Topic Analysis in the Media and the Biggest Hurdles Facing the Biden Administration: Using LDA and BERT for Topic Modeling of News Articles[1]

John Stephen, Parker Johnson, Scott Serpa, Swati Sundar & Taylor Stevenson

July 17th, 2022

[1]Address to which correspondence should be addressed:

jojo.step12@gmail.com, parkerstephenjohnson@gmail.com, scottserpa@gmail.com, sundar.swati@gmail.com, t.s.goodwin18@gmail.com

Abstract

President Biden has encountered a variety of significant issues during his presidential administration. To take a deeper look at these issues, our team has gathered a corpus of twenty-eight news articles sourced from credible organizations and applied different types of Natural Language Processing (NLP) techniques. We performed five experiments using various algorithms for vectorization, clustering, and Topic Modeling. The vectorization algorithms used are TF-IDF and Doc2Vec. We experimented with K-means and Latent Dirichlet Allocation (LDA) clustering algorithms. Furthermore, we used different flavors of BERT embedding. In parallel, we constructed an ontology to serve as a ground truth to compare the algorithms' results. In this effort, we identified the top challenges the administration has faced and the possible relationships between these issues within the limitations of our corpus.

Keywords

President Biden, Natural Language Processing, k-means, reference term vector, Latent Dirichlet Allocation, Bi-Directional Encoder Representations from Transformers, ontology, Term Frequency-Inverse Document Frequency

Introduction

Tracking discussions in media and news websites is a way to monitor the current situation and gain insights on leading topics and themes. The purpose of this study is to identify the common topic and themes that contribute to the issues faced by the Biden Administration. Using a corpus of documents focused on the challenges faced by the Biden Administration, we perform five experiments using different combinations of vectorization, clustering, and topic modeling algorithms. We used Doc2Vec and TF-IDF for vectorization. Doc2Vec is an unsupervised algorithm used to convert a document to a vector, and in our experiments, we use the Gensim implementation. TF-IDF is a statistical method used to measure how important a term is within a document relative to the entire corpus. In Experiment 3, we combined the vectorization from TF-IDF with the unsupervised clustering algorithm k-means. For k-means, the user determines the number of clusters or centroids, k. The algorithm then collects each data point and assigns it to a cluster. In simple terms, Latent Dirichlet Allocation (LDA) can automatically cluster topics within documents. It is a powerful statistical approach for organizing and understanding the relationships between topics and documents in a corpus. For topic modeling, we experimented with the transformer-based BERT algorithm.

Literature review

Natural language processing (NLP) has transitioned from rule-based to probabilistic systems. With the advance in computing power of personal computers, it is now feasible to train models that would have been impossible to create a decade ago. Today, more complex patterns train utilizing advanced algorithms to answer queries linked with multiple applications. It has been studied by researchers and used in various domains, including software engineering, political science, medicine, and linguistics. Latent Dirichlet Allocation (LDA) and BERTopic are two of many topic modeling techniques. There are many articles published on the Latent Dirichlet Allocation (LDA) and BERTopic in topic modeling, and researchers have presented a variety of models.

Some works have focused on the media in Topic modeling. For example, Hillard sought to gain a deeper comprehension of what determines the success of Latent Dirichlet Allocation (LDA) models by tracing the connection between the production of reference term vectors and the LDA's results (Hilliard 2021). Using a corpus centered on legislation proposed during the Biden administration's first months, they demonstrated that topic mapping is susceptible to even the smallest changes in its input space. They established the importance of manual feature engineering and tied it back to the difference between bag of words and TF-IDF.

A similar study by Ahmed et al., 2022 used Latent Dirichlet Allocation (LDA) techniques to explore aspects of the Pakistani economy (Ahmed 2022). This study adopted a data-driven approach to drive meaning out of a larger volume of text under consideration. They collected 3,000 articles from two Pakistani English newspapers. LDA was then applied to the corpus to extract and frame the topics. Ten topics were extracted, each consisting of ten keywords. Then they analyzed the keywords in detail to find out common themes. These themes contributed to getting the text's overall picture, which outlined Pakistan's economic dynamics.

Minghao & Mengoni, 2020 conducted a study on text analysis using LDA (Wang and Mengoni 2020). They collected a total of 1127 articles and 5563 comments from the South China Morning Post (SCMP) on COVID-19 from January 20 to May 19. After preprocessing the corpus, they trained the LDA model, and parameters were tweaked using Cv coherence as the model assessment technique. They then analyzed the results using the optimal model to determine the dominant topics as well as the representative documents of each topic.

Using a collection of articles from the New York Times from the year 2020, Vidiyala applied the Gensim open-source Python library and LDA algorithm to the corpus (Vidiyala 2021). She went through the various pre-processing of the data and applied the bag of words technique for feature extraction. Next, the vectorized corpus was passed into the LDA model using Gensim. Finally, PyLDAvis was used to visualize the model. Results showed a total of sixteen topics with the top thirty most relevant terms for the topic. They identified words like corona, virus, and pandemic are the most discussed topic in 2020 across the globe.

Sethia compared the results from various trending topic modeling techniques to machine learning techniques (Sethia, 2022). They applied a state of art topic modeling framework using BERT, LDA, and machine learning techniques using k-means for classifying news articles under similar topics. Using LDA as the baseline, they tested various experiments using simple word embedding models like Doc2Vec and a sentence transformer using BERT. The best clustering resulted from using Doc2Vec in the case of a simple word embedding model.

Data

The corpus consists of twenty-eight documents focusing on “Major Challenges that the Biden Administration is Facing.” These documents are articles written by prominent news sources in the second quarter of 2022 and trimmed to 500 words. Some reputable sources include the NY Times, NPR, Reuters, CNN, AP, Washington Post, Forbes, and the BBC.

The next step was to begin preprocessing the data. We performed standard preprocessing of lowercasing the words, removing punctuation and symbols, and removing stop words derived from the NLTK library. We applied lemmatization and stemming individually and together and found that lemmatization produced better results for our corpus. We removed the term ‘us’ to differentiate between the use of ‘us’ and the abbreviation for the United States (US). We also removed words under four letters except for a list of hand-curated important terms to the corpus. We supplemented the NLTK stopwords list with additional words that came up frequently but were not impactful, such as ‘should’ or ‘would.’ We then performed a word frequency evaluation as a preliminary analysis technique to understand the effects of the preprocessing. Figure 1 details the word frequency across all documents in the corpus. The next step was to create n-grams of our corpus using TF-IDF’s n-gram function.

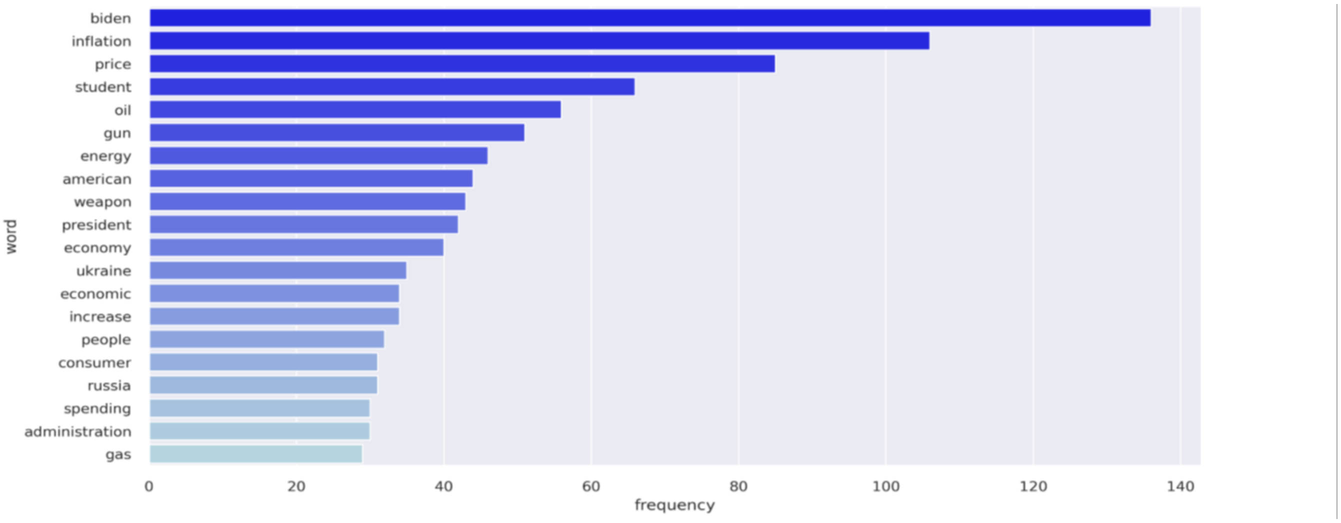


Figure 1. Corpus Word Frequency post data preprocessing

Methods

Various algorithms and methods have recently evolved in natural language processing (NLP). We explored the commonalities in topics and clustering using these algorithms. We describe the various methodologies behind our feature engineering, k-means clustering algorithm, Gensim Latent Dirichlet

Allocation (LDA), and BERTopic to discover the main themes in this section.

TF-IDF: Term Frequency - Inverse Document Frequency (TF-IDF) is a well-known algorithm to calculate word frequency for terms that are interesting within the corpus context. TF-IDF, which is robust enough to understand the text content, is useful for finding essential and related words or phrases in the text. A measure of the TF-IDF is calculated by multiplying the term frequency within the corpus and inverse document frequency resulting in a well-rounded importance score (Lane, Howard & Hapke, 2019, p.90). We used TfidfVectorizer from the scikit-learn framework to prepare the dataset for clustering and topic modeling. First, we compiled a term list of all unigrams, bigrams, and trigrams in our corpus documents and scored each term using TfidfVectorizer. This gives us an idea of which terms are more critical based on the highest scores and should drive the topic mapping for that document.

Doc2Vec: Le and Mikolov proposed Doc2Vec as a simple extension to Word2Vec to extend the learning of embeddings from words to word sequences (Le and Mikolov, 2014). The Doc2Vec model computes feature vectors for every document in a corpus. These vector representations have the advantage that they capture the semantics of the words, similar to word vectors. E.g., words such as "powerful" are more associated with "strong" than "Paris." The second advantage is that it works the same way as the n-gram model, which preserves a lot of information in the paragraph, including the word order. However, the bag of words model is very sparse and tends to generalize poorly. We trained a Gensim Doc2Vec model in our corpus and then used the infer vectors as an input for clustering.

Clustering: In the subsequent experiment, we used k-means clustering to group the documents in their clusters with TF-IDF and Doc2Vec vectors. Clustering served as an intermediate step in our analysis. K-means is an unsupervised learning algorithm that divides the dataset into a predefined number (k) of non-overlapping groups using the distances between the specified points. We must predefine the optimal number of clusters (k) in k-means clustering (W 1965). There is no one-size-fits-all method to determine k clusters. The elbow method and silhouette score are the most popular way to find the optimal number of clusters in k-means. We used the silhouette coefficient and elbow method to estimate the optimal number of clusters. A silhouette score of +1 indicates very dense and high separation, and scores near -1 suggest that the samples may have been assigned to the wrong cluster. We also used a silhouette visualizer to display the density and separation between clusters. The optimal number of clusters retrieved was nine, but many overlaps existed. So, we iteratively ran multiple experiments with different k values and evaluated the results. The result from TF-IDF vectors from experiment 1 with k=9 is shown in Appendix A figure 27.

Entity Co-Referencing/Equivalence Classes: This is a feature engineering step where we curated a list of similar terms in an equivalence class dictionary. We wanted to see how the equivalence classes would or would not improve our results. The idea is to use human intelligence to map the correlated terms with equivalence classes. This will allow us to have a higher probability of matching words into a cluster, reduce the dimensionality, and increase the tightness of our cluster formation. We went through the manual step of iteratively customizing our equivalence classes and curated a dictionary of eighteen equivalence classes having similar contexts. E.g., ‘clean energy’, ‘geothermal energy’, and ‘renewable fuels’ can map to ‘renewable energy’. Similarly, the ‘federal’, ‘congress’, and ‘senate’ can map to ‘government’.

Ontology: When addressing problems of any domain with artificial intelligence, understanding of that domain is critical to the application. Moreover, a well thought out and designed ontology will also help maintain a proper focus on the task. To acquire the former and hold the latter, we built an ontology on our corpus consisting of challenges faced by the Biden Administration, as shown in Figure 2. Flow Ontology. Ontologies help us organize concepts, like a knowledge graph, to show how they relate within a domain. The TF-IDF algorithm enabled us to understand the related terms and their connection with the documents and topics. We considered this an essential step to understanding the domain and processing the corpus with the aforementioned algorithms (Estival, Nowak, & Zschorn, 2004).

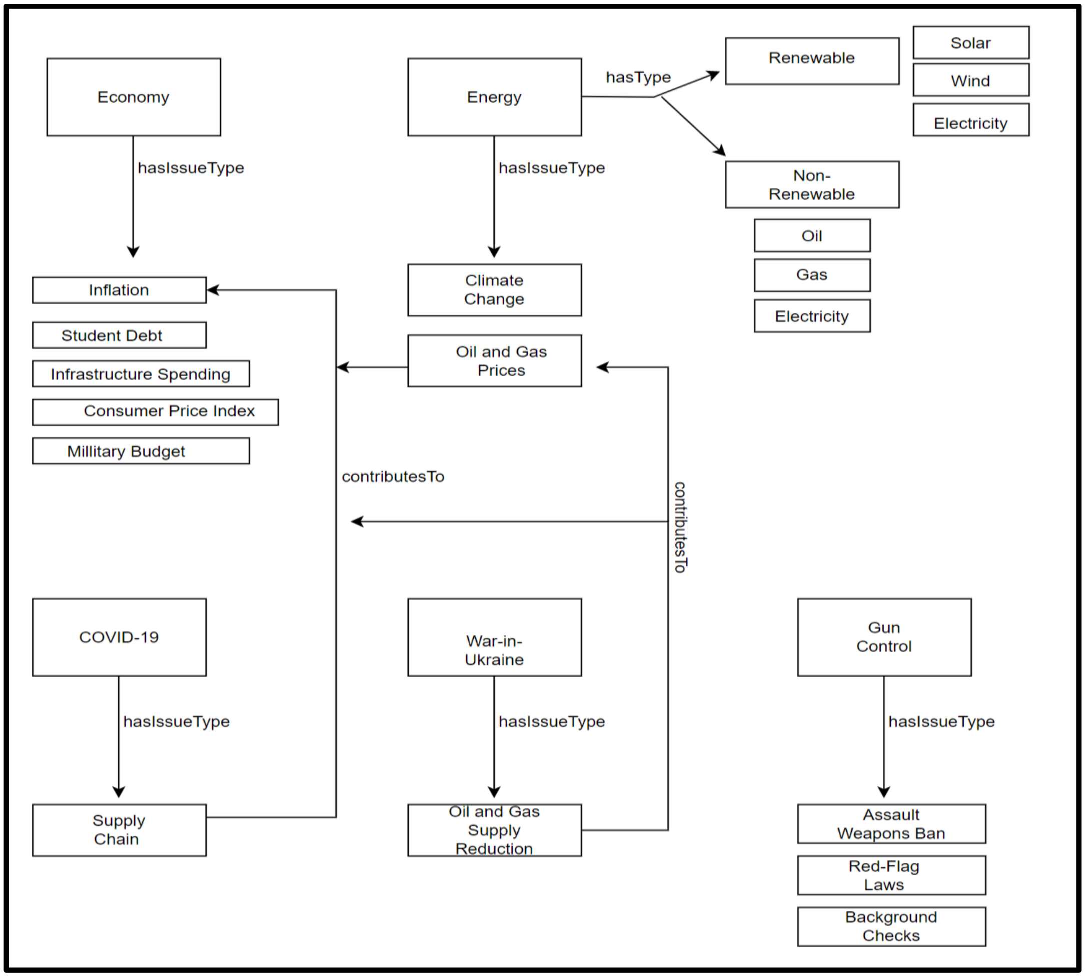


Figure. 2 Flow Ontology on Corpus of Biden Administration

LDA: We experimented with the Latent Dirichlet Allocation (LDA) algorithm for topic modeling and visualized the topics using the pyLDAvis library. Latent Dirichlet Allocation (LDA) is an unsupervised learning algorithm often used to identify topics that best represent a set of documents. LDA reduces the dimensions of text data by modeling each document as a distribution of topics and each topic as a distribution of words, mirroring Dirichlet distributions. Working backward from the prescribed number of topics (k), LDA randomly assigns each word to one of the k topics to get an initial representation of topics and terms. It then adjusts word placement by calculating the probability of each topic in each document and the likelihood of each word in each topic. LDA will adjust and move words to different topics until it reaches an optimal state of topic and document representations (Blei, Ng, & Jordan, 2003). The output of the model consists of both distributions of topics per document and distribution of terms per topic. In this experiment, we iteratively retrain the LDA model to find the optimal number of topics until the intertopic distance map in the pyLDAvis chart displays bubbles for overlapping topics which means we have overshot the number of topics. Furthermore, we looked at the word cloud for each topic to see the most common terms associated with each topic across the entire corpus and also examined the overall topic weightage distribution across the corpus.

