Semantic Topic Models: Combining Word Distributional Statistics and Dictionary Definitions

Weiwei Guo
Department of Computer Science,
Columbia University,
weiwei@cs.columbia.edu

Mona Diab
Center for Computational Learning Systems,
Columbia University,
mdiab@ccls.columbia.edu

Abstract

In this paper, we propose a novel topic model based on incorporating dictionary definitions. Traditional topic models treat words as surface strings without assuming predefined knowledge about word meaning. They infer topics only by observing surface word co-occurrence. However, the co-occurred words may not be semantically related in a manner that is relevant for topic coherence. Exploiting dictionary definitions explicitly in our model yields a better understanding of word semantics leading to better text modeling. We exploit WordNet as a lexical resource for sense definitions. We show that explicitly modeling word definitions helps improve performance significantly over the baseline for a text categorization task.

1 Introduction

Latent Dirichlet Allocation (LDA) (Blei et al., 2003) serves as a data-driven framework in modeling text corpora. The statistical model allows variable extensions to integrate linguistic features such as syntax (Griffiths et al., 2005), and has been applied in many areas.

In LDA, there are two factors which determine the topic of a word: the topic distribution of the document, and the probability of a topic to emit this word. This information is learned in an unsupervised manner to maximize the likelihood of the corpus. However, this data-driven approach has some limitations. If a word is not observed frequently enough in the corpus, then it is likely to be assigned the dominant topic in this document. For example, the word *grease (a thick fatty oil)* in a political domain document should be assigned the topic *chemicals*. However, since it is an infrequent word, LDA cannot learn its correct semantics from the observed distribution, the LDA model will assign it the dominant document topic *politics*. If we look up the semantics of the word *grease* in a dictionary, we will not find any of its meanings indicating the *politics* topic, yet there is ample evidence for the *chemical* topic. Accordingly, we hypothesize that if we know the semantics of words in advance, we can get a better indication of their topics. Therefore, in this paper, we test our hypothesis by exploring the integration of word semantics explicitly in the topic modeling framework.

In order to incorporate word semantics from dictionaries, we recognize the need to model sense-topic distribution rather than word-topic distribution, since dictionaries are constructed at the sense level. We use WordNet (Fellbaum, 1998) as our lexical resource of choice. The notion of a sense in WordNet goes beyond a typical word sense in a traditional dictionary since a WordNet sense links senses of different words that have similar meanings. Accordingly, the sense for the first verbal entry for *buy* and for *purchase* will have the same sense id (and same definition) in WordNet, while they could have different meaning definitions in a traditional dictionary such as the Merriam Webster Dictionary or LDOCE. In our model, a topic will first emit a WordNet sense, then the sense will generate a word. This is inspired by the intuition that words are instantiations of concepts.

The paper is organized as follows: In Sections 2 and 3, we describe our models based on WordNet. In Section 4, experiment results on text categorization are presented. Moreover, we analyze both qualitatively and quantitatively the contribution of modeling definitions (by teasing out the contribution of explicit sense modeling in a word sense disambiguation task). Related work is introduced in Section 5. We conclude in Section 6 by discussing some possible future directions.

Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pages 552–561,
Edinburgh, Scotland, UK, July 27–31, 2011. ©2011 Association for Computational Linguistics

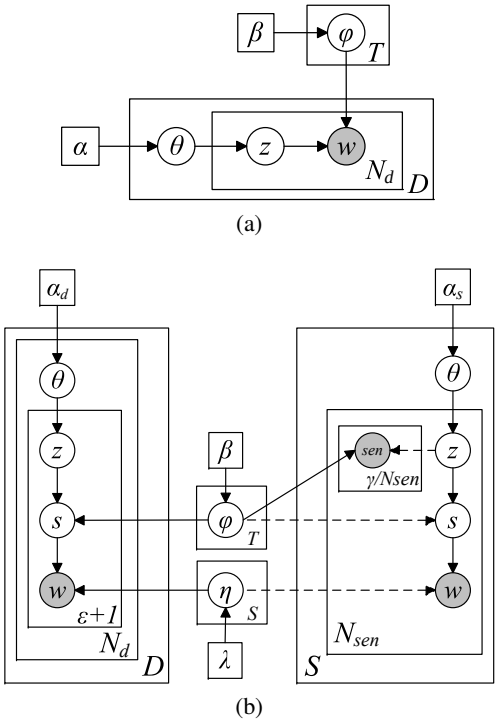


Figure 1: (a) LDA: Latent Dirichlet Allocation (Blei et al., 2003). (b) STM: Semantic topic model. The dashed arrows indicate the distributions (ϕ and η) and nodes (z) are not influenced by the values of pointed nodes.

2 Semantic Topic Model

2.1 Latent Dirichlet Allocation

We briefly introduce LDA where Collapsed Gibbs Sampling (Griffiths and Steyvers, 2004) is used for inference. In figure 1a, given a corpus with D documents, LDA will summarize each document as a normalized T-dimension topic mixture θ. Topic mixture θ is drawn from a Dirichlet distribution $Dir(\alpha)$ with a symmetric prior α. ϕ contains T multinomial distribution, each representing the probability of a topic z generating word w $p(w|z)$. ϕ is drawn from a Dirichlet distribution $Dir(\beta)$ with prior β.

