{\phi}_{w_{|s|}}^{(z)}, 0, 0, ..) \in (0,1)^W$, $p_q^{(z)} = (\hat{\phi}_{w_1}^{(z)}, ..., \hat{\phi}_{w_{|q|}}^{(z)}, 0, 0, ..) \in (0,1)^W$. Given a topic z, the similarity between $p_q^{(z)}$ and $p_s^{(z)}$ is measured via transformed information radius (*IR*). We

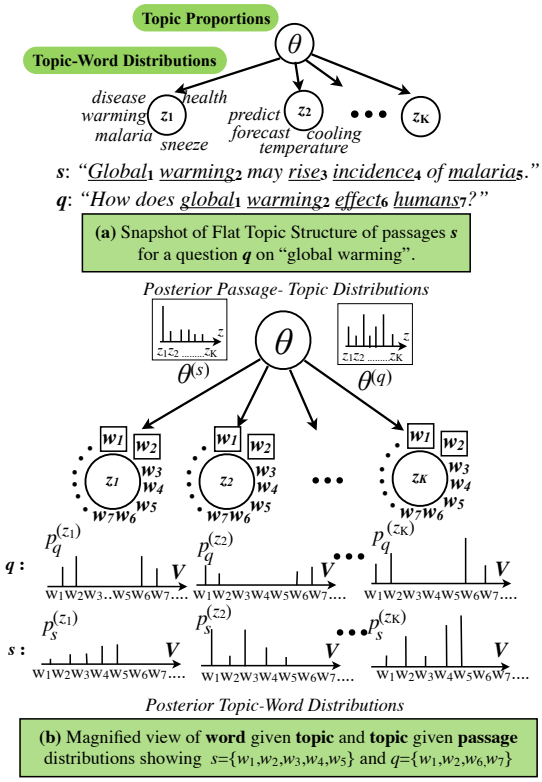

Figure 1: (a) The topic distributions of a passage s and a question q obtained from LDA. Each topic z_k is a distribution over words (Most probable terms are illustrated). (b) magnified view of (a) demonstrating sparse distributions over the vocabulary *V*, where only words in passage s and question q get values. The passage-topic distributions are topic mixtures, $\theta^{(s)}$ and $\theta^{(q)}$, for s and q.

first measure the divergence at each topic using *IR* based on Kullback-Liebler (*KL*) divergence:

$$IR(p_q^{(z)}, p_s^{(z)}) = KL(p_q^{(z)} || \frac{p_q^{(z)} + p_s^{(z)}}{2}) + KL(p_s^{(z)} || \frac{p_q^{(z)} + p_s^{(z)}}{2}) \tag{3}$$

where, $KL(p||q) = \sum_i p_i \log \frac{p_i}{q_i}$. The divergence is transformed into similarity measure (Manning and Schutze, 1999):

$$W(p_q^{(z)}, p_s^{(z)}) = 10^{-\delta IR(p_q^{(z)}, p_s^{(z)})}{}^{1} \tag{4}$$

To measure the similarity between probability distributions we opted for *IR* instead of commonly used *KL* because with *IR* there is no problem with infinite values since $\frac{p_q + p_s}{2} \neq 0$ if either $p_q \neq 0$ or $p_s \neq 0$, and it is also symmetric, *IR*(*p*,*q*)=*IR*(*q*,*p*). The similarity of q/a pairs on topic-word basis is the average

[1] In experiments $\delta = 1$ is used.

of transformed divergence over the entire *K* topics:

$$sim_1^{LDA}(q,s) = \frac{1}{K}\sum_{k=1}^{K} W(p_q^{(z=k)}, p_s^{(z=k)}) \quad (5)$$

(2) *sim*$_2^{LDA}$: We introduce another measure based on passage-topic mixing proportions in q and s to capture similarities between their topics using the transformed *IR* in Eq.(4) as follows:

$$sim_2^{LDA}(q,s) = 10^{-IR(\hat{\theta}^{(q)}, \hat{\theta}^{(s)})} \quad (6)$$

The $\hat{\theta}^{(q)}$ and $\hat{\theta}^{(s)}$ are K-dimensional discrete topic weights in question q and a passage s from Eq.(2). In summary, sim_1^{LDA} is a measure of lexical similarity on topic-word level and sim_2^{LDA} is a measure of topical similarity on passage level. Together they form the *degree of similarity* $DES^{LDA}(s,q)$ and are combined as follows:

$$DES^{LDA}(s,q) = sim_1^{LDA}(q,s) * sim_2^{LDA}(q,s) \quad (7)$$

Fig.1 shows sparse distributions obtained for sample q and s. Since the topics are not distributed hierarchially, each topic distribution is over the entire vocabulary of words in retrieved collection D. Fig.1 only shows the most probable words in a given topic. Moreover, each s and q are represented as a discrete probability distribution over all *K* topics.

