

# Dravidian Top2Vec

Deepthi Sudharsan (CB.EN.U4AIE19022)  
Kolimi Pooja Reddy (CB.EN.U4AIE19038)  
Vuyyuru Bindu Sri (CB.EN.U4AIE19069)  
Kudumula Devaraj (CB.EN.U4AIE19072)

Project Mentor: Dr. Premjith B, Asst. Prof,  
Center for Computational Engineering and Networking  
Amrita School of Computing  
Amrita Vishwa Vidyapeetham

08 December 2022

## Abstract

In a vast collection of documents, latent semantic structures often referred to as topics, can be found via topic modelling. Latent Dirichlet Allocation and Probabilistic Latent Semantic Analysis are the techniques that are most frequently utilized. Despite being well-liked, they have a few flaws. They often call for the number of topics to be understood, personalized stop-word lists, stemming, and lemmatization in order to attain the best outcomes. These techniques also use a bag-of-words representation of documents that ignores word order and meaning. Due to their capacity to capture the semantics of words and texts, distributed representations of documents and words have grown in prominence. This project is about identifying topic vectors by using joint document and word semantic embedding in Dravidian languages. With this paradigm, the number of topics is automatically determined without the need for stop-word lists, stemming, or lemmatization. The topic vectors that come from this are jointly embedded with the word and document vectors, with the distance between them signifying semantic similarity.

## Keywords

Topic Modelling, Top2Vec, Dravidian Languages

## 1 Acknowledgments

We would like to express our profound gratitude to Dr. K.P Soman who has provided us with sufficient knowledge and guidance. We would also like to thank our entire Computer Science and Engineering - Artificial Intelligence Department for giving us this opportunity to nurture and hone our skills. We would like to express our gratitude to our mentor Dr. Premjith B of CEN department, for the time, guidance, and support that he has provided throughout the semester, and during the project's completion. Furthermore, we would like to thank the management of Amrita Vishwa Vidyapeetham for providing ample resources to avail our project needs.

## 2 Introduction

In this growing digital era, with the rapid increase in technology, data has also been growing. Organizing, searching and summarising large collections of text is a ubiquitous problem. With huge data, it is simply impossible for a person to read and sort out the data. To make this strenuous task possible, Topic modelling is used. Topic modelling is usually used on a large collection of data where organization and summarization of data cannot be done manually.

### 2.1 Motivation

Topic modelling has been performed in different domains where the data is in the English language. Dravidian languages are not explored till now due to the language complexity and unavailability of large data resources.

## 2.2 Major Contribution

- This project aims at developing a novel model for extracting topics in Dravidian languages.
- The proposed model compares its performance on various embedding models and tokenizers and later adapts the best one.
- A small study on outliers has also been performed as a part of the work as much work in this area has not been performed in Top2Vec and hence, it is a first step towards the future goals of the project.

## 3 Literature Survey

To date, there have only been very few studies on Top2Vec. The first study was [1] by Dimo Angelov where he proposed Top2Vec, a topic modelling algorithm that can capture latent semantic structure to find topics. Top2Vec uses UMAP for dimensionality reduction and HDBSCAN for clustering which makes it stand out among topic modelling algorithms. In [2], a comparative analysis of different topic modelling algorithms like LDA, NMF, Top2Vec, and BERTopic was performed on the covid travel data. It was concluded between LDA and NMF, NMF model outperformed LDA in general because its results are closer to those of human judgment. Between BERTopic and Top2Vec, BERTopic appears to be more focused, with a definite distinction. Top2Vec, on the other hand, offers helpful insights, particularly for the topic naming process. Overall, Top2Vec performed the best among all the other topic modelling algorithms used. [3] uses Dimo Angelov's Top2Vec [1] to ascertain the most representative topics in each country's Covid-19 news articles dataset. Jointly embedded document and word vectors are created using Doc2Vec followed by UMAP for dimensionality reduction and HDBSCAN to find dense areas. In another work, [4], the difference in the pattern between fixed intents and split or merged intents was addressed and the model was able to find new topics or intents of the user that had not been identified. These two works, go on to prove the strength and capability of Top2Vec for various applications. Even in the paper [5] by Ryan Hodgson, Top2Vec was used to automatically identify significant trends in ITS and automatically extract relationships between several ITS topics indicating the potential for novel areas of research. Through this work, Top2Vec algorithm proved that it can manage large-scale literature analysis, without limitations presented by conventional probabilistic topic models. In the research presented in [6], LDA and Top2Vec were compared on Covid vaccination hesitancy tweets and concluded that Top2Vec is able to extract more relevant topics. Our main motivation is to utilize the advantages of Top2Vec and apply them for topic modelling in Dravidian languages, which is an unexplored area of research to date.

