A Semantic Cover Approach for Topic Modeling

Rajagopal Venkatesaramani
Computer Science, WUSTL
St. Louis MO 63130
rajagopal@wustl.edu

Douglas Downey
EECS, Northwestern University
Evanston IL 60208
ddowney@eecs.northwestern.edu

Bradley Malin
Biomedical Informatics, Vanderbilt University
Nashville TN 37209
b.malin@vanderbilt.edu

Yevgeniy Vorobeychik
Computer Science, WUSTL
St. Louis MO 63130
yvorobeychik@wustl.edu

Abstract

We introduce a novel topic modeling approach based on constructing a semantic set cover for clusters of similar documents. Specifically, our approach first clusters documents using their *Tf-Idf* representation, and then covers each cluster with a set of topic words based on semantic similarity, defined in terms of a word embedding. Computing a topic cover amounts to solving a minimum set cover problem. Our evaluation compares our topic modeling approach to Latent Dirichlet Allocation (LDA) on three metrics: 1) qualitative topic match, measured using evaluations by Amazon Mechanical Turk (MTurk) workers, 2) performance on classification tasks using each topic model as a sparse feature representation, and 3) topic coherence. We find that qualitative judgments significantly favor our approach, the method outperforms LDA on topic coherence, and is comparable to LDA on document classification tasks.

1 Introduction

Topic modeling is one of the core research problems in natural language processing. Approaches to topic modeling range from simple vector comparisons to probabilistic graphical models (Deerwester et al., 1990; Hofmann, 1999; Blei et al., 2003; Mimno and McCallum, 2012). Nevertheless, despite the many approaches proposed over the years, probabilistic topic modeling methods in general, and Latent Dirichlet Allocation (LDA) (Blei et al., 2003) in particular, have become arguably the dominant paradigm. For example, it remains the algorithm of choice in the Amazon's healthcare NLP toolkit (Amazon Web Services, 2018).

However, there have been concerns about the performance of probabilistic models, particularly in the context of datasets comprised of short documents, such as tweets (Davidson et al., 2017; Yan et al., 2013; Hong and Davison, 2010; Mittos et al., 2018; Steinskog et al., 2017). This is primarily because the sparsity posed by short texts makes it hard for the model to sufficiently account for word co-occurrences, which form the basis of the definition of a topic in the sense of a multinomial distribution over words. Additionally, the language used on Twitter is informal in nature, uses slang and non-dictionary words, and often lacks proper grammatical structure. Moreover, the complexity of the probabilistic topic modeling approaches makes it difficult to interpret the specific choices they make about topics and their constituent words.

In this paper, we propose a novel approach to topic modeling which is conceptually simple and highly interpretable. Our approach is based on two hypotheses about the nature of short texts, such as tweets: first, that such texts can be grouped into relatively few disjoint clusters representing a similar mix of subjects (nominally, we call these clusters topics, recognizing that any such cluster may be comprised of multiple topics), and second, that each such subject mix can be adequately summarized by a small number of concepts (words). Both of these are distinct from LDA, which models a topic as a probability distribution over a large number of words. While LDA models each text as a mixture of multiple topics, we assert that each tweet falls into a single cluster. A more fundamental qualitative distinction of our approach from LDA is that it is deterministic in nature, and admits a much more compact representation of the corpus, since each topic, or cluster, is represented by only a small number of words.

To operationalize our hypotheses, we propose a two-step approach to topic modeling. First, we cluster documents based on their similarity in terms of *Tf-Idf* feature representation. Second, given the clustering, we attempt to find a set of

*Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 92–102
Minneapolis, June 6–7, 2019. ©2019 Association for Computational Linguistics

words for each cluster that forms a description of the cluster. Specifically, we use a word embedding, along with a document representation in the same semantic space, to cover each cluster with a small set of topic words that are semantically similar to the documents. More precisely, we say that a word (concept) in a dictionary covers a document if it is among the k most similar words in the semantic embedding space. To cover a collection of documents thereby becomes a minimum set cover problem instance. While the set cover problem is computationally hard, it admits a fast greedy approximation algorithm (Chvatal, 1979), which we utilize to construct the topic descriptions for each document cluster.

Our evaluation combines qualitative and quantitative metrics. We first qualitatively compare our approach to LDA by asking MTurk subjects for their judgments about the quality of respective choices of topics for a random sample of documents from a cluster. We do this through two conceptually different ways, and observe a significant and systematic advantage of our approach over LDA. Quantitatively, we compare our approach and LDA in terms of standard intrinsic topic coherence and performance in text classification. On the intrinsic topic coherence metric, our approach fares significantly better than LDA for 4 out of the 5 datasets we use, and the two are comparable on the fifth dataset. Finally, we consider two classification tasks, spam and hate speech prediction, in which topic modeling is used as a sparse feature representation. In this task, we find that both approaches yield similar performance.