BERTopic: Our corpus size was relatively small, but we wanted to experiment with the latest topic modeling using BERTopic architecture. BERTopic architecture overcomes the shortcomings of LDA by considering correlated topics (Raju et al. n.d.). Figure 3 below describes the topic model architecture for LDA and BERTopic. The main three components of BERTopic are:

1. Document Embedding - This component uses sentence transformer models like BERT and DistillBert to extract document embeddings.

2. Document Clustering - Many clustering algorithms handle high dimensionality poorly. So, UMAP is used to lower the dimensionality of the embeddings first. Then HDBSCAN is performed for document clustering.

3. Topic Creation and Representation - A class-based variant of TF-IDF (c-TFIDF) is used for topic creation to extract significant words for each cluster on a class-by-class basis. This would allow us to obtain relevance scores for individual words within the cluster and generate topic representation. This is extremely useful for inferring meaning from unsupervised clustering techniques.

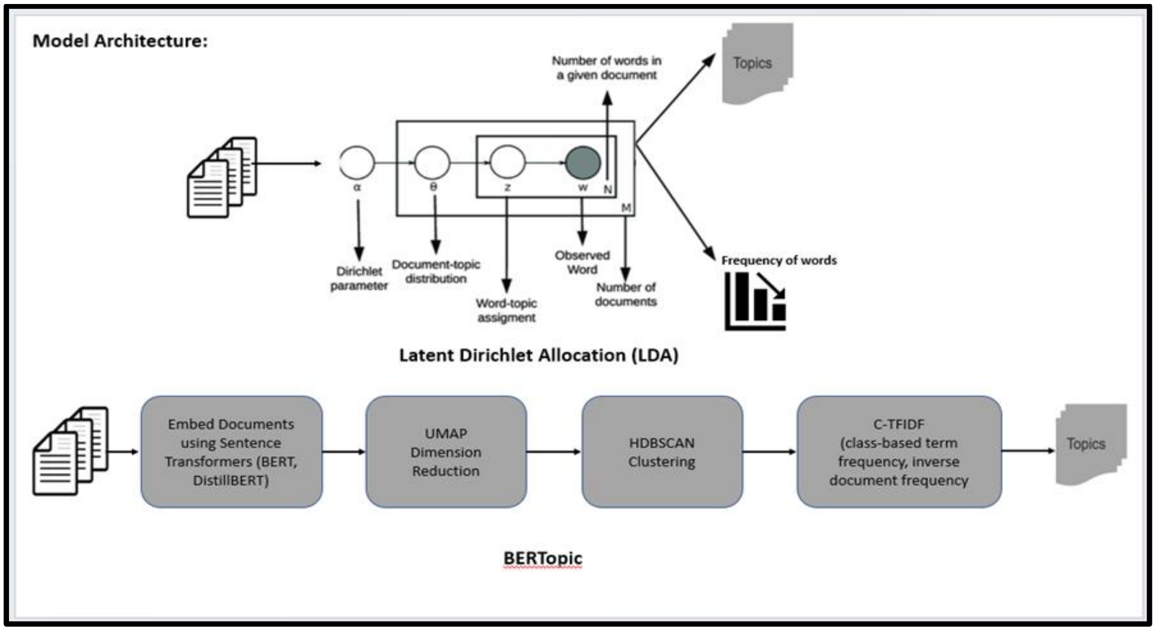

Model Architecture:
Number of words in a given document
Topics
α
θ
z
w
N
M
Dirichlet parameter
Document-topic distribution
Word-topic assigment
Observed Word
Number of documents
Frequency of words
Latent Dirichlet Allocation (LDA)
Embed Documents using Sentence Transformers (BERT, DistillBERT)
UMAP Dimension Reduction
HDBSCAN Clustering
C-TFIDF (class-based term frequency, inverse document frequency
Topics
BERTopic

Figure 3. Research Methodology: LDA (Top); BERTopic (Bottom)

Results

We ran five experiments, starting with a baseline model and successively adding feature engineering techniques to evaluate and compare the results. To assess the model's performance, we used the heat map distribution for each term calculated for each document, the word clouds showing the most common words associated with each topic, and the bar graph comparing corpus-wide topic weightage for each topic. This section will review the first pass results, underlying causes of poor algorithmic performance, and techniques employed to improve performance.

Experiment 1 (K-means Clustering using TF-IDF): Using the TF-IDF corpus matrix, we ran k-means clustering as an intermediate step to analyze the document groupings. We evaluated clusters of six all the way up to nine. The purpose of this experiment was to use the term frequency vectors generated using each document’s list of unigrams, bigrams, and trigrams as input to a clustering algorithm. K-means clustering used these term frequency vectors to split the data into varying cluster sizes. After the clustering algorithm split the documents into different clusters, our team used the cosine similarity of each document to check the validity of the placement for each document per cluster. We used the silhouette score of cluster sizes two through twelve to determine optimal k clusters. The results from these tests will guide us in considering the optimal k clusters in this experiment. To understand how well each document fits into its assigned cluster, we created a heatmap that shows the distribution of average TF-IDF values for each cluster of terms calculated for each document. Refer to Figures 21, 23, 25, and 27 in the Appendix A for the heatmaps for the final set of four clusters.

Experiment 2 (K-means Clustering using Doc2Vec): We ran Experiment 2 using k-means clustering with Doc2Vec. We used a similar combination of silhouette scores and observations based on results to determine a cluster size of six. This method was not particularly successful for topic modeling as the clusters that emerged appeared to have only some commonality with one another. Compared to Experiment 1 using TF-IDF, our review of the documents clustered showed less in common between documents using Doc2Vec than TF-IDF. Two of the examples this was most apparent in was that the gun control cluster in Experiment 1 was consistent with the gun control articles in the corpus, and the student loan cluster also contained all articles referencing those terms. However, In Experiment 2, the Doc2Vec algorithm grouped those articles in several clusters. The term clustering using TF-IDF gave clusters that better matched the content of the documents with each other.

Experiment 3 (Entity Co-Referencing using Equivalence Classes): In Experiment 3, we introduced the use of equivalence classes to see if these would improve our results from prior experiments. We created a dictionary containing multiple umbrella terms and their associated synonyms within the corpus. For example, ‘student-loan-forgiveness’ has the same meaning as ‘student loan forgiveness’, ‘studentloan forgiveness’, ‘debt cancellation’, and ‘student cancellation.’ Setting this equivalence class will enable the algorithm to identify and weigh the other terms like the umbrella term. First, we used k-means to see the kinds of clusters produced, starting with k=2 clusters and through k=12. Then, we did further analysis using a cluster size k=6 in combination with TF-IDF. K-means will return a related group of documents, and TF-IDF will return the most frequent terms within those documents. Table 1 below lists the top terms in each cluster of documents.

K-Means Clustering with Doc2Vec using Equivalence Classes - 6 Clusters

Cluster	Document Titles	Top Terms
1	MRD_Doc1_Civil-Society-Groups.docx MRD_Doc2_Not-Even-Halfway.docx	budget defense defense budget military military spending pentagon fiscal government spending us
2	JAS_D0c1_Student_Loan_Forgiveness.docx JAS_Doc2_Inflation-complicates_Biden.docx JJ_Doc2_Why_Is_Inflation.docx SS_Doc2_inflation_impacts_Student-Loan-Forgiveness.docx	student loan student loan forgiveness student loan forgiveness loan forgiveness inflation interest biden borrower
3	DT_Doc1_Biden_Administration_Renewable.docx DT_Doc2_Biden_More_Rewable.docx MCD_Doc3_Even-More-Biden-Oil.docx PSJ_doc1_President_Biden_announced.docx PSJ_doc2_The_worlds_wealthiest.docx	renewable energy renewable energy biden project oil biden administration administration world investment
4	BAC_Doc1_Biden-Rescue-Plan.docx BAC_Doc2_Biden-worsened -Inflation.docx CMP_Doc2_Voters_have_made.docx JJ_Doc1_Consumers-Are-Feeling.docx SS_Doc1_job_growth_double-edged_sword.docx Sieminski_Doc1_Inflation.docx	inflation price consumer survey american rescue plan rescue point american rescue plan american rescue

5	CMP_Doc1_Consumer_Price_Index.docx KN_Doc1_Biden-urges-G7.docx MCD_Doc1_Biden-Oil-Reserve .docx MCD_Doc2_More-Biden-Oil.docx SS_Doc1_Mission_Not_Yet.docx SS_Doc2_A_New_Task.docx Sieminski_Doc2_Recession.docx	biden oil russian russia reserve barrel sanction ukraine war
6	JS_DOC_1_lives_will_be_saved.docx JS_DOC_2_Biden_signs_gun_control.docx TSS_Doc1_Biden-Gun-Control.docx TSS_Doc2_Biden-Remarks-Gun-Violence.docx	gun weapon assault shooting violence assault weapon biden uvalde texas background check

Table 1. K-Means Clustering with Doc2Vec using Equivalence Classes - 6 Clusters

Experiment 4 (LDA using Bag of Words/BOW and TF-IDF)

Experiment 4 utilizes Latent Dirichlet Allocation (LDA) for Topic Modeling using Gensim implementation. Three kinds of experiments are tested here with different number of topics: (1) LDA based on the bag of words (BOW) method with the number of topics five and four, (2) LDA based on the bag of words (BOW) method with equivalence class (EC) applied on the number of topics as five, four and three, and (3) LDA based on the TF-IDF method with the number of topics as four and three. Bag of words (BOW) produced distinct topics with the number of topics four and five. Inflation was among the dominant topics in both bag of words (BOW) experiments, appearing in three topics each, as noticed in Figure 4. In addition to topic visualization in Figure 5 and 6, we visualized the extracted top ten keywords in Figure 7 and 8 as a word cloud.

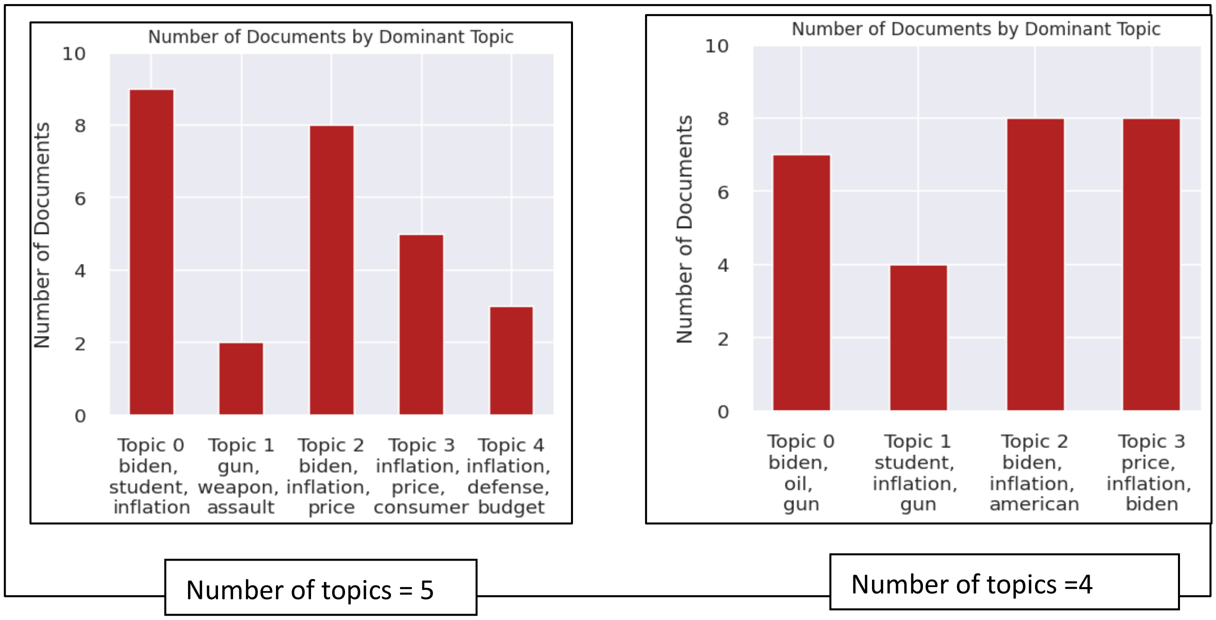
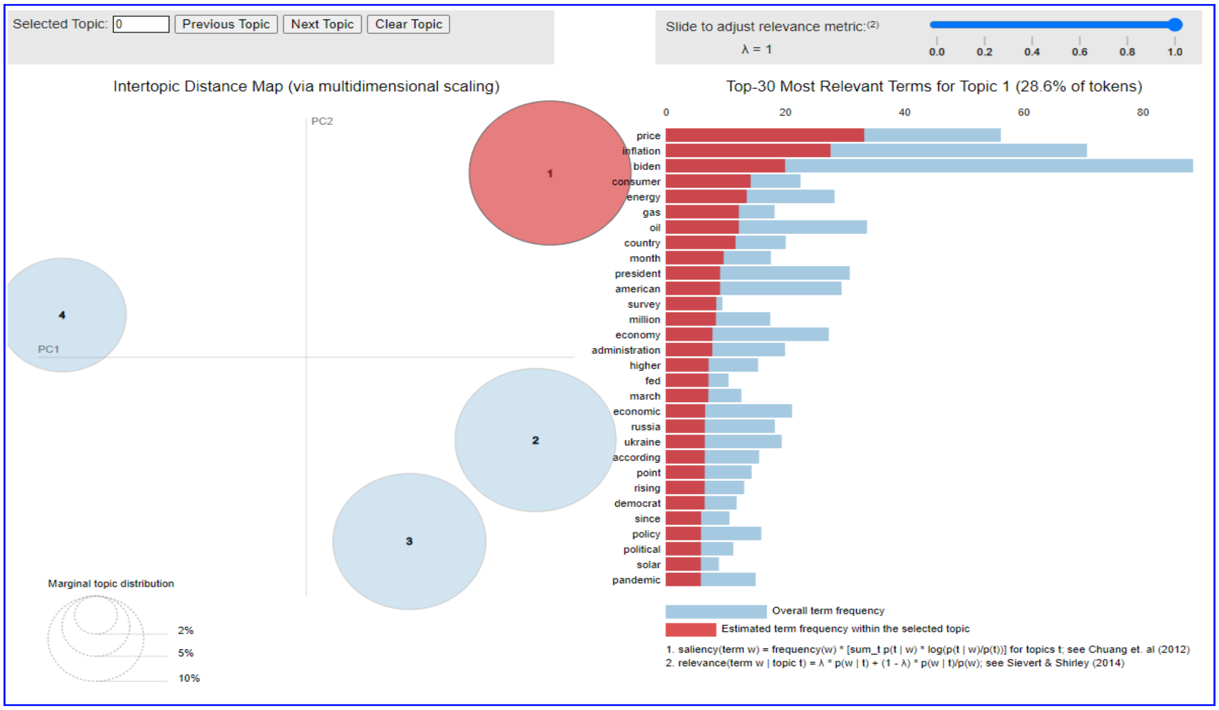