Algorithm 1 Flat Topic-Based Similarity Model

1: Given a query q and candidate passages $s \in D$
2: Build an LDA model for the retrieved passages.
3: **for** each passages $s \in D$ **do**
4: - Calculate $sim_1(q,s)$ using Eq.(5)
5: - Calculate $sim_2(q,s)$ using Eq.(6)
6: - Calculate degree of similarity between q and s:
7: $DES^{LDA}(q,s) = sim_1(q,s) * sim_2(q,s)$
8: **end for**

4 Similarity Modeling with hLDA

Given a question, we discover hidden topic distributions using hLDA (Blei et al., 2003a). hLDA organizes topics into a tree of a fixed depth L (Fig.2.(a)), as opposed to *flat* LDA. Each candidate passage s is assigned to a path c_s in the topic tree and each word w_i in s is assigned to a hidden topic z_s at a level l of c_s. Each node is associated with a topic distribution over words. The Gibbs sampler (Griffiths and Steyvers, 2004) alternates between choosing a new path for each passage through the tree and assigning each word in each passage to a topic along that path. The structure of tree is learnt along with the topics using a nested Chinese restaurant process (nCRP) (Blei et al., 2003a), which is used as a prior.

The nCRP is a stochastic process, which assigns probability distributions to infinitely branching and deep trees. nCRP specifies a distribution of words in passages into paths in an L-level tree. Assignments of passages to paths are sampled sequentially: The first passage takes the initial L-level path, starting with a single branch tree. Next, mth subsequent passage is assigned to a path drawn from distribution:

$$\begin{aligned} p(path_{old}, c|m, m_c) &= \frac{m_c}{\gamma+m-1} \\ p(path_{new}, c|m, m_c) &= \frac{\gamma}{\gamma+m-1} \end{aligned} \quad (8)$$

$path_{old}$ and $path_{new}$ represent an existing and novel (branch) path consecutively, m_c is the number of previous passages assigned to path c, m is the total number of passages seen so far, and γ is a hyperparameter, which controls the probability of creating new paths. Based on this probability each node can branch out a different number of child nodes proportional to γ. The generative process for hLDA is:
(1) For each topic $k \in T$, sample a distribution $\beta_k \backsim$ Dirichlet(η).
(2) For each passage s in retrieved documents,
(a) Draw a path $c_s \backsim$ nCRP(γ),
(b) Sample L-vector θ_s mixing weights from Dirichlet distribution $\theta_s \sim Dir(\alpha)$.
(c) For each word n, choose :
(*i*) a level $z_{s,n}|\theta_s$, (*ii*) a word $w_{s,n}|\{z_{s,n}, c_s, \beta\}$

Given passage s, θ_s is a vector of topic proportions from L dimensional Dirichlet parameterized by α (distribution over levels in the tree.) The nth word of s is sampled by first choosing a level $z_{s,n} = l$ from the discrete distribution θ_s with probability $\theta_{s,l}$. Dirichlet parameter η and γ control the size of tree effecting the number of topics. Large values of η favor more topics (Blei et al., 2003a).

Model Learning: Gibbs sampling is a common method to fit the hLDA models. The aim is to obtain the following samples from the posterior of: (*i*) the latent tree T, (*ii*) the level assignment **z** for all words, (*iii*) the path assignments **c** for all passages conditioned on the observed words **w**.