In Collapsed Gibbs Sampling, the distribution of a topic for the word $w_i = w$ based on values of other data is computed as:

$$P(z_i = z|\mathbf{z_{-i}}, \mathbf{w}) \propto \frac{n^{(d)}_{-i,z} + \alpha}{n^{(d)}_{-i} + T\alpha} \times \frac{n^{w}_{-i,z} + \beta}{n_{-i,z} + W\beta} \quad (1)$$

In this equation, $n^{(d)}_{-i,z}$ is a count of how many words are assigned topic z in document d, excluding the topic of the ith word; $n^{w}_{-i,z}$ is a count of how many words $= w$ are assigned topic z, also excluding the topic of the ith word. Hence, the first fraction is the proportion of the topic in this document $p(z|\theta)$. The second fraction is the probability of topic z emitting word w. After the topics become stable, all the topics in a document construct the topic mixture θ.

2.2 Applying Word Sense Disambiguation Techniques

We add a sense node between the topic node and the word node based on two linguistic observations: a) **Polysemy**: many words have more than one meaning. A topic is more directly relevant to a word meaning (sense) than to a word due to polysemy; b) **Synonymy**: different words may share the same sense. WordNet explicitly models synonymy by linking synonyms to the same sense. In WordNet, each sense has an associated definition.

It is worth noting that we model the sense-word relation differently from (Boyd-Graber and Blei, 2007), where in their model words are generated from topics, then senses are generated from words. In our model, we assume that during the generative process, the author picks a concept relevant to the topic, then thinks of a best word that represents that concept. Hence the word choice is dependent on the relatedness of the sense and its fit to the document context.

In standard topic models, the topic of a word is sampled from the document level topic mixture θ. The underlying assumption is that all words in a document constitute the context of the target word. However, it is not the case in real world corpora. Titov and McDonald (2008) find that using global topic mixtures can only extract global topics in online reviews (e.g., Creative Labs MP3 players and iPods) and ignores local topics (product features such as portability and battery). They design the Multi-grain LDA where the local topic of a word is only determined by topics of surrounding sentences. In word sense disambiguation (WSD), an even narrower context is taken into consideration, for instance in graph based WSD models (Mihalcea, 2005), the choice of a sense for a word only depends on a local window whose size equals the length of the sentence. Later in (Sinha and Mihalcea, 2007; Guo and Diab, 2010; Li et al., 2010), people use a fixed window size containing around 12 neighbor words for WSD.

Accordingly, we adopt the WSD inspired local window strategy in our model. However, we do

not employ the complicated schema in (Titov and McDonald, 2008). We simply hypothesize that the surrounding ϵ words are semantically related to the considered word, and they construct a local sliding window for that target word. For a document d with N_d words, we represent it as N_d local windows – a window is created for each word. The model is illustrated in the left rectangle in figure 1b. The window size is fixed for each word: it contains $\epsilon/2$ preceding words, and $\epsilon/2$ following words. Therefore, a word in the original document will have ϵ copies, existing in $\epsilon+1$ local windows. Similarly, there are $\epsilon + 1$ pairs of topics/senses assigned for each word in the original document. Each window has a distribution θ_i over topics. θ_i will emit the topics of words in the window.

This approach enables us to exploit different context sizes without restricting it to the sentence length, and hence spread topic information across sentence boundaries.

2.3 Integrating Definitions

Intuitively, a sense definition reveals some prior knowledge on the topic domain: the definition of sense [*crime, offense, offence*] indicates a *legal* topic; the definition of sense [*basketball*] indicates a *sports* topic, etc. Therefore, during inference, we want to choose a topic/sense pair for each word, such that the topic is supported by the context θ and the sense definition also matches that topic.

Given that words used in the sense definitions are strongly relevant to the sense/concept, we set out to find the topics of those definition words, and accordingly assign the sense sen itself these topics. We treat a sense definition as a document and perform Gibbs sampling on it. We normalize definition length by a variable γ. Therefore, before the topic model sees the actual documents, each sense s has been sampled γ times. The γ topics are then used as a "training set", so that given a sense, ϕ has some prior knowledge of which topic it should be sampled from.

Consider the sense [*party, political party*] with a definition "an organization to gain political power" of length 6 when $\gamma = 12$. If topic model assigns *politics* topic to the words "organization political power", then sense [*party, political party*] will be sampled from $politics$ topic for $3 * \gamma/definitionLength = 6$ times.

We refer to the proposed model as Semantic Topic Model (figure 1b). For each window v_i in the document set, the model will generate a distribution of topics θ_i. It will emit the topics of $\epsilon + 1$ words in the window. For a word w_{ij} in window v_i, a sense s_{ij} is drawn from the topic, and then s_{ij} generates the word w_i. Sense-topic distribution ϕ contains T multinomial distributions over all possible senses in the corpus drawn from a symmetric Dirichlet distribution $Dir(\beta)$. From WordNet we know the set of words $W(s)$ that have a sense s as an entry. A sense s can only emit words from $W(s)$. Hence, for each sense s, there is a multinomial distribution η_s over $W(s)$. All η are drawn from symmetric $Dir(\lambda)$.

