Contextualized Topic Models with Commonsense Knowledge

Felipe González-Pizarro
Department of Computer Science
University of British Columbia
felipegp@cs.ubc.ca

Raymond Li
Department of Computer Science
University of British Columbia
raymondl@cs.ubc.ca

1 Introduction

The vast amount of data has led to the development of various techniques designed to summarize and understand textual data (Peter et al., 2015). A well-known method is topic modeling, a robust approach for extracting core themes or *topics* from a large collection of documents. When topic modeling is applied to a document corpus (e.g., a collection of news articles), the output will include a list of topics (e.g., topics related to "politics", "economics", "sports"). Usually, each topic is represented by a collection of terms that make sense together (e.g., {"tropical", "storm", "hurricane", "cyclone", "weather", and "rain"}) (Zhao et al., 2021).

From a practical standpoint, topic modeling can be viewed as an extreme form of multi-document summarization, where it can be used to understand the underlying general themes presented in a large collection of documents (Blei et al., 2010; Boyd-Graber et al., 2017). However, studies have shown that the output of topic models do not always accurately represent the characteristics of the analyzed document collections (El-Assady et al., 2019) or make sense to the end users (Hoque and Carenini, 2015). Part of this problem is because most topic modeling approaches focus on the co-occurrence of terms as the primary signal to detect the topical structure among them (Harrando and Troncy, 2021). As a result, these methods do not capture semantic and pragmatic relations between terms in the corpus (Harrando and Troncy, 2021; Song et al., 2020; Hong et al., 2020).

Prior work has suggested using external knowledge to overcome this drawback (Hong et al., 2020), and commonsense knowledge (i.e., relations between concepts) is one promising alternative (Harrando and Troncy, 2021). In this project, we test this hypothesis by performing top modeling with commonsense knowledge using two techniques. In the first technique, we extend Contextualized Topic Model (CTM) (Bianchi et al., 2021a), a state-of-the-art topic model that implements black-box variational inference, to include commonsense-aware embeddings as input (§3.2). In the second technique, we employ a clustering-based approach using the commonsense-aware embeddings of extracted concepts in the corpus (§3.3). In §4, we perform extensive experiments on three datasets and discuss the trade-off between the two techniques.

Our contributions can be summarized in three folds. First, we systematically experimented with commonsense embeddings (i.e., COMET (Bosselut et al., 2019) and ConceptNet NumberBatch (Speer et al., 2017)) as a viable solution to incorporate commonsense knowledge into CTM. Secondly, we explored clustering as a potential alternative to LDA-inspired techniques (e.g., Neural Topic Models). Finally, from our experiment results, we discuss the trade-off between the two techniques and provide motivations to incorporate corpus-level semantic relationships between words into neural topic models.

2 Related Work - Topic models

During the last few years, several topic modeling techniques have been proposed. The most influential and popular topic modeling algorithm is *Latent Dirichlet Allocation* (LDA) (Blei et al., 2003). This Bayesian probabilistic generative model summarizes a collection of documents as a set of latent topics. These latent topics are represented by distribution over words, and a mixture of these topics represents the documents.

Over the years, LDA has been used in several domains (e.g., social media analysis (González et al., 2019; Jang et al., 2021)), showing the technique's usefulness. However, it has several limitations. First, it fails to get interpretable topics in large heavy-tailed vocabularies (Dieng et al., 2020). To

mitigate this problem, practitioners could remove the most and least frequent words from the vocabulary. However, by doing so, the scope of the topics might be restricted (Dieng et al., 2020). Secondly, their inference process is challenging to scale efficiently on large text collections or extend in parallel computing facilities (Zhao et al., 2021).

With the recent developments of deep neural networks, several *neural topic models* (NTMs) have also been proposed to address probabilistic topic model limitations in terms of performance, efficiency, and usability (Zhao et al., 2021). For instance, Srivastava and Sutton (2017) proposed one of the first NTMs: *Product-of-Experts LDA* (ProdLDA), a topic modeling algorithm based on a variational autoencoder (Blei et al., 2017).

ProdLda takes the Bag-of-Words representation of documents as input to learn two parameters μ and σ^2 of a Gaussian distribution. Then it samples a continuous latent representation from these parameters, passing through a softplus[1] to obtain a document-topic distribution. Finally, this topic-document representation is used to reconstruct the Bag-of-Words (BoW) representation of the input documents. Another difference between this approach and LDA is that it is based on Product of Experts (PoE) (Hinton, 2002). These features make ProdLDA consistently identifies more coherent and segregated topics than LDA (Srivastava and Sutton, 2017; Sridhar et al., 2022). Also, given that it is based on variational autoencoders, it can process more data in lower execution times (Srivastava and Sutton, 2017).

One of the shortcomings of LDA, and ProdLDA, is that they take as input Bag-of-Words document representations. These representations do not account for the syntactic and semantic relationships among words which might impact the performance in identifying high-quality topics (Bianchi et al., 2021a). For instance, prior work indicates that while ProdLDA learns topics of relatively high quality, they are usually redundant, which indicates poor segregation of the topics (Burkhardt and Kramer, 2019).

