Phrase Based Embedded Topic Model

Zijian Wu (zijian.wu@berkeley.edu)

Abstract

Recent advancements in topic modeling have allowed for an embedded topic model (ETM), which combines traditional topic modeling using latent Dirichlet allocation and pre-trained word embeddings. While the unigram ETM is an improvement on traditional topic modeling, we are able to further improve the quality of topics by including noun phrase n-grams in the training and testing corpuses. Furthermore, we run a series of experiments to optimize the hyperparameters that can be used to create generalizable pre-trained word embedding models that can be applied to ETMs.

1 Introduction

As the amount of data collected each day grows, it becomes increasingly important to be able to catalogue and extract information from free text for everyday use. While large state-of-the-art models such as GPT-3 or BERT have made massive strides in certain tasks such as text generation or question answering, for summary statistics of a large corpus of text in order to determine which topics have been written about and how trends have changed over time, most organizations continue to default to latent Dirichlet allocation (LDA).

While LDA can reasonably effectively determine the distribution of words within a topic and the distribution of topics within a document, the bag-of-words approach loses meaningful information carried by word order and is less effective when working with documents that have a long tail of infrequent words. Yu et al. (2013) have shown that clearer topics can be extracted from key phrases, which retain some information conveyed by word order, rather than the full text of individual words within a corpus, and Dieng et al. (2019) have been able to partially overcome the limitation of rare words by utilizing the meaning carried within word embeddings using an embedded topic model.

Despite Yu and Dieng's improvements, no research has yet combined the two approaches. Furthermore, while Dieng et al. had leveraged pre-trained skip-gram Word2Vec vectors, Levy et al. (2015) have shown that there can be large variability in the representations of words based on the hyperparameters used to train word embeddings, potentially indicating further potential to improve on the embedded topic model by tuning word embedding hyperparameters.

The aim of this project is two-fold. First, we aim to determine whether we can improve on the unigram embedded topic model by incorporating key phrase n-grams into the corpus. Secondly, as many organizations work in very specific domains, it is often necessary to rely on generalizable pre-trained word embeddings learned on a custom corpus rather than relying on open source pre-trained embeddings. Given the potential variability in results based on the hyperparameters used to create the pre-trained word embeddings for the embedded topic model, we create a series of experiments with variations in a number of hyperparameters to determine a combination of hyperparameters that can produce a word embedding model optimized for embedded topic models.

2 Background

Latent Dirichlet allocation (LDA): In the LDA model, a document is represented by a combination of topics, and a topic is represented by a combination of words (Blei et al. 2003). To generate a hypothetical document, one would first

define a distribution of topics (e.g., this document is 30% on topic A and 70% on topic B) and then sample from the words representing topic A and B (e.g., for a 100 word document, 30 words should come from topic A based on the word probability distribution within that topic and similarly 70 words should come from topic B). While this approach is certainly effective, it relies on a bag-of-words approach which ignores meaning conveyed by word order. For example, while a document on physics may mention a "unifying" or "united" framework to describe multiple physical "states", this is very different from political documents mentioning the "United States".

Phrase Based LDA: As n-grams are often more informative than individual terms, by keeping some of the meaning conveyed by word order, Yu et al. (2013) were able to show that n-grams extracted from documents presented with better topics than the traditional unigram approach. Yu et al. used the C-value method to extract noun phrases candidates by looking for nouns, adjectives + nouns, or adjectives + nouns + noun prepositions. Candidates were then scored based on the frequency of the phrase and how often longer phrases contained it before higher scoring candidates were extracted as final noun phrases. Yu et al. were able to qualitative determine that phrases produced superior topics by having 11 participants grade the topics extracted. We similarly aim to extract noun phrases, but we are able to leverage the CNN pre-trained model within the open source Spacy package to extract phrases. Furthermore, given the challenge and variability in acquiring live participants, we use topic coherence as an intrinsic quantitative measure of topic quality.

Embedded topic model (ETM): In order to incorporate more recent advancements in word embeddings, Dieng et al. (2019) created the embedded topic model. Similar to standard LDA, topic distributions (β) are estimated for each document and probability distributions (θ_β) for unique words are defined for each topic. However, the embedded topic model is also able to define a topic embedding within the same space as the word embeddings. Given a W x M matrix, called ρ, of W words with embedding size M, the embedded topic model defines a topic vector (T) of size M so that $\theta_\beta = \text{softmax}(\rho * T)$.