### 3.1 Inferences From the Literature

## 4 Proposed System

### 4.1 Data Description

#### 4.1.1 Tamil Dataset Description

The dataset that we have used for our analysis is taken from Kaggle. The data contains 5 columns News\_Id, News\_Date, News\_Category, News\_Title, and News\_Article. This dataset has 15 different news categories and the total dataset contains 1,27,000 news articles.

News_Category	Count
Spirituality	405
India	16935
The World	7477
Education	240
Crime	16290
Cinema(reelma)	9248
Tamil Nadu	53333
Editorial	1535
Profession	68
Medicine	544
Mystery	66

District Masala	9079
Sports	8230
Employment	1042
State Express	2253

The Tamil dataset was used to perform over 300 experiments with different embedding models, tokenizers, and parameters to find the optimal models and parameters for topic modelling in Dravidian languages especially, Tamil in specific.

#### 4.1.2 Telugu Dataset Description

The Telugu dataset is also taken from Kaggle and it contains 18000 news articles distributed into five different categories - Nation, Entertainment, Business, Sports, and Editorial.

News_Category	Count
Nation	6628
Entertainment	5145
Business	2572
Sports	1908
Editorial	1059

- The exceptional performance of a few models on the Tamil dataset was put to test in terms of reproducibility on the Telugu dataset and, yet again, Dravidian Top2Vec performed extremely well, generating meaningful topic vectors and expected topics.

## 4.2 Methodology

### 4.2.1 Semantic Embedding

The spatial representation of words and documents is called a semantic space which jointly retains embedded document and word vector properties like grouping similar documents together and the words are kept close to the documents that they best describe. The authors of [1] stated that the semantic space with the outlined properties is a continuous representation of topics.

The dense area of documents represents the high similarity between them in semantic space, which indicates a common underlying topic for those documents. The document vectors represent the topics of the documents hence the centroid or average of those vectors can be considered a topic vector. Words closest to this topic vector are the words that best describe the document semantically. The assumption made by the authors of [1] is that the number of dense areas of document vectors equals the number of prominent topics. This is in a sense the best way to find the number of topics without having prior knowledge of the data.

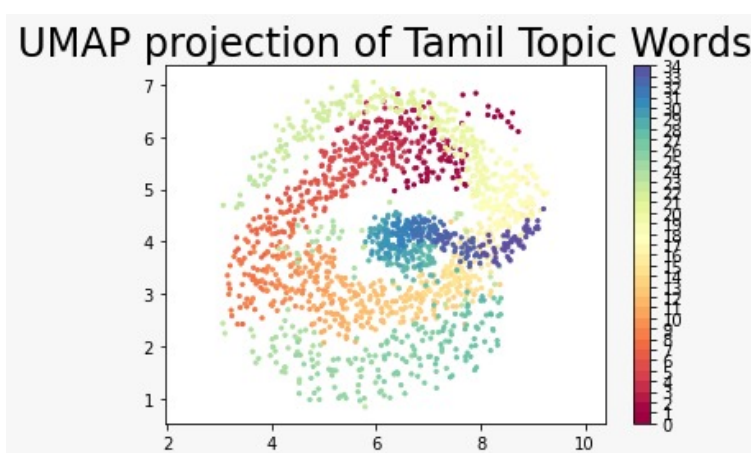
### 4.2.2 Find Number of Topics

In order to find the topic words, dense areas need to be calculated and to calculate dense areas of documents in the semantic space, we need to cluster the documents. Density-based clustering is used on the document vectors, specifically Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN).

The similarity between documents is considered based on how close the document vectors are in semantic space. But the "curse of dimensionality" takes away the interpretation of distance, because distance doesn't mean anything in that high-dimensional space which is of 300 dimensions. This introduces two main problems. In the high-dimensional semantic embedding space, the document vectors are very sparse. The document vector sparsity makes it difficult to find dense clusters and doing so comes at a high computational cost. To overcome these two problems, we perform dimension reduction on the document vectors with the algorithm Uniform Manifold Approximation and Projection for Dimension Reduction (UMAP). In the dimension-reduced space, HDBSCAN can then be used to find dense clusters of documents.

