Sentence-BERT for Interpretable Topic Modeling in Web Browsing Data

Stanford CS224N Custom Project

Natalie Cygan
Department of Computer Science
Stanford University
cygann@stanford.edu

Abstract

With the great number of articles and resources accessed online, a significant portion of one's intellectual exploration is recorded in the form of browsing history. However, the tabular nature of this data existing as a list of URLs, titles, and timestamps leaves it much neglected and difficult to semantically explore. Topic modeling and document clustering are techniques used to manipulate and search collections of text for information retrieval and potential knowledge discovery. In this work, I leverage Sentence-BERT (SBERT) to build expressive embeddings for building an interpretable space for topic modeling within browsing history data. After qualitative analysis, topic clusterings made from SBERT web page embeddings outperform those made from Doc2Vec-based document embeddings.

1 Key Information to include

- Mentor: Megan Leszczynski
- External Collaborators (if you have any): N/A
- Sharing project: N/A

2 Introduction

Nowadays, much intellectual exploration happens through a web browser. However, the breadcrumbs that trail all this activity are largely unstructured. Common browsers retain lists of browsing history, which is typically timestamped, but making use of this information requires searching by article titles or URL, which can be cumbersome to do.

When I find myself wanting to retrieve articles related to a particular topic that I read in the past from my browser history, I am limited to relying upon matching title keywords or knowing the correct temporal range in order to comb through manually. Alternatively, articles that I note to be worth saving can also be saved at the time of reading to either my bookmarks or reading list. This is nice, but still falls into the unstructured data problem. My Safari reading list has no organization, and bookmarking requires you to mark where you want to save the link. In the case that I am broadly exploring some topic, I may not yet understand the hierarchical relationship that best represents the articles. Also, bookmarking can feel tedious— a significant portion of the time, the default-chosen title for the bookmark is something unintelligible, which makes it difficult to find things later. Another issue related to retrieval of web pages includes the many articles I open yet do not read: they accumulate on my tabs list, and eventually get cleared out once they sit there for too long. However, these articles are indeed logged in my browsing history.

Having all the websites I've visited, articles I've read, pages I clicked on but maybe ran out of time to read, and Wikipedia rabbit holes I've explored organized would be incredibly valuable to me. When I want to recall sources for a particular topic for the task of writing a paper, revisiting a topic, or

sharing the information with someone else, having a semantically meaningful interpretable map of my browsing history would be incredibly useful.

Beyond my own personal motivations to have an organized browser history, powerful topic modeling techniques that can organize diverse bodies of documents and data in a semantically meaningful way is an important field within NLP for unsupervised knowledge discovery that can broadly assist engineers, designers, medical professionals, and scientists in their work [1].

In order to build a tool that accomplishes this task, I employ a method of topic modeling that leverages the expressive power of Sentence-BERT (SBERT) to create a space of rich semantic document embeddings from which documents can be clustered into unique "topics." Once these "topics" are created, I use Latent Dirichlet allocation to construct a single topic descriptor over the documents within a single cluster. The quality of this LDA-generated topic is then evaluated with the Topic Coherence metric.

Compared to the baseline of topic clusters formed from a Doc2Vec embedding model for the same input web pages, the SBERT-based embedding cluster-topics receive a better Topic Coherence score of 0.455 compared to the Doc2Vec embedding score of 0.448 for clusters created with a minimum size of 3 documents each. When the minimum cluster size increases, the Topic Coherence score for the SBERT embedding cluster-topics decreases below that of the Doc2Vec model's. However, qualitative analysis shows that SBERT embedding topics are clearly superior, with exciting results in the precision of documents included within a cluster and the interpretability between clusters.

3 Related Work

For topic modeling within bodies of documents, traditionally popular algorithms include the Latent Dirichlet Allocation (LDA) [2] and Probabilistic Latent Semantic Analysis (PLSA) [3], which are both probabilistic models that view individual documents as mixtures of some discretized set of t topics, where t is a fixed parameter. However, these models have several limitations, including the fact that the number of topics is fixed integer, and that they require significant pre-processing of the data including stop-word lists and lemmatization.