2 Related Work

One of the earlier and more influential topic modeling methods was Latent Semantic Analysis (LSA) (Deerwester et al., 1990) which performs a singular value decomposition on the term-document matrix to discover concepts. Probabilistic Latent Semantic Analysis (pLSA) Hofmann (1999) tackles the limitations of LSA – namely potential negative values in the SVD, and the lack of a proper probability distribution – using a latent variable model, where topics are the latent variables. Arguably the most influential approach to the topic modeling domain is Latent Dirichlet Allocation (Blei et al., 2003). LDA can be thought of as an extension to pLSA, where the priors are Dirichlet distributions. LDA continues to be widely used in topic modeling, and several derivatives exist – each catering to a specific task, or corpus-structure (Blei et al., 2007; Blei and Lafferty, 2006; Yan et al., 2013).

Concerns about the performance of such probabilistic topic models with short text data (eg. tweets) have been illustrated by Davidson et al. (2017); Yan et al. (2013); Hong and Davison (2010); Mittos et al. (2018); Steinskog et al. (2017). Poor performance is attributed to the sparsity of short text data, which provide insufficient information for an approach like LDA to capture word co-occurrence. Yan et al. (2013) tackle this by explicitly modeling co-occurrence throughout the corpus to enhance topic learning. However, this approach requires $\mathcal{O}(m^2)$ memory (where m is the size of the vocabulary) to maintain all biterms (2-grams) and their frequencies in the corpus, making it inefficient in practice.

Weng et al. (2010) aggregate tweets by the same user into pseudo-documents, yet this approach suffers from a dependence on the availability of user-information, or disproportionate distribution of tweets over users. Hong and Davison (2010) aggregate tweets containing the same word, which improves performance relative to LDA. Combining documents based on single words however induces heavy biases on the topics discovered. In our approach, we include a clustering step that can be thought of as an aggregation method. Documents that are semantically similar are grouped together into a cluster instead of a pseudo document, where similarity is a function of all words in the document.

Rangarajan Sridhar (2015) propose learning a vector space representation of words in a corpus using Word2Vec, similarly to our approach, except without *Tf-Idf* weights, and then fitting a mixture of gaussians on the resulting vectors using standard EM. However, the dimensionality of a Word2Vec representation is typically high (50-300 in practice), where gaussian mixtures are known to perform poorly (Krishnamurthy, 2011). Dimensionality reduction on the Word2Vec space is typically used to alleviate this problem, but it reduces the strength of the representation in the process.

In addition to probabilistic topic modeling, document clustering was successfully used in topic modeling by Aker et al. (2016), who use a supervised framework to train a learning model that predicts similarity scores between comments from news articles. A graph consisting of documents as nodes and similarity-weighted edges is then passed to the Markov Clustering Algorithm (Van Dongen, 2000). A major drawback of this approach is the dependence on availability of ground truth data to begin with.

3 Topic Modeling Using a Semantic Cover

We propose a simple topic modeling framework comprised of two steps. First, we cluster documents based on similarity. Second, we extract a set of topics from each cluster by leveraging a word embedding. The intuition behind the clustering step is that it splits a corpus into qualitatively similar groups of documents. Thus, we expect it to be possible to summarize the subject of each cluster by a small collection of topic words. The second step aims at summarizing each cluster of documents using a small set of topic words. The property we seek in this step is that the topic words chosen are *semantically* representative of the cluster. To achieve this goal, we leverage recent advances in neural word embeddings which empirically demonstrated that such embeddings are semantically meaningful (Mikolov et al., 2013a,b). Semantic similarity between words is roughly captured by cosine similarity in the embedded space. Specifically, we first represent documents in the same embedding space as words, and define the problem of the choice of topic extraction as a set cover problem instance. In the set cover instance, a potential topic word covers a collection of documents if the word is similar to these in the embedding space.

3.1 Document Clustering

Our first step is to partition the set of documents in the corpus into a collection of clusters. For this purpose, we first transform each document into its *Tf-Idf* representation. Depending on the dataset, any standard clustering approach may be used to partition the documents. In our case, we run spectral clustering (Ng et al., 2002) over the documents in their *Tf-Idf* form, where we use cosine similarity between vectors as the similarity metric.

3.2 A Set Cover Approach for Topic Extraction

Having obtained a collection of clusters, we treat them independently, with the goal of extracting a small set of representative topic words for each cluster, which adequately represents the subject of the documents in the cluster. To this end, we first represent words, as well as documents, in a vector space using a word embedding. Aiming for a small set of words is useful both in reducing the effort required for human interpretation, as well as forming a compact representation of a set of documents for quantitative tasks such as document classification.