### 4.2.3 Low Dimensional Document Embedding

Dimension reduction algorithm is chosen in such a way that it reduces feature dimension and at the same time preserves the global structure of semantic space, for clustering. The strong theoretical foundations of UMAP help in achieving this task. UMAP is a manifold learning technique for dimension reduction that overcomes the drawbacks of T-distributed Stochastic Neighbor Embedding (t-SNE), in preserving the global structure and local structure, and is able to scale to very large datasets. The most important parameter of UMAP is the number of nearest neighbors, which controls the balance between preserving global structure versus local structure in the low dimensional embedding. Larger values put more emphasis on global over local structure preservation. Since the goal is to find dense areas of documents that would be close to each other in the high dimensional space, the local structure is more important in this application. Setting the number of nearest neighbors to 20 gives the best results in preserving local structure. Another parameter is the distance metric, used to measure the distance between vectors in semantic space. Often we use cosine similarity or Euclidean distance as a metric because it measures the similarity of documents irrespective of their size. Lastly, the embedding dimension must be chosen; we find 20 dimensions to give the best results for the downstream task of density-based clustering.



### 4.2.4 Find Dense Clusters of Documents

Density-based clustering is used to find areas of highly similar documents in the semantic space, that indicate an underlying topic. This process is performed on the UMAP-reduced document vectors. The algorithm should handle varying density clusters present in semantic space, without greedily assigning noise data points to clusters. There will be sparse areas where documents are highly dissimilar in semantic space; the algorithm should be able to handle that. HDBSCAN meets the requirements as it was designed to handle both noise and variable density clusters.

The HDBSCAN hyper-parameter that needs to be tuned for Top2Vec is minimum cluster size; this parameter is at the core of how the algorithm finds clusters of varying density. This parameter represents the smallest size that should be considered a cluster by the algorithm. We find that a minimum cluster size of 40 gives the best results in our experiments.

### 4.2.5 Calculate Topic Vectors

Calculate Centroids in Semantic Space After performing HDBSCAN and UMAP processes, we are left with the dense clusters of documents and noise documents identified in the UMAP-reduced dimension. These correspond to locations in the original semantic embedding space. The use of UMAP and HDBSCAN can be seen as a process which labels each document in the semantic embedding space with either a noise label or a label for the dense cluster to which it belongs. Topic vectors can be calculated, from the obtained labels for each cluster of dense documents in the semantic embedding space. The simplest method to calculate topic vector from the document vectors is to calculate the centroid, i.e. the arithmetic mean of all the document vectors in the same dense cluster. There are

other reasonable options such as the geometric mean or using probabilities from the confidence of clusters created by HDBSCAN. The sparsity of the high dimensional space reduces every technique to result in very similar topic vectors. The number of dense areas found is the number of prominent topics identified in the corpus.

#### **4.2.6 Find Topic Words**

In the semantic space, every document vector represents a topic that is best described semantically by its nearest word vectors. Therefore the word vectors that are closest to a topic vector are those that represent the document semantically. The distance of each word vector to the topic vector will indicate how semantically similar the word is to the topic.

Common words that appear in most documents are often in a region of the semantic space that is equally distant from all documents. As a result, the words closest to a topic vector will rarely be stopwords. Therefore there is no need for stop-word removal.

#### **4.2.7 Outlier Analysis**


Outlier Analysis is performed to find whether the word vectors that are labeled as noise by the model are the most common words present in the documents like stop words or just the words that don't describe the topic of documents. Get the topic words from Top2Vec and compute word embeddings using muril-base cased model. These topic word embeddings are in high dimensional space, reducing the embedding dimension using UMAP for better computation and visualization of outliers. Compute the cosine similarity of topic words with other topic words that are outside the dense cluster and label the overlapping topic words as outliers. Visualize the outliers by plotting using UMAP embeddings.

#### **4.2.8 Streamlit GUI**

Streamlit GUI was designed to create a good user experience by leveraging the deployment and interface facilities provided by the Python package. Streamlit helps deploy python models and programs with ease using its user-friendly functionalities. Below, is an image grab of the Streamlit GUI interface that has been deployed.