Compared to standard LDA, Dieng et al. reports better results (e.g., higher topic coherence). Unlike LDA, which purely replies on frequency statistics, because the embedded topic model can rely on large pre-trained word embeddings, it's able to potentially more effectively incorporate words that may not appear very often.

Topic coherence: While Yu et al. were able to obtain the assistance of 11 respondents to provide input on the quality of topics, in order to more efficiently compare multiple models, we use topic coherence as the intrinsic quantitative metric. In our case, we decide to use positive pointwise mutual information, also used by Dieng et al. The overall idea is that the top 10 most likely words in a topic should appear together often and have high mutual information.

3 Methods

In order to determine the efficacy of combining phrasing with the embedded topic model as well as determine a combination of hyperparameters best suited to create pre-trained word embeddings for the embedded topic model, we varied three major categories of parameters in each pretrained word embedding model: phrasing, preprocessing, and word embedding parameters. To allow for a suitably large corpus for training as well as small enough to run in a reasonable time period, the first 261500 sections of the 2017 Wikipedia Dataset (~600MB of text) were used as inputs into a generalizable word embedding model. Embeddings from the resultant model were then used to create topics for the 20 Newsgroups dataset. Topic coherence was ultimately used to compare the quality of topics from each iteration.

Preprocessing: The Wikipedia and 20 Newsgroups datasets were first processed by uncasing all terms and removing punctuation and extra white spaces. Four versions of the datasets were then created. The first one had no further pre-processing and included both stop-words and the original text unchanged. The second included stop words but had lemmatized tokens. The third excluded stop words but did not lemmatize tokens, and the last version both excluded stop words and had lemmatized tokens.

Phrasing: In order to test the inclusion of phrases in the quality of the topic models, noun

chunks were extracted the document being processed (i.e., Wikipedia or 20 Newsgroups). The phrases then had their tokens linked by an underscore (e.g., 'noun phrase' becomes 'noun_phrase') and were put back into the corpus.

The pre-processing and phrasing were performed together using the Spacy en_core_web_sm model package to tokenize, clean, and lemmatize the text as well as extract noun phrases and link them together. The Spacy model is a multi-task CNN pre-trained on the OntoNotes dataset and is able to extract named entities as well as perform dependency parsing and assign part of speech tags.

Word embeddings: The processed Wikipedia texts were then used to create word embeddings using two common models (Word2Vec or FastText), number of epochs (10 to 50), and using skipgram (SG) or continuous bag of words (CBOW).

Data from the processed Wikipedia dataset was streamed into the Gensim language processing package in order to create the different types of word embedding models. In order to create the multiple models, FastText and Word2Vec embeddings were trained on multiple Google Cloud instances with 12 CPUs and 32GB of memory.

Embedded topic model: The 20 Newsgroups dataset was split into a training set of 11, 260 documents, a test set of 7, 532 documents, and a validation set of 100 documents. The embedded topic models are created using the word embeddings pre-trained on the Wikipedia dataset and applied to the 20 Newsgroups training dataset. To create the embedded topic models, we leverage the code provided by Dieng et al. in their original embedded topic model paper, which uses PyTorch to efficiently estimate the topic distributions (β).

Once topics have been extracted, topic coherence is calculated on the 20 Newsgroups testing set. Both the training and testing of the embedded topic model were computed on a Google Cloud instance with 8 CPUs, 30GB of memory, and 1 GPU.

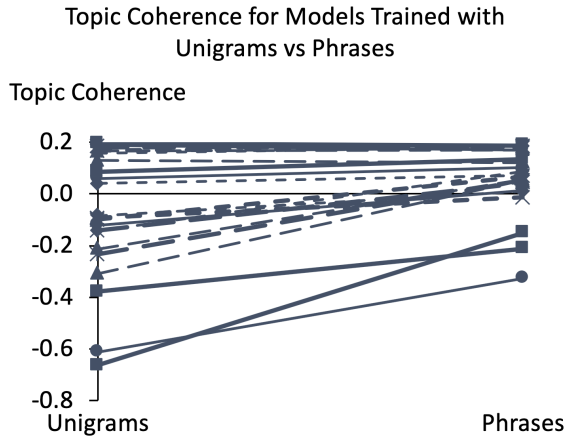

Figure 1. Ladder plot showing topic coherence for models trained with unigrams vs phrases, all other parameters held constant.