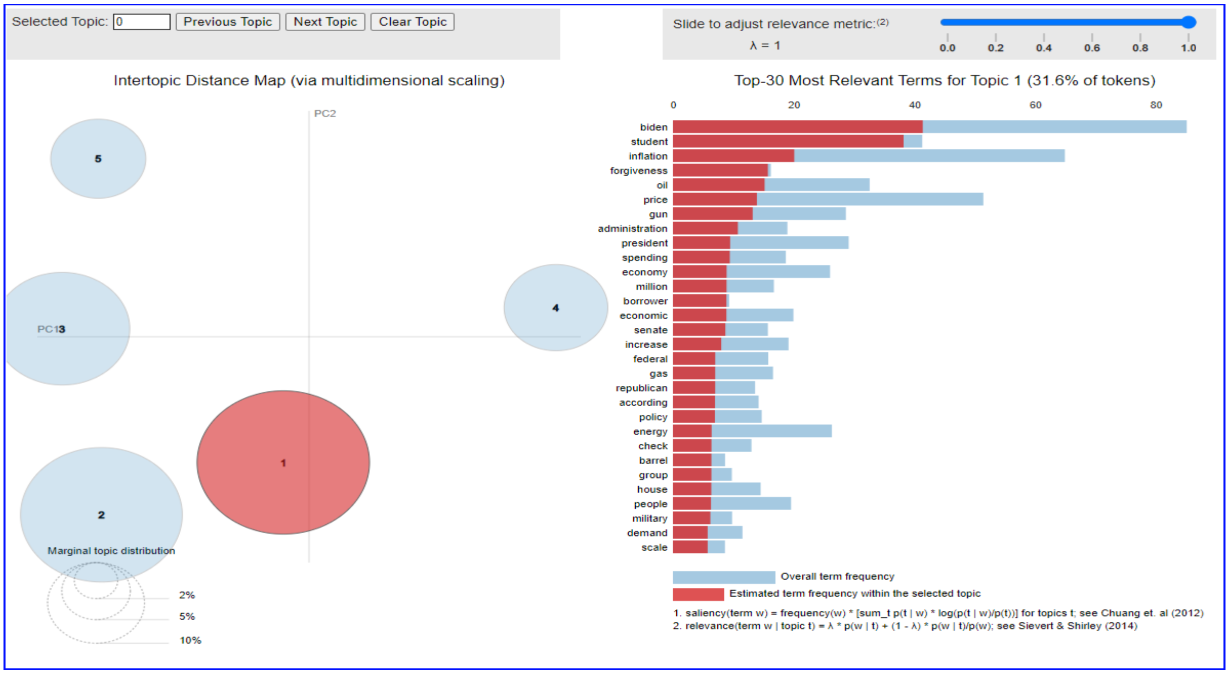
Figure 4. Dominant Topic (inflation) in LDA (BOW)


Selected Topic: 0

Slide to adjust relevance metric:(2)
λ = 1

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Figure 5. LDA (BOW) topic visualization of extracted topics (number of topics = 4)


Selected Topic: 0

Slide to adjust relevance metric:(2) λ = 1

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Figure 6. LDA (BOW) topic visualization of extracted topics (number of topics = 5)

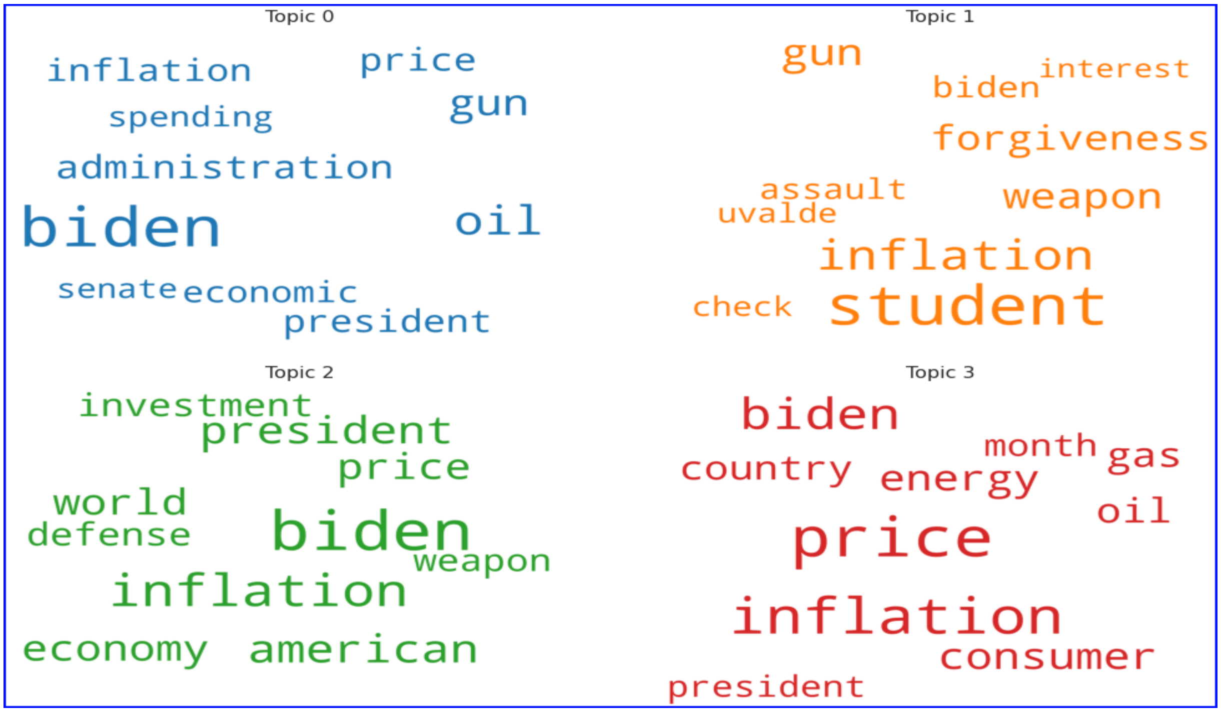


Figure 7. Word cloud LDA BOW Model (Number of topics = 4).

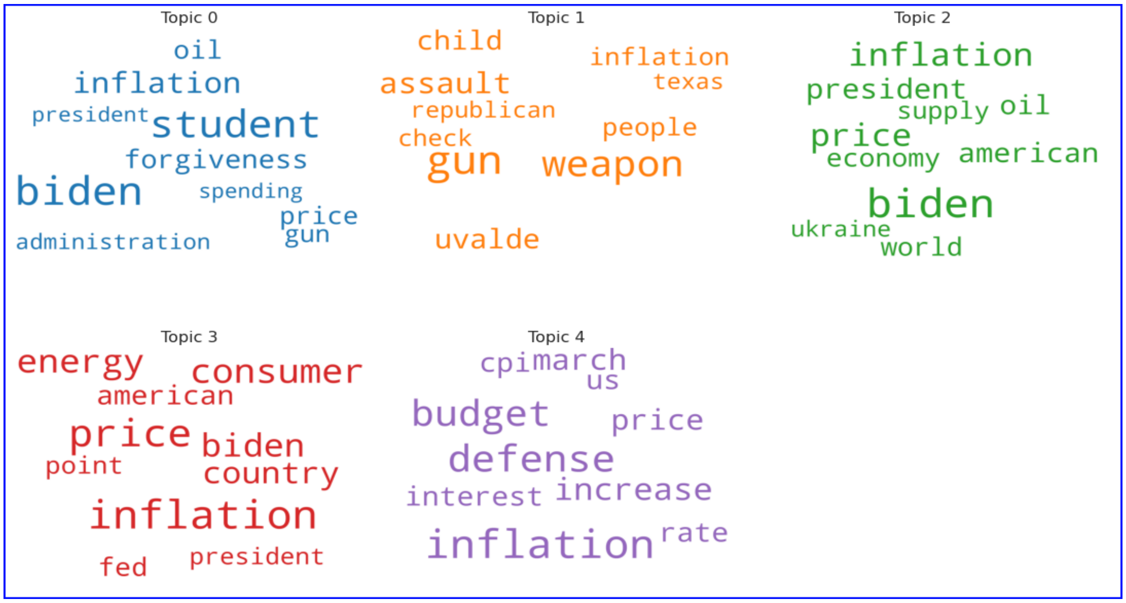
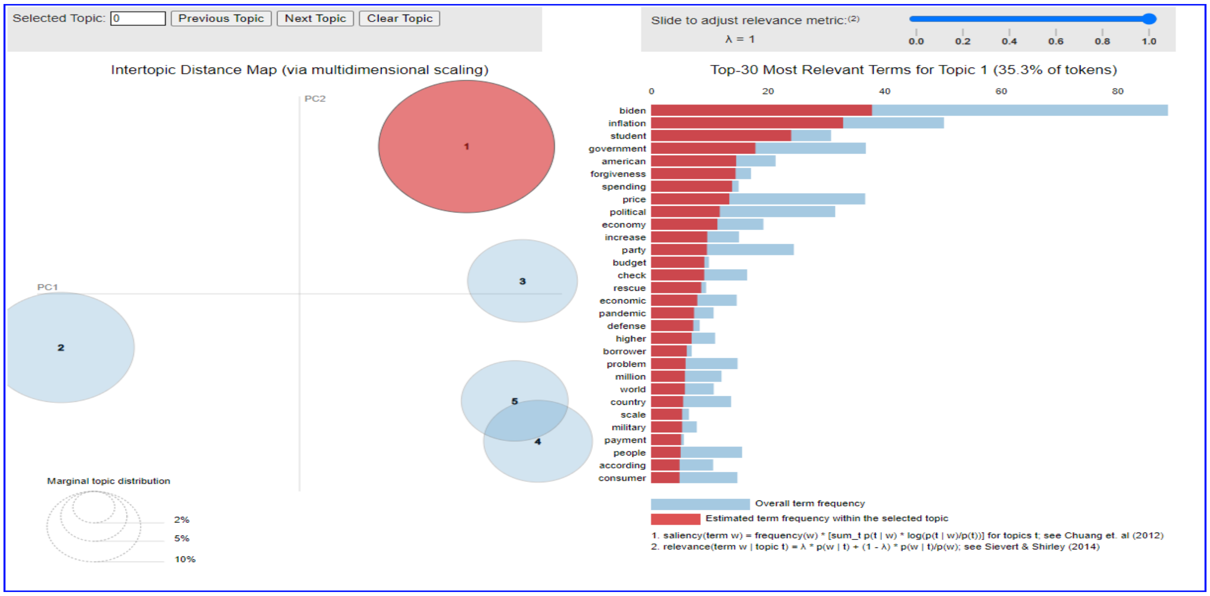


Figure 8. Word cloud LDA BOW Model (Number of topics = 5).

Furthermore, we applied the equivalence classes curated in experiment 3 to the bag of words model with the different number of topics to see if the results would improve. Cluster three shows distinct clusters with dominant terms similar to those produced by the ontology with few noises or outliers. Figures 9, 10, and 11 visualize the results.

