# Semantic Visualization for Short Texts with Word Embeddings

**Tuan M. V. Le**
School of Information Systems
Singapore Management University
vmtle.2012@phdis.smu.edu.sg

**Hady W. Lauw**
School of Information Systems
Singapore Management University
hadywlauw@smu.edu.sg

## Abstract

Semantic visualization integrates topic modeling and visualization, such that every document is associated with a topic distribution as well as visualization coordinates on a low-dimensional Euclidean space. We address the problem of semantic visualization for short texts. Such documents are increasingly common, including tweets, search snippets, news headlines, or status updates. Due to their short lengths, it is difficult to model semantics as the word co-occurrences in such a corpus are very sparse. Our approach is to incorporate auxiliary information, such as word embeddings from a larger corpus, to supplement the lack of co-occurrences. This requires the development of a novel semantic visualization model that seamlessly integrates visualization coordinates, topic distributions, and word vectors. We propose a model called GaussianSV, which outperforms pipelined baselines that derive topic models and visualization coordinates as disjoint steps, as well as semantic visualization baselines that do not consider word embeddings.

## 1 Introduction

Visualization of a text corpus is an important exploratory task. A document is represented in a high-dimensional space, where every dimension corresponds to a word in the vocabulary. Dimensionality reduction maps this to a low-dimensional latent representation, such as a 2D or 3D that is perceivable to human eye. Documents, and their relationships, can be visualized in Euclidean space via a scatterplot.

While there exist dimensionality reduction techniques that go directly from the high-dimensional to the low-dimensional space, such as MDS [Kruskal, 1964], more recent approaches recognize the value of incorporating topic modeling [Iwata *et al.*, 2008; Le and Lauw, 2014a]. This is because the synonymy and polysemy inherent in text to some degree could be modeled by topics, where each topic corresponds to words that are related by some shared meaning [Blei *et al.*, 2003].

**Problem.** *Semantic visualization* refers to jointly modeling topics and visualization. Given a corpus of documents, we learn for each document its coordinate in a 2D Euclidean space for visualization and its topic distribution. Of primary concern in this paper is semantic visualization for *short texts*, which make up an increasing fraction of texts generated today, owing to the proliferation of mobile devices and prevalence of social media. For instance, tweets are limited to 140 characters. Search snippets, news headlines, or status updates are not much longer. Their limitation in modeling semantics is well-documented in various contexts [Sriram *et al.*, 2010; Metzler *et al.*, 2007; Sun, 2012].

Existing semantic visualization models are not designed for short texts. For example, PLSV [Iwata *et al.*, 2008] represents documents as bags of words, and topic distributions are inferred from word co-occurrences in documents. This assumes sufficiency in word co-occurrences to discover meaningful topics. This may be valid for regular-length documents, but not for short texts, due to the extreme sparsity of words in such documents. Methods based on tf-idf vectors, such as SSE [Le and Lauw, 2014b] would also suffer, because tf-idf vectors are not efficient for short text analysis [Yan *et al.*, 2012]. Many words appear only once in a short document, and may appear in only a few documents. Consequently tf and idf are not very distinguishable in short texts.

**Approach.** There are several possible directions to deal with short text. Not all are suitable for semantic visualization. For instance, it is possible to combine a few short texts into a longer "pseudo-document", e.g., grouping tweets of one user. However, this would not allow us visualize individual short texts, in order to view their relationships, as they are now aggregated into one pseudo-document displayed as a single element. For another instance, we could constrain the topic model to assign one topic to all words within a short text to enforce word co-occurrences. However, this still would not fully resolve the issue of the sparsity of word co-occurrences.

The direction taken in this paper is to attack the main issue of sparsity, by supplementing short texts with auxiliary information from a larger external corpus. Outside of semantic visualization, this was explored in the context of topic modeling (without visualization), by incorporating topics learned from Wikipedia [Phan *et al.*, 2008] or jointly learning two sets of topics on short and auxiliary long texts [Jin *et al.*, 2011].

Specifically, we seek to leverage word embeddings, which have gained increasing attention for their ability to express the conceptual similarity of words. Models such as Word2Vec [Mikolov *et al.*, 2013] and GloVe [Pennington *et al.*, 2014] learn a continuous vector in an embedding space for each

word. They are trained on a large corpora (e.g., Wikipedia, Google news). We postulate that word vectors would be a useful form of auxiliary information in the context of semantic visualization for short texts, as the conceptual similarities learned from the huge corpus and encoded in word vectors can supplement lack of word co-occurrences in short-texts.

There are two potential approaches to using word vectors. The first is what we term a *pipelined* approach, by employing topic models that work with word vectors [Das *et al.*, 2015; Hu and Tsujii, 2016] to produce the topic distributions of short texts, which are then mapped to visualization coordinates using an appropriate dimensionality reduction technique. The second is what we term a *joint* approach, by designing a single model that incorporates visualization coordinates, topic distributions, and word vectors within an integrated generative process. Inspired by the precedence established by previous semantic visualization works on bag of words [Iwata *et al.*, 2008] showing the advantage of a joint approach, we surmise that joint modeling is a promising approach for semantic visualization using word embeddings.

