Member-only story

# Topic Modeling with BERT

## An Analysis of Senate Tweets

Amber Teng · Follow

Published in AI Advances · 12 min read · Feb 13, 2023



Photo by Harold Mendoza on Unsplash

Have you ever wondered what politicians *truly* care about? What do they talk about on their personal social media accounts, and what are topics that they discuss with the general public? In this blog post, we'll explore what different politicians talk about on twitter–and how that relates to their political parties.

## What is topic modeling?

To get started, let's go over a brief background of the methods we're using to determine what politicians tweet about–topic modeling. Topic modeling is a powerful tool used in natural language processing and machine learning. It's a process of identifying abstract topics that occur in a collection of documents, such as a corpus of text. This technique is used to uncover the underlying structure of a collection of text, revealing patterns and topics that might not be immediately apparent.

At its core, topic modeling algorithms use statistical methods to identify patterns in text, allowing us to understand the content of large collections of documents–such as thousands of tweets. These algorithms analyze the frequency and co-occurrence of words within documents, then assign those documents to a small number of topics. The result is a more organized and manageable view of the text, making it easier for us to understand the underlying themes and patterns present in the data.

There are a few traditional methods of doing topic modeling, including latent dirichlet allocation (LDA), latent semantic analysis, and non-negative matrix factorization. For this blog post though, we'll be using BERT for topic modeling.

## Senate Tweets

Before we dive into the code, let's first discuss the dataset we're using. For this project, I'm using a dataset of senator tweets that I manually collected. To see how I collected this data, check out my previous article here.

## What is BERT?

Earlier, I mentioned that there are different methods and algorithms to do topic modeling, particularly on a corpus of text like tweet data. For this project, we're using BERT, which stands for Bidirectional Encoder Representations from Transformers. BERT is a state-of-the-art language model developed by Google. It's considered a breakthrough in natural language processing and has quickly become one of the most popular models in use today. The reason for BERT's success lies in its innovative architecture, which allows it to understand the meaning of words in context and predict their usage in a sentence.

At its core, BERT uses a transformer-based architecture that enables it to process text in a bidirectional manner. Unlike traditional language models that only process text from left to right or right to left, BERT can understand the relationship between words in a sentence regardless of their order. This allows BERT to accurately predict the next word in a sentence, even if that word appears earlier in the sentence.