# Dravidian Top2Vec

Upload Dataset Here

 Drag and drop file here  
Limit 1GB per file • CSV

Browse files



tamilmurasu\_dataset.csv 494.8MB



	news_id	news_date	news_category	news_title
0	6	1/6/2011 2:45:49 PM	மர்மம்	தூக்கில் தொங்கும் சேவல்கள் திருடர்களை காவ
1	9	1/6/2011 2:56:51 PM	மர்மம்	பவுர்ணமி ஜாமத்தில் மாயமான கர்ப்பிணி
2	10	1/6/2011 3:02:00 PM	இந்தியா	காமன்வெல்த் ஊழல்: சுரேஷ் கல்மாடியிடம் 102 C
3	11	1/6/2011 3:08:15 PM	மர்மம்	மச்சுபிச்சு மலை ரகசியம்
4	12	1/6/2011 3:09:20 PM	மர்மம்	ரத்த பலி வாங்கும் விபரீத ஆவி!

Reading the Documents...

Select an Embedding Model of your choice

Muril-large

Selecting UMAP and HDBSCAN hyperparameters

Use Default

Please note that the tokenizer has been experimentally set to Muril-large cased

Topic Vectors generated successfully by Dravidian Top2Vec are :

[[ 'போலீசார்' 'போலீசில்' 'கைது' 'பதிந்து' 'போலீசுக்கு' 'விசாரணை' 'விசாரணையில்' 'போலீஸ்' 'மருத்துவமனைக்கு' 'தெரியவந்தது' 'போலீசாருக்கு' 'சேர்ந்தவர்' 'அப்பகுதி' 'கொலை' 'சம்பவ' 'விசாரித்தனர்' 'வழக்கு' 'இறந்தார்' 'புகார்' 'சம்பவம்' 'வேலை' 'பிரேத' 'இதுகுறித்து' 'உயிருக்கு' 'அப்பகுதியில்' 'என்று' 'அறிந்ததும்' 'கீழே' 'முன்பு' 'மீட்டு' 'இன்ஸ்பெக்டர்' 'திடீரென' 'சேர்ந்த' 'இறந்து' 'விரைந்து' 'வீட்டில்' 'வந்து' 'நடவடிக்கை' 'உறவினர்கள்' 'தொடர்பாக' 'அருகே' 'உடனடியாக' 'வீட்டுக்கு' 'மருத்துவமனையில்' 'உள்ள' 'கும்பல்' 'அவர்கள்' 'சிலர்' 'சடலத்தை' 'நகை' ] [ 'அணியின்' 'ஆட்டம்' 'ஆட்டத்தில்' 'அணி' 'பேட்டிங்' 'ஆட்டத்தை' 'சுழற்பந்து' 'அணியை' 'ஆட்டங்களில்' 'உலககோப்பை' 'பந்துவீச்சு' 'பேட்டிங்கில்' 'சச்சின்' 'ஆஸி' 'பந்து' 'கேப்டன்' 'வேகப்பந்து' 'காலிறுதி' 'ஆடி' 'தொடரில்' 'விக்ரெட்' 'உலககோப்பையில்' 'சேவாக்' 'டோனி' 'ஜாகீர்கான்' 'ரெய்னா' 'அணிக்கு' 'போட்டியில்' 'ரன்கள்' 'வென்று' 'ஆட்டநாயகன்' 'வீச்சாளர்' 'பந்துவீச்சில்' 'இண்டிஸ்' 'விக்ரெட்டுகளை' 'யுவராஜ்சிங்' 'வெஸ்ட்' 'ரன்' 'தோல்வியை' 'பாக்' 'சதம்' 'ஆஸ்திரேலிய' 'அணிகள்' 'வீரர்' 'வீழ்த்தி' 'அப்ரிடி' 'தென்' 'வீரர்கள்' 'விளையாடி' 'உலககோப்பையை' ] [ 'நடிக்க' 'ஹீரோயின்' 'ஷூட்டிங்' 'படத்தில்' 'படத்துக்கு' 'படம்' 'பட' 'கதை' 'நடிக்கும்' 'ஹீரோ' 'நடிக்கிறார்' 'நடித்த' 'மலையாள' 'படங்களில்' 'படத்தின்' 'தமிழில்' 'படத்தை' 'இயக்கும்' 'பாடல்' 'கூறியது' 'படங்கள்' 'பொறுக்குது' 'நடித்து' 'இந்தி' 'படத்துல' 'இசை' 'ரொம்ப' 'நிறைய' ] [ 'UNK' 'ஆகிறது' 'காதல்' 'நடிகை' 'காட்சி' 'நல்ல' 'இயக்குனர்' 'இருக்கிறார்' 'சினிமா' 'காலம்' 'காமெடி' 'சமீபத்தில்' 'தெலுங்கு' 'எனக்கு' 'மாட்டேன்' 'என்கிறார்' 'சொல்லி' 'தமிழ்' 'எனது' 'நான்' 'விஷயம்' ] ]