**Contributions.** We make the following contributions. *Firstly*, as far as we are aware, we are the first to propose semantic visualization for short texts. *Secondly*, we design a novel semantic visualization model that leverages word embeddings. Our model, called *Gaussian Semantic Visualization* or GaussianSV, assumes that each topic is characterized by a Gaussian distribution on the word embedding space. Section 3 presents the model in detail including its generative process as well as how to learn its parameters based on MAP estimation. *Thirdly*, we evaluate our model on two public real-life short text datasets in Section 4. To validate our joint modeling, one class of baselines consist of pipelined approaches that apply dimensionality reduction to the outputs of topic models with word embeddings. To validate our modeling of word embeddings, the other class of baselines consist of semantic visualization models not using word vectors.

## 2 Related Work

An early work in semantic visualization was PLSV [Iwata *et al.*, 2008], which extended PLSA [Hofmann, 1999] for bag of words. Follow-on works include Semafore [Le and Lauw, 2014a], which leveraged on neighborhood graphs, and SSE [Le and Lauw, 2014b], which worked with tf-idf vectors. These models were not designed with short texts in mind. They would suffer from sparsity when applied to short texts.

Topic models, such as LDA [Blei *et al.*, 2003], PLSA [Hofmann, 1999], LSA [Deerwester *et al.*, 1990] and those based on Non-negative Matrix Factorization [Arora *et al.*, 2012], worked with bag of words. Some recent models incorporated word embeddings. GLDA [Das *et al.*, 2015] modeled a topic as a distribution over word vectors. LCTM [Hu and Tsujii, 2016] modeled a topic as a distribution of concepts, where each concept defined another distribution of word vectors. GPUDMM [Li *et al.*, 2016] and LFDMM [Nguyen *et al.*, 2015] extended DMM [Nigam *et al.*, 2000] that assigned all words in a short text to only one topic. While these topic models were not meant for visualization, their output topic distributions could be mapped to a 2D space using the dimensionality reduction meant for probability distributions, i.e., Parametric Embedding or PE [Iwata *et al.*, 2007].

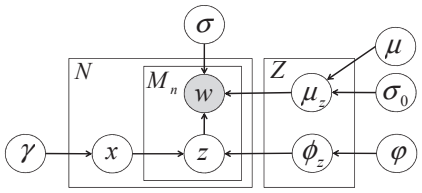


Figure 1: Graphical Model of GaussianSV

Generic dimensionality reduction techniques could be used to map any high-dimensional data to low-dimensions, by preserving some notion of similarity among data points [Kruskal, 1964; Roweis and Saul, 2000; Tenenbaum *et al.*, 2000; der Maaten and Hinton, 2008]. They were not designed for text, nor short texts. For one reason, they do not incorporate topic modeling, which provides semantic interpretability.

Our work is also different from those that sought to derive embeddings for documents or sentences [Le and Mikolov, 2014; Kiros *et al.*, 2015]. They were not meant for visualization, as they would still operate at high dimensions (e.g., 400). They were not concerned with topic modeling either.

## 3 Gaussian Semantic Visualization

In this section, we describe our proposed model GaussianSV, whose graphical model is shown in Figure 1.

Our input is a corpus of documents $\mathcal{D} = \{d_1, \ldots, d_N\}$. Each document $d_n$ is a bag of words. Denote $w_{nm}$ to be the $m^{\text{th}}$ word in document $d_n$, and $M_n$ to be the number of words in $d_n$. Each word $w$ in the vocabulary $W$ is represented as a $p$-dimensional continuous vector, which has been learned from an external corpus using some word embedding model. For popular word embeddings [Mikolov *et al.*, 2013; Pennington *et al.*, 2014], $p$ is usually in the hundreds.

Our objective is two-fold. First, we seek to derive as output the visualization coordinate $x_n$ for each document $d_n$. Without loss of generality, in the following, we assume $x_n$ is 2-dimensional for visualization. Second, we also seek to derive each document's topic distribution over $Z$ topics $\{\mathrm{P}(z|d_n)\}_{z=1}^{Z}$. Each topic $z$ is associated with a probability distribution $\{\mathrm{P}(w|z)\}_{w \in W}$ over words in the vocabulary $W$. The words with the highest probabilities given a topic usually help to provide some interpretable meaning to a topic.

### 3.1 Generative Process

In a conventional topic model, such as LDA [Blei *et al.*, 2003] or PLSA [Hofmann, 1999], a topic is represented by a multinomial distribution over words. Some previous works on semantic visualization [Iwata *et al.*, 2008; Le and Lauw, 2014a] are also based on such topic representation.

The key difference is that in our context a word is not just a discrete outcome of a multinomial process, but rather a continuous vector in the embedding space. We need another way to characterize a topic, as well as to model the generation of words due to that topic. Inspired by [Das *et al.*, 2015], we associate each topic $z$ with a continuous vector $\mu_z$ resident in the same $p$-dimensional word embedding space. This allows

us to model the word generation due to a topic as a Gaussian distribution, centered at the $\mu_z$ vector, with spherical covariance. In other words, a word $w_{nm}$ belonging to topic $z$ will be drawn according to the following probability:

$$\mathrm{P}(w_{nm}|\mu_z,\sigma) = \big(\frac{\sigma}{2\pi}\big)^{\frac{p}{2}} \exp(-\frac{\sigma}{2} \parallel w_{nm} - \mu_z \parallel^2), \quad (1)$$

where $\sigma$ is a hyper-parameter.