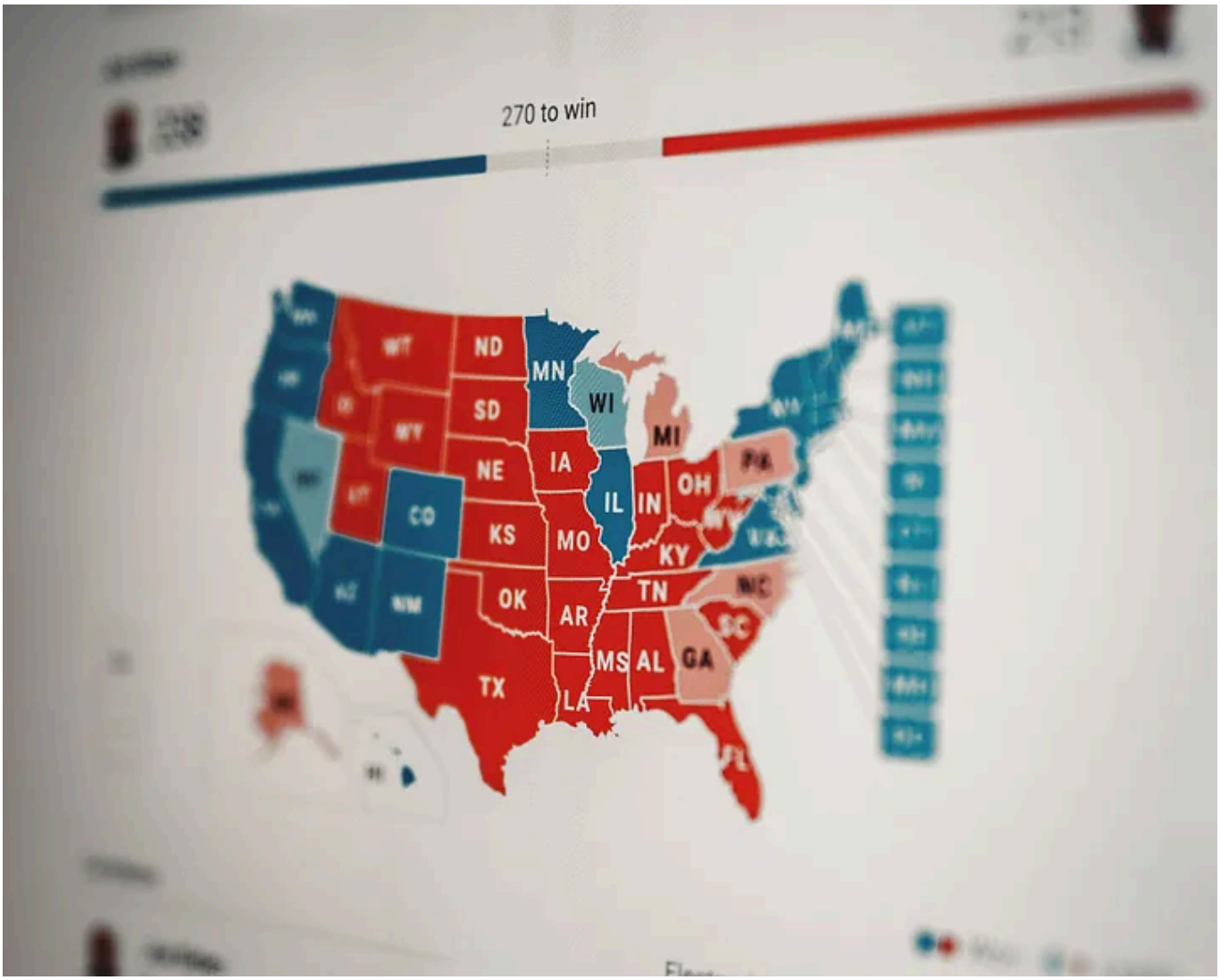



Photo by Clay Banks on Unsplash

BERT's architecture is also unique in that it uses a technique called pre-training. During pre-training, BERT is fed a massive corpus of text and is trained to predict missing words in a sentence, known as masked language modeling. This pre-training allows BERT to learn and understand the relationship between words in a large amount of text, allowing it to be fine-tuned for specific tasks, such as sentiment analysis or question answering, and in our case, topic modeling. The pre-training process also enables BERT to capture the nuances of language, such as idioms and sarcasm, allowing it to provide more accurate results for a wider range of language-related tasks. These nuances are particularly important to capture as we look through our

dataset of tweets, which are generally less formal and closer to conversational language than other types of text corpora.

## Using BERTopic for Senate Tweet Topic Modeling

BERT is becoming increasingly popular for topic modeling due to its ability to capture the context of words in a sentence. Traditional topic models typically consider words in isolation, making it difficult for them to understand the relationships between words. BERT, on the other hand, uses the deep learning architecture discussed above that allows it to grasp the context of words by considering both the words that come before and after them.

The impact of BERT on topic modeling has been significant, leading to improved accuracy and greater interpretability in the topics generated. By better capturing the relationships between words, BERT models are able to identify topics that are more closely related to the underlying content of a document. This can be particularly useful for our application of content categorization, where it is important to accurately identify the topic of a tweet in order to categorize it correctly–particularly since our goal is to better understand what politicians are discussing.

Furthermore, BERT's ability to handle out-of-vocabulary words and rare terms makes it a powerful tool for topic modeling in domains where specialized language is used. For example, in scientific domains, researchers often use specific technical terms that are not commonly used outside their field. BERT models trained on scientific documents can better identify topics related to these specialized terms, making it possible to generate more accurate and relevant topics for this domain. This is useful for political tweets because politicians sometimes refer to new memes, or current news such as COVID-19 that may be considered out-of-vocabulary words or rare

terms. Hashtags that are particular to Twitter could potentially also be treated as out-of-vocabulary words, depending on the context.