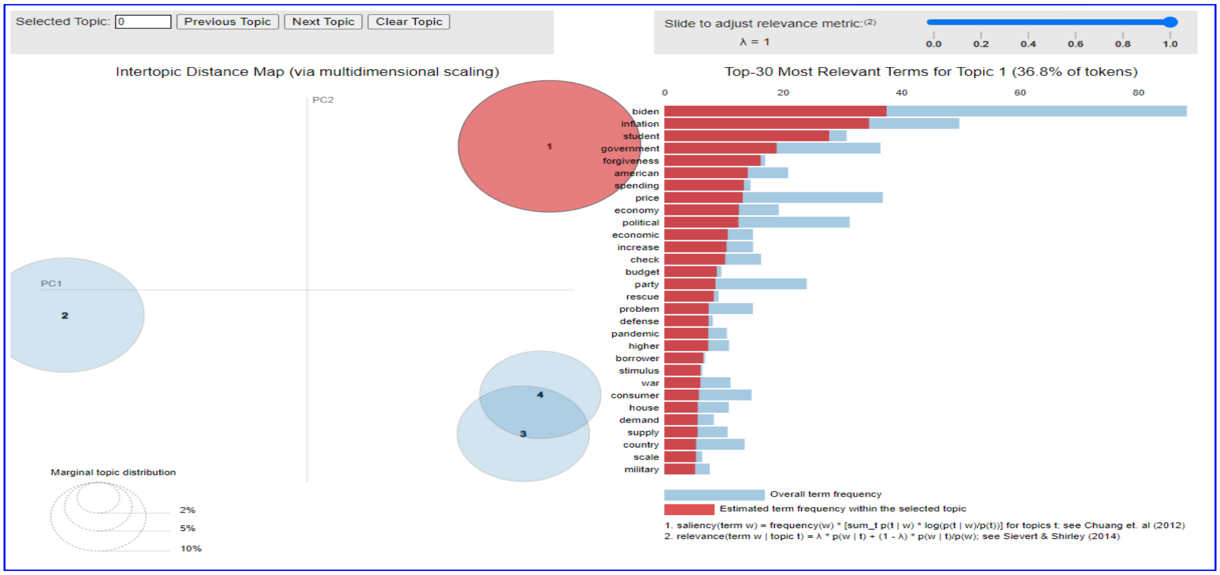

Selected Topic: 0

Slide to adjust relevance metric:(2)

λ = 1

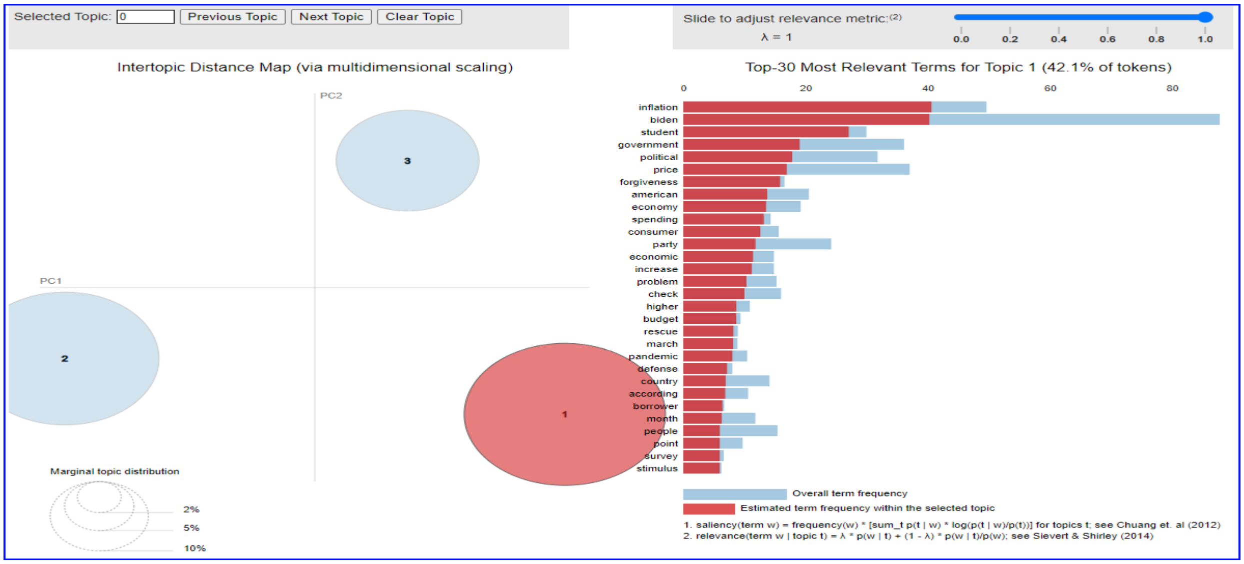
1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Figure 9. LDA BOW Model topic visualization with Equivalence class (Number of topics = 5).



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

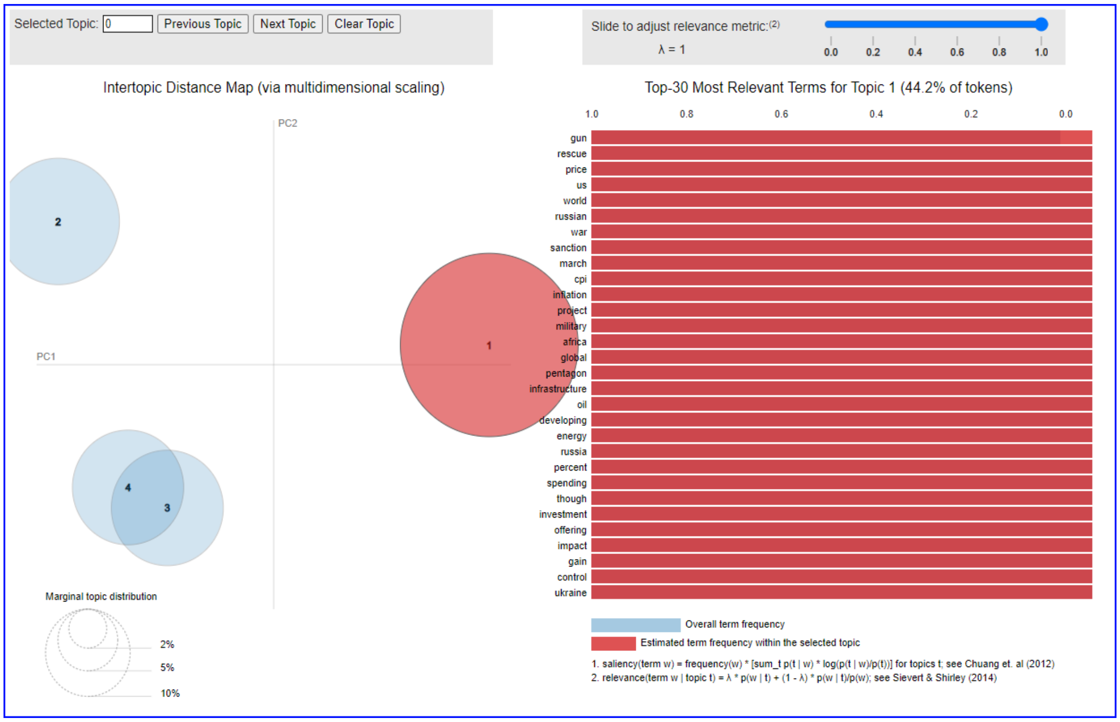
Figure 10. LDA BOW Model topic visualization with Equivalence class (Number of topics = 4).



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Figure 11. LDA BOW Model topic visualization with Equivalence class (Number of topics = 3).

TF-IDF vectorizer applied on the corpus showed a big difference in topic assignment compared to bag of words (BOW). The inflation-related topic was less dominant using TF-IDF for both number of topics as four and five. With the number of topics set to four, gun violence found its way into two different topics. Figures 12 and 13 show the top three dominant terms in the topics for both number of topics as four and five. Figures 14 and 15 show the word cloud representation of the extracted topics and the keywords in each topic.



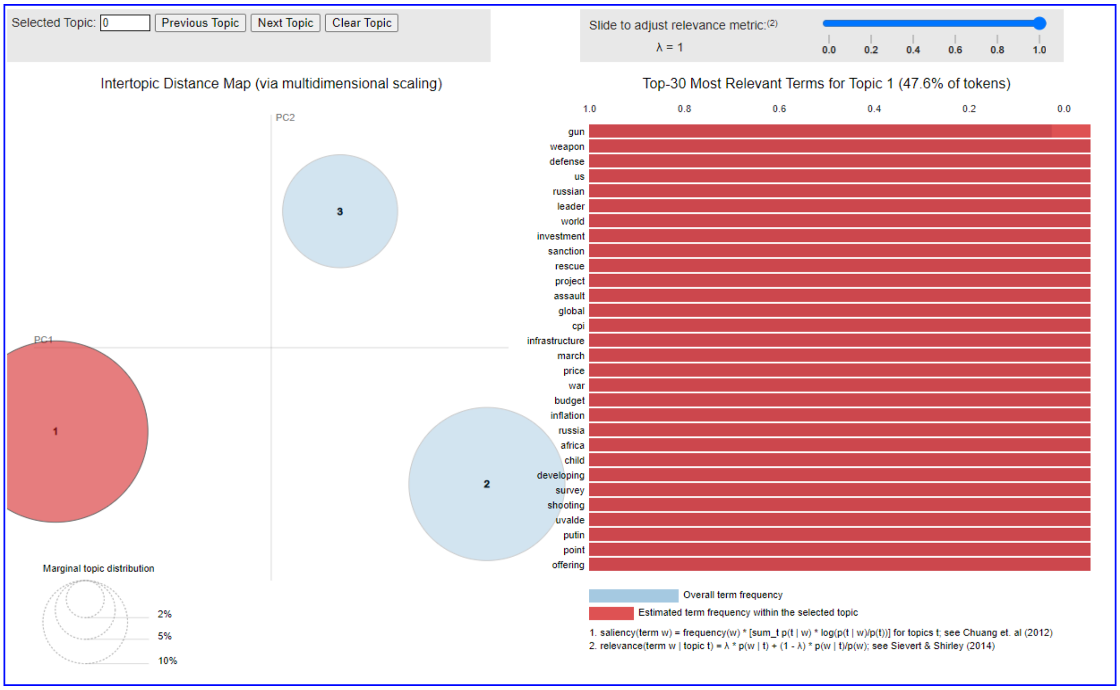
Selected Topic: 0 Previous Topic Next Topic Clear Topic

Slide to adjust relevance metric:(2)

λ = 1

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Figure 12. LDA (TF-IDF) topic visualization of extracted topics (Number of topics = 4).



Selected Topic: 0 Previous Topic Next Topic Clear Topic

Slide to adjust relevance metric:(2)

λ = 1

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Figure 13. LDA (TF-IDF) topic visualization of extracted topics (Number of topics = 3).

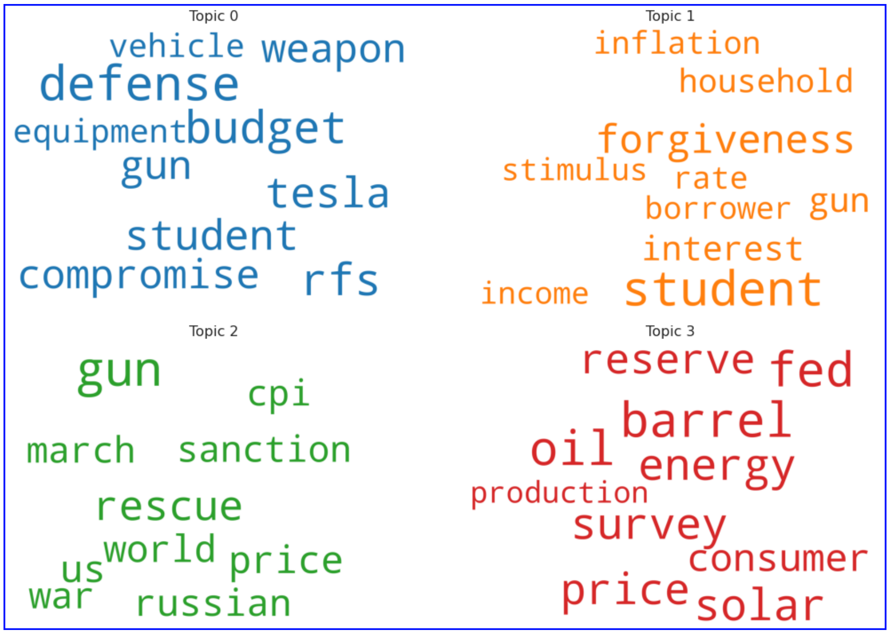


Figure 13. Word cloud LDA TF-IDF Model (Number of topics = 4).

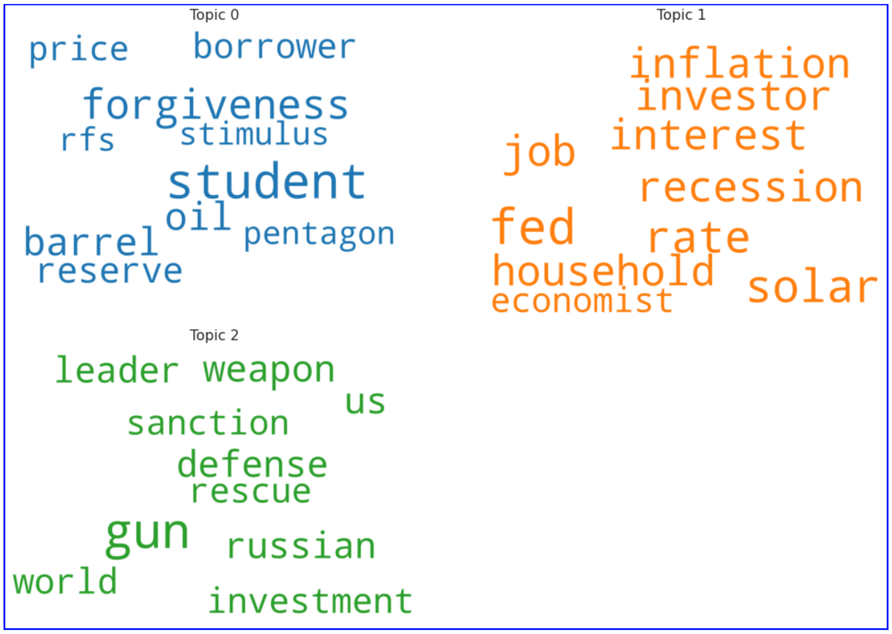


Figure 14. Word cloud LDA TF-IDF Model (Number of topics = 3).

Evaluation Metrics

Two evaluation metrics are typically used in the domain of topic modeling: Perplexity and coherence. Coherence measures the degree of semantic similarity between high scoring words in the topic. Among the several coherence metrics in the Gensim library, "c_v" has the highest average correlation to human assessments in a study by Röder. A proposed range of 0.50-0.60 is deemed sufficient. (Röder et al., 2015). On the other hand, Perplexity assesses the model's fitness on the number of topics. Generally, the lower perplexity scores provide a better prediction of model performance. However, from a qualitative viewpoint, low perplexity measures do not always indicate profound insights. Contrary to popular belief, perplexity does not directly correspond with the exploratory goals of topic modeling (Chang et al., 2009). In reality, the semantics of an LDA model with many subjects might appear nonsensical (Chang et al., 2009). We, therefore, did not give much attention to the perplexity scores. Table 2 below shows the various scores for Experiment 4.

LDA Evaluation Metrics			
Experiment 4 - LDA Topic Modeling	**Number of topics**	**Perplexity**	**Coherence**
Bag of words	5	-7.36	0.42
	4	-7.35	0.5
TF-IDF	4	-10.05	0.44
	3	-9.46	0.46
Bag of words using EC	5	-7.18	0.42
	4	-7.15	0.41
	3	-7.16	0.35

Table 2. LDA Evaluation Metric

Experiment 5 (Bert Embedding): Both LDA and BERTopic modeling extracted the relevant terms and categorized them into the same topic groups. A significant difference between the two methods is that LDA sometimes returns clustered terms that seem unrelated and thus require further interpretation. In Experiment 5, we employed BERT embedding using DistilBERT and then used cosine-based k-means clustering to cluster the documents. K-means clustering with BERT embedding vectors was an intermediate step to help us better understand our document groupings and compare them with TF-IDF and Doc2Vec vectorization. With an optimal cluster k=5 obtained using the elbow method, the inflation related documents end up in three different clusters. We also see a cluster overlap between student loan forgiveness and renewable energy documents highlighted in Table 3 below.