To derive the visualization, in addition to the coordinate $x_n$ associated with each document $d_n$, we also assign each topic $z$ a latent coordinate $\phi_z$ in the same visualization space. With documents and topics residing in the same Euclidean space, spatial distances between documents and topics can represent their relationship. Intuitively, documents close to each other would tend to talk about the same topics (that are also located near those documents). We thus express a document $d_n$'s distribution over topics, in terms of the Euclidean distances between $x_n$ and topic coordinate $\phi_z$, as follows:

$$\mathrm{P}(z|x_n,\Phi) = \frac{\exp(-\frac{1}{2}||x_n - \phi_z||^2)}{\sum_{z'=1}^{Z}\exp(-\frac{1}{2}||x_n - \phi_{z'}||^2)} \quad (2)$$

where $\mathrm{P}(z|x_n,\Phi)$ is the probability of topic $z$ in document $d_n$ and $\Phi = \{\phi_z\}_{z=1}^{Z}$ is the set of topic coordinates.

Our objective is to derive the coordinates of documents and topics in the visualization space, as well as the distribution over $Z$ topics $\{\mathrm{P}(z|d_n)\}_{z=1}^{Z}$ for each document $d_n$. We also derive the mean $\mu_z$ for each topic $z$. Note that we do not derive word vectors, but consider them as input to our model.

The generative process is now described as follows:

1. For each topic $z = 1,\ldots,Z$:
   (a) Draw $z$'s mean: $\mu_z \sim \mathrm{Normal}(\boldsymbol{\mu}, \sigma_0^{-1}\mathrm{I})$
   (b) Draw $z$'s coordinate: $\phi_z \sim \mathrm{Normal}(0, \varphi^{-1}I)$
2. For each document $d_n$, where $n = 1,\ldots,N$:
   (a) Draw $d_n$'s coordinate: $x_n \sim \mathrm{Normal}(0, \gamma^{-1}I)$
   (b) For each word $w_{nm} \in d_n$:
      i. Draw a topic: $z \sim \mathrm{Multi}(\{\mathrm{P}(z|x_n,\Phi)\}_{z=1}^{Z})$
      ii. Draw a word: $w_{nm} \sim \mathrm{Normal}(\mu_z, \sigma^{-1}\mathrm{I})$

The first step concerns the generation of topics' mean vectors and visualization coordinates. The second step concerns the generation of documents' coordinates, and words (represented as word vectors) within each document.

Notably, by representing documents and topics in the same visualization space, as well as words and topics in the same word embedding space, the topics play a crucial role as conduits between the two spaces. Therefore, documents that contain similar words are more likely to share similar topics. Here, "similar" words could be the same words, frequently co-occurring words, and owing to the use of word embeddings: also different words that are close in the word embedding space. For short texts in particular, the latter is expected to be especially significant, because of lower word frequencies and weaker role of word co-occcurrences.

### 3.2 Parameter Estimation

The parameters are estimated based on maximum a posteriori estimation (MAP) using EM algorithm [Dempster *et al.*, 1977]. The unknown parameters that need to be estimated include document coordinates $\chi = \{x_n\}_{n=1}^{N}$, topic coordinates $\Phi = \{\phi_z\}_{z=1}^{Z}$, and topic mean vectors $\Pi = \{\mu_z\}_{z=1}^{Z}$, collectively denoted as $\Psi = \{\chi, \Phi, \Pi\}$.

Given the generative process described earlier, the log likelihood can be expressed as follows:

$$\mathcal{L}(\Psi|\mathcal{D}) = \sum_{n=1}^{N}\sum_{m=1}^{M_n}\log\sum_{z=1}^{Z}\mathrm{P}(z|x_n,\Phi)\mathrm{P}(w_{nm}|\mu_z,\sigma) \quad (3)$$

The conditional expectation of the complete-data log likelihood with priors is as follows:

$$\mathcal{Q}(\Psi|\hat{\Psi}) = \sum_{n=1}^{N}\sum_{m=1}^{M_n}\sum_{z=1}^{Z}\mathrm{P}(z|n,m,\hat{\Psi})\log\big[\mathrm{P}(z|x_n,\Phi)\mathrm{P}(w_{nm}|\mu_z,\sigma)\big] + \sum_{n=1}^{N}\log(\mathrm{P}(x_n)) + \sum_{z=1}^{Z}\log(\mathrm{P}(\phi_z)) + \sum_{z=1}^{Z}\log(\mathrm{P}(\mu_z)),$$

where $\hat{\Psi}$ is the current estimate. $\mathrm{P}(z|n,m,\hat{\Psi})$ is the class posterior probability of the $n^{\mathrm{th}}$ document and the $m^{\mathrm{th}}$ word in the current estimate. $\mathrm{P}(x_n)$ and $\mathrm{P}(\phi_z)$ are Gaussian priors with a zero mean and a spherical covariance for the document coordinates $x_n$ and topic coordinates $\phi_z$:

$$\mathrm{P}(x_n) = \Big(\frac{\gamma}{2\pi}\Big)^{\frac{D}{2}}\exp\Big(-\frac{\gamma}{2}\parallel x_n \parallel^2\Big), \quad (4)$$

$$\mathrm{P}(\phi_z) = \Big(\frac{\varphi}{2\pi}\Big)^{\frac{D}{2}}\exp\Big(-\frac{\varphi}{2}\parallel \phi_z \parallel^2\Big), \quad (5)$$

where we set the hyper-parameters to $\gamma = 0.1Z$ and $\varphi = 0.1N$ following PLSV [Iwata *et al.*, 2008].