Overall, the use of BERT for topic modeling offers significant benefits for applications where it is important to accurately identify the topics present in a document. By capturing the context of words in a sentence, BERT models are able to generate topics that are more closely related to the underlying content, making them a powerful tool for topic modeling political tweet data.

To apply BERT for topic modeling, we'll be using a cool package created by Maarten Grootendorst. In summary, BERTopic uses HuggingFace transformers and the c-TF-IDF algorithm to create "dense clusters" which then allow users to view an easily interpretable list of topics, while keeping important or defining words in the topic description. To view the official package website and Quick Start, check out their homepage here.

There are five core steps to the BERTopic algorithm.

1. Embedding Generation
2. Dimensionality Reduction
3. Clustering
4. Tokenization
5. Weighting Scheme Application

First, our list of senate tweets is converted to numerical representation via sentence-based transformer embeddings. According to the package

documentation, this specific embedding method was chosen because of its optimal use when prioritizing semantic similarity and thus clustering.

Second, the BERTopic runs dimensionality reduction via the UMAP algorithm. Although principal component analysis is more popular, uniform manifold approximation and projection (UMAP) is set as the default method used in BERTopic. UMAP can help keep some of our dataset's local and global structure while preventing the curse of dimensionality–this is significant because these representations of the local and global structures contain information that may be necessary to create clusters of semantically similar documents.

Third, after embedding and dimensionality reduction, we can now cluster the documents. The BERTopic algorithm uses a density-based clustering technique called HDBSCAN. HDBSCAN can find clusters that have different shapes and can also identify outliers, when possible–this is helpful because it avoids "forcing" documents into a given cluster when they might not have the same topic.

Photo by Aditya Joshi on Unsplash

Fourth, we can then work on our bag-of-words tokenization. To create a topic representation in BERTopic, we must choose a technique that offers modularity. I mentioned earlier that BERTopic uses the HDBSCAN algorithm as a cluster model, which may have clusters of varying densities and shapes. Thus, a centroid-based topic representation might not be suitable. To address this, we combine all documents in a cluster into one document, then count the frequency of each word in the cluster to generate a bag-of-words representation. This bag-of-words representation is L1-normalized to account for clusters of different sizes and does not make any assumptions about the structure of the clusters, making it ideal for our topic representation needs.

Finally, we can now generate topic representations. From the fourth step of the BERTopic algorithm, we now want to understand the differences between the clusters generated from the bag-of-words representation. To do this, we modify the traditional TF-IDF to focus on *topics* instead of documents. Instead of comparing the importance of words between documents, BERTopic's c-TF-IDF algorithm treats all documents within a cluster as a single document and then applies TF-IDF. This provides us with importance scores for words within a cluster, allowing us to extract the most important words and get descriptions of the topics. In essence, the most important words in a cluster represent the topic it represents. BERTopic's c-TF-IDF method converts each cluster into a single document and extracts the frequency of each word in that cluster. This gives us a class-based term-frequency representation, which is L1-normalized to account for differences in topic sizes. To find the importance score per word in each cluster, we take the logarithm of the average number of words per class divided by the frequency of the word across all classes, plus one. This creates a class-based

inverse document-frequency representation, which we then multiply with its term frequency to get the importance score per word. The classic TF-IDF procedure is not used here, but a modified version of the algorithm that provides a better representation.

## Generating Senate Tweet Topics

So how does this look like in code? Actually, all of this happens under the hood using the BERTopic package, so the code is pretty simple.