K-means Clustering with BERT Embedding Vectors	
Cluster	**Documents**
0	KN_Doc1_Biden-urges-G7.docx
	PSJ_doc1_President_Biden_announced.docx
	PSJ_doc2_The_worlds_wealthiest.docx
	SS_Doc1_Mission_Not_Yet.docx
1	DT_Doc1_Biden_Administration_Renewable.docx
	DT_Doc2_Biden_More_Rewable.docx
	JAS_D0c1_Student_Loan_Forgiveness.docx
	JAS_Doc2_Inflation-complicates_Biden.docx
	JJ_Doc2_Why_Is_Inflation.docx
2	MRD_Doc1_Civil-Society-Groups.docx
	MRD_Doc2_Not-Even-Halfway.docx
	SS_Doc2_inflation_impacts_Student-Loan-Forgiveness.docx
3	BAC_Doc1_Biden-Rescue-Plan.docx
	BAC_Doc2_Biden-worsened -Inflation.docx
	CMP_Doc1_Consumer_Price_Index.docx
	CMP_Doc2_Voters_have_made.docx
	JJ_Doc1_Consumers-Are-Feeling.docx
	MCD_Doc1_Biden-Oil-Reserve .docx
	MCD_Doc2_More-Biden-Oil.docx
	MCD_Doc3_Even-More-Biden-Oil.docx
	SS_Doc1_job_growth_double-edged_sword.docx
	SS_Doc2_A_New_Task.docx
	Sieminski_Doc1_Inflation.docx
	Sieminski_Doc2_Recession.docx
4	JS_DOC_1_lives_will_be_saved.docx
	JS_DOC_2_Biden_signs_gun_control.docx
	TSS_Doc1_Biden-Gun-Control.docx
	TSS_Doc2_Biden-Remarks-Gun-Violence.docx

Table 3. K-Means Clustering with DistilBERT Embedding in Experiment 5

In subsequent models, we used BERTopic for topic modeling. We implemented BERTopic modeling from scratch using BERT embeddings to create our own topic model 6. In model 7, we leveraged the BERTopic model for dynamic topic modeling. The results in model 6 yield only two main topics, economy and gun violence, as shown in the word cloud of the top ten words in Figure 15.

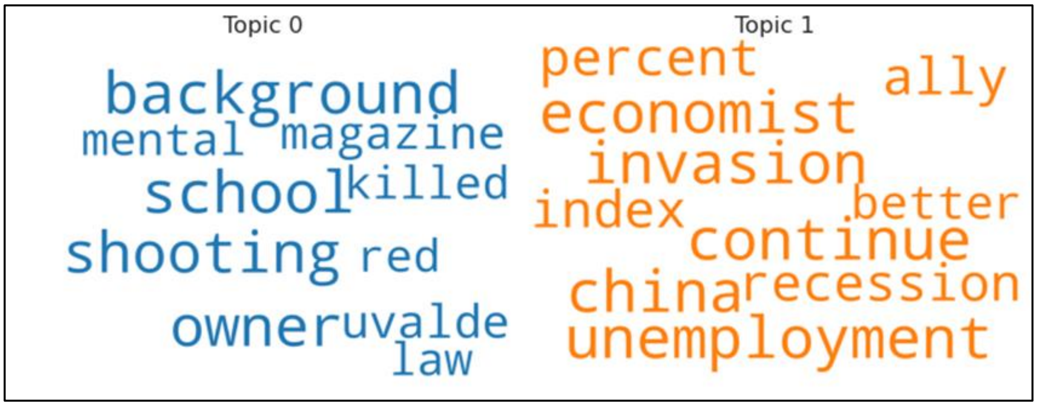


Figure 15. Bert Topic Modeling in Experiment 6

BERTopic modeling flagged the defense budget (MRD_Doc2_Not-Even-Halfway.docx) related document as an outlier. However, the rest of the news articles resulted in three distinct topics: Inflation, oil/energy, and gun violence, as shown in topic word scores in Figure 16.

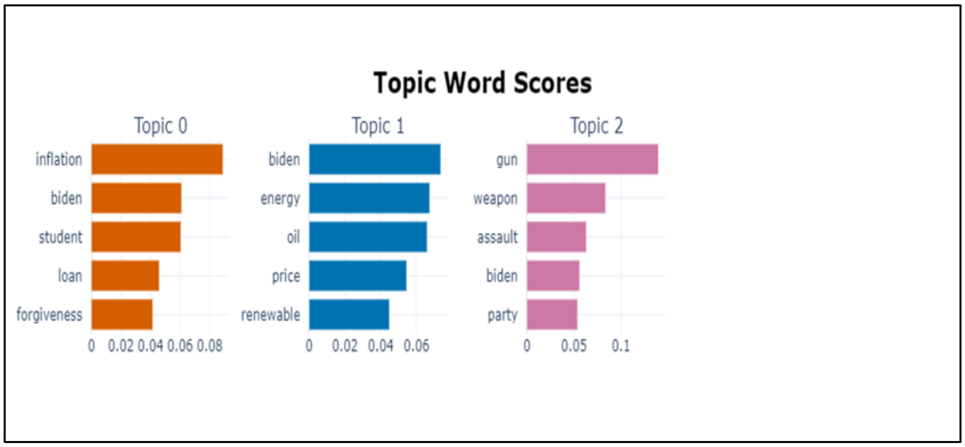


Figure 16. BERTopic Word Scores in Experiment 7

Analysis and Interpretation

Experiment 1 (K-means Clustering using TF-IDF): The results from the clustering analysis details the cosine similarity between the document terms for each cluster. Figure 17 shows the similarity for six clusters; there aren't many distinguishable groupings to make out. Cluster 6 (red) was the only cluster with all the documents aligned within the same quadrant of the cosine similarity graph.

As far as increasing the cluster size to 7, 8, and 9, a similar lack of pattern occurred where there weren't any clear groupings of documents for each cluster. However, there are still observations showing which document groupings split into smaller groups after increasing cluster size. For instance, when increasing the cluster size from 6 to 7, a foreign policy cluster emerged and allowed for the distinction between problems for Biden based on foreign and domestic.

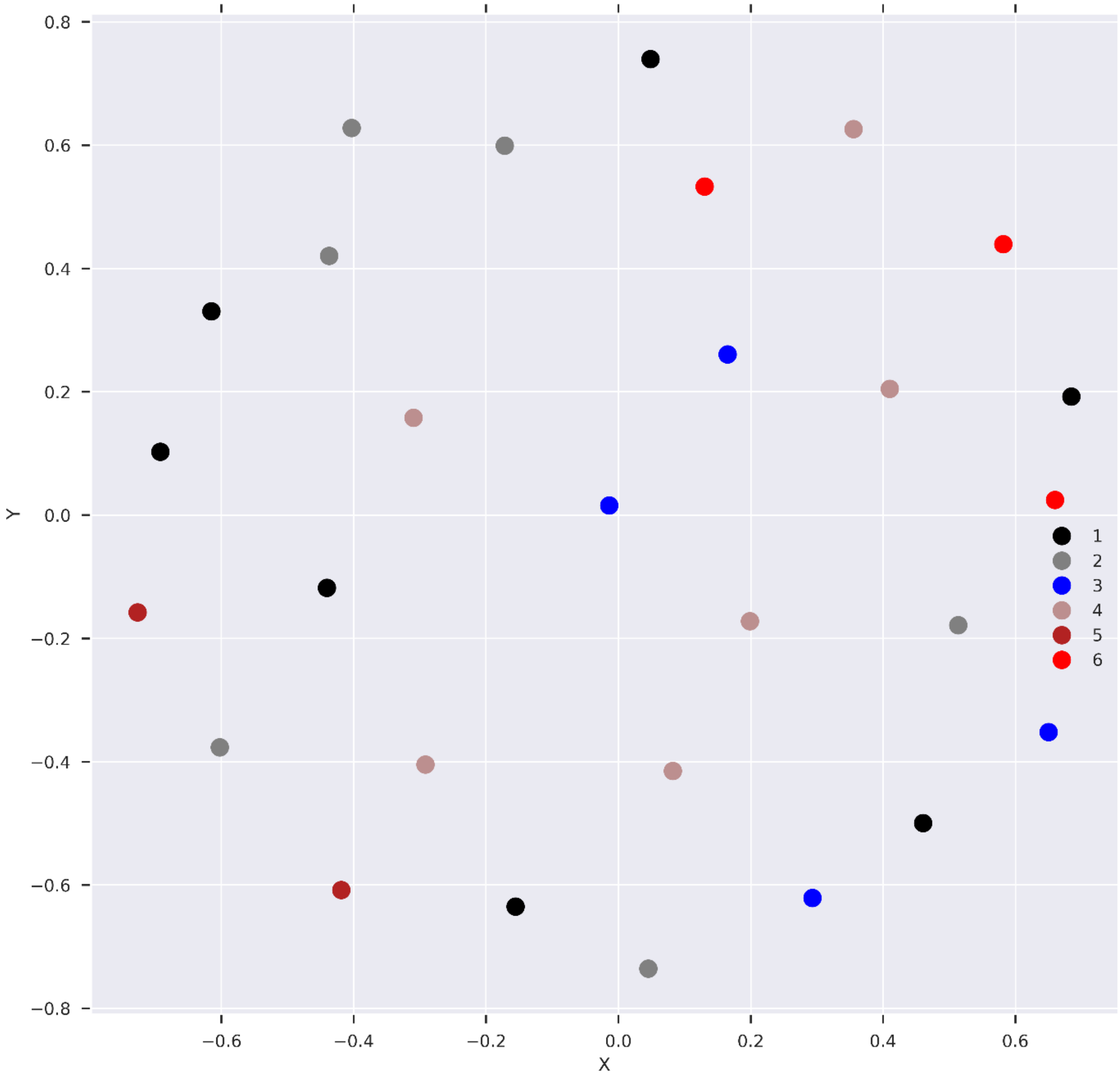


Figure 17. K-means Clustering with TF-IDF (K=6)

Experiment 2 (K Means Clustering Doc2Vec): In Experiment 2, we tested multiple cluster sizes, measured silhouette scores, and compared them against an elbow graph. The elbow didn't show a distinct point for the number of clusters, but the best silhouette score was six or ten clusters. Upon review of the data, the six cluster model appeared to be the best. Table 4 arranges the six clusters that emerged from the corpus:

K-means Clustering with Doc2Vec Vectors	
Cluster	**Documents**
0	DT_Doc2_Biden_More_Rewable.docx
	JAS_D0c1_Student_Loan_Forgiveness.docx
	JS_DOC_2_Biden_signs_gun_control.docx
	MCD_Doc1_Biden-Oil-Reserve.docx
	PSJ_doc1_President_Biden_announced.docx
	PSJ_doc2_The_worlds_wealthiest.docx
	Sieminski_Doc1_Inflation.docx
	TSS_Doc2_Biden-Remarks-Gun-Violence.docx
1	BAC_Doc1_Biden-Rescue-Plan.docx
	CMP_Doc2_Voters_have_made.docx
	SS_Doc1_Mission_Not_Yet.docx
	SS_Doc2_A_New_Task.docx
2	DT_Doc1_Biden_Administration_Renewable.docx
	KN_Doc1_Biden-urges-G7.docx
	MRD_Doc1_Civil-Society-Groups.docx
	SS_Doc1_job_growth_double-edged_sword.docx
	Sieminski_Doc2_Recession.docx
	TSS_Doc1_Biden-Gun-Control.docx
3	JAS_Doc2_Inflation-complicates_Biden.docx
	JJ_Doc1_Consumers-Are-Feeling.docx
	JJ_Doc2_Why_Is_Inflation.docx
	MCD_Doc2_More-Biden-Oil.docx
4	BAC_Doc2_Biden-worsened-Inflation.docx
	CMP_Doc1_Consumer_Price_Index.docx
	MCD_Doc3_Even-More-Biden-Oil.docx
	SS_Doc2_inflation_impacts_Student-Loan-Forgiveness.docx
5	JS_DOC_1_lives_will_be_saved.docx
	MRD_Doc2_Not-Even-Halfway.docx

Table 4. K-Means Clustering with Doc2Vec.

Based on the clustering results, some clusters don’t appear to have an immediately apparent commonality. In cluster 0, for example, there are articles about Ukraine and Russia, inflation, and oil supply. In cluster 1, we observe gun control and inflation articles clustered together. Cluster 5 has a broad mix of gun control, economics, and renewable energy. Some of the other clusters have clear groupings, such as in cluster 3, which has to do with renewable energy and oil, and cluster 4, which has to do with student loans. Cluster 2 also falls in this category, with articles mainly related to inflation and economic concerns.

Experiment 3 (Entity Co-Referencing using Equivalence Classes): Through observation of the previous experiments, Experiment 2 improved by introducing equivalence classes. In Table 5 below, the number of documents and their topics changed significantly. For example, in Cluster 0, Experiment 2, there is a mix of topics such as oil, renewable energy, global infrastructure, inflation, and gun control. However, the algorithm was insufficient: Each of these documents can still break down into different clusters such as energy, economy, and gun control. Interestingly, when we applied equivalence classes in Experiment 3, Cluster 0 completely differed from previous experiments. This time only two documents emerged in the cluster, whereas previously, there were six. Both documents are intra-related, primarily covering military spending in relation to the war in Ukraine. Based on the ontology, it is easier to validate if Cluster 0 matches the real world understanding of the article content. In this case, adding equivalence classes enabled the algorithm to match the ontology more accurately. In Figure 18, depicting the cluster level ontology, both documents returned from the algorithm in Cluster 0 of Experiment 3 are mapped in a similar way. By this, we can confirm that these two documents are intra-related. However, the algorithm did not account for overlapping clusters. Human understanding of these articles reveals that their content is about both the economy and the war in Ukraine.

Another improvement the equivalence classes provided is clustering all of the documents about gun control in an isolated cluster. Referencing the ontology, this is again confirmed to be an accurate result as each of those documents focuses solely on that topic; there is no overlap with any other documents in our corpus.