We put a Gaussian prior over $\mu_z$ with hyper-parameter $\sigma_0$ and mean $\boldsymbol{\mu}$ which is set to the average of all word vectors in the vocabulary.

$$\mathrm{P}(\mu_z) = \big(\frac{\sigma_0}{2\pi}\big)^{\frac{p}{2}}\exp(-\frac{\sigma_0}{2}\parallel \mu_z - \boldsymbol{\mu} \parallel^2) \quad (6)$$

We use EM algorithm to estimate the parameters. In the E-step, we compute $\mathrm{P}(z|n,m,\hat{\Psi})$ as in Equation 7. We then update $\Psi = \{\chi, \Phi, \Pi\}$ in the M-step. $\mu_z$ is updated using Equation 8. To update $\phi_z$ and $x_n$, we use gradient-based numerical optimization method such as the quasi-Newton method [Liu and Nocedal, 1989] because the gradients cannot be solved in a closed form. We alternate the E- and M-steps until some appropriate convergence criterion is reached.

**E-step:**

$$\mathrm{P}(z|n,m,\hat{\Psi}) = \frac{\mathrm{P}(z|\hat{x}_n,\hat{\Phi})\mathrm{P}(w_{nm}|\hat{\mu}_z,\hat{\Sigma}_z)}{\sum_{z'=1}^{Z}\mathrm{P}(z'|\hat{x}_n,\hat{\Phi})\mathrm{P}(w_{nm}|\hat{\mu}_{z'},\hat{\Sigma}_{z'}))} \quad (7)$$

**M-step:**

$$\frac{\partial\mathcal{Q}(\Psi|\hat{\Psi})}{\partial\phi_z} = \sum_{n=1}^{N}\sum_{m=1}^{M_n}\big(\mathrm{P}(z|x_n,\Phi) - \mathrm{P}(z|n,m,\hat{\Psi})\big)(\phi_z - x_n) - \beta\phi_z$$

$$\frac{\partial\mathcal{Q}(\Psi|\hat{\Psi})}{\partial x_n} = \sum_{m=1}^{M_n}\sum_{z=1}^{Z}\big(\mathrm{P}(z|x_n,\Phi) - \mathrm{P}(z|n,m,\hat{\Psi})\big)(x_n - \phi_z) - \gamma x_n$$

$$\mu_z = \frac{\sum_{n=1}^{N}\sum_{m=1}^{M_n}\big(\mathrm{P}(z|n,m,\hat{\Psi})\sigma w_{nm}\big) + \sigma_0\boldsymbol{\mu}}{\sum_{n=1}^{N}\sum_{m=1}^{M_n}\mathrm{P}(z|n,m,\hat{\Psi})\sigma + \sigma_0} \quad (8)$$

| | Visualization | Topic model | Joint model | Word vectors |
|---|---|---|---|---|
| GaussianSV | ✓ | ✓ | ✓ | ✓ |
| PLSV | ✓ | ✓ | ✓ | |
| SEMAFORE | ✓ | ✓ | ✓ | |
| SSE | ✓ | ✓ | ✓ | |
| GLDA/PE | ✓ | ✓ | | ✓ |
| LCTM/PE | ✓ | ✓ | | ✓ |
| GPUDMM/PE | ✓ | ✓ | | ✓ |

Table 1: Comparative Methods

# 4 Experiments

The objective is to evaluate the effectiveness of GaussianSV for visualizing short texts and the quality of its topic model.

## 4.1 Experimental Setup

**Datasets.** We use short texts from two public datasets. The first is $BBC$[1] [Greene and Cunningham, 2006], which consists of 2,225 BBC news articles from 2004-2005, divided into 5 classes. We only use the title and headline of an article. The second is $SearchSnippet$[2] [Phan *et al.*, 2008], which consists of 12,340 Web search snippets belonging to 8 classes. We use the pre-trained 300-d word vectors from *Word2Vec* trained on Google News[3]. We remove stopwords, perform stemming, and remove words that do not have pre-trained word vectors. The average document length is 14.1 words for $BBC$ and 14.9 words for $SearchSnippet$.

Following [Iwata *et al.*, 2008; Le and Lauw, 2014a], for each dataset, we sample 50 documents per class to create a well-balanced dataset. Each sample of $SearchSnippet$ has 400 documents, and that of $BBC$ has 250 documents respectively. As the methods are probabilistic, we create 5 samples for each dataset, and run each sample 5 times. The reported performance numbers are averaged across 25 runs.

**Comparative Methods.** We compare our GaussianSV[4] model to two classes of baselines that generate both topic model and visualization coordinates, as listed in Table 1. Their differences to GaussianSV are discussed in Section 2.

The first class of baselines are semantic visualization techniques that do not rely on word vectors. These include PLSV[5], SEMAFORE[6], and SSE[7]. Comparison to these models help to validate the contributions of word vectors.

The second class of baselines are not semantic visualization models per se. Rather they are a pipeline of topic models that incorporate word vectors, i.e., GLDA[8], LCTM[9], and GPUDMM[10], followed by PE [Iwata *et al.*, 2007] for mapping topic distributions into visualization space. Comparison to these help to validate the contributions of joint modeling.