First, I load in the tweet data I have and then merge my tweet dataset into another table that contains the appropriate political party of each politician.

```
Tweets_df = pd.read_csv('https://raw.githubusercontent.com/angelaaaateng/politic
party_df = pd.read_csv('https://raw.githubusercontent.com/angelaaaateng/politics

party_df['username'] = party_df['Link'].str.replace('https://twitter.com/','')
# join party_df and tweet_df to get the political party of the senators
master = pd.merge(
tweets_df,
party_df,
how="left",
left_on='username',
right_on='username'
)
master.rename(columns = {'State ':'State', 'Party ': 'Party', 'Name ': 'Name'},
```

Now that I have my master dataset, we can now proceed with BERTopic modeling. For this use case, we don't need to manually remove stopwords from the data because BERT uses transformer-embedding models that need the full context of our text in order to create accurate embeddings.

I then define a model with the appropriate specifications, fit the model to each of my datasets (Republican, Democrat, and Independent), and then get information about each topic before finally generating appropriate visualizations.

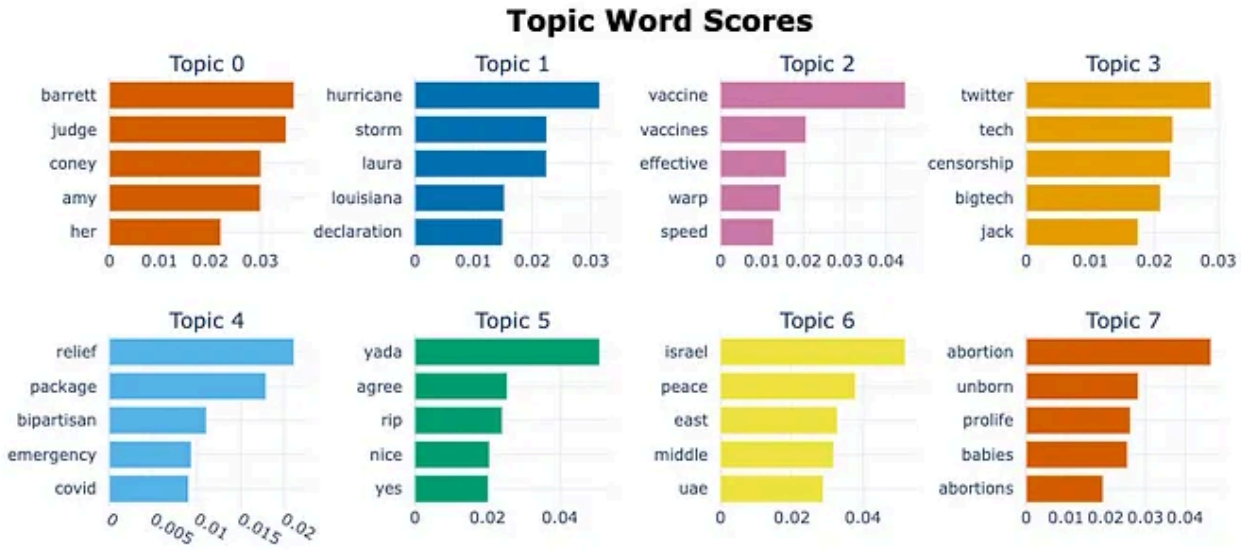
```
topic_model = BERTopic()
topics, probs = topic_model.fit_transform(content)
topics, probs = topic_model.fit_transform(content)
topic_model.get_representative_docs(0)
topic_model.visualize_barchart()
```

To view my full code, please view the Jupyter Notebook here.