Having an interpretable space of topics is incredibly useful for exploration purposes, which is another major limitation of traditional topic modeling methods. Angelov proposes the method Top2Vec, which leverages Word2Vec and Doc2Vec to simultaneously create an interpretable space of word, documents, and topics vectors [4]. In this method, the word vectors closest to a topic vector provide a textual description for a topic within the learned joint topic-document space.

A primary limitation of training-based document embedding methods that is that they of course must be retrained for specific datasets (which will take awhile as the dataset size scales). Additionally, this means that the model must also learn about the semantic relationships between words, phrases, concepts, and ideas exclusively through the input text, which is greatly limiting.

For many NLP tasks outside of topic modeling, such as classification, generation, translation, etc., this has been a challenge as well. However, the paradigm shift of pretrained large language models such as the GPT family and BERT has introduced incredible gains on a variety of tasks due to their ability to expressively represent complex semantic relationships from being trained on massive datasets [5].

While fine-tuning large transformer models such as BERT yields far-superior results for a variety of specific tasks, this method is not suitable for semantic cluster- and comparison-based tasks for the reason that independent sentence embeddings are never computed [5]. However, in 2019 Reimers and Gurevych [5] introduced the Sentence-Bert (SBERT) model that is a modification of the BERT architecture that computes sentence and paragraph embeddings. Specifically, they add a mean pooling operation to the output of BERT/RoBERTa to compute a fixed size sentence embedding. Next, BERT/RoBERTa is fine-tuned using siamese and triplet networks to update the weights. The result is a system that now produces state-of-the-art sentence embeddings, which means that the expressive power of BERT can now be leveraged for a new subset of tasks such as clustering, or semantic similarity.

4 Approach

4.1 Doc2Vec Document Embeddings Baseline

The baseline model uses Doc2Vec to create an embedding of a web page document in the browser history. Doc2Vec is a method that deploys distributed memory and distributed bag of words models, techniques which are widely used in topic modeling [6]. An implementation for this method is accessible via the Gensim Python libraries [7].

4.2 SBERT Document Embeddings

My primary method for creating embeddings of web page documents is using Sentence-BERT.

Since SBERT has a limitation to the input text size (512 characters, enough for most sentences, short paragraphs), my procedure for constructing document involves tokenizing the entire document by sentences and then truncating any sentence that is too long (over 512 characters). Next, I use the SBERT model to compute a sentence embedding for each of these sentences. Finally, I average the embeddings for all sentences in the document to form a document embedding. This is repeated for each document/webpage in my dataset.

4.3 Creating Document Clusters

Once all documents have been translated into 768-dimension document embeddings, they are ready to be clustered into topics. Before clustering, the dimensionality of the embedding space is first reduced with the Uniform Manifold Approximation and Projection (UMAP) algorithm [8]. Next, documents are assigned to a cluster using Hierarchical Density-Based Spatial Clustering of Applications with Noise (HBSCAN) [9].

I ran the clustering step 7 times each on the Doc2Vec and SBERT embeddings for 7 different minimum cluster sizes: 3, 5, 7, 9, 12, 15, 20. For each of these clustering results, LDA was used on each cluster to find the salient topics associated with the document cluster (Discussed in the next subsection). This is done in order to see how well a topic could be modeled from a more precise set of documents.

4.4 Topic Descriptions with Latent Dirichlet Allocation

For each cluster of document, I use Latent Dirichlet Allocation (LDA) to infer meaningful topic descriptors over the cluster. LDA is a statistical generative process that computes a vector-like representation for a number of topics in a set of documents with a probability distribution of the words that best describe the topic [2]. The implementation for LDA is provided by the Gensim library [7].

Unlike how LDA is traditionally used to perform topic modeling over a large space of documents to generate some number of topics, LDA is used in this experiment solely to generate a single topic from a set of documents. This is useful because a single LDA topic is represented by a short list of words to "describe" each document cluster. The generated topic is then evaluated by the Topic Coherence Metric (Described in the evaluation method section). The main hypothesis is that by presenting a better cluster (with SBERT document embeddings) of a few documents, the topic generated by LDA would be more coherent and easy to understand.