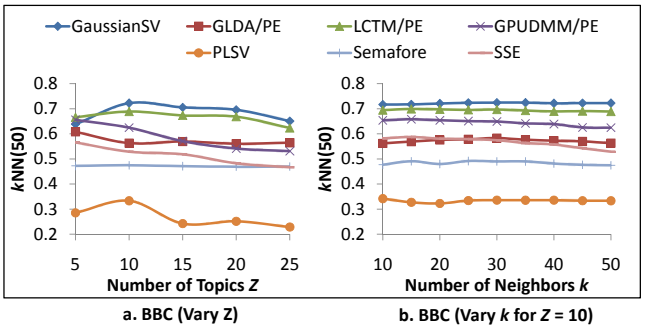


Figure 2: $k$NN Accuracy Comparison on $BBC$

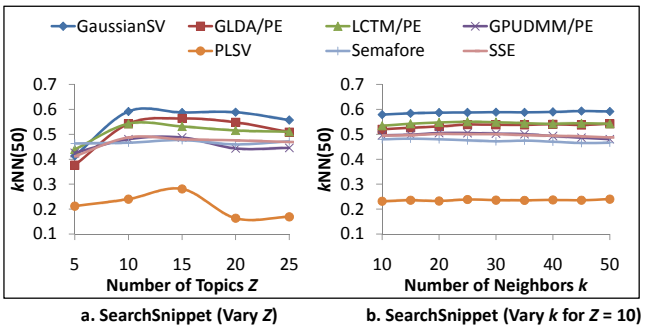


Figure 3: $k$NN Accuracy Comparison on $SearchSnippet$

## 4.2 Visualization Quality

**Metric.** A good visualization is expected to keep similar documents close, and keep different documents far in the visualization space. We rely on $k$ nearest neighbors ($k$NN) classification accuracy to measure the visualization quality. This is an established metric for semantic visualization [Iwata *et al.*, 2008; Le and Lauw, 2014a; 2014b] for objectivity and repeatability. For each document, we hide its true class and assign it to the majority class determined by its $k$ nearest neighbors in the visualization space. The accuracy is the fraction of documents that are assigned correctly to its true class.

**Results.** We report $k$NN accuracy on $BBC$ in Figure 2 and on $SearchSnippet$ in Figure 3. At first, we set $k$ = 50 as the datasets contain 50 documents from each class. Later, we also show $k$NN accuracy at different $k$.

In Figure 2a and Figure 3a, we vary the number of topics $Z$. The results show that methods with word vectors (i.e., GaussianSV, GLDA/PE, LCTM/PE and GPUDMM/PE) deal with short texts better than conventional semantic visualization techniques (i.e., PLSV, Semafore and SSE). The latter suffer due to the sparsity of word co-occurrences.

[1] http://mlg.ucd.ie/datasets/bbc.html

[2] http://jwebpro.sourceforge.net/data-web-snippets.tar.gz

[3] https://code.google.com/archive/p/word2vec/

[4] We choose appropriate values for $\sigma_0$ and $\sigma$. $\sigma_0 = 10000$ and $\sigma = 100$ work well for most of the cases in our experiments.

[5] We use the implementation by `https://github.com/tuanlvm/SEMAFORE`.

[6] We use the author implementation in `https://github.com/tuanlvm/SEMAFORE`.

[7] We use the implementation obtained from the authors.

[8] We use the author implementation at `https://github.com/rajarshd/Gaussian_LDA`, set degree of freedom $\nu = 1000p$, and use default values for other parameters.

[9] We use the author implementation at `https://github.com/weihua916/LCTM`. The number of concepts is 500, and the noise of each concept is 0.001. Other parameters are set to default.

[10] We use the author implementation at `https://github.com/NobodyWHU/GPUDMM` with default parameters.

Among those leveraging word vectors, our method GaussianSV performs significantly better than the others. For $BBC$, comparing to LCTM/PE that has the closest performance, we gain 4-5% improvement for 10 to 25 topics. Paired samples t-test indicate that the improvement is significant at 0.05 level or lower in all cases, except for $Z = 25$. At 5 topics, LCTM/PE is slightly better, but it is not significant even at 0.1 level. For $SearchSnippet$, except for $Z = 5$, we beat the two closest baselines GLDA/PE and LCTM/PE by 4-14% with statistical significance at 0.05 level or lower. These improvements show that joint modeling to leverage word embeddings is better for semantic visualization of short texts.

In Figures 2b and 3b, we vary $k$ while fixing $Z = 10$. The performances are not affected much by $k$. Similar observations regarding the comparisons can be drawn as before.

**Example Visualizations.** Figure 5 shows the visualization of each method on $BBC$. Documents are represented as colored points placed according to their coordinates. Topic coordinates are represented as hollow circles. GaussianSV separates the 5 classes well. PLSV tends to mix the classes together. Semafore is better than PLSV, as it produces some clusters, although it cannot differentiate documents belonging to $business$ and $tech$. SSE differentiates those two classes better, but the $business$ documents are spread all over instead of being grouped together like in GaussianSV's visualization. SSE also mixes some documents belonging to $entertainment$, $politics$ and $sport$ at the bottom, which is not the case in GaussianSV's visualization. The classes are not separated well in GLDA/PE's visualization, especially for those documents at the center. GPUDMM/PE separates $business$ and $tech$ well, but it divides $politics$ into two subclusters which could reduce the $k$NN accuracy. In addition, it also mixes some documents of $entertainment$ and $sport$, while GaussianSV can differentiate them. LCTM/PE provides a good visualization, however it still mixes some documents of $business$ and $politics$ together near the center. GaussianSV is better than LCTM/PE at separating them.