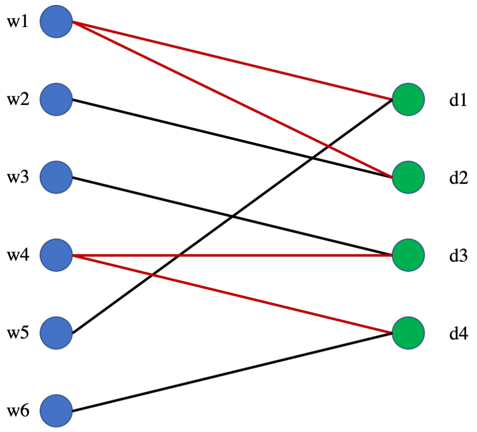


Figure 1: Extracting topic words with *set cover*

Each document (green) in a cluster is connected to its 2 most similar words (blue). The aim is to find the smallest set of words such that the union of the edges originating from them covers all documents in the cluster. In this case, $w1$ and $w2$ form the cover.

Suppose that we have a dictionary W (a collection of words, which is a superset of words that actually occur in the cluster), with each word embedded in a real vector space, i.e., for each $w \in W$, $w \in \mathbb{R}^n$. Moreover, suppose that each document d is represented in the same embedding space. First, we associate each word $w \in W$ with a set of documents, $D(w)$, based on their similarity in the embedding space. Let $s(w, d)$ be a similarity score between a word w and a document d. Given a document d, let $W_k(d)$ be the set of k most similar words to d in terms of $s(w, d)$.

Definition 1 *A word* w k-covers *a document* d *if* $w \in W_k(d)$.

Now, the set $D_k(w)$ is the set of all documents d in the cluster k-covered by a word w. Next, we define our topic representation for a cluster C of documents as a set cover.

Definition 2 *A collection of words* W_C k-covers *a cluster of documents* C *if* $C \subseteq \cup_{w \in W_C} D_k(w)$. *A collection of words* W_C $(k, 1-\delta)$-covers *a cluster* C *if* $|C \cap (\cup_{w \in W_C} D_k(w))|/|C| \geq 1 - \delta$.

In words, a collection of words W_C covers a cluster if each document in the cluster is covered by some word $w \in W_C$. If the cover is partial, in the sense that at least a fraction $1-\delta$ (i.e., most) of the documents are covered, we call it the $1-\delta$ cover. At this point, it is important to note that in principle the cover W_C *need not include solely words found in the documents in cluster* C.

Having defined what it means for a collection of topical words to cover (exactly or approximately) a document cluster (really, an arbitrary collection of documents), we now observe that our aim is to find a *small* cover—that is, the smallest number of topic words that adequately cover a document cluster. Next, we define this notion precisely.

Definition 3 *Given a* k *and* δ*, a* minimum $(k, 1-\delta)$ *cover for a document cluster* C *is a collection* W_C^* *which is a* $1-\delta$ *cover such that* $|W_C^*| \leq |W_C|$ *for any other* $(k, 1-\delta)$ *cover* W_C *of a document cluster* C.

Embedding Words and Documents

To derive a word embedding, we can use one of the standard embedding approaches which has been demonstrated to roughly correspond to semantic relationships among words. We chose Word2Vec for this purpose, although other such embedding approaches can presumably be used in its place. While we used the *Tf-Idf* representation of documents in clustering, this is not well-suited to topic extraction using set cover, since it does not embed documents in the same semantic space as words.

To address this, we represent the documents in a new embedded space by computing a weighted average of Word2Vec (Mikolov et al., 2013a) representations of words occurring in the document, with *Tf-Idf* as the weighting scheme. Using *Tf-Idf* weighting in conjunction with a Word2Vec representation helps alleviate issues that the individual representations face when used independently. Used in isolation, the standard *Tf-Idf* representation only allows us to compute similarities between documents, but not between words - given that words in this case are simply orthonormal one-hot vectors. Using only the Word2Vec representation allows us to compare similarity between words, but does not, by itself, represent documents. As *Tf-Idf* is an information-measure of how important a word is to a document, it is naturally an apt weighting scheme to represent a document as the weighted centroid of the vectors corresponding to the words in the document.

As we describe in the sections to follow, this also allows us to find topic-words for documents that might not necessarily be contained in the documents themselves. To define this representation precisely, suppose that t is the *Tf-Idf* representation of a document over a word dictionary W, and let V be the matrix with columns corresponding to words embedded in real space using Word2Vec. Then the embedded document representation is defined by

$$d = Vt/m,$$

where m is the number of words in the document.

Computing the Minimum Semantic Set Cover

Given the definition of the minimum semantic cover for a cluster of documents, along with an embedding of both words and documents in the same space, we can now extract the topics for each cluster using a greedy algorithm inspired by the $\mathcal{O}(n \log n)$ greedy solution for set-cover (Chvatal, 1979), as follows.