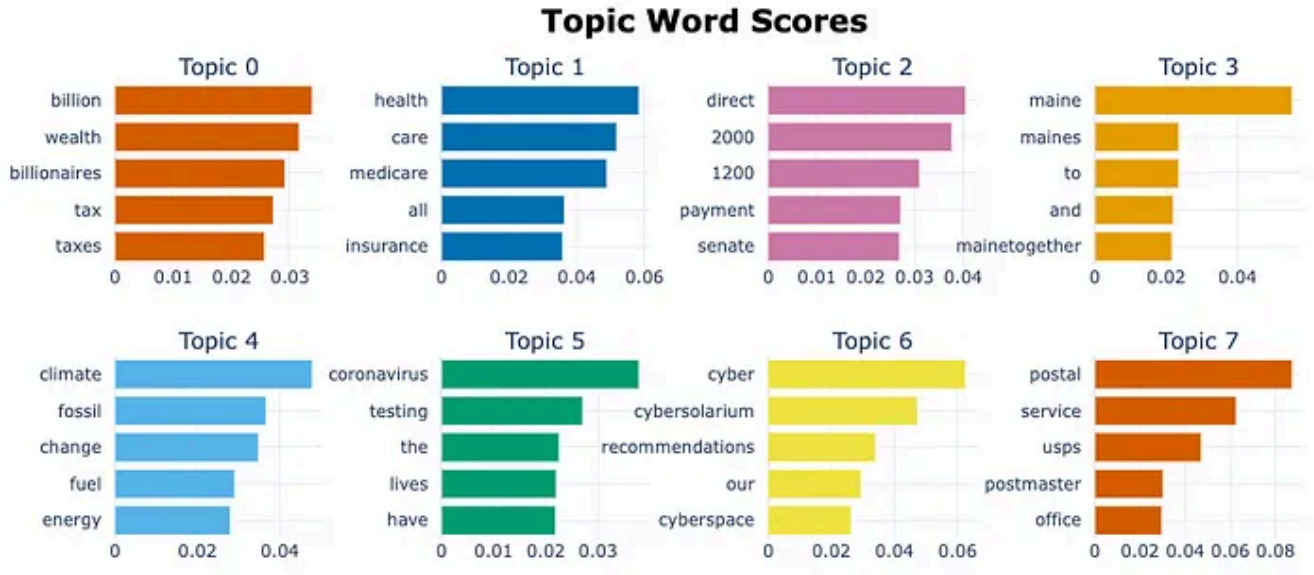
## Senate Tweet Topics

There are a few ways that I thought it would be helpful to look at the dataset, and the topics generated by our BERTopic model. The first is by analyzing the *overall topics* discussed by *all politicians*, irrespective of their political inclination. For this, we'll take a look at visualizations of each data slice's topic word scores. In the next few barcharts below, we'll see how selected terms for a few topics are visualized by creating bar charts out of the c-TF-IDF scores for each topic representation. This visualization is particularly useful because it allows us to easily compare topic representations with each other, and view the most defining words for each topic.


As seen in the barchart visualization above, overall, the top 8 topics discussed by all politicians could be described as (1) the Affordable Care Act (2) Associate Justice Amy Coney Barrett, (3) Senate Majority Leader Mitch McConnell's rejection of the COVID stimulus plan, (4) Louis DeJoy, the 75th US Postmaster General and his involvement with USPS, (5) Operation Warp Speed for Accelerated COVID-19 Vaccine Development, (6) a certain politician's impeachment and trial (probably Trump's, given the timestamp of the tweets being in mid 2020), (7) a topic related to religion, possibly, and (8) wildfires in California.




Looking specifically at the Republicans, we see that some topics are similar, which makes sense given that Republicans are a large subset of the total population. However, we also see a few new topics, namely Topic 1, Topic 3, Topic 4, Topic 6 and Topic 7. Topic 1 seems to be about Hurricane Laura which hit the Gulf of the United States in August 2020–which is aligned with our dataset timeframe. Topic 3 seems to be about big tech censorship, particularly focused on Twitter and Jack (Dorsey), the former CEO of Twitter. Topic 4 seems to be another topic focused on COVID, particularly on the COVID Bipartisan Relief Bill. Topic 6 seems to be about the Israel-United Arab Emirates Normalization Agreement, officially the “Abraham Accords Peace Agreement: Treaty of Peace, Diplomatic Relations and Full Normalization Between the United Arab Emirates and the State of Israel” that was signed by the US, Israel, and UAE in August 2020. And lastly, Topic 7 seems to be about Republicans and their pro life stance on abortion.