On the definition side, we use a different prior α_s to generate a topic mixture θ. Aside from generating s_i, z_i will deterministically generate the current sense sen for γ/N_{sen} times (N_{sen} is the number of words in the definition of sense sen), so that sen is sampled γ times in total.

The formal procedure of generative process is the following:

For the definition of sense sen:
- choose topic mixture $\theta \sim Dir(\alpha_s)$.
- for each word w_i:
  - choose topic $z_i \sim Mult(\theta)$.
  - choose sense $s_i \sim Mult(\phi_{z_i})$.
  - deterministically choose sense $sen \sim Mult(\phi_{z_i})$ for γ/N_{sen} times.
  - choose word $w_i \sim Mult(\eta_{s_i})$.

For each window v_i in a document:
- choose local topic mixture $\theta_i \sim Dir(\alpha_d)$.
- for each word w_{ij} in v_i:
  - choose topic $z_{ij} \sim Mult(\theta_i)$.
  - choose sense $s_{ij} \sim Mult(\phi_{z_{ij}})$.
  - choose word $w_{ij} \sim Mult(\eta_{s_{ij}})$.

2.4 Using WordNet

Since definitions and documents are in different genre/domains, they have different distributions on senses and words. Besides, the definition sets contain topics from all kinds of domains, many of which are irrelevant to the document set. Hence we prefer ϕ and η that are specific for the document set, and we do not want them to be "corrupted" by the text in the definition set. Therefore, as in figure 1b, the dashed lines indicate that when we estimate ϕ and η, the topic/sense pair and sense/word pairs in the definition set are not considered.

WordNet senses are connected by relations such as synonymy, hypernymy, similar attributes, etc.

We observe that neighboring sense definitions are usually similar and are in the same topic domain. Hence, we represent the definition of a sense as the union of itself with its neighboring sense definitions pertaining to WordNet relations. In this way, the definition gets richer as it considers more data for discovering reliable topics.

3 Inference

We still use Collapsed Gibbs Sampling to find latent variables. Gibbs Sampling will initialize all hidden variables randomly. In each iteration, hidden variables are sequentially sampled from the distribution conditioned on all the other variables. In order to compute the conditional probability $P(z_i = z, s_i = s|\mathbf{z_{-i}}, \mathbf{s_{-i}}, \mathbf{w})$ for a topic/sense pair, we start by computing the joint probability $P(\mathbf{z}, \mathbf{s}, \mathbf{w}) = P(\mathbf{z})P(\mathbf{s}|\mathbf{z})P(\mathbf{w}|\mathbf{s})$. Since the generative processes are not exactly the same for definitions and documents, we need to compute the joint probability differently. We use a type specific subscript to distinguish them: $P_s(\cdot)$ for sense definitions and $P_d(\cdot)$ for documents.

Let sen be a sense. Integrating out θ we have:

$$P_s(\mathbf{z}) = \left(\frac{\Gamma(T\alpha_s)}{\Gamma(\alpha_s)^T}\right)^S \prod_{sen=1}^{S} \frac{\prod_z \Gamma(n_z^{(sen)} + \alpha_s)}{\Gamma(n^{(sen)} + T\alpha)} \tag{2}$$

where $n_z^{(sen)}$ means the number of times a word in the definition of sen is assigned to topic z, and $n^{(sen)}$ is the length of the definition. S is all the potential senses in the documents.

We have the same formula of $P(\mathbf{s}|\mathbf{z})$ and $P(\mathbf{w}|\mathbf{s})$ for definitions and documents. Similarly, let n_z be the number of words in the documents assigned to topic z, and n_z^s be the number of times sense s assigned to topic z. Note that when s appears in the superscript surrounded by brackets such as $n_z^{(s)}$, it denotes the number of words assigned to topics z in the definition of sense s. By integrating out ϕ we obtain the second term:

$$P(\mathbf{s}|\mathbf{z}) = \left(\frac{\Gamma(S\beta)}{\Gamma(\beta)^S}\right)^T \prod_{z=1}^{T} \frac{\prod_s \Gamma(n_z^s + n_z^{(s)}\gamma/n^{(s)} + \beta)}{\Gamma(n_z + \sum_{s'} n_z^{(s')}\gamma/n^{(s')} + S\beta)} \tag{3}$$

At last, assume n_s denotes the number of sense s in the documents, and n_s^w denotes the number of sense s to generate the word w, then integrating out η we have:

$$P(\mathbf{w}|\mathbf{s}) = \prod_{s=1}^{S} \frac{\Gamma(|W(s)|\lambda)}{\Gamma(\lambda)^{|W(s)|}} \frac{\prod_w^{W(s)} \Gamma(n_s^w + \lambda)}{\Gamma(n_s + |W(s)|\lambda)}$$