We first convert the documents and words in the embedded space to an unweighted bipartite graph, using our notion of $(k, 1-\delta)$-cover. Let V_1 be a set of vertices where each vertex corresponds to a word w in the corpus dictionary, W. Let V_2 be a set of vertices, where each vertex corresponds to a document d in the corpus D. We add an edge between a word w and a document d if w $(k, 1-\delta)$-covers d in the sense of cosine similarity between words and documents, $s(w, d)$, in the embedded space. Thus, the graph $G = \{(V_1 \cup V_2), E\}$. We also have for each document, a cluster assignment from the spectral-clustering step, i.e. $D = \cup_{i=1,\ldots,n} C_i$, where n is the total number of clusters (topics), such that each document belongs to exactly one cluster C_i.

Then, to construct a minimum semantic set cover for a cluster, we proceed as follows. Let the set of topic words, T_i for the i^{th} cluster, C_i be an empty set. Let $V_{2,i} = \{d \in V_2 : d \in C_i\}$, i.e. $V_{2,i}$ is the subset of vertices in V_2 corresponding to documents in the i^{th} cluster. Let $V_{1,i} = \cup_{d \in V_{2,i}} N(d)$, where $N(d)$ represents node neighborhood. In words, $V_{1,i}$ is the subset of corpus words that cover at least one document in cluster C_i, i.e. the set $\cup_{d \in C_i} W_k(d)$.

Let G_i be the subgraph of G induced on $V_{i,1} \cup V_{2,i}$. The greedy algorithm to find the minimum set-cover for a cluster C_i proceeds by picking the node in $V_{1,i}$ that covers the maximum number of documents in $V_{2,i}$. In the case of a tie, we pick all nodes with maximum degree. The words corresponding to the selected vertices are placed in T_i, then the selected nodes, their neighbors in G_i and the edges between them are removed from the graph. We then recompute the degrees of all nodes affected by this removal of edges. This process is repeated until we have covered a desired fraction $(1 - \delta)$ of the cluster. Algorithm 1 details topic-word extraction using set-cover.

Algorithm 1 Greedy Set Cover

```
for Cluster $C_i$ do
    Label Set $T_i \leftarrow \emptyset$
    $V_{2,i} = \{v \in V_2 : v \in C_i\}$
    $V_{1,i} = \{N(v)\ \forall v \in V_{2,i}\}$
    $G_i$ = Subgraph of $G$ induced on $V_{2,i} \cup V_{1,i}$
    $k = \delta|V_{2,i}|$
    while $|V_{2,i}| > k$ do
        Sort $V_{1,i}$ in descending order of degree
        Remove the highest degree node(s), $v^*$ and place in $T_i$
        Remove all neighbors of $v^*$ and corresponding edges from $G_i$
        Recompute degrees for $V_{1,i}$
    end while
end for
```

4 Evaluation Methodology

We evaluate our approach in comparison with LDA—the de facto standard in topic modeling—both in qualitative and quantitative terms. Our qualitative evaluation involves human judgments about the appropriateness of topic choices for a subsample of texts. We complement this with two quantitative metrics, one with respect to a standard topic coherence measure, and the second in using topic models for text classification tasks. Throughout, we refer to our approach as *set cover*. Moreover, in our experiments, the Word2Vec vectors are derived by training a skip-gram model on the corpus, with a sliding window of size 4 and the number of dimensions set to 500. Additionally, we compute the minimum 1-cover (i.e. $\delta = 0$), that is, we ensure that all documents in the cluster are covered.

4.1 Qualitative Evaluation

Given the common use of topic modeling in obtaining qualitative insight from text, our first evaluation approach involves human judgments of quality. This evaluation echos other human evaluations of topic modeling, such as by Steinskog et al. (2017) for the topic-intrusion detection task. Also noteworthy is the work by Chang et al. (2009), who demonstrated the poor correlation of the popular perplexity metric (Blei et al., 2003) with human judgments.

For our qualitative evaluation, we set up a series of experiments on Amazon Mechanical Turk (MTurk). For these tasks, we use 4 sets from the health news tweets collected by (Karami et al., 2018) and YouTube comments about 23andMe (we provide specific details in a later section). To ensure fairness to LDA—our chosen baseline—we do this in two different settings based on how we group documents into topically related subsets.

Matched Clusters

In the first setup, we take the document clusters produced by spectral clustering as given, and focus the comparison between LDA and *set cover* on the particular choice of topical words these generate. In this case, we produce a correspondence between a given cluster and an LDA topic by choosing an LDA topic which maximizes the likelihood that the cluster was produced by the topic. More precisely, we assign a cluster C to the topic j which maximizes

$$\sum_{i \in C} P(i|j),$$

where $P(i|j)$ is the LDA-derived likelihood that a document i reflects a topic j. We then generate the collection of topic words for a given cluster using LDA in a standard way. Specifically, we choose the n most probable words in the associated LDA topic, where n is set as the number of topic words produced by the *set cover*.