Now, looking at the Democrats' top 8 topics, we see that they are somewhat different compared to the Republican topics. Topic 0 seems to be about billionaires and taxes–possibly due to Democrats proposing a taxation scheme that would target billionaires' gains. Topic 1 appears to be about Medicare-for-All, and the Democrats' stance on healthcare in 2020. Topic 2 seems to be about $2,000 USD COVID-19 stimulus checks passed by the White House. Topic 3 seems to be about #MaineTogether, a nonpartisan coalition of Maine organizations that supports Maine's economy and people. Topic 4 seems to be about the Democratic stance on fossil fuels, and climate change. Topic 5 seems to be about pushing for more COVID testing. Topic 6 appears to be about the Cyberspace Solarium Commission. And lastly, Topic 7 seems to be about news relating to USPS postmaster general, which is also discussed by Republicans. One interesting thing we notice about the topics for the Democrats is that there are a lot of stopwords that typically are cleaned out during text pre-processing.


Finally, we also have the top 8 topics discussed by Independents. Most of these are topics with defining words we've seen before. A few topics that seem to be unique to independents are Topic 4 and Topic 5. Topic 4 seems to be about the Yemen War, which escalated again in 2020. Topic 5 seems to be about providing (frontline) workers with masks and other protective gear at the height of the COVID pandemic.

As we've seen, topic modeling particularly using BERT can be a valuable tool when analyzing data, and digging deeper into understanding the story that our data holds. When looking at senate tweets, we saw different ways that we can slice the data, and we also observed how different granularities of our corpus affected the topics that seemed most significant.

## What's next?

In this article, we saw the basics of using topic modeling on senator Tweet data. We also looked at how to use BERTopic, and we dove deeply into why BERT can be useful for an application such as topic modeling. We saw a few insights about how the types of content that Republicans versus Democrats discuss on social media platforms like Twitter differ. But our work doesn't

end here. There's still plenty of room to expand this study. In the future, we can look into answering questions such as:

- How does BERTopic compare to other topic modeling methods like LDA?
- What "portion" of tweets does each politician contribute to? For example, is there a specific Republican or Democrat whose tweets contribute more to a certain topic?
- Are there "divides" within each political party? Do some Republicans tweet about topics that are, on average, stereotypical Democrat? What about for Independents–do they tweet more like Republicans or Democrats? Or a mix of both?

Purchase my book here and please do report any bugs or suggestions to this article via email.

Connect with me via LinkedIn or Twitter.

Follow me on Medium.

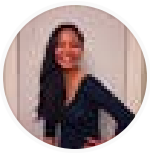
For more of my projects, check out my personal website here.

To keep up with my 100 days of Code videos, check out my TikTok here.

## Sources:

- https://iq.opengenus.org/topic-modelling-techniques/
- https://www.tidytextmining.com/topicmodeling.html#:~:text=Topic%20modeling%20is%20a%20method,for%20fitting%20a%20topic%20model
- https://maartengr.github.io/BERTopic/getting_started/tips_and_tricks/tips_and_tricks.html#gpu-acceleration
- https://maartengr.github.io/BERTopic/index.html


Data Science NLP Code Politics Twitter



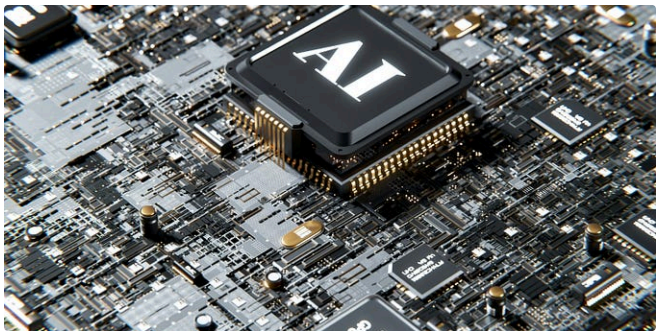
# Written by Amber Teng

Follow

600 Followers · Writer for AI Advances

A writer, learner, and explorer, Angela Teng spends most of her time thinking about how interdisciplinary collaboration can galvanize innovations in technology.