5 Experiments

5.1 Data

The data used for this project is my own personal Safari browsing history of 65k different URLs with timestamps, extracted from Safari's locally stored SQL-like database. Each URL represents a document. I excluded URLs from the dataset that I either (1) did not want in the model (such as shopping services like Amazon) or (2) thought would not provide meaningful text (Services that require logins such as Slack, Notion, Gradescope, etc. or sites with minimal text data like Youtube). After excluding such URLs, I had 12k valid websites to use

For each of these valid URLs, I used the Trafilatura web-scraping library [10] to obtain the primary text from all of the web pages, thus forming the corpus of documents.

5.2 Evaluation method

To evaluate the results of this system, the outputed topics for each document cluster is scored with the Topic Coherence Metric. Topic Coherence is computed as:

$$C(t; V^{(t)}) = \sum_{m=2}^{M} \sum_{l=1}^{m-1} \log \frac{D(v_m^{(t)}, v_l^{(t)}) + 1}{D(v_l^{(t)})}$$

Where $D(v)$ is the document frequency of word type v, and $D(v, v')$ is the co-document frequency of word types v and v', and $V^{(t)}$ is a list of the M most probable words in topic t [11]. Intuitively, a higher topic coherence score means that the most meaningful words found for a topic have a high rate of co-occurrence in the documents the topic was developed over. This metric is provided by the topic modeling library, Gensim. [7]

5.3 Experimental details

Baseline Doc2Vec Embedding Details: For creating the Doc2Vec embeddings, Doc2Vec was trained on the input data for 100 epochs and the output vector size was set to be 768 (The same as the SBERT embedding size). All the other parameters use the Gensim `models.doc2vec` defaults.

SBERT Embedding Details: For creating the SBERT-based document embeddings for the web pages, I used `paraphrase-distilroberta-base-v1` as the underlying BERT model, which was pretrained from the SBERT library. No fine-tuning was applied to the model.

Reduction and Clustering Details: The UMAP dimensionality reduction parameters used were: 15 for the number of neighboring points used in local approximations of manifold structure (`n_neighbors`), cosine as the distance metric, and the dimensionality was reduced to 2 components.

LDA details: The Gensim LDA model used was `models.ldamodel`, with `chunk_size=100`, `passes=10`, `alpha='auto'`, and the rest set to default parameters. Only a single topic was generated from each cluster of documents, as a single cluster should represent a single topic.

5.4 Results

As discussed earlier, the clustering procedure was run 7 times each on the Doc2Vec and SBERT embeddings for 7 different minimum cluster sizes. For each of these clustering results, LDA was used to generate a single topic representation, and these topics were evaluated using the Topic Coherence Metric. Specifically, for each clustering assignment, each cluster gets its own Topic Coherence score. For each clustering assignment (One is assigned for each pair of (Embedding type, minimum cluster size), I computed the average Topic Coherence score for clusters within the assignment.

Analysis of these quantitative results is explored in-depth within the Analysis section.

6 Analysis

6.1 Qualitative Results

Qualitatively, the clusterings created from the SBERT are quite superior to the Doc2Vec ones. I was amazed at how precise the the clusters in the SBERT embedding space represented a particular topic of web pages. One such example is a cluster I found that included many links to lidar and self-driving car related articles and lidar company web pages. (See Figures 4 and 5, appendix). Notably, the clusters surrounding this one include clusters for topics I would describe as "AI companies" and "articles about Apple technologies." Moving further from this cluster, I found a whole region of clusters on different Python libraries.