4 Results and discussion

Effect of noun phrases: Figure 1 shows a ladder plot with the change in topic coherence between embedded topic models that use unigrams vs those that include noun phrases, all other hyperparameters held constant. The average improvement in topic coherence is about 0.13, from a baseline average of -0.061 to 0.071. A paired T-test indicates of p-value of 0.003.

Nevertheless, despite the increase in topic coherence, we do acknowledge that the overall coherence score across all models is markedly lower than a simple LDA model on unigrams (0.475). In part, this is due to the relatively size of the training set used to create the word embeddings. While standard LDA can rely purely on token statistics within the testing corpus, the embedded topic model also relies on the quality of the underlying word embeddings. When the training corpus is too small or too unrelated to the documents the model is evaluated on, the quality of topics may decrease. Furthermore, as noun phrase tokens are even more rare than unigram tokens, a sparsity problem becomes more apparent as certain noun phrases within the 20 newgroups testing set may not be found within the Wikipedia dataset used for training the word embeddings.

Despite the small subset of data used to train the word embeddings and the fact that Wikipedia is relatively unrelated to the 20 Newsgroups data, there is still a noticeable improvement to topic coherence by including noun phrases. Even for the

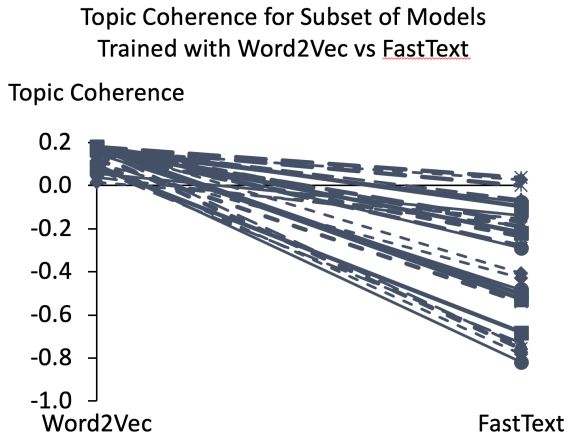

Figure 2. Ladder plot showing topic coherence for models trained with Word2Vec vs FastText, all other parameters held constant.

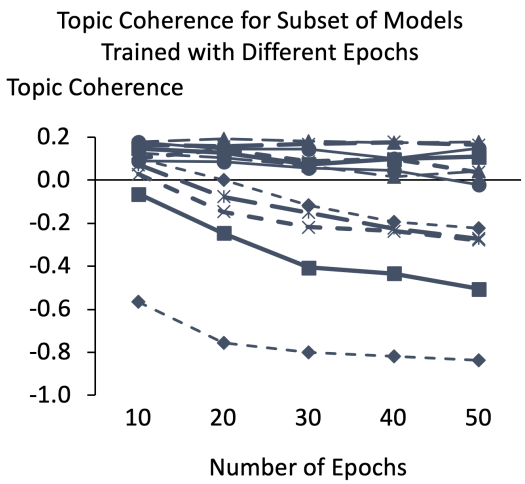

Figure 3. Plot of topic coherence for models trained with different number of epochs, all other parameters held constant.

standard LDA with noun phrases, topic coherence increases from 0.475 to 0.509. This indicates that regardless of ultimate topic model technique used, the incorporation of phrases into the tokenization rather than simply breaking the corpus into unigrams can improve the model. Pre-trained word embedding models using a large enough corpus or a smaller corpus directly related to the testing set may also create a superior topic model; however, if there is insufficient resources to create a large pre-trained model or if there is no word embedding training corpus directly related to the documents the topic model will be applied on, a standard LDA approach may work better.

Effect of lemmatization and stop words: Overall, it appears that neither lemmatization nor the removal of stop words has an impact on the quality of the embedded topic model. A paired t-test shows that utilizing lemmatization creates almost no consistent difference in topic coherence with a p-value of 0.148. Removing stop words has a p-value of 0.962 and also has no effect.