In the experiment, we assign a random cluster to a subject, who is then presented with the documents in this cluster (or a random subsample of these, if the cluster is too large), the choice of topic words based on LDA, and the choice of topic words based on *set cover*. Additionally, we also ask the subjects for judgments of a collection of n randomly chosen words from the cluster to calibrate the results. We then ask participants to judge how well a topic (i.e., the collection of topic words) describes the given set of documents, and score each result on a 5-point Likert scale, with 1 being very poor and 5 very good.

Independent Clusters

One may naturally object that the above comparison is unfair to LDA insofar as we are choosing the clusters and then retrofitting LDA topics to these. We therefore ran a second set of qualitative experiments in which LDA topics were used to derive clusters of similar documents. Specifically, we clustered all documents based on their associated likelihood given a topic; that is, a document i was assigned to an LDA topic j which maximizes $P(i|j)$. This gives us a collection of document clusters, which we can then present to human subjects for judgment. As before, we used the top most probable n words from an LDA topic as the topic description presented to human subjects. The *set cover* approach, on the other hand, used spectral clustering as before. Since the set of documents presented for judgment is now different for the two approaches, we omitted the random words for calibration. Consequently, while we still presented the subjects with the same 5-point Likert scale as before, this scale is now calibrated differently, as will be made evident in the results section.

4.2 Quantitative Evaluations

To quantitatively compare our algorithms, we use the standard intrinsic topic coherence metric, and two classification tasks to compare the strengths of the sparse representation produced. The topic coherence (Stevens et al., 2012) for a set T of topic words is defined as follows:

$$Coherence(T) = \sum_{(w_i,w_j)\in T} \log \frac{N(w_i,w_j)+\lambda}{N(w_j)},$$

where $N(w)$ is the number of documents that contain the word w and λ is a smoothing factor. We compute average coherence scores over 5 runs, varying cluster sizes between 5 and 25.

Document Classification Accuracy

We set up two classification tasks. The first task is to classify short text messages as Ham or Spam. The second task is to classify tweets as offensive, hate speech or neither. For both, we use topic modeling approaches to arrive at a sparse feature representation of a document. For LDA, the feature vector for a document is comprised of the probabilities that a document was generated by each of the topics. For the *set cover* approach, we construct binary feature vectors that represent the occurrence of topic words in the cluster to which the document is assigned. Given the above feature representations, we use a Linear Support Vector Classifier. For the multi-class problem, we use a One-vs-Rest approach with Linear Support Vector Classifiers for both classification tasks. We maintain a 60%-40% train-test split over the corpus, and average accuracy over 5 runs, varying the number of topics between 5 and 25.

4.3 Data

Our evaluation used several datasets which we describe briefly below.

Twitter - Health News tweets from more than 15 health news agencies were collected by Karami et al. (2018). The dataset contains separate files for tweets collected from each source. Each source is observed to have had trends in tweets, which implicitly form topic clusters.

YouTube Comments - 23andMe We collected a sample of 800 YouTube comments from the top 50 YouTube video results for the search term '23andMe'. This dataset is qualitatively different from the Twitter corpus, showing greater variation in document length and significantly more noise.

Twitter - Hate and Offensive Speech A set of 24802 tweets based on a hate speech lexicon were collected and labelled into 3 categories - *hate speech, offensive speech* and *neither* by Davidson et al. (2017). This dataset is used in one of our two classification tasks.

Ham/Spam Short Messages 5574 short text messages were classified as legitimate (ham) or spam by Almeida et al. (2013). This forms the basis of the second classification task.

5 Results

5.1 Qualitative Evaluations

In the first set of MTurk experiments, where topics from both algorithms were shown in the same task, we asked human judges to score topics from 4 document clusters, collecting 20 responses for each. For instance, the Fox News dataset contains a set of tweets posted in 2015 about the measles outbreak in California, linked to Disney theme parks. The topic words for the measles outbreak cluster identified by the two algorithms are shown in Table 1. Here, we see that LDA picks certain irrelevant terms for the shown cluster sample ('u', 'rare'), while completely missing the term 'measles', which is a key subject of the documents (we note that we remove stop-words during preprocessing using NLTK (Loper and Bird, 2002)). The *set cover* approach, on the other hand, is able to identify highly pertinent words. We refer to this experiment as *Matched Clusters*. This can be thought of as reflecting the propriety of the chosen topic words *conditional* on the clusters of similar documents. The average scores are shown in Table 2.