K-Means Clustering with Doc2Vec - 6 Clusters	
With Equivalence Class	**No Equivalence Class**
Experiment 3	**Experiment 2**
MRD_Doc1_Civil-Society-Groups.docx **MRD_Doc2_Not-Even-Halfway.docx**	**DT_Doc2_Biden_More_Rewable.docx** **MCD_Doc2_More-Biden-Oil.docx** **PSJ_doc2_The_worlds_wealthiest.docx** **SS_Doc1_job_growth_double-edged_sword.docx** **Sieminski_Doc1_Inflation.docx** **TSS_Doc2_Biden-Remarks-Gun-Violence.docx**
JAS_D0c1_Student_Loan_Forgiveness.docx **JAS_Doc2_Inflation-complicates_Biden.docx** **JJ_Doc2_Why_Is_Inflation.docx** **SS_Doc2_inflation_impacts_Student-Loan-Forgiveness.docx**	**DT_Doc1_Biden_Administration_Renewable.docx** **JAS_D0c1_Student_Loan_Forgiveness.docx** **JAS_Doc2_Inflation-complicates_Biden.docx** **JJ_Doc1_Consumers-Are-Feeling.docx** **JS_DOC_2_Biden_signs_gun_control.docx**
DT_Doc1_Biden_Administration_Renewable.docx **DT_Doc2_Biden_More_Rewable.docx** **MCD_Doc3_Even-More-Biden-Oil.docx** **PSJ_doc1_President_Biden_announced.docx** **PSJ_doc2_The_worlds_wealthiest.docx**	**JJ_Doc2_Why_Is_Inflation.docx** **JS_DOC_1_lives_will_be_saved.docx** **KN_Doc1_Biden-urges-G7.docx** **Sieminski_Doc2_Recession.docx**

BAC_Doc1_Biden-Rescue-Plan.docx BAC_Doc2_Biden-worsened -Inflation.docx CMP_Doc2_Voters_have_made.docx JJ_Doc1_Consumers-Are-Feeling.docx SS_Doc1_job_growth_double-edged_sword.docx Sieminski_Doc1_Inflation.docx	BAC_Doc1_Biden-Rescue-Plan.docx BAC_Doc2_Biden-worsened -Inflation.docx MCD_Doc3_Even-More-Biden-Oil.docx MRD_Doc1_Civil-Society-Groups.docx MRD_Doc2_Not-Even-Halfway.docx SS_Doc1_Mission_Not_Yet.docx SS_Doc2_A_New_Task.docx
CMP_Doc1_Consumer_Price_Index.docx KN_Doc1_Biden-urges-G7.docx MCD_Doc1_Biden-Oil-Reserve .docx MCD_Doc2_More-Biden-Oil.docx SS_Doc1_Mission_Not_Yet.docx SS_Doc2_A_New_Task.docx Sieminski_Doc2_Recession.docx	CMP_Doc1_Consumer_Price_Index.docx PSJ_doc1_President_Biden_announced.docx
JS_DOC_1_lives_will_be_saved.docx JS_DOC_2_Biden_signs_gun_control.docx TSS_Doc1_Biden-Gun-Control.docx TSS_Doc2_Biden-Remarks-Gun-Violence.docx	CMP_Doc2_Voters_have_made.docx MCD_Doc1_Biden-Oil-Reserve .docx SS_Doc2_A_New_Task.docx TSS_Doc1_Biden-Gun-Control.docx

Table 5. K-means Clustering with Doc2Vec.

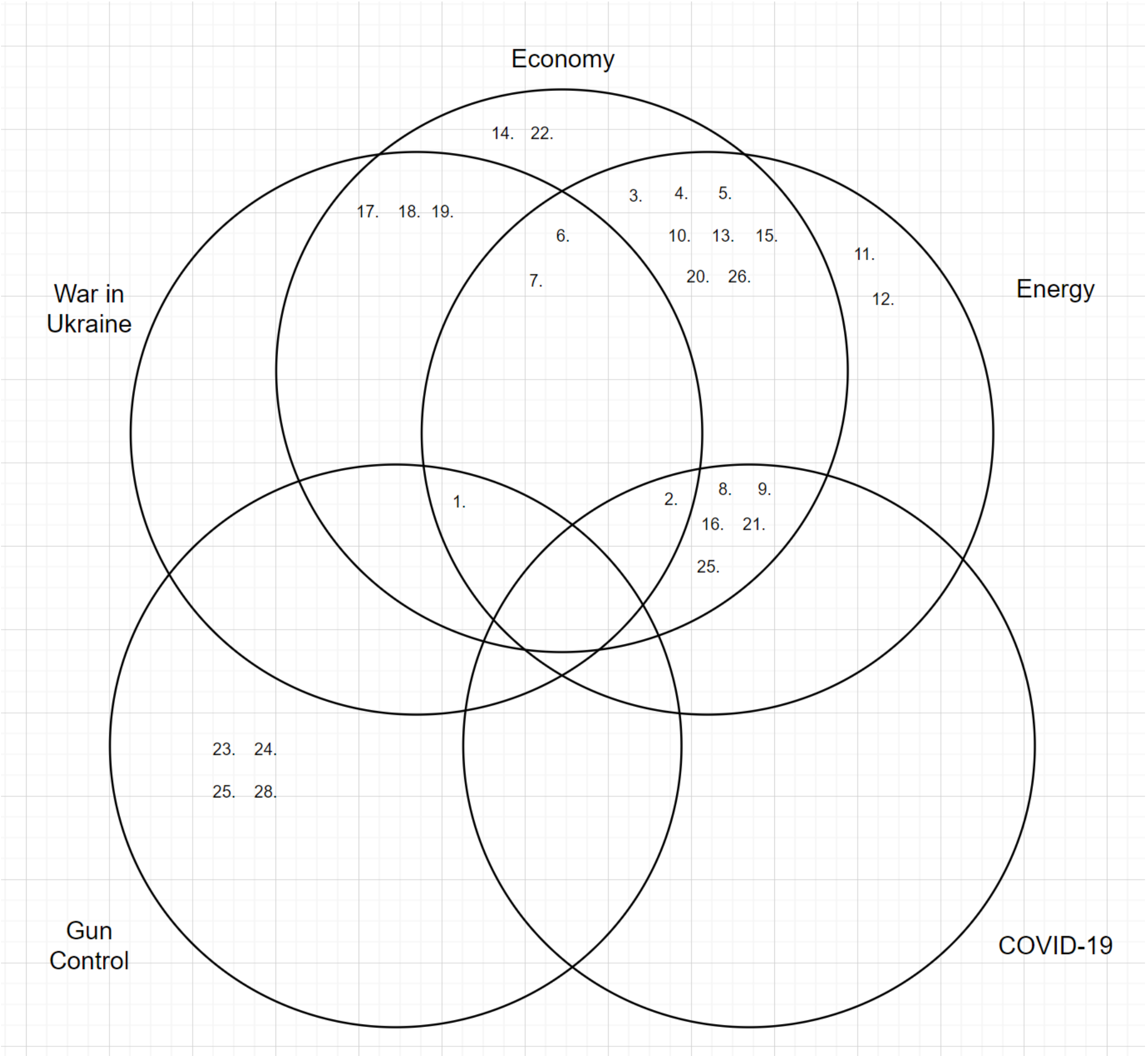


Figure 18. Cluster Level Ontology

Ontology Reference Table				
1. SS_Doc2_A_New_Task - Scott S	**7.PSJ_doc1_President_Biden_announced - Parker J**	**13. JJ_Doc2_Why_Is_Inflation-Juannan J**	**19. JAS_D0c1_Student_Loan_Forgiveness - John S**	**25. Sieminski_Doc2_Recession - Dominic S**
2. SS_Doc1_Mission_Not_Yet - Scott S	**8.PSJ_doc2_The_worlds_wealthiest - Parker J**	**14. JJ_Doc1_Consumers-Are-Feeling - Juannan J**	**20. JAS_Doc2_Inflation-complicates_Biden - John S**	**26.Sieminski_Doc1_Inflation - Dominic S**

3. MCD_Doc2_More-Biden-Oil - Marc D	**9. BAC_Doc1_Biden-Rescue-Plan - Brent C**	**15. CMP_Doc1_Consumer_Price_Index- Chris P**	**21. SS_Doc1_job_growth_double-edged_sword-Swati S**	**27. TSS_Doc2_Biden-Remarks-Gun-Violence-Taylor S**
4. MCD_Doc1_Biden-Oil-Reserve - Marc D	**10. BAC_Doc2_Biden-worsened -Inflation-Brent C**	**16.CMP_Doc2_Voters_have_made - Chris P**	**22. SS_Doc2_inflation_impacts_Student-Loan-Forgiveness-Swati S**	**28. TSS_Doc1_Biden-Gun-Control - Taylor S**
5. MCD_Doc3_Even-More-Biden-Oil - Marc D	**11. DT_Doc1_Biden_Administration_Renewable - David T**	**17. MRD_Doc2_Not-Even-Halfway - Michael D**	**23. JS_DOC_1_lives_will_be_saved - Josh S**	
6. G7 to Stay Together - Katya N	**12. DT_Doc2_Biden_More_Rewable - David T**	**18. MRD_Doc1_Civil-Society-Groups - Michael D**	**24. JS_DOC_2_Biden_signs_gun_control - Josh S**	

Table 6. Ontology Key Reference

Experiment 4 (LDA using Bag of words/BOW and TF-IDF): Experiment 4 tested multiple cluster sizes for both TF-IDF and bag of words (BOW) and measured each model's performance using coherence and perplexity scores. The proposed LDA approach created coherent and semantically meaningful topics/clusters from the corpus. The created topics provided a human-readable deconstruction of the documents. Even though the scores were quite misleading when we applied the equivalence class to the corpus, the results generated were similar to the ontology.

We can draw some general conclusions by looking at the term frequencies among the Bag of words (BOW) documents. First, there are some terms that are topic-specific, such as budget. Other terms appear almost evenly distributed across all topics, such as inflation. This results from the fact that BOW scores word frequencies. A major problem with scoring word frequency is that highly frequent words start to dominate in the documents but may not contain as much "informational content" to the model as rarer but perhaps domain specific words. This accounts for unrelated terms like gun and inflation clustered together. Another key finding was the larger topic size had irrelevant terms appearing in multiple clusters compared to the smaller cluster sizes.

The application of equivalence classes on the corpus produced an improved result. The topic size three almost resembles our ontology's ground truth, as shown in Figure 18. There was some expected noise from less dominant themes. Overall, equivalence classes on the corpus grouped terms that are equivalent to each other, which normalized the corpus and produced improved results after applying the LDA model. There were overlaps in topics with the number of topics four and above. Distinct topics were however emerged with a number of topics four and below.

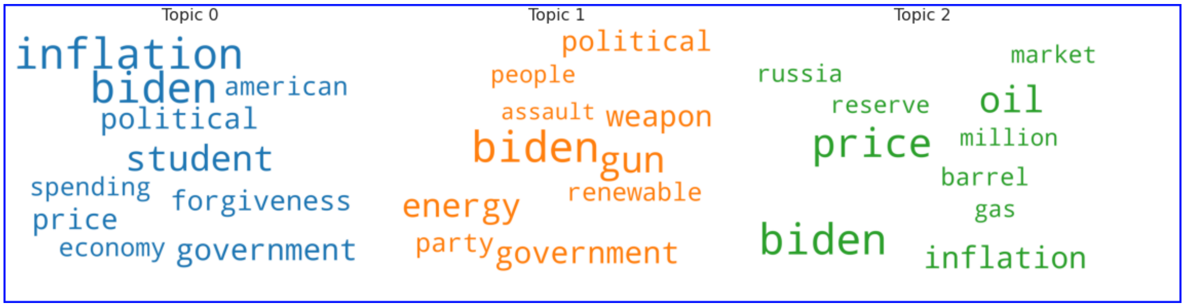


Figure 19. Word Cloud LDA (BOW) with Equivalence Class (Number topics = 3)

Experiment 5 (Bert Embedding): Compared to LDA, the clustered terms returned by the BERTopic model in Experiment 5 were more precise and closely related. However, we observed a few noises in the BERTopic model. One document about the defense budget got flagged as an outlier. The LDA model is very susceptible to data processing, as shown in Table 7. We see improvement in LDA after iterations of creating and modifying equivalence classes. One key finding of our research was just how essential the entity co-referencing step is in using equivalence classes for topic modeling. LDA, with the equivalence class dictionary, created three major topics similar to BERTopic modeling. However, we see a few unrelated terms like gun and energy clustered together in LDA with the equivalence class dictionary, as shown in Table 8. However, BERTopic clustered the relevant terms under each topic. All the terms related to economy and inflation clustered together. Gun violence and assault are in one topic, and renewable and non-renewable energy are clustered together in energy as a parent topic. BERTopic modeling almost resembles the ontology based representation. The only exception was infrastructure spending related documents clustered under the energy topic, and documents for the defense budget got flagged as an outlier. We color-coded our flow ontology to show the overlap with our BERTopic model's three main topics in Figure 20 below.

Common Topics	**LDA (BOW)**	**LDA (TF-IDF)**	**LDA (Equivalence Class)**	**BERTopic**
1	**Student forgiveness Inflation Gun Weapon**	**Student Forgiveness oil Reserve rfs**	**Inflation Biden Student Government Political**	**Inflation Biden Student Loan Student Loan**
2	**Biden Oil Gun Administration Price**	**Gun Weapon Defense Russian Leader**	**Biden Gun Energy Government Weapon**	**Gun Weapon Biden Assault Assault weapon**

3	**Biden Inflation Price Consumer Energy Gas**	**Fed Investor Recession Unemployment Economist coronavirus**	**Biden Price Oil Inflation Barrel Russia**	**Biden Energy Oil Price Renewable**
4	**Biden Inflation American World Economy Defense**			

Table 7. Comparison of topics created by LDA, and BERT topic models

Topic Number	Document Title	Topic Name
-1	MRD_Doc2_Not-Even-Halfway.docx	OUTLIER
1	BAC_Doc1_Biden-Rescue-Plan.docx	ECONOMY
	BAC_Doc2_Biden-worsened -Inflation.docx	
	CMP_Doc1_Consumer_Price_Index.docx	
	CMP_Doc2_Voters_have_made.docx	
	JAS_D0c1_Student_Loan_Forgiveness.docx	
	JAS_Doc2_Inflation-complicates_Biden.docx	
	JJ_Doc1_Consumers-Are-Feeling.docx	
	JJ_Doc2_Why_Is_Inflation.docx	
	MRD_Doc1_Civil-Society-Groups.docx	
	SS_Doc1_job_growth_double-edged_sword.docx	
	SS_Doc2_inflation_impacts_Student-Loan-Forgiveness.docx	
	Sieminski_Doc1_Inflation.docx	
	Sieminski_Doc2_Recession.docx	
2	DT_Doc1_Biden_Administration_Renewable.docx	ENERGY
	DT_Doc2_Biden_More_Rewable.docx	
	KN_Doc1_Biden-urges-G7.docx	
	MCD_Doc1_Biden-Oil-Reserve .docx	
	MCD_Doc2_More-Biden-Oil.docx	
	MCD_Doc3_Even-More-Biden-Oil.docx	
	PSJ_doc1_President_Biden_announced.docx	

	PSJ_doc2_The_worlds_wealthiest.docx	
	SS_Doc1_Mission_Not_Yet.docx	
	SS_Doc2_A_New_Task.docx	
3	**JS_DOC_1_lives_will_be_saved.docx**	**GUN CONTROL**
	JS_DOC_2_Biden_signs_gun_control.docx	
	TSS_Doc1_Biden-Gun-Control.docx	
	TSS_Doc2_Biden-Remarks-Gun-Violence.docx	