The above snapshot demonstrates the working of the GUI for the tamil dataset. Users can upload the dataset of their choice, and select the embedding models. The user gets to choose whether they would like to use the default best parameters that we experimentally got for UMAP and HDBSCAN or they can give their very own parameters. Then, the dataset is processed accordingly and the topic vectors are generated and printed.

## 5 Results and Analysis

In this novel work, different tokenizers, embeddings, and different parameters of HDBSCAN and UMAP have been experimented with. For tokenizers, Muril Base Cased, Muril Large Cased, Indic Bert, Opus-mt-en-dra autotokenizers were analysed. Of all these tokenizers, MurilLarge Cased tokenizer gave us meaningful words by considering the matras which are quite important in Dravidian language. Embedding models like Muril Base Cased, Muril Large Cased, Indic bert, and, Opus-mt-en-dra were trained with all the combinations of tokenizers. Out of all the 16 combinations, Muril Large Cased tokenizer with Muril Large Cased embedding with the following parameters gave us a concise number of topics with meaningful topic vectors.

UMAP Arguments		HDBSCAN Arguments	
n_neighbors	20	min_cluster_size	60
min_dist	0.2	min_samples	5
n_componenets	20	cluster_selection_method	eom
metric	euclidean	metric	euclidean

[ 'பேசியதாவது' 'கருணாநிதி' 'ஆதரித்து' 'ஆட்சிக்கு' 'திமுக' 'பேசினார்' 'திட்டங்கள்' 'நிறைவேற்றி' 'வடிவேலு' 'ஆட்சி' 'ஆட்சியில்' 'திட்டங்களை' 'குஷ்பு' 'வேட்பாளர்' 'ஜெயலலிதா' 'முதல்வராக' 'கிரைண்டர்' 'கூட்டணி' 'முதல்வர்' 'ரூபாய்க்கு' 'சாதனைகளை' 'பிரசாரம்' 'உங்களை' 'திட்டம்' 'விஜயகாந்த்' 'ஏழை' 'உங்களுக்கு' 'கருணாநிதியை' 'வேட்பாளரை' 'இவ்வாறு' 'தொடர்' 'நீங்கள்' 'கூட்டணிக்கு' 'பாக்யராஜ்' 'அறிக்கையில்' 'இலவச' 'கூட்டணியை' 'அரிசி' 'வந்தால்' 'உங்கள்' 'சொல்லி' 'மு' 'எல்லாம்' 'கலைஞர்' 'ஸ்டாலின்' 'தெரியும்' 'க' 'நடிகர்' 'வெது' 'வாக்கு' ]

Above is a sample output topic vector that the Dravidian Top2Vec model generated. In the dataset, as mentioned, there are articles related to crime. The model was able to identify crime-related topic words and club them into one vector. Tamil words that translate to 'Policeman', 'To Hospital', 'Death', 'To the Police', 'recovered', 'discovered', 'corpse', 'inspector', 'at investigation', 'incident' etc. were yielded as topic words for the topic vector "Crime".

With all the same arguments, the model has also been tested on the Telugu dataset with just minute changes (due to differences in dataset size) in the HDBSCAN and UMAP parameters. We got exceptional results from the designed Dravidian Top2Vec model. Following are the parameters that were one of the best when used for the Telugu dataset.