Cluster Sample	· Disneyland measles outbreak linked to low vaccination rates · More measles cases tied to Disneyland Illinois day care · Amid US measles outbreak few rules on teacher vaccinations · US measles count rises to 121; most linked to Disneyland · Measles cases turn attention to bounty of childhood vaccines · FDA Commissioner says measles outbreak alarming
LDA	study, cancer, say, vaccine, died, disneyland, u, rare, woman, treatment
Set Cover	measles, cases, linked, disney, disneyland, alarming, almost, amid, amidst, bounty

Table 1: Topic Words - Measles Outbreak, 2015

Dataset (Cluster)	**Random**	**LDA**	**Set Cover**
US News (Superfoods)	2.9	3	4.35
Fox News (Disneyland Measles)	1.95	3.05	4.45
US News (Parenting)	1.95	1.6	4.5
YouTube (23andMe, Sale of Info)	2.35	2.7	4.1

Table 2: Average Turker scores for Matched Clusters on a 5-point Likert scale.

Our second set of experiments for clusters chosen independently for the two algorithms was conducted on a significantly larger scale. We uploaded 5 clusters per dataset, and collected 40 responses per cluster, resulting in a total of 2000 data points, 1000 for each algorithm. We refer to this experiment as *Independent Clusters*. Table 3 shows example topic words identified by LDA and Set Cover. The advantages of the clustering step in our approach are evident in this example - the *set cover* cluster contains documents that are more closely related to one another.

More importantly, it is worth noting the choice of the term 'delay' in the *set cover* topic words - while the term does not itself appear in the entire cluster, it is semantically related to documents in the cluster referring to the long wait Maryland residents had to endure to sign up for Obamacare. This is precisely the reason for using a word-embedding such as Word2Vec in our approach - topic words are not restricted to words in the cluster and yet appear to be semantically meaningful. The average judgments from MTurk for these experiments are reported in Table 4.

In both experiments, we can see that *set cover* consistently outperforms LDA, often by a large margin. We also performed a two sided independent samples t-test on the scores. The differences between the means in Table 4 are statistically significant; all but 23andMe for $p < 0.0001$, and the significance for 23andMe is for $p < 0.05$. It is interesting to note that *set cover* performs slightly better in terms of evaluation scores in the *Matched Clusters* study, suggesting that it is judged favorably *particularly* in the context of random calibration and LDA.

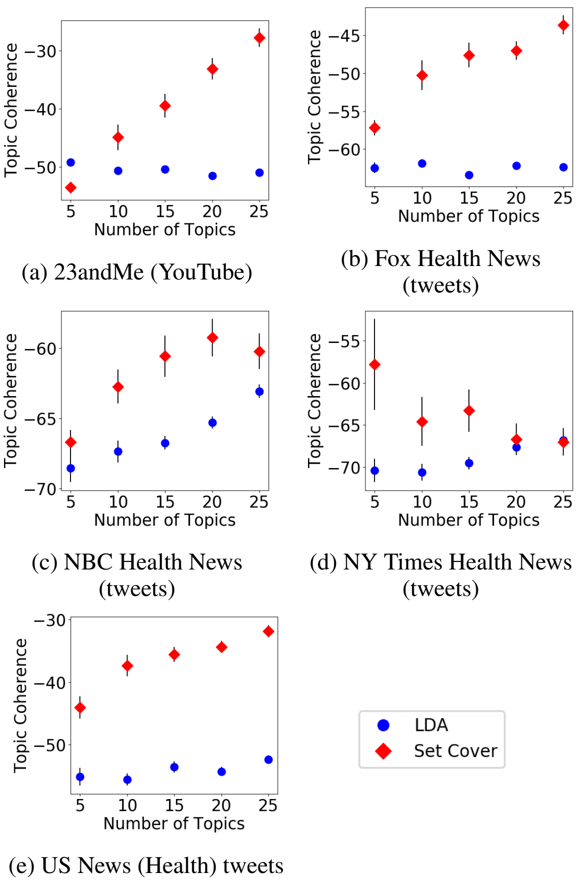
Since the clusters are fixed in these experiments, the results reflect the particular advantage of the *set cover* method itself in choosing descriptive words for a collection of similar documents. The *Independent Clusters* study, in contrast, serves more as an evaluation of each approach in an end-to-end fashion, and here, too, the difference is substantial. However, the LDA scores in this case are generally comparable or higher than in the *Matched Clusters* experiments, which suggests that the advantage of *set cover* over LDA may be primarily due to its better choice of topic words, which is its main novelty, rather than the clustering approach.