With equation 2-4, we can compute the conditional probability $P_s(z_i = z, s_i = s|\mathbf{z_{-i}}, \mathbf{s_{-i}}, \mathbf{w})$ for a sense-topic pair in the sense definition. Let sen_i be the sense definition containing word w_i, then we have:

$$P_s(z_i = z, s_i = s|\mathbf{z_{-i}}, \mathbf{s_{-i}}, \mathbf{w}) \propto \frac{n_{-i,z}^{(sen_i)} + \alpha_s}{n_{-i}^{(sen_i)} + T\alpha_s} \frac{n_z^s + n_{-i,z}^{(s')}\gamma/n^{(s')} + \beta}{n_z + \sum_{s'} n_{-i,z}^{(s')}\gamma/n^{(s')} + S\beta} \frac{n_s^w + \lambda}{n_s + |W(s)|\lambda} \tag{5}$$

The subscript $-i$ in expression n_{-i} denotes the number of certain events excluding word w_i. Hence the three fractions in equation 5 correspond to the probability of choosing z from θ_{sen}, choosing s from z and choosing w from s. Also note that our model defines s that can only generate words in $W(s)$, therefore for any word $w \notin W(s)$, the third fraction will yield a 0.

The probability for documents is similar to that for definitions except that there is a topic mixture for each word, which is estimated by the topics in the window. Hence $P_d(\mathbf{z})$ is estimated as:

$$P_d(\mathbf{z}) = \prod_i \frac{\Gamma(T\alpha_d)}{\Gamma(\alpha_d)^T} \frac{\prod_z \Gamma(n_z^{(v_i)} + \alpha_d)}{\Gamma(n^{(v_i)} + T\alpha_d)} \tag{6}$$

Thus, the conditional probability for documents can be estimated by cancellation terms in equation 6, 3, and 4:

$$P_d(z_{ij} = z, s_{ij} = s|\mathbf{z_{-ij}}, \mathbf{s_{-ij}}, \mathbf{w}) \propto \frac{n_{-ij,z}^{(v_i)} + \alpha_d}{n_{-ij}^{(v_i)} + T\alpha_d} \frac{n_{-ij,z}^s + n_z^{(s')}\gamma/n^{(s')} + \beta}{n_{-ij,z} + \sum_{s'} n_z^{(s')}\gamma/n^{(s')} + S\beta} \frac{n_{-ij,s}^w + \lambda}{n_{-ij,s} + |W(s)|\lambda} \tag{7}$$

3.1 Approximation

In current model, each word appears in $\epsilon + 1$ windows, and will be generated $\epsilon + 1$ times, so there will be $\epsilon + 1$ pairs of topics/senses sampled for each word, which requires a lot of additional computation (proportional to context size ϵ). On the other hand, it can be imagined that the set of values $\{z_{ij}, s_{ij}|j - \epsilon/2 \leq i \leq j + \epsilon/2\}$ in different windows v_i should roughly be the same, since they are hidden values for the same word w_j. Therefore, to reduce computation complexity during Gibbs sampling, we approximate the values of $\{z_{ij}, s_{ij}|\ i \neq j\}$ by the topic/sense (z_{jj}, s_{jj}) that are generated from window v_j. That is, in Gibbs sampling, the algorithm does not actually sample the values of $\{z_{ij}, s_{ij}, |\ i \neq j\}$; instead, it directly assumes the sampled values are z_{jj}, s_{jj}.

4 Experiments and Analysis

Data: We experiment with several datasets, namely, the Brown Corpus (Brown), New York Times (NYT) from the American National Corpus, Reuters (R20) and WordNet definitions. In a preprocessing step, we remove all the non-content words whose part of speech tags are not one of the following set {*noun, adjective, adverb, verb*}. Moreover, words that do not have a valid lemma in WordNet are removed. For WordNet definitions, we remove stop words hence focusing on relevant content words.

Corpora statistics after each step of preprocessing is presented in Table 1. The column *WN token* lists the number of word#pos tokens after preprocessing. Note that now we treat word#pos as a word token. The column *word types* shows corresponding word#pos types, and the total number of possible sense types is listed in column *sense types*. The DOCs size for WordNet is the total number of senses defined in WordNet.

Experiments: We design two tasks to test our models: (1) text categorization task for evaluating the quality of values of topic nodes, and (2) a WSD task for evaluating the quality of the values of the sense nodes, mainly as a diagnostic tool targeting the specific aspect of sense definitions incorporation and distinguish that component's contribution to text categorization performance. We compare the performance of four topic models. (a) LDA: the traditional topic model proposed in (Blei et al., 2003) except that it uses Gibbs Sampling for inference. (b) LDA+def: is LDA with sense definitions. However they are not explicitly modeled; rather they are treated as documents and used as augmented data. (c) STM0: the topic model with an additional explicit sense node in the model, but we do not model the sense definitions. And finally (d) STMn is the full model with definitions explicitly modeled. In this setting n is the γ value. We experiment with different γ values in the STM models, and investigate the semantic scope of words/senses by choosing different window size ϵ. We report mean and standard deviation based on 10 runs.