In this context, Dieng et al. (2020) proposed *Embedding Topic Models* (ETM) to incorporate semantic relationships into topic models. ETM is a generative probabilistic model that relies on word embeddings (Mikolov et al., 2013b) to identify interpretable topics. Word embeddings capture the semantic relationship between terms in a lower-dimensional vector space, which also helps mitigate the shortcomings of LDA regarding its poor performance in large heavy-tailed vocabularies.

Like LDA, the ETM algorithm model each document as a mixture of topics, and each word is assigned to a particular topic. The differences between these algorithms rely on the per-topic conditional probability of a term. In ETM, the topics are modeled as points in the embedding space, and the topic-term distributions are proportional to the inner product of the topic's embeddings and each term's embeddings.

A shortcoming of ETM is that do not consider syntactic relations among words. Prior work suggests that the performance of the neural topic models might be boosted by injecting contextual information. Bianchi et al. (2021a) and Bianchi et al. (2021b) proposed *Contextualized Topic Models* (CTM) to address this limitation. CTM is a family of neural topic models that combines external word representation with a bag of words representation to get more coherent topics than previous approaches. While both ETM and CTM rely on vector word representations, the first one uses static word representation of words (e.g., word2vec). In contrast, CTM uses contextual embeddings (e.g., SBERT (Reimers and Gurevych, 2019a)) to get a richer syntactic and semantic representation between tokens (Zhao et al., 2021).

2.1 Topic modeling and Commonsense

Topic modeling algorithms such as CTM show that adding contextual information to neural topic models significantly improves the resulting topics' coherence. Recent work has explored expanding these contextualized representations by injecting external knowledge, such as commonsense knowledge (see (Bosselut et al., 2019)), to improve their performance. Injecting commonsense knowledge into topic modeling algorithms might help to obtain a more semantically meaningful representation of the input document (Shah et al., 2021) and, therefore, topics more aligned with commonsense relations.

Commonsense knowledge has already been used for different tasks such as question answering (Bauer et al., 2018), sentiment analysis (Ghosal et al., 2020; Ravi et al., 2021), and dialogue (Young et al., 2018). However, only a few attempts to incorporate commonsense knowledge into topic

[1] SoftPlus is a smooth approximation to the ReLU function

modeling algorithms exist (Rajagopal et al., 2013; Harrando and Troncy, 2021).

One of these scarce approaches is the *Commonsense Topic Model* (CSTM) (Harrando and Troncy, 2021). This recently proposed topic modeling technique augments clustering with knowledge extracted from ConceptNet (Speer et al., 2017) to find more interpretable topics by humans. After evaluating this approach on several datasets, the authors claim their proposal generally finds more coherent topics than the traditional LDA.

Considering these promising results, in this project, we explore if we find more coherent and diverse topics by injecting common-sense reasoning into a neural topic modeling algorithm. To the best of our knowledge, we are the first to inject common-sense knowledge into a neural topic modeling algorithm. Moreover, we also explore clustering commonsense-based embeddings as a potential alternative to Neural topic models.

3 Methodology

We use the dense representation from embeddings trained on commonsense knowledge graphs to avoid dealing with problems such as missing edges and relationships between unnormalized concepts. In §3.1 we describe resources to obtain commonsense embeddings. Then, we explain how we incorporate those embeddings into CTM (see §3.2), and in a clustering-based technique (see §3.3).

3.1 Commonsense Embeddings

We use two well-known methods to generate commonsense-aware embedding representations, namely, Conceptnet Numberbatch (Speer et al., 2017) and COMET (Bosselut et al., 2019).

ConceptNet Numberbatch embeddings are pretrained word vectors (i.e. Word2Vec (Mikolov et al., 2013a)) retrofitted on the ConceptNet knowledge graph (Speer et al., 2017). This is done by optimizing the vectors with an objective function such that the vectors are close to their original values and also close to their neighbors in the ConceptNet knowledge graph represented as a sparse symmetric term-term matrix.

COMmons**E**nse **T**ransformers (COMET) (Bosselut et al., 2019) is a transformer language model trained on two commonsense knowledge graphs: (ConceptNet (Speer et al., 2017) and ATOMIC (Sap et al., 2019)), to generate the object token in the <subject, relation, object> tuple. The hidden state representations of the transformer encoders can be used as contextualized embeddings for the input document.

3.2 Incorporating Commonsense Embeddings into CTM

Using the embeddings described in §3.1, we first create a commonsense-based document representation by following these steps. First, we extract all the tokens from a document. Then, for each token, we obtain their respective embedding in ConceptNet Numberbatch. We represent a document as the mean of those commonsense embeddings. Note that in our experiments, we also consider applying max pooling to those vector representations.

We follow a similar pipeline when using COMET. First, we encode the document's entire text using the COMET encoder. Then, we obtain the last hidden states of the encoder. Again, we represent a document as the mean of those hidden states. During our experiments, we also consider applying max pooling to those representations.

Contextualized Topic Models (CTM) use contextualized embeddings to account for word order and contextual information, overcoming the limitations of Bag-of-Word models. In their experiments, Bianchi et al. (2021b) use Sentence-BERT (SBERT) (Reimers and Gurevych, 2019b) embeddings to get a representation of the documents. SBERT is a modification of the BERT network using siamese and triplet networks that is able to derive semantically meaningful sentence embeddings.