Figure 6 for $SearchSnippet$ shows similar trends. Semafore and SSE are better than PLSV but still mix some documents from different classes. GPUDMM/PE, by leveraging word embeddings, provides better clusters in the visualization but cannot differentiate $culture-arts-entertainment$ and $sports$ on the top. This is not case in GaussianSV's visualization. Similar to GaussianSV, GLDA/PE and LCTM/PE can separate well $engineering$ and $health$. However, GLDA/PE does not separate well $culture - arts - entertainment$ by letting some documents overlap with other documents from other classes at the center. LCTM/PE has the same problem. It mixes $culture - arts - entertainment$ with some documents from other classes such as $computers$.

### 4.3 Topic Coherence

We investigate whether while providing better visualization, our method still maintains the quality of the topic model.

**Metric.** One measure for topic model quality that has some agreement with human judgment is topic coherence [Newman *et al.*, 2010], which looks at how the top keywords in each topic are related to each other in terms of semantic meaning. As suggested by [Newman *et al.*, 2010], we rely on Pointwise Mutual Information (PMI) to evaluate topic coherence. For a pair of words, PMI is defined as $log\frac{p(w_1,w_2)}{p(w_1)p(w_2)}$. For each topic, we take the top 10 words to compute the pairwise PMI. For a topic model, PMI is computed by the average of all pairwise PMIs across all pairs and topics. The more the words in topics are correlated, the higher the PMI tends to be. Following [Le and Lauw, 2014b], we estimate $p(w)$ and $p(w_1, w_2)$ based on the frequencies of 1-grams and 5-grams from Google Web 1T 5-gram Version 1 [Brants and Franz, 2006], a corpus of n-grams from 1 trillion word tokens.

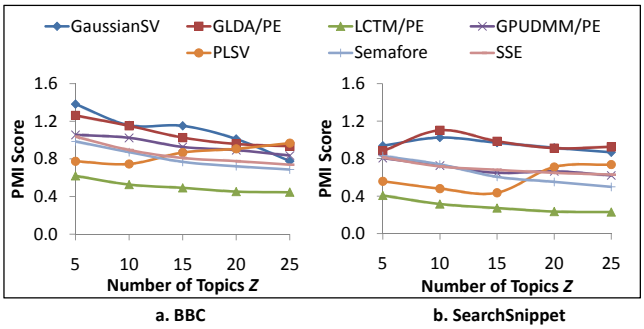


Figure 4: Topic Coherence (PMI Score)

| *BBC* | | *SearchSnippet* | |
|---|---|---|---|
| ID | Top 5 words | ID | Top 5 words |
| 0 | government, election, vote, proposal, referendum | 0 | software, technology, database, computer, system |
| 1 | player, star, boss, manager, director | 1 | game, sport, football, tournament, basketball |
| 2 | film, music, movie, musical, musician | 2 | democratic, political, democracy, government, politics |
| 3 | market, company, economy, price, economic | 3 | engine, cylinder, piston, turbine, compressor |
| 4 | internet, mobile, computer, digital, browser | 4 | health, medical, cancer, diagnosis, doctor |
| 5 | win, season, victory, championship, game | 5 | market, business, export, industry, manufacturing |
| 6 | gordon, thompson, alex, bryan, bennett | 6 | science, university, mathematics, academic, faculty |
| 7 | bring, leave, push, accept, seek | 7 | kind, type, aspect, work, approach |
| 8 | big, good, real, great, major | 8 | usa, carl, bryant, donnie, eric |
| 9 | man, woman, girl, boy, teenager | 9 | news, web, website, blog, online |

Table 2: Top Words in Each Topic by GaussianSV for $Z = 10$

**Results.** Figure 4 shows the PMI scores for various number of topics $Z$. Evidently, GaussianSV has comparable PMI score to GLDA/PE, and performs better than the other methods across different $Z$, which shows that GaussianSV produces at least a comparable topic model, while having better visualization. As examples, Table 2 shows the top 5 words of each topic for $Z = 10$ for $BBC$ and $SearchSnippet$.

## 5 Conclusion

We propose GaussianSV model, a semantic visualization model for short text, which leverages word vectors obtained from a larger external corpus to supplement the sparsity of short texts. The model performs well on real-life short text datasets against semantic visualization baselines, as well as against pipelined baselines, validating both the value of incorporating word embeddings and that of joint modeling.