It is worth noting that a larger window size ϵ suggests documents have larger impact on the model (ϕ, η) than definitions, since each document word has ϵ copies. This is not a desirable property when we want to investigate the weight of definitions by choosing different γ values. Accordingly, we only use z_{jj}, s_{jj}, w_{jj} to estimate ϕ, η, so that the impact of documents is fixed. This makes more sense, in that after the approximation in section 3.1, there is no need to use $\{z_{ij}, s_{ij}, \mid i \neq j\}$ (they have the same values as z_{jj}, s_{jj}).

4.1 Text Categorization

We believe our model can generate more "correct" topics by looking into dictionaries. In topic models, each word is generalized as a topic and each document is summarized as the topic mixture θ, hence it is natural to evaluate the quality of inferred topics in a text categorization task. We follow the classification framework in (Griffiths et al., 2005): first run topic models on each dataset **individually** without knowing label information to achieve document level topic mixtures, then we employ Naive Bayes and SVM (both implemented in the WEKA Toolkit (Hall et al., 2009)) to perform classification on the topic mixtures. For all document, the features are the percentage of topics. Similar to (Griffiths et al., 2005), we assess inferred topics by the classification accuracy of 10-fold cross validation on each dataset.

We evaluate our models on three datasets in the cross validation manner: The Brown corpus which comprises 500 documents grouped into 15 categories (same set used in (Griffiths et al., 2005)); NYT comprising 800 documents grouped into the 16 most frequent label categories; Reuters (R20) comprising 8600 documents labeled with the most frequent 20 categories. In R20, combination of categories is treated as separate category labels, so $money, interest$ and $interest$ are considered different labels.

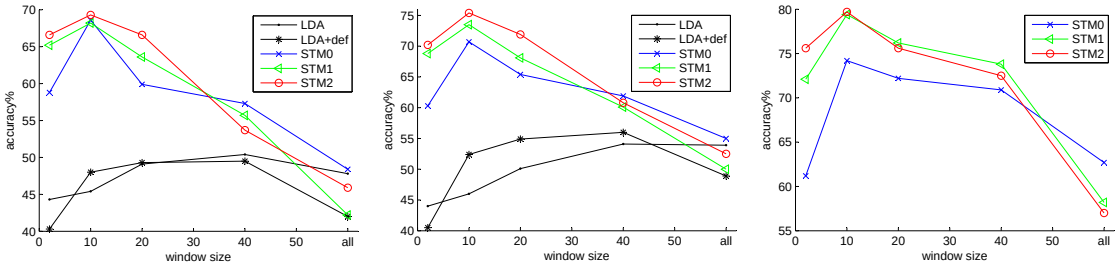
For the three datasets, we use the Brown corpus only as a tuning set to decide on the topic model parameters for all of our experimentation, and use the optimized parameters directly on NYT and R20 without further optimization.

4.1.1 Classification Results

Searching γ and ϵ on Brown: The classification accuracy on the Brown corpus with different ϵ and γ values using Naive Bayes and SVM are presented in figure 2a and 2b. In this section, the number of topics T is set to 50. The possible ϵ values in the horizontal axis are *2, 10, 20, 40, all*. The possible γ values are *0, 1, 2*. Note that $\epsilon = all$ means that no local window is used, and $\gamma = 0$ means definitions are not used. The hyper-

Corpus	DOCs size	orig tokens	content tokens	WN tokens	word types	sense types
Brown	500	1022393	580882	547887	27438	46645
NYT	800	743665	436988	393120	19025	37631
R20	8595	901691	450935	417331	9930	24834
SemCor	352	676546	404460	352563	28925	45973
WordNet	117659	1447779	886923	786679	42080	60567

Table 1: Corpus statistics



(a) Naive Bayes on Brown (b) SVM on Brown (c) SVM on NYT

Figure 2: Classification accuracy at different parameter settings

parameters are tuned as $\alpha_d = 0.1, \alpha_s = 0.01, \beta = 0.01, \lambda = 0.1$.

From figure 2, we observe that results using SVM have the same trend as Naive Bayes except that the accuracies are roughly 5% higher for SVM classifier. The results of LDA and LDA+def suggest that simply treating definitions as documents in an augmented data manner does not help. Comparing SMT0 with LDA in the same ϵ values, we find that explicitly modeling the sense node in the model greatly improves the classification results. The reason may be that words in LDA are independent isolated strings, while in STM0 they are connected by senses.

STM2 prefers smaller window sizes (ϵ less than 40). That means two words with a distance larger than 40 are not necessarily semantically related or share the same topic. This ϵ number also correlates with the optimal context window size of 12 reported in WSD tasks (Sinha and Mihalcea, 2007; Guo and Diab, 2010).

Classification results: Table 2 shows the results of our models using best tuned parameters of $\epsilon = 10, \gamma = 2$ on 3 datasets. We present three baselines in Table 2: (1) WEKA uses WEKA's classifiers directly on bag-of-words without topic modeling. The values of features are simply term frequency. (2) WEKA+FS performs feature selection using information gain before applying classification. (3) LDA, is the traditional topic model. Note that Griffiths et al.'s (2005) implementation of LDA achieve 51% on Brown corpus using Naive Bayes . Finally the Table illustrates the results obtained using our proposed models STM0 (γ=0) and STM2 (γ = 2).