To inject commonsense knowledge, we decide to concatenate the commonsense embeddings (see Section 3.1) to the SBERT embeddings. The intuition is that a neural topic model such as CTM can perform better by having a better representation of documents.

3.3 Clustering Commonsense Embeddings

An alternative view to the traditional LDA-based methods is directly congregating embeddings to get semantically similar word clusters (Sia et al., 2020). For this method, we first extract normalized ConceptNet nodes from the input documents using CoCo-Ex (Becker et al., 2021) before clustering the aggregated representations to obtain the topical clusters. Finally, we construct the topic-word distribution $p(w|t)$ based on the distance between word embeddings and the topic cluster centroids.

Since ConceptNet nodes are not normalized (e.g., *bake cake, baking cakes*), they often consist of several different nested phrase types (e.g., *buying the ingredients of the recipe, a friend was celebrating a birthday*), and usually contain uninformative, over-specific, or misspelled concepts, we first extract the normalized concept mapping using an off-the-shelf tool, CoCo-Ex.

In a nutshell, CoCo-Ex first extracts candidate phrases from the input text based on part-of-speech (noun, verb, adjective phrases) before lemmatizing the candidate phrases to create a dictionary based on ConceptNet nodes. For each entry in the resulting dictionary, the key is a lemmatized word contained in the concept nodes (e.g., *dog*), while the value is a list of ConceptNet nodes containing this lemma (e.g., *dog, dogs, my dog, my neighbor's dog*). After a filtering step that removes matched phrases based on embedding similarity, we use the dictionary key (word lemma) as the extracted normalized concept.

To obtain the embedding representation of the normalized concept, we take the average embedding vector over its list of phrases (i.e., value in the dictionary). We then perform clustering on concept embeddings based on their cosine similarity. In this work, we employ Spectral Clustering (Ng et al., 2001) to divide the fully-connected graph (defined by the pairwise cosine similarity matrix) into sub-graphs. Note that this is identical to the normalized cuts algorithm (Shi and Malik, 2000) used in previous work (Joty et al., 2013). To account for the word importance, we also weigh the normalized concepts by their occurrence frequency in the corpus. In practice, this is done by assigning each concept to a frequency bin (1-50). Finally, to directly compare against LDA-based methods, we create the topic-word distribution $p(w|t)$ by normalizing the distance between each concept embedding to the cluster centroid for topic t.

4 Experiments

In this section, we start by describing the datasets (§4.1) and automatic evaluation metrics (§4.2) used in our experiments before presenting and discussing our results in §4.3.

4.1 Datasets

To evaluate our models across a variety of domains, we train and evaluate our models on three datasets: 20NewsGroups[2], Wiki20K (a collection of 20,000 English Wikipedia abstracts), and Tweets2011[3]. Table 1 shows the domain and statistics of the three datasets.

Dataset	Domain	Docs	Vocabulary
20Newsgroups	Email	18,173	2,000
Wiki20K	Article	20,000	2,000
Tweets2011	Microblog	2,471	5,098

Table 1: Statistics of the datasets used

4.2 Metrics

We evaluate the quality of topic models based on topic coherence (topic keywords must shame some level of semantic relatedness) and topic segregation, which measures the lexical and semantic overlap between topics. Note that a higher value indicates a better performance in all of the metrics mentioned below.

Normalized Pointwise Mutual Information (NPMI) (Lau et al., 2014) is one of the most well-known automatic coherence metric. It measures how much more likely the most representative terms of a topic co-occur than if they were independent. NPMI returns a high score when the top N words that describe a topic, summed over all pairs w_i and w_j, have high joint probability $P(w_j, w_i)$ compared to their marginal probability (Hoyle et al., 2021). The range of NPMI is between [-1,1], whereas a higher value indicates a more coherent topic. Usually, NPMI is calculated by using a sliding window of 10 words to identify co-occurrences.

C_v (Röder et al., 2015) uses a variation of NPMI to calculate the coherence over a sliding window with size 110. It calculates the co-occurrence of a word of a given topic against all words of the same topic. It ranges between [0,1], where a higher value suggest more coherent topics.

External word embeddings topic coherence (WECO) (Ding et al., 2018) provides an additional measure of how similar the words in a topic are. It is based on word embeddings (Mikolov et al., 2013b). First, it is computed as the average pairwise cosine similarity of the word embeddings of the top 10 words in a topic. Then, the overall average of those values for all the topics is reported.

[2]http://qwone.com/ jason/20Newsgroups/
[3]https://trec.nist.gov/data/tweets/

Topic diversity (TD) (Dieng et al., 2020) is computed as the percentage of unique words in the top 25 words of all topics. Its range is between [0,1]. A value near zero suggests redundant topics, while a value near one suggests more different topics.

Inversed Rank-Biased Overlap (I-RBO) (Bianchi et al., 2021a) evaluates how diverse the topics generated by a single model are. When comparing topics, the rank of each term matters (Terragni et al., 2021); it is not the same if topics share words at high ranks as if they do at low ranks. I-RBO is the reciprocal of the standard RBO (Webber et al., 2010), and it is computed by considering the top 10 words of topics. In this metric, two topics that share some of the keywords, although at different rankings, are penalized less than two topics that share the same keywords at the highest ranks. It ranges between [0,1].