Given the assignment of words **w** to levels **z** and assignments of passages to paths **c**, the expected

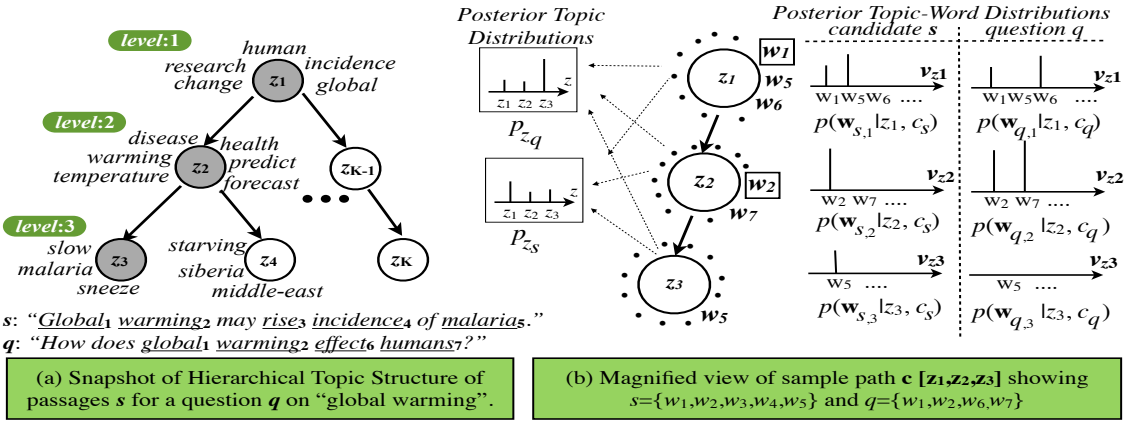

Figure 2: (a) A sample 3-level tree using hLDA. Each passage is associated with a path c through the hierarchy, where each node $z_s = l$ is associated with a distribution over terms (Most probable terms are illustrated). (b) magnified view of a path (darker nodes) in (a). Distribution of words in given passage s and a question (q) using sub-vocabulary of words at each level topic v_l. Discrete distributions on the left are topic mixtures for each passage, p_{z_q} and p_{z_s}.

posterior probability of a particular word w at a given topic $\mathbf{z}$=l of a path $\mathbf{c}$=c is proportional to the number of times w was generated by that topic:

$$p(w|\mathbf{z}, \mathbf{c}, \mathbf{w}, \eta) \propto n_{(\mathbf{z}=l,\mathbf{c}=c,\mathbf{w}=w)} + \eta \qquad (9)$$

Similarly, posterior probability of a particular topic z in a given passage s is proportional to number of times z was generated by that passage:

$$p(z|s, \mathbf{z}, \mathbf{c}, \alpha) \propto n_{(\mathbf{c}=c_c,\mathbf{z}=l)} + \alpha \qquad (10)$$

$n_{(.)}$ is the count of elements of an array satisfying the condition. Posterior probabilities are normalized with total counts and their hyperparameters.

4.1 Tree-Based Similarity Model

The hLDA constructs a hierarchical tree structure of candidate passages and given question, each of which are represented by a path in the tree, and each path can be shared by many passages/question. The assumption is that passages sharing the same path should be more similar to each other because they share the same topics (Fig.2). Moreover, if a path includes a question, then other passages on that path are more likely to entail the question than passages on the other paths. Thus, the similarity of a candidate passage s to a question q sharing the same path is a measure of semantic similarity (Algorithm 2). Given a question, we build an hLDA model on retrieved passages. Let c_q be the path for a given q. We identify the candidate passages that share the same path with q, $M = \{s \in D | c_s = c_q\}$. Given path c_q and M, we calculate the degree of similarity $DES^{hLDA}(s, q)$ between q and s by calculating two similarity measures:

(1) sim_1^{hLDA}: We define two sparse (discrete) unigram distributions for candidate s and question q at each node l to define lexical similarities on topic level. The distributions are over a vocabulary of words generated by the topic at that node, $v_l \subset V$. Note that, in hLDA the topic distributions at each level of a path is sampled from the vocabulary of passages sharing that path, contrary to LDA, in which the topics are over entire vocabulary of words. This enables defining a similarity measure on specific topics. Given $\mathbf{w}_q = \{w_1, ..., w_{|q|}\}$, let $\mathbf{w}_{q,l} \subset \mathbf{w}_q$ be the set of words in q that are generated from topic $\mathbf{z}_q$ at level l on path c_q. The discrete unigram distribution $p_{ql} = p(\mathbf{w}_{q,l}|\mathbf{z}_q = l, c_q, v_l)$ represents the probability over all words v_l assigned to topic $\mathbf{z}_q$ at level l, by sampling only for words in $\mathbf{w}_{q,l}$. The probability of the rest of the words in v_l are set 0. Similarly, $p_{s,l} = p(\mathbf{w}_{s,l}|\mathbf{z}_s, c_q, v_l)$ is the probability of words $\mathbf{w}_s$ in s extracted from the same topic (see Fig.2.b). The word probabilities in $p_{q,l}$ and $p_{s,l}$ are obtained using Eq. (9) and then normalized.