## More from Amber Teng and AI Advances




Amber Teng in AI Advances

### An Introduction to NLP and LLMs in the Age of AI

AI First Course Notes and Supporting Blog Post — Lecture 1 — Winter 2024

5d ago 113 1


Gavin Li in AI Advances

### Breakthrough: Running the New King of Open-Source LLMs...

New King of Open-Source LLM: QWen 2.5 72B

Sep 21 1.1K 14


Tarun Singh in AI Advances

Amber Teng in Towards Data Science

## AI-Powered OCR with Phi-3-Vision-128K: The Future of Document...

In the fast-evolving world of artificial intelligence, multimodal models are setting...

Oct 9 622 14

## An Introduction to Loading Large Language Models

Mastering Megamodels: An Introductory Guide to Loading Llama2 and HuggingFace'...

Oct 12, 2023 104

See all from Amber Teng

See all from AI Advances

# Recommended from Medium

Mariya Mansurova in Towards Data Science

## Topics per Class Using BERTopic

How to understand the differences in texts by categories

Sep 8, 2023 648 4

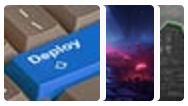

DhanushKumar

## Topic Modelling with BERTopic

BERTopic is a topic modeling technique that leverages BERT (Bidirectional Encoder...

Jul 1 13

## Lists



**Predictive Modeling w/ Python**

20 stories · 1599 saves

**Natural Language Processing**

1759 stories · 1358 saves

**Practical Guides to Machine Learning**

10 stories · 1947 saves

**The New Chatbots: ChatGPT, Bard, and Beyond**

12 stories · 484 saves

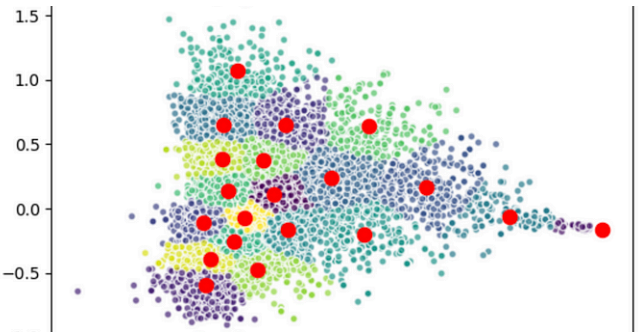
---


Shrinivasan Sankar in Level Up Coding

### TF-IDF and BM25 for RAG— a complete guide

TF-IDF and BM25 are commonly used techniques in information retrieval. while TF...

Oct 7 1

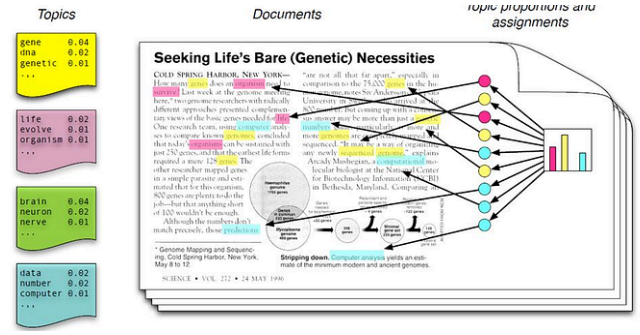



Kartheepan G

### Unveiling Text Clustering: Exploring Algorithms and Text...

Introduction In today's data-driven world, the ability to effectively analyze and organize...

Apr 22 7 1




ALSHARGI


Shashank Agarwal

## LDA Topic Modeling of the Holy Quran Text

The Holy Quran, the central religious text of Islam, is a rich source of linguistic and...

May 16 4

## Understanding TF-IDF and c-TF-IDF in Topic Modeling

Topic modeling is crucial for extracting meaningful insights from large volumes of...

4d ago

See more recommendations