4.3 Results

We compute all the metrics for 25, 50, and 75 topics. We average results for each metric over 30 runs of each model. Table 2 shows the effects of incorporating commonsense embeddings into CTM. We identify that incorporating commonsense embeddings into model input does not strongly impact the performance of CTM. We hypothesize that the model is less sensitive to the quality of the sentence embedding (in contrast to BoW), which has been found in prior work using VAE for text modeling (Kamoi and Fukutomi, 2018). This is confirmed in our results found in Table 3, where replacing the input with CLIP embeddings (Radford et al., 2021) (which were trained on text-image pairs) achieves similar performance in two of the three datasets. We leave the detailed analysis of this approach as an exciting direction for future work.

When comparing clustering-based models with CTM (Table 2), we notice a significant improvement in three out of the five metrics (WECO, TD, and I-RBO), while trailing behind CTM in the other two (NPMI and C_v). Note that since NPMI and C_v are coherency metrics based on the co-occurrences of words in the document, it is natural that the clustering-based method performs poorly as it is simply performing semantic matching on a word level. More interestingly, the superior performance of the clustering-based model on the three other metrics reveals a potential weakness in the CTM. Since CTM is trained on document-level input, it does not explicitly model the semantic relationships between words in the corpus. Moreover, since the input document is represented as a single vector, it is hard for the model to distinguish the semantic representation between individual words when the squished representation of the input document is mapped to the latent topic variables. In future work, we would like to explicitly model this relationship by incorporating the corpus-level semantic relationship between words into CTM.

Model	NPMI	C_v	WECO	TD	I-RBO
Results for the 20NewsGroups Dataset					
CTM	**0.11**	**0.68**	0.17	**0.83**	0.99
CTM+CLIP	**0.11**	0.67	0.17	**0.83**	0.99
CTM+COMET⋆	0.10	0.67	0.17	0.80	0.99
CTM+COMET	**0.11**	0.67	0.17	0.82	0.99
CTM+Numberbatch⋆	0.10	0.67	0.17	0.82	0.99
CTM+Numberbatch	**0.11**	0.67	0.17	**0.83**	0.99
Clustering	0.08	0.64	**0.30**	**0.92**	1.00
Clustering†	0.06	0.60	**0.30**	0.82	1.00
Results for the Wiki20K Dataset					
CTM	**0.19**	0.74	0.21	0.90	1.00
CTM+CLIP	**0.19**	0.74	0.21	**0.91**	1.00
CTM+COMET⋆	0.18	**0.75**	0.21	**0.91**	1.00
CTM+COMET	0.18	0.74	0.21	**0.91**	1.00
CTM+Numberbatch⋆	**0.19**	**0.75**	0.21	**0.91**	1.00
CTM+Numberbatch	**0.19**	0.74	0.21	**0.91**	1.00
Clustering	0.04	0.50	**0.35**	**0.95**	1.00
Clustering†	0.02	0.48	**0.35**	**0.93**	1.00
Results for the Tweets2011 Dataset					
CTM	0.07	0.49	**0.16**	0.85	0.99
CTM+CLIP	0.07	0.49	0.15	0.86	0.99
CTM+COMET⋆	**0.08**	**0.50**	**0.16**	**0.89**	0.99
CTM+COMET	0.07	0.49	**0.16**	**0.89**	0.99
CTM+Numberbatch⋆	0.07	0.49	**0.16**	0.87	0.99
CTM+Numberbatch	**0.08**	**0.50**	**0.16**	0.86	0.99
Clustering	-0.45	**0.54**	**0.33**	**0.99**	**1.00**
Clustering†	-0.42	0.50	**0.33**	**0.98**	**1.00**

Table 2: Averaged results over 25 topics. The best results are marked in bold. ⋆ indicates max pooling, † indicates weighted by keywords-frequency. We use Numberbatch as an abbreviation for ConceptNet Numberbatch

In the Appendix, we include the experiment results for 50 and 75 topics in Table 4 and Table 5, respectively. In summary, the performance degrades as the number of topics increases across all models. However, the differences between models follow the same patterns as displayed in Table 2.

5 Lessons Learned

In this project, we gained a better understanding of the advantages and disadvantages between LDA-based topic models (NTMs) and clustering-based

Embeddings	NPMI	C_v	WECO	TD	I-RBO
Results for the 20NewsGroups Dataset					
SBERT	**0.11**	**0.68**	0.17	**0.83**	**0.99**
CLIP	-0.01	0.57	**0.19**	0.44	0.89
COMET⋆	0.10	0.64	0.17	0.73	0.98
COMET	0.10	0.66	0.17	0.82	**0.99**
Numberbatch⋆	0.07	0.62	0.17	0.70	0.97
Numberbatch	0.10	0.66	0.18	0.74	0.98
Results for the Wiki20K Dataset					
SBERT	**0.19**	**0.74**	**0.21**	0.90	1.00
CLIP	0.18	**0.74**	**0.21**	0.89	1.00
COMET⋆	0.18	**0.74**	0.20	0.90	1.00
COMET	0.18	**0.74**	**0.21**	**0.91**	1.00
Numberbatch⋆	0.18	**0.74**	**0.21**	0.87	1.00
Numberbatch	**0.19**	0.73	**0.21**	0.90	1.00
Results for the Tweets2011 Dataset					
SBERT	0.07	0.49	0.16	0.85	**0.99**
CLIP	**0.09**	**0.52**	0.16	0.77	0.98
COMET⋆	0.07	0.50	0.16	0.85	**0.99**
COMET	0.07	0.50	0.16	**0.87**	**0.99**
Numberbatch⋆	0.08	0.50	0.16	0.73	0.98
Numberbatch	0.08	0.50	0.16	0.72	0.98