The similarity between $p_{q,l}$ and $p_{s,l}$ at each level is obtained by transformed information radius:

$$W_{c_{q,l}}(p_{q,l}, p_{s,l}) = 10^{\delta\text{-}IR_{c_{q,l}}(p_{q,l},p_{s,l})} \qquad (11)$$

where the $IR_{c_{q,l}}(p_{q,l}, p_{s,l})$ is calculated as in Eq.(3) this time for $p_{q,l}$ and $p_{s,l}$ ($\delta = 1$). Finally sim_1^{hLDA} is obtained by averaging Eq.(11) over different levels:

$$sim_1^{hLDA}(q,s) = \frac{1}{L}\sum_{l=1}^{L} W_{c_q,l}(p_{q,l}, p_{s,l}) * l \quad (12)$$

The similarity between $p_{q,l}$ and $p_{s,l}$ is weighted by the level l because the similarity should be rewarded if there is a specific word overlap at child nodes.

Algorithm 2 Tree-Based Similarity Model

1: Given candidate passages s and question q.
2: Build hLDA on set of s and q to obtain tree T.
3: Find path c_q on tree T and candidate passages
4: on path c_q, i.e., $M = \{s \in D | c_s = c_q\}$.
5: **for** candidate passage $s \in M$ **do**
6: Find $DES^{hDLA}(q,s) = sim_1^{hLDA} * sim_2^{hLDA}$
7: using Eq.(12) and Eq.(13)
8: **end for**
9: **if** $s \notin M$, **then** $DES^{hDLA}(q,s)$=0.

(2) sim_2^{hLDA}: We introduce a concept-base measure based on passage-topic mixing proportions to calculate the topical similarities between q and s. We calculate the topic proportions of q and s, represented by $p_{z_q} = p(\mathbf{z}_q|c_q)$ and $p_{z_s} = p(\mathbf{z}_s|c_q)$ via Eq.(10). The similarity between the distributions is then measured with transformed IR as in Eq.(11) by:

$$sim_2^{hLDA}(q,s) = 10^{-IR_{c_q}(p_{z_q}, p_{z_s})} \quad (13)$$

In summary, sim_1^{hLDA} provides information about the similarity between q and s based on topic-word distributions, and sim_2^{hLDA} is the similarity between the weights of their topics. The two measures are combined to calculate the degree of similarity:

$$DES^{hLDA}(q,s) = sim_1^{hLDA}(q,s) * sim_2^{hLDA}(q,s) \quad (14)$$

Fig.2.b depicts a sample path illustrating sparse unigram distributions of a q and s at each level and their topic proportions, p_{z_q}, and p_{z_s}. The candidate passages that are not on the same path as the question are assigned $DES^{hLDA}(s,q) = 0$.

5 Discriminitive Model for QA

In (Celikyilmaz et al., 2009), the QA task is posed as a textual entailment problem using lexical and semantic features to characterize similarities between q/a pairs. A discriminative classifier is built to predict the existence of an answer in candidate sentences. Although they show that semi-supervised methods improve accuracy of their QA model under limited amount of labeled data, they suggest that with sufficient number of labeled data, supervised methods outperform semi-supervised methods. We argue that there is a lot to discover from unlabeled text to help improve QA accuracy. Thus, we propose using Bayesian probabilistic models. First we briefly present the baseline method:

`Baseline`: We use the supervised classifier model presented in (Celikyilmaz et al., 2009) as our baseline QA model. Their datasets, provided in http://www.eecs.berkeley.edu/∼asli/asliPublish.html, are q/a pairs from TREC task. They define each q/a pair as a d dimensional feature vector $x_i \in \Re^d$ characterizing entailment information between them. They build a support vector machine (SVM) (Drucker et al., 1997) classifier model to predict the entailment scores for q/a pairs.