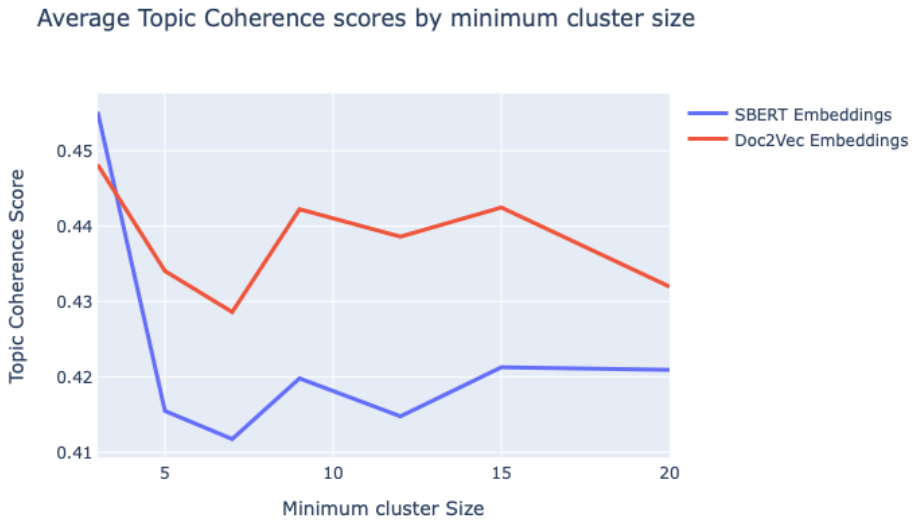


Figure 1: Average Topic Coherence scores for Word2Vec (Baseline) SBERT embedding clusters by minimum document cluster size.

On the other hand, I could not easily find many precise clusters in the Doc2Vec embedding space—the documents within most clusters appeared more random or primarily structurally related.

A good general topic to compare between the embedding spaces is where the cooking and recipe related articles ended up. In the SBERT embedding space, there remarkably was a significant subsection of the embedding space that contained almost exclusively recipe and food-related pages (See Figure 2). Not only did this large grouping of such links exist, but the labeled clusters that it comprised of were extremely interpretable subdivisions of food-related links. For example, there was a dedicated cluster for over 8 unique garlic store related websites, a cluster of tomato-related recipes, one for bread recipes, and one that mostly contained vegan dinner recipes (See Figures 6 and 7, Appendix). On the other hand, many of these same links that were logically clustered under the SBERT embeddings were not clustered together in the Doc2Vec embedding space.

For example, I found a cluster in the Doc2Vec embedding space that contained links both in the tomato-recipe cluster and lidar article cluster from the SBERT embeddings. Interestingly, this cluster received a Topic Coherence score (of 0.44) comparable to the SBERT tomato cluster (Topic Coherence: 0.46) but much higher than the SBERT lidar cluster (Topic Coherence: 0.24). This is a surprising result as some of the words I wouldn't expect to necessarily have a high co-document frequency between these two subjects (Such as "velodyne" or "tomato") (See Figure 8, Appendix).

Another interesting behavior of the Doc2Vec embedding clusters is that most of the Wikipedia articles appeared within a small sub-section of the entire embedding space (See Figure 3). Within this sub-space, the individual clusters contain sensible groups of Wikipedia articles (For example, a cluster with biology-related Wikipedia articles such as "Mitochondrial Eve", "Egg cell", "Ribosome", and "Neuron"). On the other hand, the Wikipedia articles are spread out across the SBERT embeddings based on their semantic content. As a result, the SBERT embedding space is more useful as it will put Wikipedia articles on topics such as "Neuron" and "Axon" in a cluster with links to neuroscience labs, research papers, and articles. This suggests that Word2Vec relies heavily on learning the structural similarity of articles to create embeddings. This is supported by the fact that when looking at the words that contribute most to the LDA-generated topic, all of the Wikipedia clusters have the word "Retrieve", which is the lemmatized form of the word "Retrieved", which is found many times at the bottom of almost every Wikipedia article in the citation section.