Table 3: Performance of using different embeddings to represent the input documents in CTM. Averaged results over 25 topics. Best results are bold. ⋆ indicates max pooling.

topic models. Also, we learned more about how to estimate the quality of topics and identify automatic metrics' limitations (e.g., measuring topic diversity only on the top ten keywords is not very helpful).

We have also developed skills to improve our algorithms in terms of space and time complexity. Currently, we can evaluate topic models five times faster than during our project update. While the initial research question was promising, we believe a more thorough literature review prior to the start of our project would be much more beneficial, as it would provide a clearer direction for us to proceed.

Regarding the logistical planning, since we have divided the approach into two non-overlapping components (CTM vs. clustering), we find it a lot easier to work asynchronously. Lastly, from a time-management perspective, we find that we underestimated the efforts of changing the inner workings of neural topic models.

6 Reflections and Future Work

Overall, we believe our project was somewhat successful in answering our initial research question. From the initial stages of this project, we were able to gain a better understanding of the inner workings of the Variational Autoencoders used for topic modeling, as well as the technical details of LDA and Variational Inference. Through our experiments, we were also able to understand the trade-off between clustering-based and NTM-based approaches which provides motivation for our future work.

Moreover, we were able to run our experiments in three full datasets, and we evaluated our results considering multiple topics' coherence and diversity metrics. We have also evaluated our models several times under different settings (e.g., different number of topics). Our current pipeline makes it easier to expand this work.

While there are some obvious weaknesses in our work, including the lack of human evaluation and the failure to propose a novel approach to integrate these two lines of work (due to time constraints), our extensive evaluation does provide a clear indication of the trade-off between the two methods, which serves as a strong motivation for future work.

For future work, we would like to perform a more detailed analysis of the two methods (e.g., the entropy of topic distribution over vocab, the geometry of latent topic variables) to gather detailed insights regarding the models' behavior. Further, we would like to use visualizations to conduct qualitative evaluations using human participants (e.g., measuring topic coverage and topics' usefulness for real-world applications) and perform experiments on other datasets (e.g., GoogleNews, BBC). In the appendix, we include a figure to show how an interactive topic modeling visualization tool can help humans to interpret and analyze intermediate results.

Finally, we would like to propose a novel approach combining the advantage of clustering and NTM-based models. In this vain, we are currently experimenting with penalizing the topic-word distribution of the CTM, so the keywords from each topic are more semantically similar.

References

Lisa Bauer, Yicheng Wang, and Mohit Bansal. 2018. Commonsense for generative multi-hop question answering tasks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4220–4230, Brussels, Belgium. Association for Computational Linguistics.

Maria Becker, Katharina Korfhage, and Anette Frank.

2021. COCO-EX: A tool for linking concepts from texts to ConceptNet. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 119–126, Online. Association for Computational Linguistics.

Federico Bianchi, Silvia Terragni, and Dirk Hovy. 2021a. Pre-training is a hot topic: Contextualized document embeddings improve topic coherence. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 759–766, Online. Association for Computational Linguistics.

Federico Bianchi, Silvia Terragni, Dirk Hovy, Debora Nozza, and Elisabetta Fersini. 2021b. Cross-lingual contextualized topic models with zero-shot learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1676–1683, Online. Association for Computational Linguistics.

David Blei, Lawrence Carin, and David Dunson. 2010. Probabilistic topic models. *IEEE signal processing magazine*, 27(6):55–65.

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. 2017. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877.

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

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. Comet: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779.

Jordan Boyd-Graber, Yuening Hu, David Mimno, et al. 2017. Applications of topic models. *Foundations and Trends® in Information Retrieval*, 11(2-3):143–296.

Sophie Burkhardt and Stefan Kramer. 2019. Decoupling sparsity and smoothness in the dirichlet variational autoencoder topic model. *J. Mach. Learn. Res.*, 20(131):1–27.

Adji B Dieng, Francisco JR Ruiz, and David M Blei. 2020. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.

Ran Ding, Ramesh Nallapati, and Bing Xiang. 2018. Coherence-aware neural topic modeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 830–836, Brussels, Belgium. Association for Computational Linguistics.

Mennatallah El-Assady, Rebecca Kehlbeck, Christopher Collins, Daniel Keim, and Oliver Deussen. 2019. Semantic concept spaces: Guided topic model refinement using word-embedding projections. *IEEE transactions on visualization and computer graphics*, 26(1):1001–1011.