To characterize the similarity between q/a pairs they use: (*i*) features represented by similarities between semantic components, e.g., subject, object, verb, or named-entity types discovered in q/a pairs, and (*ii*) lexical features represented by lexico-syntactic alignments such as n-gram word overlaps or cause and entailment relations discovered from WordNet (Miller, 1995). For a given question q, they rank the candidate sentences s based on predicted entailment scores from the classifier, $TE(q,s)$.

We extend the baseline by using the degree of similarity between question and candidate passage obtained from LDA, $DES^{LDA}(q,s)$, as well as hLDA $DES^{hLDA}(q,s)$, and evaluate different models:

`Model M-1`: Degree of Similarity as Rank Scores: In this model, the QA is based on a fully generative approach in which the similarity measures of Eq.(7) in §3 and Eq.(14) in §4 are used to obtain ranking scores. We build two separate models, `M-1.1` using $DES^{LDA}(q,s)$, and `M-1.2` using $DES^{hLDA}(q,s)$ as rank scores and measure accuracy by re-ranking candidate passages accordingly. Given a question, this model requires training individual LDA and hLDA models.

`Model M-2`: Interpolation Between Classifier-Based Entailment Scores and Generative Model Scores: In this model, the underlying

mechanism of QA is the discriminative method presented in baseline. We linearly combine the probabilistic similarity scores from generative models, *DES* scores in M-1, with the baseline scores. We build two additional models to calculate the final rank scores; M-2.1 using:

$$score(s|q) = a*TE(q,s)+b*DES^{LDA}(q,s) \quad (15)$$

and M-2.2 using:

$$score(s|q) = a*TE(q,s)+b*DES^{hLDA}(q,s) \quad (16)$$

where $0 \leq a \leq 1$ and $0 \leq b \leq 1$ and $a+b=1$. We find the optimum a^* and b^* based on the validation experiments on training dataset. The candidate sentences are re-ranked based on these scores.

Model M-3: Degree of Similarity as Entailment Features: Another way to incorporate the latent information into the discriminitive QA model is to utilize the latent similarities as explanatory variables in the classifier model. Particularly we build M-3.1 by using sim_1^{LDA}, sim_2^{LDA} as well as $DES^{LDA}(q,s)$ as additional features for the SVM, on top of the the existing features used in (Celikyilmaz et al., 2009). Similarly, we build M-3.2 by using sim_1^{hLDA}, sim_2^{hLDA} as well as $DES^{hLDA}(q,s)$ as additional features to the SVM classifier model to predict entailment scores. This model requires building two new SVM classifier models with the new features.

6 Experiments and Discussions

We demonstrate the results of our experiments on exploration of the effect of different generative models presented in §5 on TREC QA datasets.

We performed experiments on the datasets used in (Celikyilmaz et al., 2009). Their train dataset composes of a set of 1449 questions from TREC-99-03. For each question, the 5 top-ranked candidate sentences are extracted from a large newswire corpora (Acquaint corpus) through a search engine, i.e., Lucene [2]. The q/a pairs are labeled as true/false depending on the containment of the true answer string in retrieved passages. Additionally, to calculate the LDA and hLDA similarity measures for each candidate passage, we also extract around 100 documents in the same fashion using Lucene and identify passages to build the probabilistic models. We calculate the probabilistic similarities, i.e., sim_1^{LDA}, sim_2^{LDA}, sim_1^{hLDA}, sim_2^{hLDA}, and the degree of similarity values, i.e., $DES^{LDA}(q,s)$ and $DES^{hLDA}(q,s)$ for each of the 5 top-ranked candidate sentences in training dataset at inference time. Around 7200 q/a pairs are compiled accordingly.

The provided testing data contains a set of 202 questions from TREC2004 along with 20 candidate sentences for each question, which are labeled as true/false. To calculate the similarities for the 20 candidate sentences, we extract around 100 documents for each question and build LDA and hLDA models. 4037 testing q/a pairs are compiled.