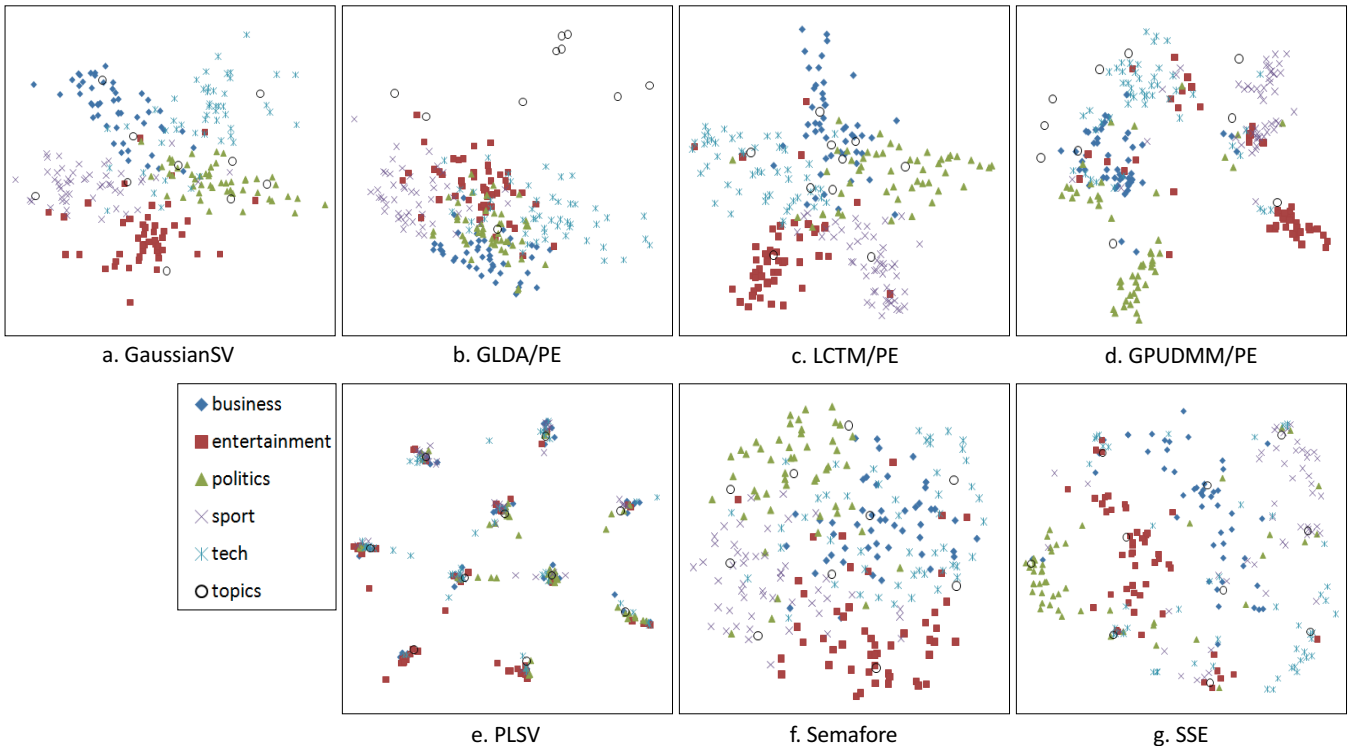


Figure 5: Visualization of $BBC$ for $Z = 10$ (*best seen in color*)

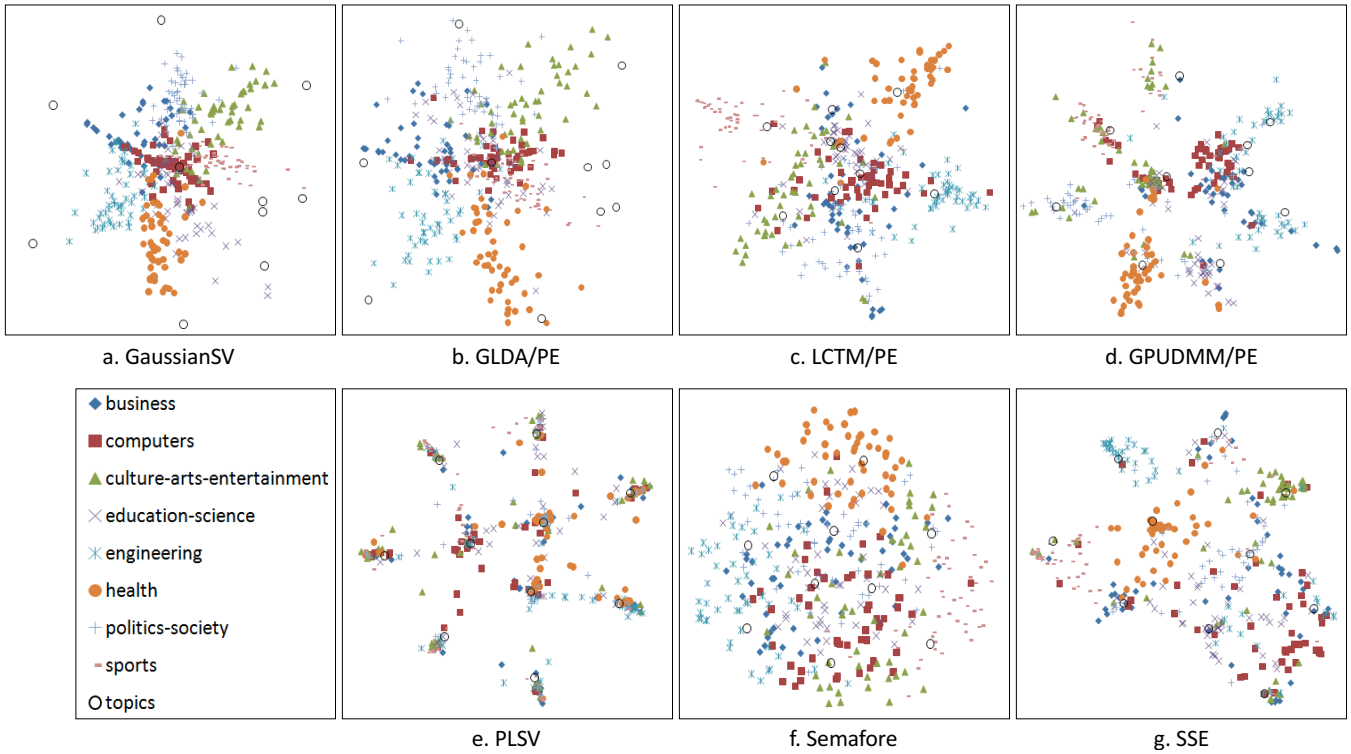


Figure 6: Visualization of $SearchSnippet$ for $Z = 10$ (*best seen in color*)

## References


[Arora *et al.*, 2012] Sanjeev Arora, Rong Ge, and Ankur Moitra. Learning topic models–going beyond svd. In *FOCS*, pages 1–10. IEEE, 2012.