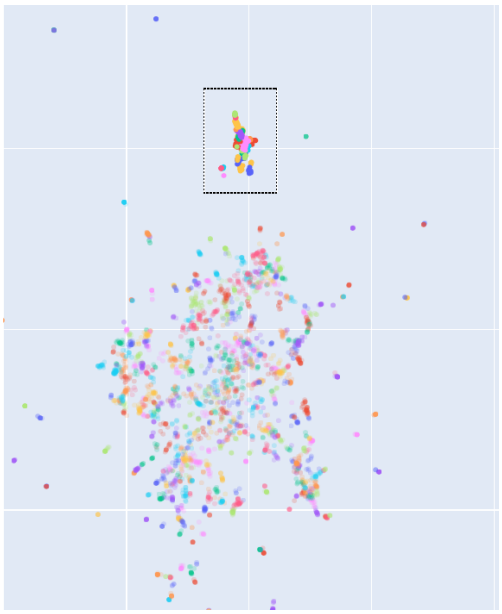


Figure 2: Area of SBERT embedding space with almost exclusively food-related clusters.

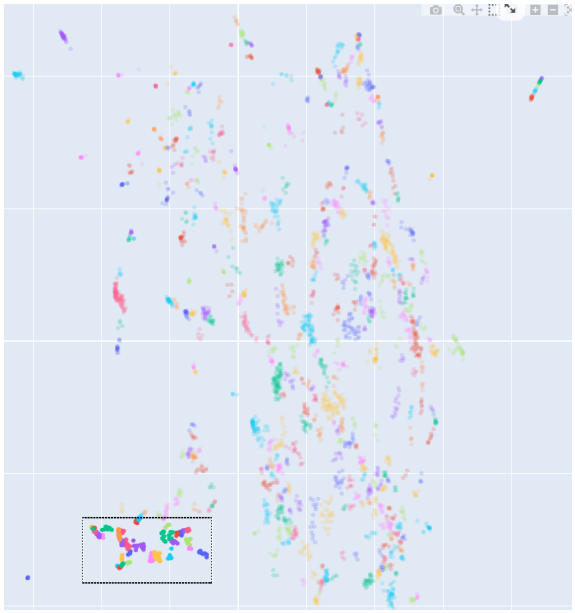


Figure 3: Area of Doc2Vec embedding space containing mostly Wikipedia articles.

6.2 Analysis of the Topic Coherence Metric Results

Based on the qualitative results, the relatively low Topic Coherence score for the SBERT embeddings is surprising. However, the comparative trends between the Topic Coherence scores for the two embeddings as the minimum cluster size increases are interesting. First, it is notable that the highest average Topic Coherence score, of 0.455, was achieved by the SBERT embeddings with the minimum cluster size of 3 documents. This suggests that when extremely small cluster sizes are possible, the documents clustered together by the SBERT embeddings are a tight selection of documents from which a highly coherent topic can be generated.

However, the Topic Coherence scores of the SBERT embedding clusters fell more dramatically when the minimum cluster size jumped from 3 to 5 compared to the Doc2Vec embeddings. This may be

explained by the fact that as the minimum cluster size increases, the SBERT embedding clusterings are more likely to include more documents that are semantically very similar, but differ structurally. In other words, including more documents in SBERT clusters introduces more words that ultimately drive down the co-document frequency of words in the LDA topic representation, whereas while the Doc2Vec embedding cluster sizes grow, they are more likely to include structural words that are maintained across a growing cluster size, thus preserving a higher co-document frequency of LDA topic words.

6.3 Limitations

For both embedding representations, there exist clusters of documents with extremely high Topic Coherence scores that are largely meaningless. One such example of this correspond to sets of documents that all display the same type of error message because the page itself was not accessible (required authentication or login), or similarly had most of its content embedded in a format that was not able to be fetched with the Trafilatura library. While I attempted to clean out similar types of URLs during my initial data cleaning stage by excluding known problematic URLs, it is difficult to find them all. As a result, any such embedding representation of diverse documents is embedded by the quality of the data corpus. The major challenge here is that is it difficult to determine what is valid and useful text from a webpage.

7 Conclusion

7.1 Summary of Results

After qualitative exploration of the SBERT document embedding topic clusterings, it is evident that these embeddings are very useful for the organization of a diverse set of documents in spite of a relatively low Topic Coherence score compared to the Doc2Vec baseline. Remarkably, the SBERT document embedding model required no training or knowledge of the document corpus and was able to produce better embeddings than the Doc2Vec model, which had to be trained over the input dataset. SBERT-based embeddings succeed wonderfully by employing an expressive understanding of the semantic concepts conveyed within the documents, whereas the Doc2Vec model appears to favor structural similarity over semantic similarity in most instances. Overall, this suggests that training over the input dataset for document topic modeling is not favorable due to the potential of over-valuing aspects of the input documents related to structural composition instead of semantic meaning.