We report the retrieval performance of our models in terms of Mean Reciprocal Rank (MRR), top 1 (Top1) and top 5 prediction accuracies (Top5) (Voorhees, 2004). We performed parameter optimization during training based on prediction accuracy to find the best $C = \{10^{-2}, .., 10^{2}\}$ and $\Gamma = \{2^{-2}, .., 2^{3}\}$ for RBF kernel SVM. For the LDA models we present the results with 10 topics. In hLDA models, we use four levels for the tree construction and set the topic Dirichlet hyperparameters in decreasing order of levels at $\eta = \{1.0, 0.75, 0.5, 0.25\}$ to encourage as many terms in the mid to low levels as the higher levels in the hierarchy, for a better comparison between q/a pairs. The nested CRP parameter γ is fixed at 1.0. We evaluated n-sliding-window size of sentences in sequence, $n = \{1, 3, 5\}$, to compile candidate passages for probabilistic models (Table 1). The output scores for SVM models are normalized to [0,1].

⋆ As our baseline (in §5), we consider supervised classifier based QA presented in (Celikyilmaz et al., 2009). The baseline MRR on TREC-2004 dataset is MRR=%67.6, Top1=%58, Top5=%82.2.

⋆ The results of the new models on testing dataset are reported in Table 1. Incorporating the generative model output to the classifier model as input features, i.e., M-3.1 and M-3-2, performs consistently better than the rest of the models and the baseline, where MRR result is statistically significant based on t-test statistics (at $p = 0.95$ confidence level). When combined with the textual entailment scores, i.e., M-2.1 and M-2.2, they provide a slightly better ranking, a minor improvement compared to the baseline. However, using the generative model outcome as sole ranking scores in

[2] http://lucene.apache.org/java/

	Window-size	1-window			3-window			5-window		
	MRR categories	MRR	Top1	Top5	MRR	Top1	Top5	MRR	Top1	Top5
Models	M-1.1 (with LDA)	42.7	30.2	64.4	42.1	30.2	64.4	42.1	30.2	64.4
	M-1.1 (with hLDA)	55.8	45.5	71.0	55.8	45.5	71.0	54.9	45.5	71.0
	M-2.1 (with LDA)	66.2	55.1	82.2	65.2	54.5	80.7	65.2	54.5	80.7
	M-2.2 (with hLDA)	68.2	58.4	82.2	67.6	58.0	82.2	67.4	58.0	81.6
	M-3.1 (with LDA)	68.0	61.0	82.2	68.0	58.1	82.2	68.2	58.1	82.2
	M-3.2 (with hLDA)	68.4	**63.4**	82.2	68.3	**61.0**	82.2	68.3	**61.0**	82.2

Table 1: The MRR results of the models presented in §5 on testing dataset (TREC 2004) using different window sizes of candidate passages. The statistically significant model results in each corresponding MRR category are bolded. Baseline MRR=%67.6, Top1=%58, Top5=%82.2.

M-1.1 and M-1.2 do not reveal as good results as the other models, suggesting room for improvement.

⋆ In Table 1, Top1 MRR yields better improvement compared to the other two MRRs, especially for models M-3.1 and M-3.2. This suggests that the probabilistic model outcome rewards the candidate sentences containing the true answer by estimating higher scores and moves them up to the higher levels of the rank.

⋆ The analysis of different passage sizes suggest that the 1-window size yields best results and no significant performance improvement is observed when window size is increased. Thus, the similarity between q/a pairs can be better explained if the candidate passage contains less redundant sentences.

⋆ The fact that the similarity scores obtained from the hLDA models are significantly better than LDA models in Table 1 indicates an important property of hierarchal topic models. With the hLDA specific and generic topics can be identified on different levels of the hierarchy. Two candidate passages can be characterized with different abstract and specific topics (Fig. 2) enabling representation of better features to identify similarity measures between them. Whereas in LDA, each candidate passage has a proportion in each topic. Rewarding the similarities on specific topics with the hLDA models help improve the QA rank performance.

⋆ In M-3.1 and M-3.2 we use probabilistic similarities and *DES* as inputs to the classifier. In Table 2 we show the individual effects of these features on the MRR testing performance along with other lexical and semantic features of the baseline. Although the effect of each feature is comparable, the DES^{LDA} and DES^{hLDA} features reveal slightly better results.

Features	M-3.1	Features	M-3.1
$sim1^{LDA}$	67.7	$sim1^{hLDA}$	67.8
$sim2^{LDA}$	67.5	$sim2^{hLDA}$	68.0
DES^{LDA}	67.9	DES^{hLDA}	68.1

Table 2: The MRR results of the similarity measures on testing dataset (TREC 2004) when used as input features.