Cluster Sample	Algorithm (Data set)	Topic Words
· People are having sex after heart bypass surgery, and USN is ON IT: · Most Americans dont know what causes cancer. Do you? WorldCancerDay · Can a fitness tracker help me with my diet as well? USNTechChat · In honor of World Cancer Day, reports on 7 Innovations in Cancer Therapy · How to call a truce & build a healthy relationship with food: · Check out our 2015 BestDiets rankings! Wed love your feedback	LDA (USNews)	surgery cancer know usntech- -chat child say reports lose medical like
· Obamacare Bump: 10 Million Got Insurance, Survey Shows · #AskNBCNews: Obamacare Deadline Day Questions · Obamacare draws last-minute shoppers; site gets nearly 2 million visits · Supreme Court Hears Argument on Charged Obamacare Case · Communty health centers at center of Obamacare · The Longest Wait: Maryland Residents Wait in LIne for Last-Ditch Obamacare	Set Cover (NBC)	obamacare million get new deadline health may questions court delay

Table 3: Topic Words - Independent Clusters

Dataset		LDA	Set Cover
23andMe	Mean	3.94	4.16
	Var	0.95	0.79
NY Times	Mean	2.42	3.515
	Var	1.493	1.039
NBC	Mean	2.315	4.06
	Var	1.265	0.836
Fox	Mean	2.835	3.965
	Var	1.388	1.023
US News	Mean	3.085	3.78
	Var	1.327	1.09

Table 4: Average Turker scores for Independent Clusters on a 5-point Likert scale.



(a) 23andMe (YouTube)

(b) Fox Health News (tweets)

(c) NBC Health News (tweets)

(d) NY Times Health News (tweets)

(e) US News (Health) tweets

Figure 2: Topic Coherence.

5.2 Quantitative Evaluations

Topic Coherence

The first quantitative comparison between *set cover* and LDA is in terms of the topic coherence metric. For each dataset, we plot topic coherence as a function of the number of topics ranging from 5 to 25 (for the *set cover* approach, the number of topics corresponds to the number of clusters). Figure 2 presents the topic coherence results. In nearly all of these cases (with the few apparent exceptions), *set cover* scores significantly better on this metric than LDA. It is also notable that *set cover* tends to improve as we increase the number of topics, whereas this is typically not the case for LDA (New York Times Health News tweets is an exception, where *set cover* scores decrease with the number of topics, while LDA scores increase slightly, so that for a large number of topics the two approaches are indistinguishable).

Classification

The final evaluation uses two objective document classification tasks to compare the effectiveness of *set cover* and LDA in producing a sparse feature representation for such tasks. We present classification accuracy by varying number of topics again from 5 to 25. Figure 3 shows classification results. While LDA appears to be slightly better in the Ham/Spam email classification case, and is occasionally better in the Hate/Offensive speech classification task, the differences are quite small, with both achieving accuracy in the 87-89% range in the former, and 77-78% in the latter.

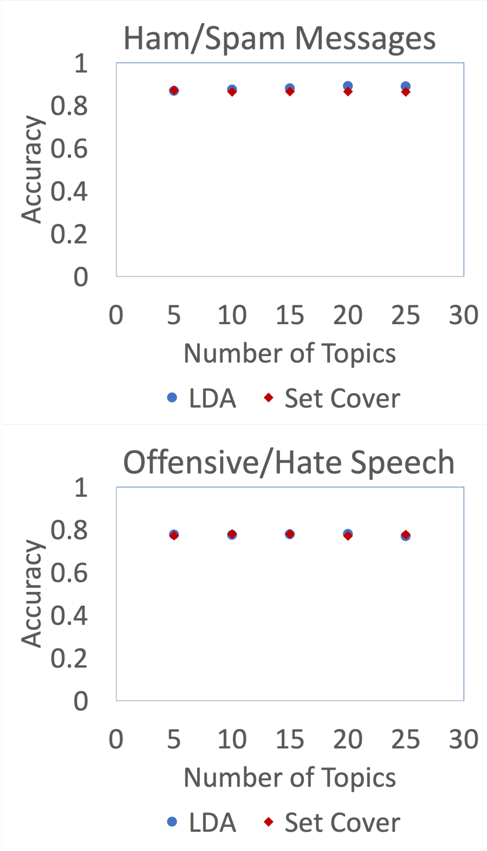


Figure 3: Classification accuracy comparison.

6 Discussion

The reason for LDA's observed inferiority in the qualitative experiments can be traced back to the fact that LDA allows each document to be generated from a mix of topics. However, in most short-text corpora, documents usually pertain to a single topic. Additionally, the number of documents belonging to each topic in a corpora is not (explicitly) captured by LDA.

With the *set cover* approach, the clustering step provides us this information - clusters need not be of uniform size, and such a clustering is easy to learn. This may explain, for instance, why LDA completely misses the word 'measles' in the Matched Clusters sample shown in Table 1. The number of documents about the measles outbreak in the corpora are relatively few, and treating this set of documents independently of other documents in the corpus makes it easier to identify this theme.

The topic coherence experiments show that the topic words learnt using *set cover* are more likely to co-occur across the corpus as compared to those learnt with LDA, thereby suggesting that *set cover*'s choice of topic words is more meaningful. The results of the classification task are noteworthy, given that our model is far less complex than LDA, and yet produces almost as effective a sparse representation.