It is worth noting that R20 (compared to NYT) is a harder condition for topic models. This is because fewer words (10000 distinct words versus 19000 in NYT) are frequently used in a large training set (8600 documents versus 800 in NYT), making the surface word feature space no longer as sparse as in the NYT or Brown corpus, which implies simply using surface words without considering the words distributional statistics – topic modeling – is good enough for classification. In (Blei et al., 2003) figure 10b they also show worse text categorization results over the SVM baseline when more than 15% of the training labels of Reuters are available for the SVM classifiers, indicating that LDA is less necessary with large training data. In our investigation, we report results on SVM classifiers trained on the whole Reuters training set. In our experiments, LDA fails to correctly classify nearly 10% of the Reuters documents compared to the WEKA baseline, however STM2 can still achieve significantly better accuracy (+4%) in the SVM classification condition.

Table 2 illustrates that despite the difference between NYT, Reuters and Brown (data size, genre, domains, category labels), exploiting WSD techniques (namely using a local window size coupled with explicitly modeling a sense node) yields

	Brown		NYT		R20	
	NB	SVM	NB	SVM	NB	SVM
WEKA	48	47.8	57	54.1	72.4	82.9
WEKA+FS	50	47.2	56.9	55.1	72.9	83.4
LDA	47.8±4.3	53.9±3.8	48.5±5.5	53.8±3.5	61.0±3.3	72.5±2.5
STM0	68.6±3.5	70.7±3.9	66.7±3.8	74.2±4.0	72.7±3.5	85.2±0.9
STM2	69.3±3.3	75.4±3.7	74.6±3.3	79.3±2.5	73±3.7	86.9±1.2

Table 2: Classification results on 3 datasets using hyperparameters tuned on Brown.

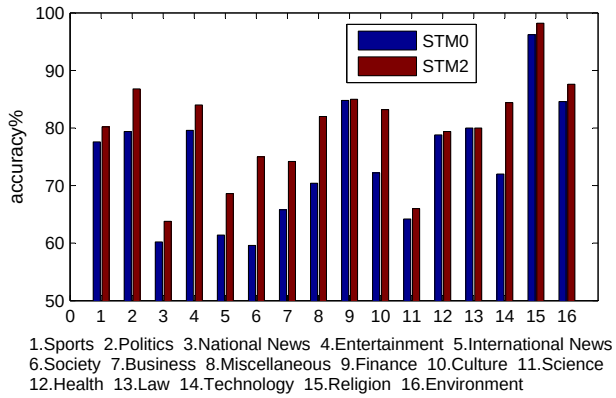


Figure 3: SVM accuracy on each category of NYT

significantly better results than all three baselines including LDA. Furthermore, explicit definition modeling as used in STM2 yields the best performance consistently overall.

Finally, in Figure 2c we show the SVM classification results on NYT in different parameter settings. We find that the NYT classification accuracy trend is consistent with that on the Brown corpus for each parameter setting of $\epsilon \in \{2, 10, 20, 40, all\}$ and $\gamma \in \{0, 1, 2\}$. This further proves the robustness of STMn.

4.2 Analysis on the Impact of Modeling Definitions

4.2.1 Qualitative Analysis

To understand why definitions are helpful in text categorization, we analyze the SVM performance of STM0 and STM2 ($\epsilon = 10$) on each category of NYT dataset (figure 3). We find STM2 outperforms STM0 in all categories. However, the largest gain is observed in *Society*, *Miscellaneous*, *Culture*, *Technology*. For *Technology*, we should credit WordNet definitions, since *Technology* may contain many infrequent technical terms, and STM0 cannot generalize the meaning of words only by distributional information due to their low frequency usage. However in some other domains, fewer specialized words are repeatedly used, hence STM0 can do as well as STM2.

For the other 3 categories, we hypothesize that these documents are likely to be a mixture of multiple topics. For example, a *Culture* news could contain topics pertaining to *religion, history, art*; while a *Society* news about crime could relate to *law, family, economics*. In this case, it is very important to sample a true topic for each word, so that ML algorithms can distinguish the *Culture* documents from the *Religion* ones by the proportion of topics. Accordingly, adding definitions should be very helpful, since it specifically defines the topic of a sense, and shields it from the influence of other "incorrect/irrelevant" topics.

4.2.2 Quantitative Analysis with Word Sense Disambiguation

A side effect of our model is that it sense disambiguates all words. As a means of analyzing and gaining some insight into the exact contribution of explicitly incorporating sense definitions (STMn) versus simply a sense node (STM0) in the model, we investigate the quality of the sense assignments in our models. We believe that the choice of the correct sense is directly correlated with the choice of a correct topic in our framework. Accordingly, a relative improvement of STMn over STM0 (where the only difference is the explicit sense definition modeling) in WSD task is an indicator of the impact of using sense definitions in the text categorization task.