7.2 Future Research Directions

Improving Evaluation Metrics: First, given how poorly the Topic Coherence Metric represents the overall quality of the topics derived from the SBERT embedding-produced document clusters compared to the Doc2Vec ones, exploration different evaluation metrics for SBERT-based document embedding clusters would be useful to aid the development of even better document embedding methods.

Data cleaning techniques: Because the quality of the embeddings is ultimately dependent upon the quality of the source corpus, it would also be worthwhile to investigate techniques for obtaining cleaner data of web pages that can intelligently discriminate between useful text and artifacts such as error messages.

Embedding usability and cluster interpretation: The interpretable space of document embeddings from my browsing history already serves as an interesting mind-map that I can use as retrieval for articles and links related to specific topics. There are several ways in which this system can be extended to increase its utility. One such future direction would be to develop a method for searching for a specific term or set of terms and find the cluster(s) most salient to the query.

Another such direction would be exploring alternate methods of describing the clusters to replace how LDA topic representations from a cluster were used in this experiment. This would be useful since the LDA topics are all represented as a sum of various words, and it is difficult for a single word to carry a complete yet concise description of a cluster. For example, in the case of the cluster of

tomato-related recipes, it would be wonderful to form a descriptive label for it such as "Recipes with tomato." Ideally, a topic could be described directly from the SBERT embeddings.

Finally, finding a method of hierarchical cluster descriptions would be useful for exploration of the embedding space. Both this and more concise cluster labeling would allow for building navigational tools that clearly illustrate the landscape of documents.

References

[1] Rui Wang, Xuemeng Hu, Deyu Zhou, Yulan He, Yuxuan Xiong, Chenchen Ye, and Haiyang Xu. Neural topic modeling with bidirectional adversarial training. 2020.

[2] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3(null):993–1022, March 2003.

[3] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, page 50–57, New York, NY, USA, 1999. Association for Computing Machinery.

[4] Dimo Angelov. Top2vec: Distributed representations of topics, 2020.

[5] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks, 2019.

[6] Quoc V. Le and Tomas Mikolov. Distributed representations of sentences and documents, 2014.

[7] Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA. `http://is.muni.cz/publication/884893/en`.

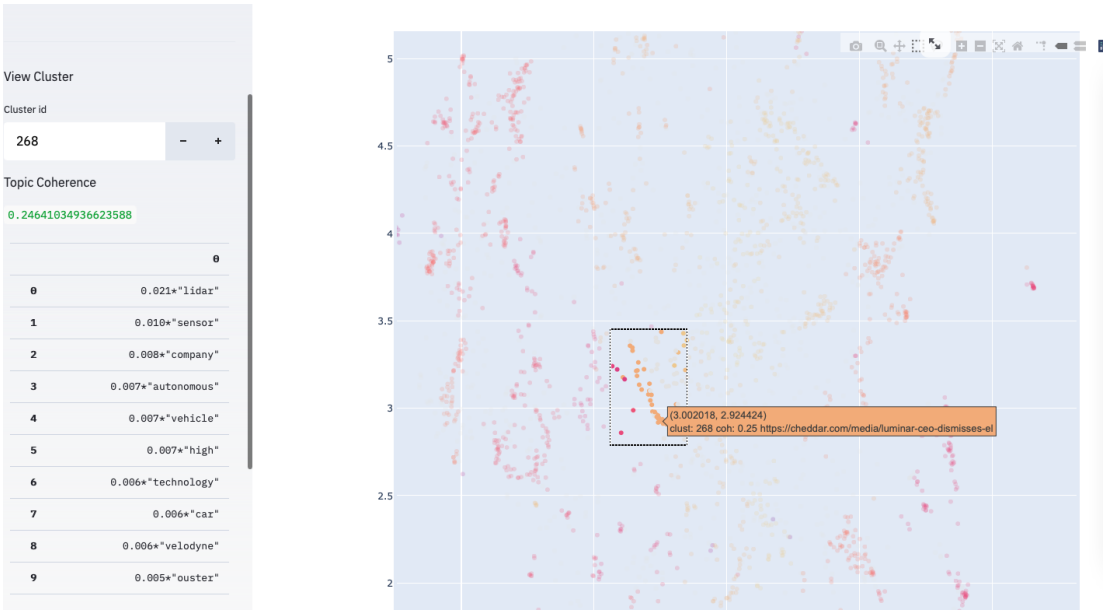
[8] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software*, 3(29):861, 2018.