Deepanway Ghosal, Devamanyu Hazarika, Abhinaba Roy, Navonil Majumder, Rada Mihalcea, and Soujanya Poria. 2020. KinGDOM: Knowledge-Guided DOMain Adaptation for Sentiment Analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3198–3210, Online. Association for Computational Linguistics.

Felipe González, Yihan Yu, Andrea Figueroa, Claudia López, and Cecilia Aragon. 2019. Global reactions to the cambridge analytica scandal: A cross-language social media study. In *Companion Proceedings of the 2019 world wide web conference*, pages 799–806.

Ismail Harrando and Raphaël Troncy. 2021. Discovering interpretable topics by leveraging common sense knowledge. In *Proceedings of the 11th on Knowledge Capture Conference*, pages 265–268.

Geoffrey E Hinton. 2002. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800.

Yang Hong, Xinhuai Tang, Tiancheng Tang, Yunlong Hu, and Jintai Tian. 2020. Enhancing topic models by incorporating explicit and implicit external knowledge. In *Asian Conference on Machine Learning*, pages 353–368. PMLR.

Enamul Hoque and Giuseppe Carenini. 2015. Convisit: Interactive topic modeling for exploring asynchronous online conversations. In *Proceedings of the 20th International Conference on Intelligent User Interfaces*, pages 169–180.

Alexander Hoyle, Pranav Goel, Andrew Hian-Cheong, Denis Peskov, Jordan Boyd-Graber, and Philip Resnik. 2021. Is automated topic model evaluation broken? the incoherence of coherence. *Advances in Neural Information Processing Systems*, 34:2018–2033.

Hyeju Jang, Emily Rempel, David Roth, Giuseppe Carenini, Naveed Zafar Janjua, et al. 2021. Tracking covid-19 discourse on twitter in north america: Infodemiology study using topic modeling and aspect-based sentiment analysis. *Journal of medical Internet research*, 23(2):e25431.

Shafiq Joty, Giuseppe Carenini, and Raymond T Ng. 2013. Topic segmentation and labeling in asynchronous conversations. *Journal of Artificial Intelligence Research*, 47:521–573.

Ryo Kamoi and Hiroyasu Fukutomi. 2018. Variational autoencoders for text modeling without weakening the decoder.

Jey Han Lau, David Newman, and Timothy Baldwin. 2014. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 530–539.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26.

Andrew Ng, Michael Jordan, and Yair Weiss. 2001. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 14.

Jessica Peter, Steve Szigeti, Ana Jofre, and Sara Diamond. 2015. Topicks: Visualizing complex topic models for user comprehension. In *2015 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 207–208. IEEE.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR.

Dheeraj Rajagopal, Daniel Olsher, Erik Cambria, and Kenneth Kwok. 2013. Commonsense-based topic modeling. In *Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining*, WISDOM '13, New York, NY, USA. Association for Computing Machinery.

Sahithya Ravi, Aditya Chinchure, Leonid Sigal, Renjie Liao, and Vered Shwartz. 2021. Vlc-bert: Visual question answering with contextualized commonsense knowledge.

Nils Reimers and Iryna Gurevych. 2019a. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019b. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992.

Michael Röder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the space of topic coherence measures. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, WSDM '15, page 399–408, New York, NY, USA. Association for Computing Machinery.

Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A Smith, and Yejin Choi. 2019. Atomic: An atlas of machine commonsense for if-then reasoning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3027–3035.

Adnan Muhammad Shah, Xiangbin Yan, Samia Tariq, and Mudassar Ali. 2021. What patients like or dislike in physicians: Analyzing drivers of patient satisfaction and dissatisfaction using a digital topic modeling approach. *Information Processing & Management*, 58(3):102516.

Jianbo Shi and Jitendra Malik. 2000. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905.

Suzanna Sia, Ayush Dalmia, and Sabrina J. Mielke. 2020. Tired of topic models? clusters of pretrained word embeddings make for fast and good topics too! In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1728–1736, Online. Association for Computational Linguistics.

Carson Sievert and Kenneth Shirley. 2014. Ldavis: A method for visualizing and interpreting topics. In *Proceedings of the workshop on interactive language learning, visualization, and interfaces*, pages 63–70.

Dandan Song, Jingwen Gao, Jinhui Pang, Lejian Liao, and Lifei Qin. 2020. Knowledge base enhanced topic modeling. In *2020 IEEE International Conference on Knowledge Graph (ICKG)*, pages 380–387. IEEE.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-first AAAI conference on artificial intelligence*.

Dhanya Sridhar, Hal Daumé III, and David Blei. 2022. Heterogeneous supervised topic models. *Transactions of the Association for Computational Linguistics*, 10:732–745.

Akash Srivastava and Charles Sutton. 2017. Autoencoding variational inference for topic models. In *5th International Conference on Learning Representations*.

Silvia Terragni, Elisabetta Fersini, and Enza Messina. 2021. Word embedding-based topic similarity measures. In *International Conference on Applications of Natural Language to Information Systems*, pages 33–45. Springer.

William Webber, Alistair Moffat, and Justin Zobel. 2010. A similarity measure for indefinite rankings. *ACM Trans. Inf. Syst.*, 28(4).