WSD Data: We choose the all-words WSD task in which an unsupervised WSD system is required to disambiguate all the content words in documents. Our models are evaluated against the SemCor dataset. We prefer SemCor to all-words datasets available in Senseval-3 (Snyder and Palmer, 2004) or SemEval-2007 (Pradhan et al., 2007), since it includes many more documents than either set (350 versus 3) and therefore allowing more reliable results. Moreover, SemCor is also the dataset used in (Boyd-Graber et al., 2007), where a WordNet based topic model for WSD is introduced. The

	Total	Noun	Adjective	Adverb	Verb
sense annotated words	225992	86996	31729	18947	88320
polysemous words	187871	70529	21989	11498	83855
TF-IDF	-	0.422	0.300	0.153	0.182

Table 3: Statistics of SemCor per POS

statistics of SemCor is listed in table 3.

We use hyperparameters tuned from the text categorization task: α_d=0.1, α_s=0.01, β=0.01, δ=1, T=50, and try different values of $\epsilon \in \{10, 20, 40\}$ and $\gamma \in \{0, 2, 10\}$. The Brown corpus and WordNet definitions corpus are used as augmented data, which means the dashed line in figure 1c will become bold. Finally, we choose the most frequent answer for each word in the last 10 iterations of a Gibbs Sampling run as the final sense choice.

WSD Results: Disambiguation per POS results are presented in table 4. We only report results on polysemous words. We can see that modeling definitions (STM2 and STM10) improves performance significantly over STM0's across the board per POS and overall. The fact that STMn picks more correct senses helps explain why STMn classifies more documents correctly than STM0. Also it is interesting to see that unlike in the text categorization task, larger values of γ generate better WSD results. However, the window size ϵ, does not make a significant difference, yet we note that ϵ=10 is still the optimal value, similar to our observation in the text categorization task.

STM10 achieves similar results as in LDAWN (Boyd-Graber et al., 2007) which was specifically designed for WSD. LDAWN needs a fine grained hypernym hierarchy to perform WSD, hence they can only disambiguate nouns. They report different performances under various parameter setting. We cite their best performance of 38% accuracy on nouns as a comparison point to our best performance for nouns of 38.5%.

An interesting feature of STM10 is that it performs much better in nouns than adverbs and verbs, compared to a random baseline in Table 4. This is understandable since topic information content is mostly borne by nouns and adjectives, while adverbs and verbs tend to be less informative about topics (e.g., *even, indicate, take*), and used more across different domain documents. Hence topic models are weaker in their ability to identify clear cues for senses for verbs and adverbs. In support of our hypothesis about the POS distribution, we compute the average TF-IDF scores for each POS (shown in Table 3 according to the equation illustrated below). The average TF-IDF clearly indicate the positive skewness of the nouns and adjectives (high TF-IDF) correlates with the better WSD performance.

$$\text{TF-IDF}(pos) = \frac{\sum_i \sum_d \text{TF-IDF}(w_{i,d})}{\#\ of\ w_{i,d}}$$
$$where\ w_{i,d} \in pos.$$

At last, we notice that the most frequent sense baseline performs much better than our models. This is understandable since: (1) most frequent sense baseline can be treated as a supervised method in the sense that the sense frequency is calculated based on the sense choice as present in sense annotated data; (2) our model is not designed for WSD, therefore it discards a lot of information when choosing the sense: in our model, the choice of a sense s_i is only dependent on two facts: the corresponding topic z_i and word w_i, while in (Li et al., 2010; Banerjee and Pedersen, 2003), they consider all the senses and words in the context words.

5 Related work

Various topic models have been developed for many applications. Recently there is a trend of modeling document dependency (Dietz et al., 2007; Mei et al., 2008; Daume, 2009). However, topics are only inferred based on word cooccurrence, while word semantics are ignored.

Boyd-Graber et al. (2007) are the first to integrate semantics into the topic model framework. They propose a topic model based on WordNet noun hierarchy for WSD. A word is assumed to be generated by first sampling a topic, then choosing a path from the root node of hierarchy to a sense node corresponding to that word. However, they only focus on WSD. They do not exploit word definitions, neither do they report results on text categorization.

Chemudugunta et al. (2008) also incorporate a sense hierarchy into a topic model. In their framework, a word may be directly generated from a topic (as in standard topic models), or it can be

		Total	Noun	Adjective	Adverb	Verb
random		22.1	26.2	27.9	32.2	15.8
most frequent sense		64.7	74.7	77.5	74.0	59.6
STM0	$\epsilon = 10$	24.1±1.4	29.3±4.3	28.7±1.1	34.1±3.1	17.1±1.6
	$\epsilon = 20$	24±1.3	30.2±3.3	29.1±1.4	34.9±3.1	15.9±0.7
	$\epsilon = 40$	24±2.4	28.4±4.3	28.7±1.1	**36.4±4.7**	17.3±2.4
STM2	$\epsilon = 10$	27.5±1.1	36.1±3.8	34.0±1.2	33.4±1.8	17.8±1.4
	$\epsilon = 20$	25.7±1.3	32.0±4.2	33.5±0.7	34.2±3.4	17.3±0.7
	$\epsilon = 40$	26.1±1.3	32.5±3.9	33.6±0.9	34.2±3.4	17.5±1.4
STM10	$\epsilon = 10$	**28.8±1.1**	**38.5±2.3**	**34.7±0.8**	34.0±3.3	**18.4±1.2**
	$\epsilon = 20$	27.7±1.0	36.8±2.2	34.5±0.7	33.0±3.1	17.6±0.7
	$\epsilon = 40$	28.1±1.5	38.4±3.1	34.0±1.0	35.1±5.4	17.0±0.9

Table 4: Disambiguation results per POS on polysemous words.

generated by choosing a sense path in the hierarchy. Note that no topic information is on the sense path. If a word is generated from the hierarchy, then it is not assigned a topic. Their models based on different dictionaries improve perplexity.