[9] Ricardo J. G. B. Campello, Davoud Moulavi, Arthur Zimek, and Jörg Sander. Hierarchical density estimates for data clustering, visualization, and outlier detection. 10(1), 2015.

[10] Adrien Barbaresi, LukasBBAW, Vincent Barbaresi, Phong Nguyen, Ellie Lockhart, Ashik Paul, François Schmidts, Raphael Geronimi, and Guillaume Plique. adbar/trafilatura: trafilatura-0.8.1, March 2021.

[11] David Mimno, Hanna M. Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. Optimizing semantic coherence in topic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, page 262–272, USA, 2011. Association for Computational Linguistics.

A Appendix (optional)



(a) Lidar article cluster

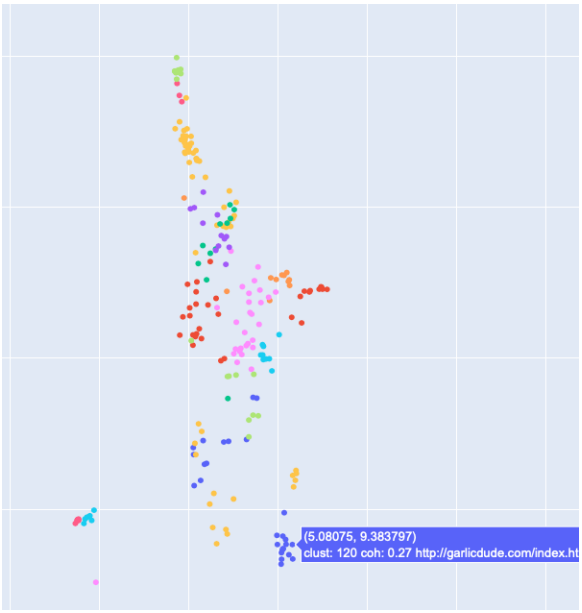
Figure 4: Area of SBERT embedding space Lidar and self-driving vehicle related articles