While lemmatization can act as a form of dimensionality reduction as it reduces the number of unique tokens and helps limit the sparsity issue, given a large enough corpus, the embeddings for a lemmatized and original word are likely to be highly similar. While the parts of speech and original inflections may be useful and important for word syntactic analogy tasks, for the purposes of topic modeling, words with different inflections would still be similar semantically and would thus have similar effects on the topic model.

While stop words may not contribute to the meaningful concepts within a document, they can potentially contribute to how specific word embeddings are defined for an individual token. The lack of difference in coherence with the inclusion or removal of stop words seems to concur with results by Dieng et al. They had indicated that the embedded topic model is able to be robust to the inclusion of stopwords as would potentially be grouped into their own cluster; thus, regardless of whether stop words or removed or included in the Wikipedia training set or 20 Newsgroups datasets, the results should be similar.

Word2Vec and FastText: The use of Word2Vec rather than FastText as the underlying model to create the word embeddings also creates an increase of 0.233 in topic coherence, from an average of -0.088 to 0.145 (Figure 2).

As FastText creates embeddings for character n-grams (e.g., '-ing'), its embeddings are much better able to carry syntactic information about a word. However, as validated in experiments by Hartmann et al. (2017), semantically, FastText falls short compared to Word2Vec. As topic modeling focuses more on the semantic meaning within words and documents, it makes sense that Word2Vec would perform better.

Training epochs: As seen in figure 3, additional training iterations on the Wikipedia data set actually reduces the topic coherence of the resulting model. A paired t-test between models

run at 10 epochs and 50 epochs have a p-value of <0.0001. In part, this reduction in topic coherence may be due to a combination of the small size of the embedding training corpus and high number of epochs leading to potential overfitting on the Wikipedia corpus. As the word embeddings must be generalizable to the 20 Newsgroups dataset or any other corpus that one wishes to apply topic modeling to, if these embeddings are overfit on specific Wikipedia articles, the resulting topic quality is likely to decrease.

Continuous bag of words (CBOW) and skipgram (SG): In general, the relative advantage of can vary depending on the context and the task. Levy et al. indicated a few examples of contrasting research where CBOW or SG had performed better. For the purposes of the embedded topic model, it appears that training a model using CBOW is able to increase topic coherence from an average of -0.342 using SG to 0.116, an increase of about 0.459.

5 Conclusion

For the purpose of topic modeling, we were able to confirm that addition of phrasing within pre-processing is able to increase topic coherence whether one uses standard LDA or an embedded topic model. For domain specific applications where one is able to pre-trained a word embedding model to be generalized across topic modeling applications, the ETM is not better than LDA when the word embedding training corpus is small or too different from the documents being topic modeled.

If a suitably large and relevant training corpus is available, we have shown that training word embeddings using Word2Vec with continuous bag of words for a relatively low number of epochs (10) allows for greater generalizability. The removal of stopwords and lemmatization in the pre-processing steps have a negligible impact on overall topic quality.

References

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. J. Mach. Learn. Res.3, null (3/1/2003), 993–1022.

Adji B. Dieng and Francisco J. R. Ruiz and David M. Blei. 2019. Topic Modeling in Embedding Spaces. arXiv. 1907.04907

Hartmann, Nathan & Fonseca, Erick & Shulby, Christopher & Treviso, Marcos & da Silva, Jéssica & Aluisio, Sandra. (2017). Portuguese Word Embeddings: Evaluating on Word Analogies and Natural Language Tasks.

Levy, Omer, et al. “Improving Distributional Similarity with Lessons Learned from Word Embeddings.” Transactions of the Association for Computational Linguistics, vol. 3, 2015, pp. 211–225., doi:10.1162/tacl_a_00134.

David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. 2010. Automatic evaluation of topic coherence. In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT ’10). Association for Computational Linguistics, USA, 100–108.

Z. Yu, T. R. Johnson and R. Kavuluru, "Phrase Based Topic Modeling for Semantic Information Processing in Biomedicine," 2013 12th International Conference on Machine Learning and Applications, Miami, FL, 2013, pp. 440-445, doi: 10.1109/ICMLA.2013.89.