Umap Arguments		HDBSCAN Arguments	
n_neighbors	20	min_cluster_size	40
min_dist	0.2	min_samples	5
n_components	20	cluster_selection_method	eom
metric	euclidean	metric	euclidean

```
[ 'కెప్టెన్', 'మ్యాచ్', 'బ్యాటింగ్', 'వికెట్లు', 'టీమిండియా',
  'జట్టు', 'మ్యాచ్', 'ఓపెనర్', 'పరుగులు', 'వికెట్', 'సెంచరీ',
  'బంతుల్లో', 'ఓవర్లో', 'కోట్లీ', 'వన్డే', 'బౌలింగ్', 'బౌలింగ్',
  'స్కోరు', 'ఆటగాడు', 'జట్టుకు', 'ఫోర్లు', 'ధోనీ', 'నాటౌట్',
  'ఇంగ్లండ్', 'ధవన్', 'పాండ్యా', 'పరుగులు', 'ఠిక్', 'విరాట్', 'లంక',
  'ఇన్నింగ్స్', 'జడేజా', 'అశ్విన్', 'జట్టులో', 'ఆతిథ్య', 'ఆసీస్',
  'ఓవర్లో', 'శ్రీలంక', 'ఘోషా', 'టీ', 'అర్ధ', 'మూడో', 'సాధించాడు',
  'ఆట', 'సిరీస్', 'సిరీస్లో', 'క్రికెట్', 'ప్రత్యర్థి', 'ఘోషా',
  'లీగ్']], dtype='<U14')
```

Above picture is output topic vector that the Dravidian Top2Vec model generated for telugu dataset. In the dataset, as mentioned, there are 5 articles and one of them is sports. The model is able to identify sports-related topic words and club them into one vector. Telugu words that translate to 'Captain', 'in match', 'Batting', 'wickets', 'Team India', 'Team', 'match', 'opener', 'runs', 'century' etc. were yielded as topic words for the topic vector "Sports".

In the Outlier Analysis, it can be depicted that there are some overlapping words for different topics which can be seen in UMAP projection picture since they are from different topics and have similarity it is understood that they are the most common words in the documents. These can be studied further to find how they are related to topic words.

## 6 Conclusion and Future Work

Analysis of different embedding models, tokenizers, and UMAP-HDBSCAN parameters was performed on the Tamil dataset and the experiments yielded great results. Using the designed models, Dravidian Top2Vec was applied to the Telugu dataset and it is observed that great results are achievable with the best parameters identified, validating the reproducibility of the designed structure for other Dravidian languages. Muril-large cased tokenizer was the best performing tokenizer in all the experiments performed. A small analysis of the outliers was performed as a precursor to the future scope of the project. Streamlit GUI has been designed for the work as well, which allows users to try Top2Vec with their customized parameters and models of choice.

The original Top2Vec consists of a get\_chunks function that performs the data chunking part for the English dataset and also identifies phrases. Phrase detection and chunking are huge tasks by themselves as identifying phrases and chunking based on words are difficult due to the unavailability of already available phrase detection models and, in Dravidian languages, meaningful words can be as small as one character long and as big as over 25 characters, so these two tasks will be taken forward as a part of Phase 2. Based on our sample experimentation with outlier analysis, in-depth outlier analysis will be demonstrated as a part of the next phase. High-Performance Computing will also be experimented with to speed up the topic modelling process and the Dravidian Top2Vec model will be released as a Python package. Reproducibility on Malayalam and Kannada datasets will also be performed in parallel.



## References

- [1] Dima Angelov. Top2vec: Distributed representations of topics. *arXiv preprint arXiv:2008.09470*, 2020.
- [2] Roman Egger and Joanne Yu. A topic modeling comparison between lda, nmf, top2vec, and bertopic to demystify twitter posts. *Frontiers in Sociology*, 7, 2022.
- [3] Piyush Ghasiya and Koji Okamura. Investigating covid-19 news across four nations: a topic modeling and sentiment analysis approach. *Ieee Access*, 9:36645–36656, 2021.
- [4] Darell Hendry, Fariz Darari, Raditya Nurfadillah, Gaurav Khanna, Meng Sun, Paul Constantine Condylis, and Natanael Taufik. Topic modeling for customer service chats. In *2021 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, pages 1–6. IEEE, 2021.
- [5] Ryan Hodgson, Alexandra Cristea, Lei Shi, and John Graham. Wide-scale automatic analysis of 20 years of its research. In *International Conference on Intelligent Tutoring Systems*, pages 8–21. Springer, 2021.
- [6] Phillip Ma, Qing Zeng-Treitler, and Stuart J Nelson. Use of two topic modeling methods to investigate covid vaccine hesitancy. In *Int. Conf. ICT Soc. Hum. Beings*, volume 384, pages 221–226, 2021.