7 Conclusion and Future Work

In this paper we introduced a set of methods based on Latent Dirichlet Allocation (LDA) to characterize the similarity between the question and the candidate passages, which are used as ranking scores. The results of our experiments suggest that extracting information from hidden concepts improves the results of a classifier-based QA model.

Although unlabeled data exploration through probabilistic graphical models can help to improve information extraction, devising a machinery with suitable generative models for the given natural language task is a challenge. This work helps with such understanding via extensive simulations and puts forward and confirms a hypothesis explaining the mechanisms behind the effect of unsupervised pre-training for the final discriminant learning task.

In the future, we would like to further evaluate the models presented in this paper for larger datasets and for different tasks such as question paraphrase retrieval or query expansion. Moreover, we would like to enhance the similarities with other semantic components extracted from questions such as question topic and question focus.

References

C. M. Bishop and J. Lasserre. Generative or discriminative? getting the best of both worlds. In *In Bayesian Statistics 8, Bernardo, J. M. et al. (Eds), Oxford University Press*, 2007.

D. Blei, T. Griffiths, M. Jordan, and J. Tenenbaum. Hierarchical topic models and the nested chinese restaurant process. In *In Neural Information Processing Systems [NIPS]*, 2003a.

D. M. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. In *Jrnl. Machine Learning Research, 3:993-1022*, 2003b.

G. Bouchard and B. Triggs. The tradeoff between generative and discriminative classifiers. In *Proc. of COMPSTAT'04*, 2004.

A. Celikyilmaz, M. Thint, and Z. Huang. Graph-based semi-supervised learning for question answering. In *Proc. of the ACL-2009*, 2009.

C.L.A. Clarke, G. V. Cormack, R. T. Lynam, and E. L. Terra. Question answering by passage selection. In *In: Advances in open domain question answering, Strzalkowski, and Harabagiu (Eds.)*, pages 259–283. Springer, 2006.

H. Drucker, C.J.C. Burger, L. Kaufman, A. Smola, and V. Vapnik. Support vector regression machines. In *NIPS 9*, 1997.

A. Echihabi and D. Marcu. A noisy-channel approach to question answering. In *ACL-2003*, 2003.

T. Griffiths and M. Steyvers. Finding scientific topics. In *PNAS, 101(Supp. 1): 5228-5235*, 2004.

S. Harabagiu and A. Hickl. Methods for using textual entailment in open-domain question answering. In *In Proc. of ACL-2006*, pages 905–912, 2006.

Z. Huang, M. Thint, and A. Celikyilmaz. Investigation of question classifier in question answering. In *In EMNLP'09*, 2009.

W.-Y. Ma and K. McKeowon. Where's the verb? correcting machine translation during question answering. In *In ACL-IJCNLP'09*, 2009.

C. Manning and H. Schutze. Foundations of statistical natural language processing. In *MIT Press. Cambridge, MA*, 1999.

A. McCallum, C. Pal, G. Druck, and X. Wang. Multi-conditional learning: Generative/discriminative training for clustering and classification. In *AAAI 2006*, 2006.

G.A. Miller. Wordnet: A lexical database for english. In *ACM*, 1995.

D. Molla, M.V. Zaanen, and D. Smith. Named entity recognition for question answering. In *In ALTW2006*, 2006.

H.T. Ng, J.L.P. Kwan, and Y. Xia. Question answering using a large text database: A machine learning approach. In *EMNLP-2001*, 2001.

H. Saggion and R. Gaizauskas. Experiments in passage selection and answer extraction for question answering. In *In: Advances in open domain question answering, Strzalkowski, and Harabagiu (Eds.)*, pages 291–302. Springer, 2006.

T. Schmah, G. E Hinton, R. Zemel, S. L. Small, and S. Strother. Generative versus discriminative training of rbms for classification of fmri images. In *Proc. NIPS 2009*, 2009.

Dan Shen and Dietrich Klakow. Exploring correlation of dependency relation paths for answer extraction. In *Proc. of ACL-2006*, 2006.

J.L. Vicedo and A. Ferrandez. Applying anaphora resolution to question answering and information retrieval systems. In *In LNCS*, volume 1846, pages 344–355, 2000.

Ellen M. Voorhees. Overview of trec2004 question answering track. 2004.

J. Weston, F. Rattle, and R. Collobert. Deep learning via semi-supervised embedding. In *ICML*, 2008.