Tom Young, Erik Cambria, Iti Chaturvedi, Hao Zhou, Subham Biswas, and Minlie Huang. 2018. Augmenting end-to-end dialogue systems with commonsense knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

He Zhao, Dinh Phung, Viet Huynh, Yuan Jin, Lan Du, and Wray Buntine. 2021. Topic modelling meets deep neural networks: A survey. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 4713–4720. International Joint Conferences on Artificial Intelligence Organization. Survey Track.

7 Appendix

Our source code is available under request at `https://github.com/gonzalezf/532g_project`.

We provide a high-level schema of the architecture of CTM in Figure 1. We also provide intermediate results of a topic model visualized on pyLDAvis (Sievert and Shirley, 2014) in Figure 2.

We also run our experiments considering a different number of topics (i.e., 50, and 75). Table 4 and Table 5 show the result of adding commonsense embeddings to CTM to topic models with 50 and 75 topics, respectively. Table 6 and Table 7 shows the results of using a different type of embeddings to represent the input documents into CTM for 50 and 75 topics, respectively.

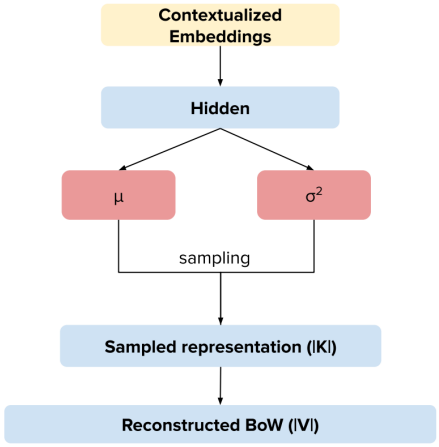

Figure 1: High-level schema of the architecture for CTM.

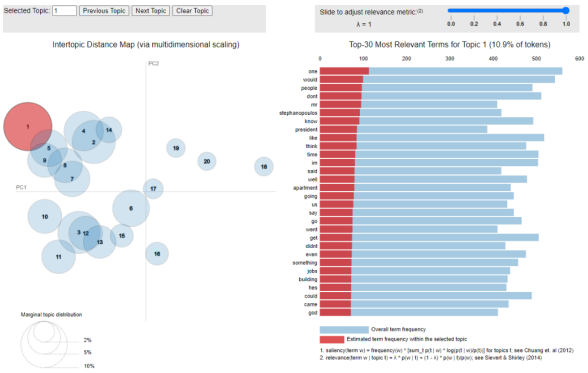

Figure 2: We used pyLDAvis (Sievert and Shirley, 2014), an interactive topic modeling visualization tool, to interpret topics and analyze the quality of our intermediate results.

Model	NPMI	C_v	WECO	TD	I-RBO
Results for the 20NewsGroups Dataset					
CTM	0.11	**0.68**	0.17	0.70	0.99
CTM+CLIP	0.11	0.67	0.17	0.71	0.99
CTM+COMET⋆	0.11	0.67	0.17	0.66	0.99
CTM+COMET	0.11	0.67	0.17	**0.72**	0.99
CTM+Numberbatch⋆	0.11	0.67	0.17	0.70	0.99
CTM+Numberbatch	0.11	0.67	0.17	**0.72**	0.99
Clustering	0.06	0.61	**0.29**	**0.77**	**1.00**
Clustering†	0.05	0.60	**0.29**	0.71	**1.00**
Results for the Wiki20K Dataset					
CTM	0.18	**0.72**	0.20	0.75	1.00
CTM+CLIP	0.18	**0.72**	0.20	0.77	1.00
CTM+COMET⋆	0.18	0.71	0.20	0.76	1.00
CTM+COMET	0.18	0.71	0.20	**0.78**	1.00
CTM+Numberbatch⋆	0.18	**0.72**	0.20	0.75	1.00
CTM+Numberbatch	0.18	**0.72**	0.20	0.76	1.00
Clustering	0.01	0.49	**0.36**	**0.83**	1.00
Clustering†	0.05	0.60	**0.29**	0.71	1.00
Results for the Tweets2011 Dataset					
CTM	0.11	0.51	0.16	0.59	0.98
CTM+CLIP	0.11	0.51	0.16	0.62	0.98
CTM+COMET⋆	**0.12**	**0.52**	0.16	**0.66**	0.98
CTM+COMET	0.11	0.51	0.16	0.65	0.98
CTM+Numberbatch⋆	0.11	**0.52**	0.16	0.64	0.98
CTM+Numberbatch	0.11	0.51	0.16	0.62	0.98
Clustering	-0.44	**0.54**	**0.34**	**0.97**	**1.00**
Clustering†	-0.42	0.51	**0.33**	**0.93**	**1.00**

Table 4: Averaged results over 50 topics. The best results are marked in bold. ⋆ indicates max pooling, † indicates weighted by keywords-frequency