Table 8. BERT Topic Model Representation and Descriptions

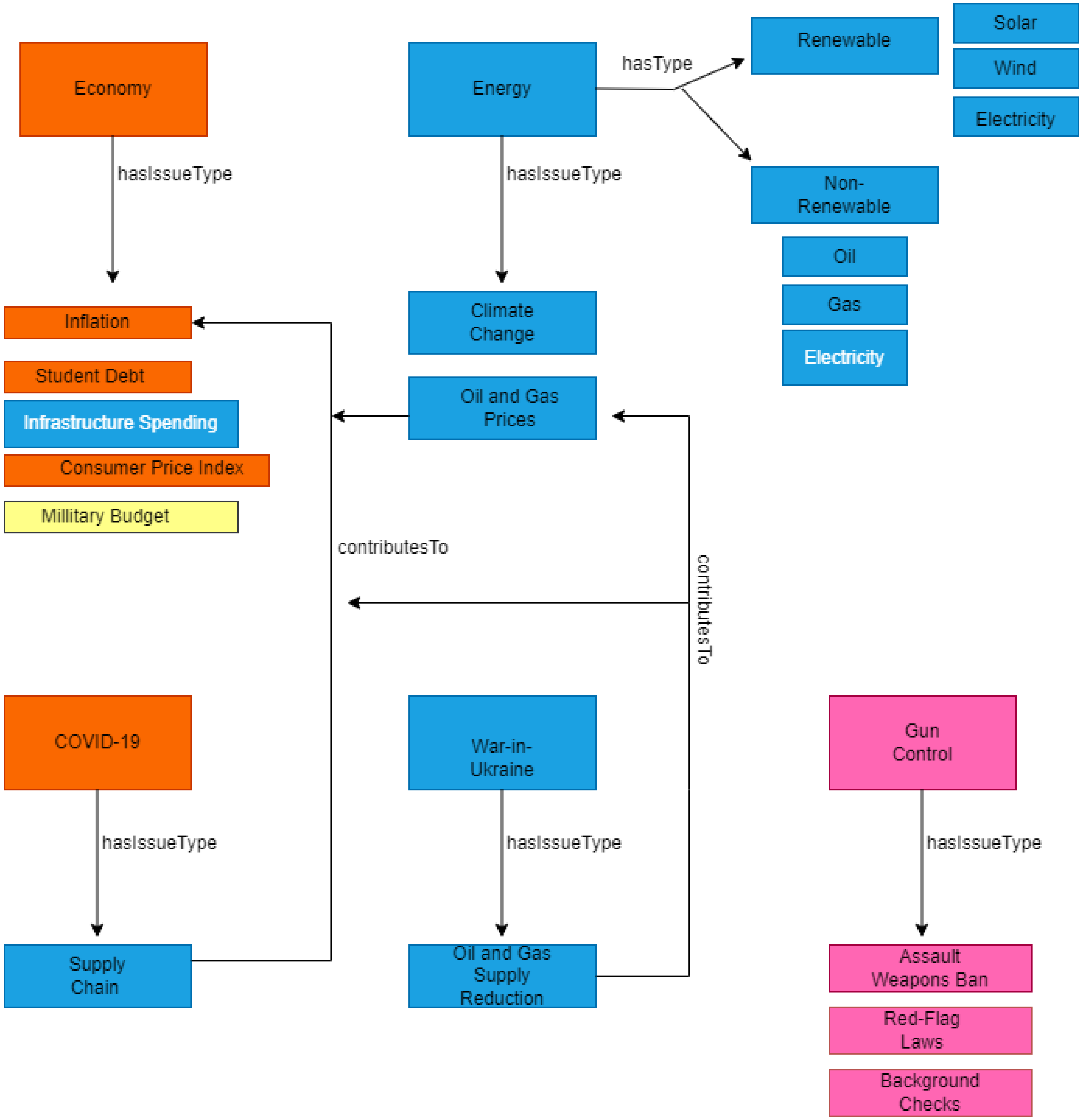


Figure 20. Color Coded Flow Ontology on corpus of Biden Administration with Table 8 above.

Conclusions

This paper described our work in a natural language pipeline as applied to a corpus of articles listing challenges faced by the Biden Administration. We have demonstrated a thorough understanding of feature engineering, selected algorithms, and analysis of algorithmic results. In doing so, we successfully addressed each question listed in the beginning, discovering noteworthy and significant connections between the reference term vector preparation and topic model results (LDA and BERTopic). We have demonstrated the performance of our topic models and how it resembles the flow ontology with and without the application of equivalence classes. Our manual effort of customizing the equivalence class dictionary emphasized the importance of feature engineering in natural language processing applications. Furthermore, we also compared the LDA and BERTopic modeling with bag of words and TF-IDF vectorization. Our LDA model suffered from the same disadvantages as the bag of words, ignoring syntactic information (e.g., word order) and semantic information (e.g., the multiplicity of meanings of a given word). We also inferred from LDA model results that it is very susceptible to any modification of input vector space. This is due to the probabilistic nature of the LDA algorithm, as it can generate associations between words and topics by assigning probabilities. So, LDA regenerates the probabilities between words and topics with any manipulation to the input vector space. As for BERTopic modeling, the cluster terms were more precise and tailored to the specific clusters. However, there is one instance where BERTopic modeling grouped unrelated terms such as infrastructure and energy. While BERTopic modeling did have these outliers, the clusters more closely resemble the ground truth of our corpus ontology representation.

Directions for Future Work

One thing we would look to do in the future is to expand the corpus and sample more evenly from all issues faced by the Biden administration. We had a small, hand selected corpus of documents. Taking what we have and expanding it to a much larger selection of articles would give a broader picture of current issues the media believes are most important to focus on for the Biden administration or future administrations. Ideally, we would scrape the data from a collection of trusted news sources. Another expansion of this would be tracking issues over time, so we could see which topics are important during the presidency.

Another potential avenue is comparing topic clusters from our corpus, or a larger corpus, to topic clusters of white house tweets and press releases to determine whether the current administration addresses the concerns most talked about in the media. This would require scraping tweets and articles, matching dates, and determining how similar the topics are between those two groups. This would also help inform how effectively the current administration approaches the issues that concern most Americans.

A third possible option for expanding our research would be to compare topic clusters on media and Biden Administration press releases to tweets and Facebook posts. This would add the dimension of accounting for what the average American is concerned by and how well that matches with our topic clusters from media sources.

References

Hilliard, Marcus. 2021. "CORPUS ANALYSIS of POLITICAL ASSOCIATIONS UTILIZING NATURAL LANGUAGE PROCESSING." Medium. March 15, 2021. https://marcus-hilliard.medium.com/corpus-analysis-of-political-associations-utilizing-natural-language-processing-149cd43bfe8d.

Ahmed, Fasih, Muhammad Nawaz, and Aisha Jadoon. "Topic Modeling of the Pakistani Economy in English Newspapers via Latent Dirichlet Allocation (LDA)." SAGE Open 12, no. 1 (2022): 215824402210799–. https://doi.org/10.1177/21582440221079931.

Wang, Minghao, and Paolo Mengoni. 2020. "How Pandemic Spread in News: Text Analysis Using Topic Model." IEEE Xplore. December 1, 2020. https://doi.org/10.1109/WIIAT50758.2020.00118.

Vidiyala, Ramya. 2021. "Topic Modelling on NYT Articles Using Gensim, LDA." Medium. June 9, 2021. https://towardsdatascience.com/topic-modelling-on-nyt-articles-using-gensim-lda-37caa2796cd9.

Sethia, Kashi, Madhur Saxena, Mukul Goyal, and R.K. Yadav. "Framework for Topic Modeling Using BERT, LDA and K-Means." 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE) 2022. https://doi.org/10.1109/icacite53722.2022.9823442.

Lane, H., Howard, C., and Hapke, H.M. (2019). Natural Language Processing in Action (Shelter Island, NY: Manning Publications). Chapter 3: Math with Words (tf-idf vectors)

Le, Quoc V, and Tomas Mikolov. 2014. "Distributed Representations of Sentences and Documents." ArXiv.org. 2014. https://arxiv.org/abs/1405.4053.

W, Forgy E. 1965. "Cluster Analysis of Multivariate Data : Efficiency versus Interpretability of Classifications." Biometrics 21: 768–69. https://cir.nii.ac.jp/crid/1571980074621944832.

Estival, Dominique, Chris Nowak, and Andrew Zschorn. 2004. "Towards Ontology-Based Natural Language Processing." Proceeedings of the Workshop on NLP and XML (NLPXML-2004): RDF/RDFS and OWL in Language Technology on - NLPXML '04. https://doi.org/10.3115/1621066.1621075.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. JMLR 3, pp. 993-1022. Retrieved from http://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf

Raju, Vasudeva, Bharath Kumar Bolla, Deepak Nayak, and Jyothsna Kh. n.d. "Topic Modelling on Consumer Financial Protection Bureau Data: An Approach Using BERT Based Embeddings." Accessed July 6, 2022. https://arxiv.org/ftp/arxiv/papers/2205/2205.07259.pdf.

Röder, Michael, Andreas Both, and Alexander Hinneburg. "Exploring the Space of Topic Coherence Measures." Proceedings of the Eighth ACM International Conference on Web Search and Data Mining 2015. https://doi.org/10.1145/2684822.2685324.

Chang, Jonathan, Sean Gerrish, Chong Wang, Jordan Boyd-graber, and David Blei. “Reading Tea Leaves: How Humans Interpret Topic Models.” In Advances in Neural Information Processing Systems (2009): 288-296. https://papers.nips.cc/paper/2009/fileel/f92586a25bb3145facd64ab20fd554ff-Paper.pdf.

Appendix A

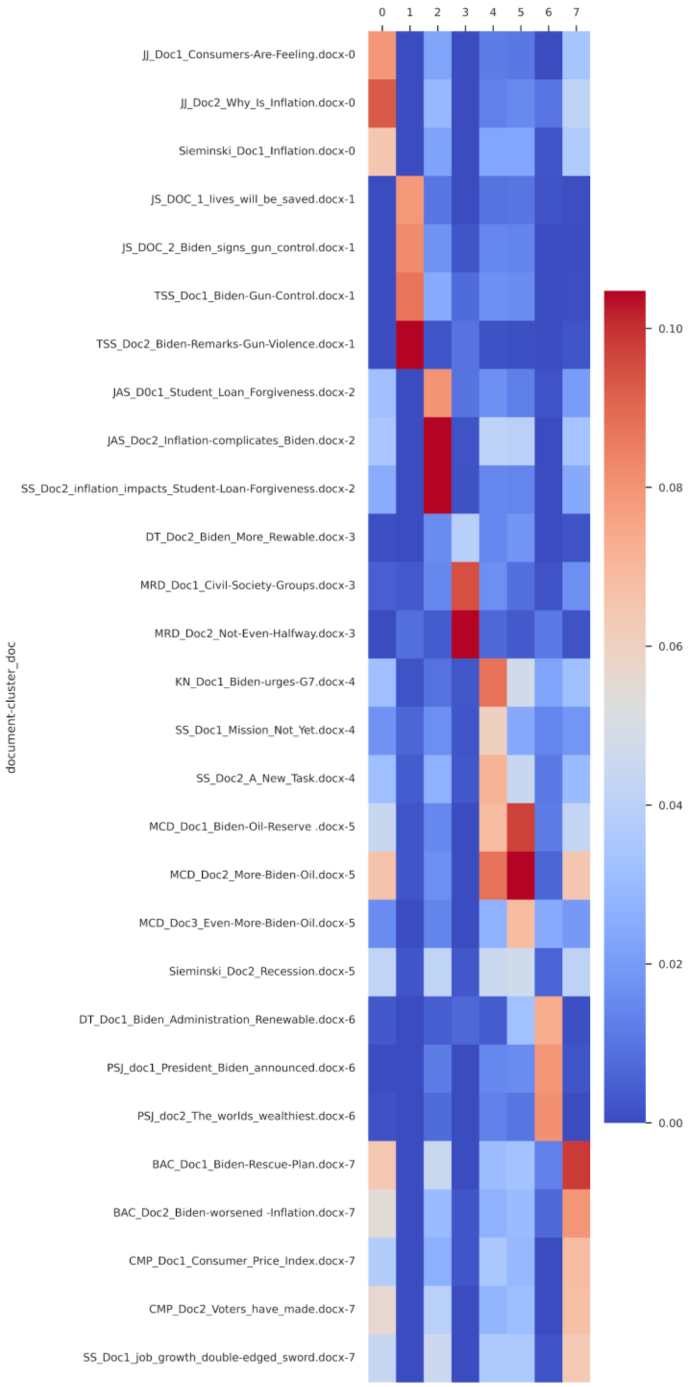


Figure 21. K-means Clustering with TF-IDF (K=8) - Heatmap

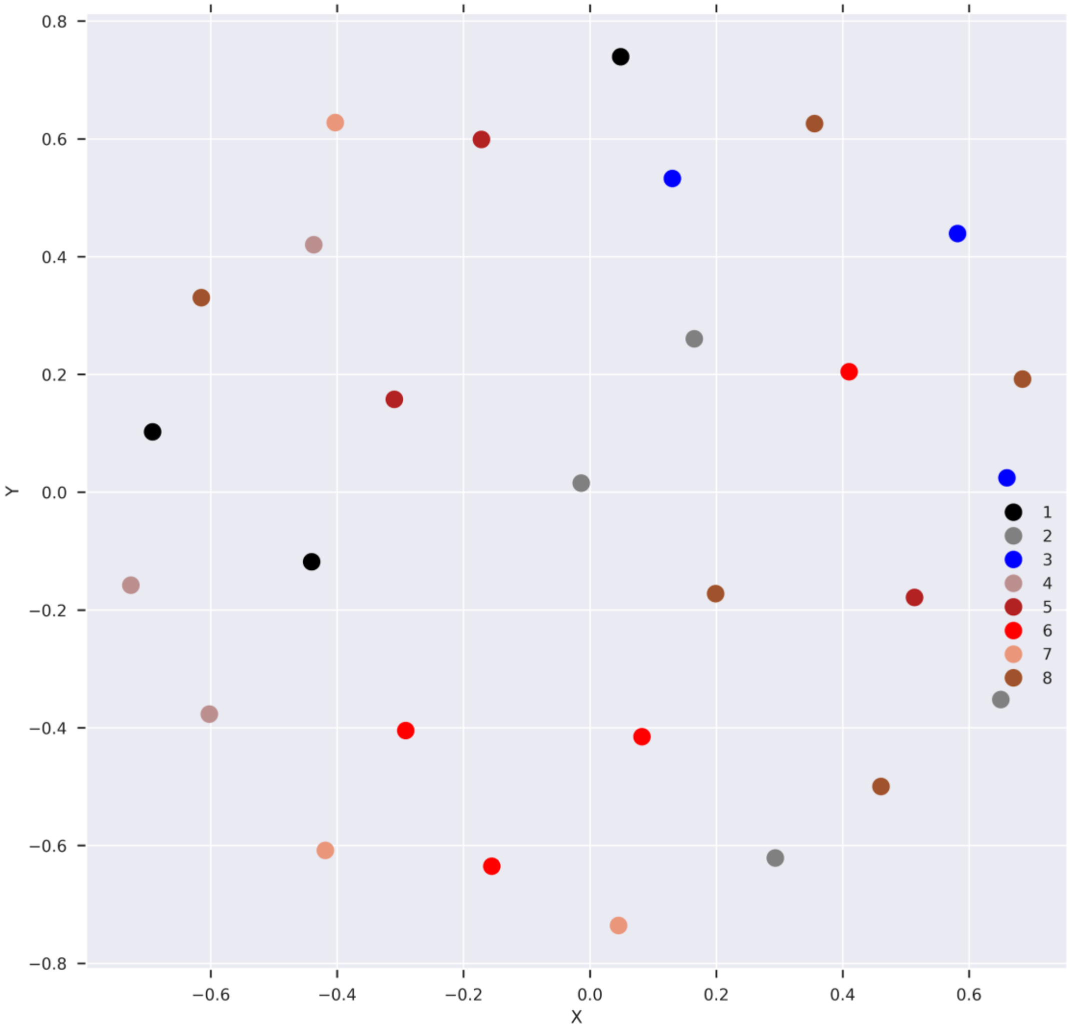


Figure 22. K-means Clustering with TF-IDF (K=8)