Recently, several systems have been proposed to apply topic models to WSD. Cai et al. (2007) incorporate topic features into a supervised WSD framework. Brody and Lapata (2009) place the sense induction in a Baysian framework by assuming each context word is generated from the target word's senses, and a context is modeled as a multinomial distribution over the target word's senses rather than topics. Li et al. (2010) design several systems that use latent topics to find a most likely sense based on the sense paraphrases (extracted from WordNet) and context. Their WSD models are unsupervised and outperform state-of-art systems.

Our model borrows the local window idea from word sense disambiguation community. In graph-based WSD systems (Mihalcea, 2005; Sinha and Mihalcea, 2007; Guo and Diab, 2010), a node is created for each sense. Two nodes will be connected if their distance is less than a predefined value; the weight on the edge is a value returned by sense similarity measures, then the PageRank/Indegree algorithm is applied on this graph to determine the appropriate senses.

6 Conclusion and Future Work

We presented a novel model STM that combines explicit semantic information and word distribution information in a unified topic model. STM is able to capture topics of words more accurately than traditional LDA topic models. In future work, we plan to model the WordNet sense network. We believe that WordNet senses are too fine-grained, hence we plan to use clustered senses, instead of current WN senses, in order to avail the model of more generalization power.

Acknowledgments

This research was funded by the Ofce of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), through the U.S. Army Research Lab. All statements of fact, opinion or conclusions contained herein are those of the authors and should not be construed as representing the ofcial views or policies of IARPA, the ODNI or the U.S. Government.

References

Satanjeev Banerjee and Ted Pedersen. 2003. Extended gloss overlaps as a measure of semantic relatedness. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, pages 805–810.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Jordan Boyd-Graber and David M. Blei. 2007. Putop: turning predominant senses into a topic model for word sense disambiguation. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 277–281.

Jordan Boyd-Graber, David M. Blei, and Xiaojin Zhu. 2007. A topic model for word sense disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 1024–1033.

Samuel Brody and Mirella Lapata. 2009. Bayesian word sense induction. In *Proceedings of the 12th Conference of the European Chapter of the ACL*, pages 103–111.

Jun Fu Cai, Wee Sun Lee, and Yee Whye Teh. 2007. Improving word sense disambiguation using topic features. In *Proceedings of 2007 Joint Conference on Empirical Methods in Natural Language*

Processing and Computational Natural Language Learning, pages 1015–1023.

Chaitanya Chemudugunta, Padhraic Smyth, and Mark Steyvers. 2008. Combining concept hierarchies and statistical topic models. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 1469–1470.

Hal Daume. 2009. Markov random topic fields. In *Proceedings of the ACL-IJCNLP Conference*, pages 293–296.

Laura Dietz, Steffen Bickel, and Tobias Scheffer. 2007. Unsupervised prediction of citation influence. In *Proceedings of the 24th international conference on Machine learning*, pages 233–240.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Thomas L. Griffiths and Mark Steyvers. 2004. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101:5228–5235.

Thomas L. Griffiths, Mark Steyvers, David M. Blei, and Joshua B. Tenenbaum. 2005. Integrating topics and syntax. In *Advances in Neural Information Processing Systems*.

Weiwei Guo and Mona Diab. 2010. Combining orthogonal monolingual and multilingual sources of evidence for all words wsd. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1542–1551.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11:10–18.

Linlin Li, Benjamin Roth, and Caroline Sporleder. 2010. Topic models for word sense disambiguation and token-based idiom detection. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1138–1147.

Qiaozhu Mei, Deng Cai, Duo Zhang, and Chengxiang Zhai. 2008. Topic modeling with network regularization. In *Proceedings of the 17th international conference on World Wide Web*, pages 101–110.

Rada Mihalcea. 2005. Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labeling. In *Proceedings of the Joint Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 411–418.

Sameer S. Pradhan, Edward Loper, Dmitriy Dligach, and Martha Palmer. 2007. Semeval-2007 task 17: English lexical sample, srl and all words. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 87–92. ACL.

Ravi Sinha and Rada Mihalcea. 2007. Unsupervised graph-based word sense disambiguation using measures of word semantic similarity. In *Proceedings of the IEEE International Conference on Semantic Computing*, pages 363–369.

Benjamin Snyder and Martha Palmer. 2004. The english all-words task. In *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 41–43. ACL.

Ivan Titov and Ryan McDonald. 2008. Modeling online reviews with multi-grain topic models. In *Proceedings of the 17th international conference on World Wide Web*, pages 111–120.