https://venturebeat.com/2018/11/27/volvo-and-luminar-demo-advanced-lidar-tech-that-gives-autonomous-cars-detailed-view-of-pedestrian-movements/
https://venturebeat.com/2020/09/16/intel-leans-hard-on-advanced-packaging-technologies-in-battle-for-computing-supremacy/
https://arstechnica.com/cars/2020/09/lidar-is-becoming-a-real-business/
https://ouster.com/company/
https://www.luminartech.com/technology/
https://www.luminartech.com/productshttps://www.luminartech.com/products/
https://blog.luap.info/category/startup.html
https://www.newsweek.com/hacked-billboards-can-make-teslas-see-phantom-objects-1539478
https://venturebeat.com/2018/11/19/aeye-raises-40-million-for-sensor-that-merges-camera-and-lidar-data/
https://venturebeat.com/2018/05/21/aeyes-idar-sensor-combines-camera-and-lidar-data-into-a-3d-point-cloud/
https://velodynelidar.com/press-release/velodyne-lidar-launches-vls-128-the-worlds-highest-resolution-lidar-for-autonomous-vehicles/
https://www.scmp.com/tech/innovation/article/3110694/chinese-tesla-rival-xpeng-will-add-lidar-their-self-driving
https://www.cnbc.com/2020/12/03/luminar-ipo-mints-a-25-year-old-autonomous-driving-billionaire.html
https://www.luminartech.com/
https://www.luminartech.com/updates/
https://cheddar.com/media/luminar-ceo-dismisses-elon-musk-s-critiques-of-lidar-for-self-driving
https://www.spar3d.com/news/lidar/luminar-debuts-comprehensive-and-affordable-iris-lidar/
https://venturebeat.com/2020/01/07/luminar-unveils-hydra-a-lidar-sensor-sold-on-subscription/
https://devpost.com/software/autocross
https://velodynelidar.com/
https://velodynelidar.com/products/velarray-m1600/
https://velodynelidar.com/products/alpha-prime/
https://velodynelidar.com/press-release/velarray-m1600-lidar-sensor-product-announcement/
https://www.forbes.com/sites/sabbirrangwala/2020/08/01/amazons-zoox-acquisition-is-lidar-next/
https://www.forbes.com/sites/sabbirrangwala/2020/08/01/amazons-zoox-acquisition-is-lidar-next/?sh=6d1f84b7eda8
https://zoox.com/https://zoox.com/about/
https://zoox.com/autonomy
https://www.uhnder.com/
https://news.crunchbase.com/news/austins-uhnder-raises-45m-series-c/
https://innoviz.tech/
https://innoviz.tech/innoviztwo
https://williamgibson.fandom.com/wiki/Microsoft

Figure 5: List of URLs included in the cluster of lidar-related documents

0	0.061*"garlic"
1	0.020*"wine"
2	0.019*"gilroy"
3	0.015*"highway"
4	0.015*"world"
5	0.015*"california"
6	0.014*"shoppe"
7	0.012*"rapazzini"
8	0.010*"monterey"
9	0.009*"winery"

(a) Garlic Topic Words

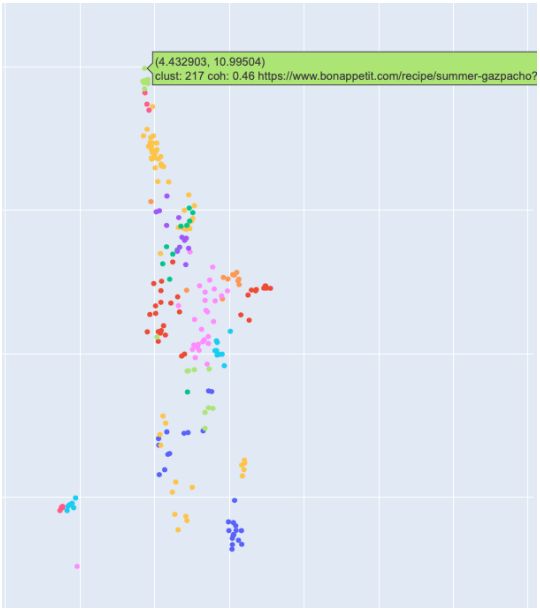


(b) Garlic Cluster (Violet cluster)

Figure 6: Example of a specific SBERT embedding space cluster that represents articles related to garlic and garlic stores near Gilroy.

0	0.029*"tomato"
1	0.021*"step"
2	0.017*"minute"
3	0.015*"egg"
4	0.015*"salt"
5	0.014*"add"
6	0.013*"heat"
7	0.013*"cook"
8	0.012*"pepper"
9	0.011*"bowl"

(a) Tomato Topic Words



(b) Tomato Cluster (Lime green cluster)

Figure 7: Example of a specific SBERT embedding space cluster that represents recipes that feature tomatoes such as eggs in purgatory, gazpacho, and buccatini with tomato sauce.

Topic Coherence

0.4405632606241718

	0
0	0.011*"time"
1	0.010*"call"
2	0.010*"hold"
3	0.009*"lidar"
4	0.009*"velodyne"
5	0.007*"high"
6	0.007*"resolution"
7	0.007*"car"
8	0.006*"tomato"
9	0.006*"safety"

Figure 8: Most salient topic words for a cluster in the Doc2Vec embedding space that contained a tomato buccatini recipe link and two lidar articles.