7 Conclusion

In this paper, we introduced a conceptually simple and highly interpretable deterministic topic modeling algorithm based on constructing a semantic set cover over clusters of documents in a corpus. Unlike popular probabilistic topic modeling methods, our algorithm performed well on short text data, thereby overcoming the limitations imposed by corpus-sparsity. We demonstrated that our approach significantly outperforms LDA on qualitative scores by human judges as well as the standard topic coherence metric, and that it is comparable to LDA for document classification.

One limitation of our approach is the dependence on a good clustering of documents, in the sense that documents are meaningfully grouped together by the clustering algorithm used, given a dataset. Additionally, we rely on a word embedding, which may not be easy to learn over datasets where terms do not recur in the same contexts frequently. A potential solution to this is to learn the embedding on the union of said dataset with another corpus of similar (thematic and structural) nature, where term co-occurrences are more frequent.

Finally, as future work, we aim to explore set-cover based topic modeling where the covering threshold set as the top-k similar words to a document varies for each topic. Hopefully, this will allow us to capture the notion that some topics are sufficiently captured by a smaller set of words whereas others may need a larger threshold to fully capture their semantics.

References

Ahmet Aker, Emina Kurtic, AR Balamurali, Monica Paramita, Emma Barker, Mark Hepple, and Rob Gaizauskas. 2016. A graph-based approach to topic clustering for online comments to news. In *European Conference on Information Retrieval*, pages 15–29. Springer.

Tiago Almeida, José María Gómez Hidalgo, and Tiago Pasqualini Silva. 2013. Towards sms spam filtering: Results under a new dataset. *International Journal of Information Security Science*, 2(1):1–18.

Amazon Web Services. 2018. Amazon comprehend - developer guide.

David M Blei and John D Lafferty. 2006. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pages 113–120. ACM.

David M Blei, John D Lafferty, et al. 2007. A correlated topic model of science. *The Annals of Applied Statistics*, 1(1):17–35.

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

Jonathan Chang, Sean Gerrish, Chong Wang, Jordan L Boyd-Graber, and David M Blei. 2009. Reading tea leaves: How humans interpret topic models. In *Advances in Neural Information Processing Systems*, pages 288–296.

Vasek Chvatal. 1979. A greedy heuristic for the set-covering problem. *Mathematics of Operations Research*, 4(3):233–235.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media*, pages 512–515.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391.

Thomas Hofmann. 1999. Probabilistic latent semantic analysis. In *Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence*, pages 289–296.

Liangjie Hong and Brian D Davison. 2010. Empirical study of topic modeling in twitter. In *Proceedings of the 1st Workshop on Social Media Analytics*, pages 80–88. ACM.

Amir Karami, Aryya Gangopadhyay, Bin Zhou, and Hadi Kharrazi. 2018. Fuzzy approach topic discovery in health and medical corpora. *International Journal of Fuzzy Systems*, 20(4):1334–1345.

Akshay Krishnamurthy. 2011. High-dimensional clustering with sparse gaussian mixture models. *Unpublished paper*.

Edward Loper and Steven Bird. 2002. Nltk: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1*, ETMTNLP '02, pages 63–70, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013a. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013b. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751. Association for Computational Linguistics.

David Mimno and Andrew McCallum. 2012. Topic models conditioned on arbitrary features with dirichlet-multinomial regression. *arXiv preprint arXiv:1206.3278*.

Alexandros Mittos, Jeremy Blackburn, and Emiliano De Cristofaro. 2018. "23andme confirms: I'm super white" – analyzing twitter discourse on genetic testing. *arXiv preprint arXiv:1801.09946*.

Andrew Y Ng, Michael I Jordan, and Yair Weiss. 2002. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, pages 849–856.

Vivek Kumar Rangarajan Sridhar. 2015. Unsupervised topic modeling for short texts using distributed representations of words. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 192–200. Association for Computational Linguistics.

Asbjørn Steinskog, Jonas Therkelsen, and Björn Gambäck. 2017. Twitter topic modeling by tweet aggregation. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 77–86. Association for Computational Linguistics.

Keith Stevens, Philip Kegelmeyer, David Andrzejewski, and David Buttler. 2012. Exploring topic coherence over many models and many topics. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 952–961. Association for Computational Linguistics.

Stijn Marinus Van Dongen. 2000. *Graph clustering by flow simulation*. Ph.D. thesis, Utrecht University.

Jianshu Weng, Ee-Peng Lim, Jing Jiang, and Qi He. 2010. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining*, pages 261–270. ACM.

Xiaohui Yan, Jiafeng Guo, Yanyan Lan, and Xueqi Cheng. 2013. A biterm topic model for short texts. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 1445–1456. ACM.