Model	NPMI	C_v	WECO	TD	I-RBO
Results for the 20NewsGroups Dataset					
CTM	0.10	**0.66**	0.17	0.59	0.99
CTM+CLIP	0.10	**0.66**	0.17	0.59	0.99
CTM+COMET⋆	0.09	0.65	0.17	0.54	0.99
CTM+COMET	0.10	0.65	0.17	**0.61**	0.99
CTM+Numberbatch⋆	0.10	**0.66**	0.17	0.58	0.99
CTM+Numberbatch	**0.11**	**0.66**	0.17	**0.61**	0.99
Clustering	0.03	0.57	**0.27**	**0.64**	**1.00**
Clustering†	0.04	0.58	**0.28**	0.57	**1.00**
Results for the Wiki20K Dataset					
CTM	**0.17**	**0.71**	0.19	0.59	0.99
CTM+CLIP	0.17	0.70	0.19	0.60	0.99
CTM+COMET⋆	0.17	0.70	0.19	0.61	**1.00**
CTM+COMET	0.17	0.70	0.19	**0.62**	**1.00**
CTM+Numberbatch⋆	0.17	**0.71**	0.19	0.60	0.99
CTM+Numberbatch	**0.17**	0.70	0.19	0.61	**1.00**
Clustering	-0.03	0.45	**0.33**	**0.67**	**1.00**
Clustering†	-0.03	0.44	**0.32**	**0.63**	**1.00**
Results for the Tweets2011 Dataset					
CTM	**0.11**	0.52	**0.16**	0.44	0.98
CTM+CLIP	**0.12**	0.52	0.15	0.45	0.98
CTM+COMET⋆	**0.12**	0.52	0.15	**0.50**	**0.98**
CTM+COMET	0.11	0.51	**0.16**	0.47	**0.98**
CTM+Numberbatch⋆	**0.12**	0.52	0.15	0.48	0.98
CTM+Numberbatch	**0.12**	**0.53**	**0.16**	0.46	**0.98**
Clustering	-0.44	**0.54**	**0.33**	**0.91**	**1.00**
Clustering†	-0.43	0.52	**0.32**	**0.86**	**1.00**

Table 5: Averaged results over 75 topics. The best results are marked in bold. ⋆ indicates max pooling, † indicates weighted by keywords-frequency

Embeddings	NPMI	C_v	WECO	TD	I-RBO
Results for the 20NewsGroups Dataset					
SBERT	**0.11**	**0.68**	0.17	0.70	**0.99**
CLIP	0.00	0.58	**0.20**	0.31	0.87
COMET⋆	0.10	0.64	0.17	0.56	0.98
COMET	0.10	0.66	0.17	0.70	**0.99**
Numberbatch⋆	0.08	0.62	0.17	0.53	0.97
Numberbatch	0.10	0.66	0.19	0.53	0.98
Results for the Wiki20K Dataset					
SBERT	0.18	0.72	**0.20**	0.75	**1.00**
CLIP	0.19	0.72	0.19	0.70	0.99
COMET⋆	0.19	0.73	0.19	0.74	**1.00**
COMET	0.18	0.72	**0.20**	**0.77**	**1.00**
Numberbatch⋆	**0.20**	**0.74**	**0.20**	0.65	0.99
Numberbatch	0.18	0.72	**0.20**	0.71	**1.00**
Results for the Tweets2011 Dataset					
SBERT	**0.11**	0.51	0.16	0.59	**0.98**
CLIP	0.08	0.51	0.16	0.47	0.97
COMET⋆	**0.11**	**0.52**	0.16	**0.61**	**0.98**
COMET	**0.11**	**0.52**	0.16	**0.61**	**0.98**
Numberbatch⋆	0.08	**0.52**	0.16	0.47	0.97
Numberbatch	0.08	0.50	0.16	0.45	0.97

Table 6: Performance of using different embeddings to represent the input documents in CTM. Averaged results over 50 topics. Best results are bold. ⋆ indicates max pooling.

Embeddings	NPMI	C_v	WECO	TD	I-RBO
Results for the 20NewsGroups Dataset					
SBERT	**0.10**	**0.66**	0.17	**0.59**	**0.99**
CLIP	-0.01	0.57	**0.19**	0.26	0.86
COMET⋆	0.09	0.64	0.18	0.46	0.98
COMET	0.09	0.64	0.17	**0.59**	**0.99**
Numberbatch⋆	0.09	0.63	0.17	0.43	0.97
Numberbatch	**0.10**	**0.66**	**0.19**	0.43	0.97
Results for the Wiki20K Dataset					
SBERT	0.17	0.71	0.19	0.59	0.99
CLIP	0.18	0.72	0.19	0.51	0.99
COMET⋆	0.17	0.71	0.19	0.56	0.99
COMET	0.17	0.70	0.19	**0.61**	**1.00**
Numberbatch⋆	**0.19**	**0.73**	0.19	0.47	0.99
Numberbatch	0.18	0.71	0.19	0.54	0.99
Results for the Tweets2011 Dataset					
SBERT	**0.11**	**0.52**	**0.16**	**0.44**	**0.98**
CLIP	0.08	0.51	**0.16**	0.35	0.97
COMET⋆	0.10	**0.52**	**0.16**	**0.44**	0.97
COMET	0.09	0.51	**0.16**	0.43	**0.98**
Numberbatch⋆	0.07	0.51	0.15	0.35	0.97
Numberbatch	0.06	0.50	**0.16**	0.32	0.97

Table 7: Performance of using different embeddings to represent the input documents in CTM. Averaged results over 75 topics. Best results are bold. ⋆ indicates max pooling.