[Blei *et al.*, 2003] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *JMLR*, 3, 2003.

[Brants and Franz, 2006] Thorsten Brants and Alex Franz. *Web 1T 5-gram Version 1*. Linguistic Data Consortium, Philadelphia, 2006.

[Das *et al.*, 2015] Rajarshi Das, Manzil Zaheer, and Chris Dyer. Gaussian LDA for topic models with word embeddings. In *ACL*, 2015.

[Deerwester *et al.*, 1990] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *JASIS*, 41(6):391, 1990.

[Dempster *et al.*, 1977] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[der Maaten and Hinton, 2008] L. Van der Maaten and G. Hinton. Visualizing data using t-SNE. *JMLR*, 9, 2008.

[Greene and Cunningham, 2006] Derek Greene and Pádraig Cunningham. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *ICML*, pages 377–384, 2006.

[Hofmann, 1999] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, 1999.

[Hu and Tsujii, 2016] Weihua Hu and Junichi Tsujii. A latent concept topic model for robust topic inference using word embeddings. In *ACL*, page 380, 2016.

[Iwata *et al.*, 2007] T. Iwata, K. Saito, N. Ueda, S. Stromsten, T. L. Griffiths, and J. B. Tenenbaum. Parametric embedding for class visualization. *Neural Computation*, 19(9), 2007.

[Iwata *et al.*, 2008] Tomoharu Iwata, Takeshi Yamada, and Naonori Ueda. Probabilistic latent semantic visualization: topic model for visualizing documents. In *KDD*, pages 363–371, 2008.

[Jin *et al.*, 2011] Ou Jin, Nathan N Liu, Kai Zhao, Yong Yu, and Qiang Yang. Transferring topical knowledge from auxiliary long texts for short text clustering. In *CIKM*, pages 775–784, 2011.

[Kiros *et al.*, 2015] Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Skip-thought vectors. In *NIPS*, pages 3294–3302, 2015.

[Kruskal, 1964] J. B. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1), 1964.

[Le and Lauw, 2014a] Tuan M V Le and Hady W Lauw. Manifold learning for jointly modeling topic and visualization. In *AAAI*, 2014.

[Le and Lauw, 2014b] Tuan M V Le and Hady W Lauw. Semantic visualization for spherical representation. In *KDD*, pages 1007–1016, 2014.

[Le and Mikolov, 2014] Quoc V Le and Tomas Mikolov. Distributed representations of sentences and documents. In *ICML*, volume 14, pages 1188–1196, 2014.

[Li *et al.*, 2016] Chenliang Li, Haoran Wang, Zhiqian Zhang, Aixin Sun, and Zongyang Ma. Topic modeling for short texts with auxiliary word embeddings. In *SIGIR*, pages 165–174, 2016.

[Liu and Nocedal, 1989] Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45:503–528, 1989.

[Metzler *et al.*, 2007] Donald Metzler, Susan Dumais, and Christopher Meek. Similarity measures for short segments of text. In *ECIR*, pages 16–27, 2007.

[Mikolov *et al.*, 2013] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *Proceedings of Workshop at ICLR*, 2013.

[Newman *et al.*, 2010] David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. Automatic evaluation of topic coherence. In *NAACL HLT*, pages 100–108, 2010.

[Nguyen *et al.*, 2015] Dat Quoc Nguyen, Richard Billingsley, Lan Du, and Mark Johnson. Improving topic models with latent feature word representations. *TACL*, 3:299–313, 2015.

[Nigam *et al.*, 2000] Kamal Nigam, Andrew Kachites McCallum, Sebastian Thrun, and Tom Mitchell. Text classification from labeled and unlabeled documents using em. *Machine Learning*, 39(2-3):103–134, 2000.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher D Manning. GloVe: Global vectors for word representation. *EMNLP*, 12, 2014.

[Phan *et al.*, 2008] Xuan-Hieu Phan, Le-Minh Nguyen, and Susumu Horiguchi. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *WWW*, pages 91–100, 2008.

[Roweis and Saul, 2000] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290, 2000.

[Sriram *et al.*, 2010] Bharath Sriram, Dave Fuhry, Engin Demir, Hakan Ferhatosmanoglu, and Murat Demirbas. Short text classification in twitter to improve information filtering. In *SIGIR*, pages 841–842, 2010.

[Sun, 2012] Aixin Sun. Short text classification using very few words. In *SIGIR*, pages 1145–1146. ACM, 2012.

[Tenenbaum *et al.*, 2000] J. B. Tenenbaum, V. De Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290, 2000.

[Yan *et al.*, 2012] Xiaohui Yan, Jiafeng Guo, Shenghua Liu, Xue-qi Cheng, and Yanfeng Wang. Clustering short text using ncut-weighted non-negative matrix factorization. In *CIKM*, pages 2259–2262, 2012.