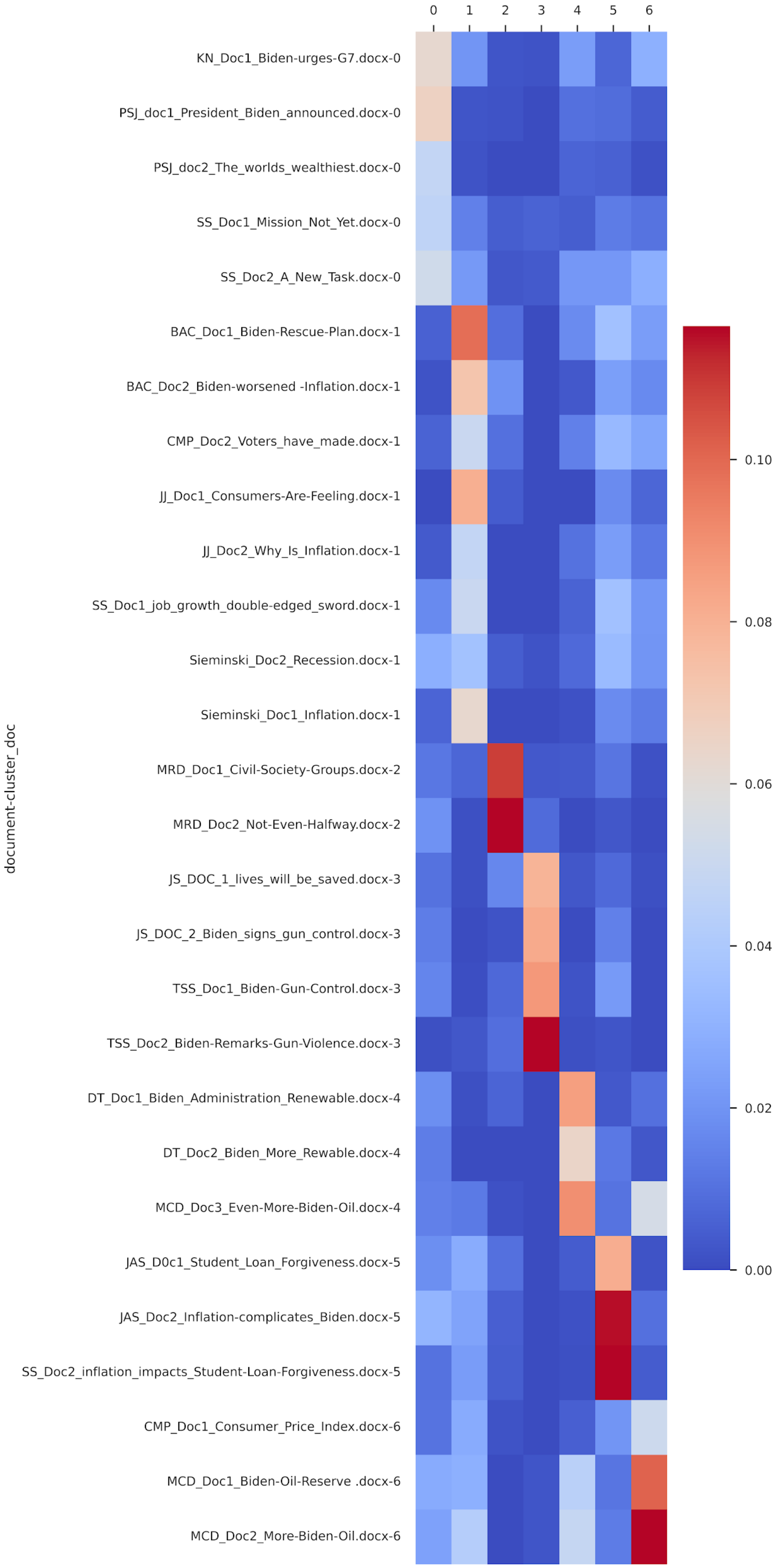


Figure 23. K-means Clustering with TF-IDF (K=7) - Heatmap

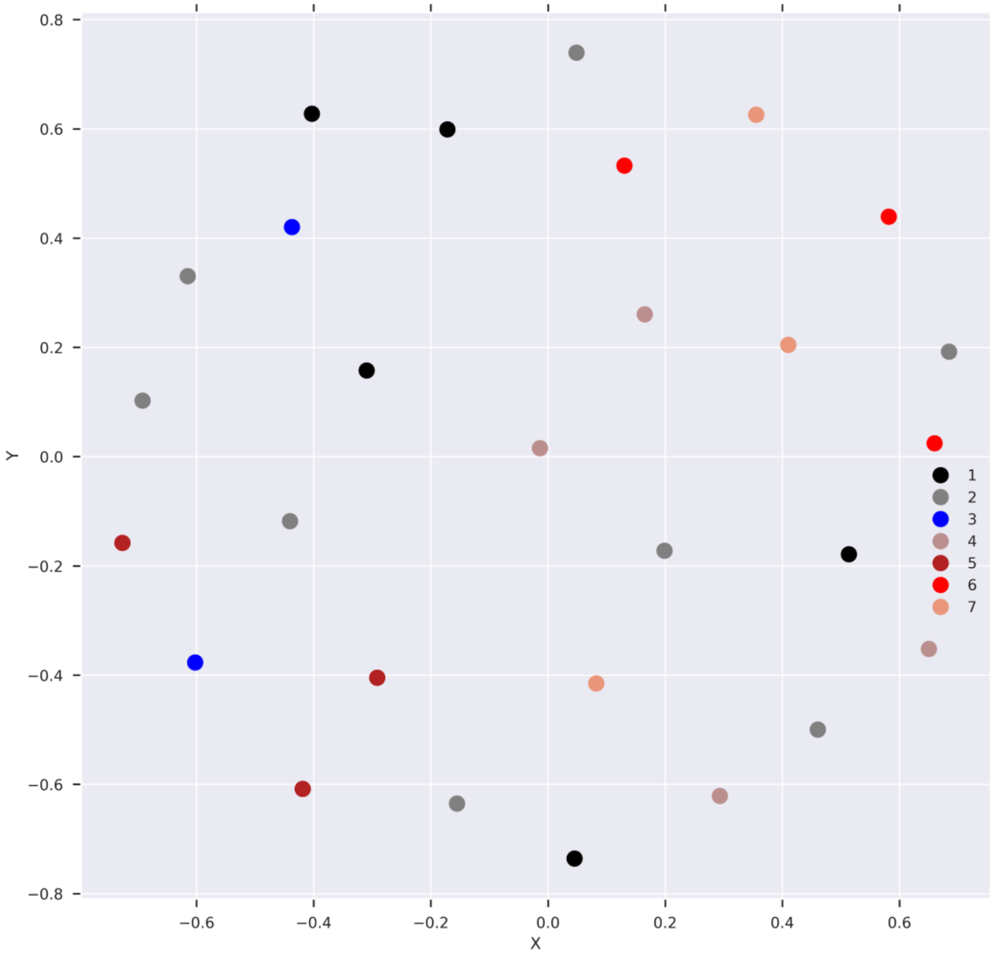


Figure 24. K-means Clustering with TF-IDF (K=7)

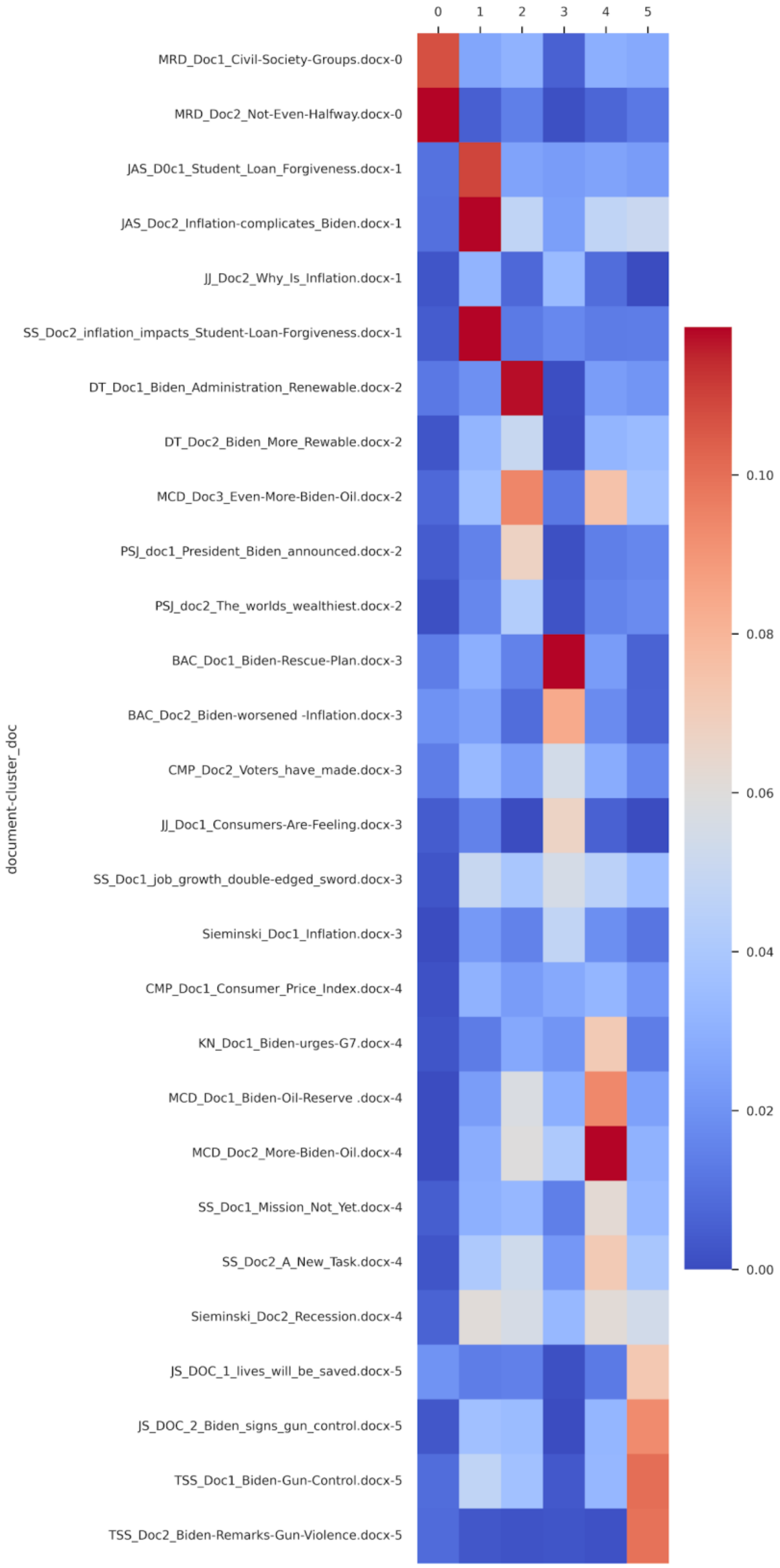


Figure 25. K-means Clustering with TF-IDF (K=6) - Heatmap

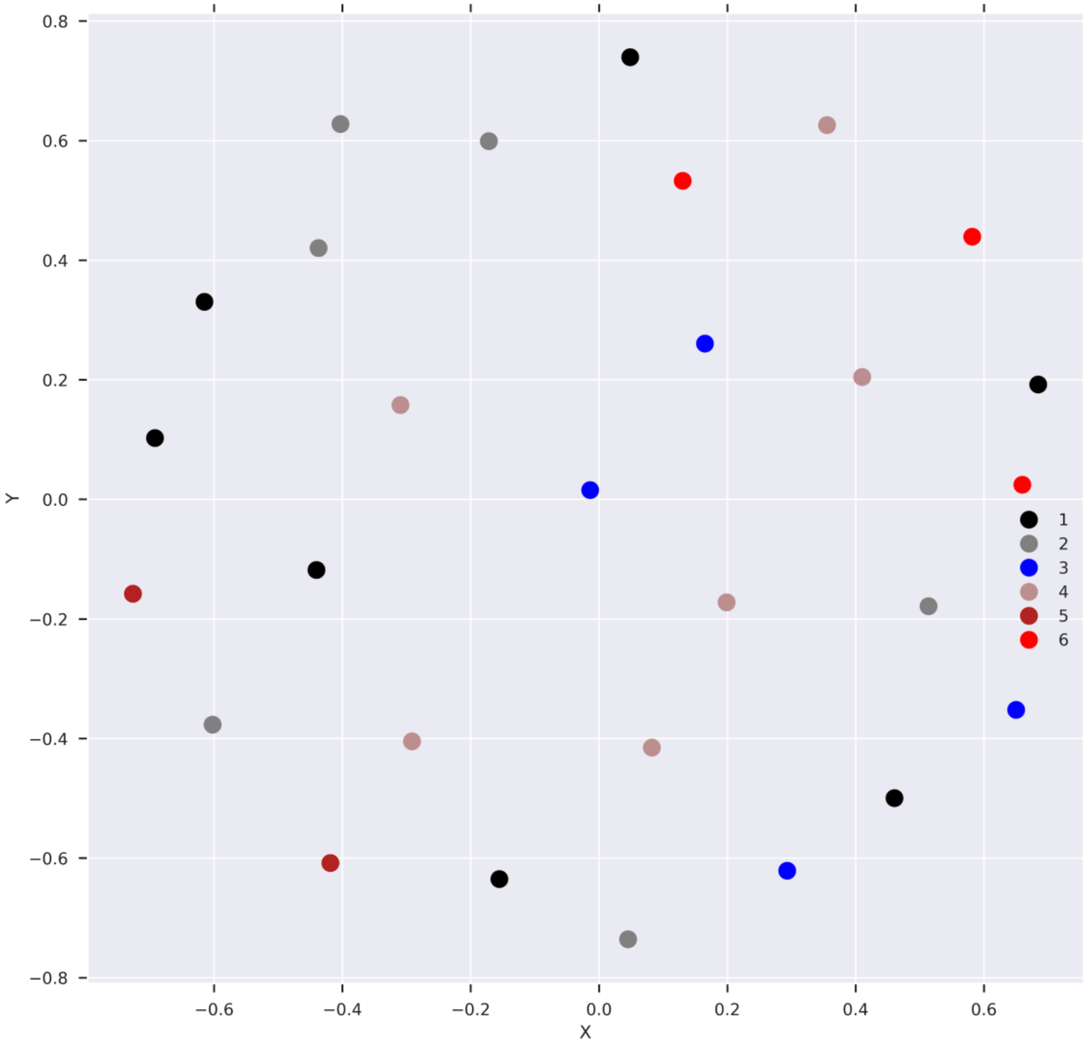


Figure 26. K-means Clustering with TF-IDF (K=6)

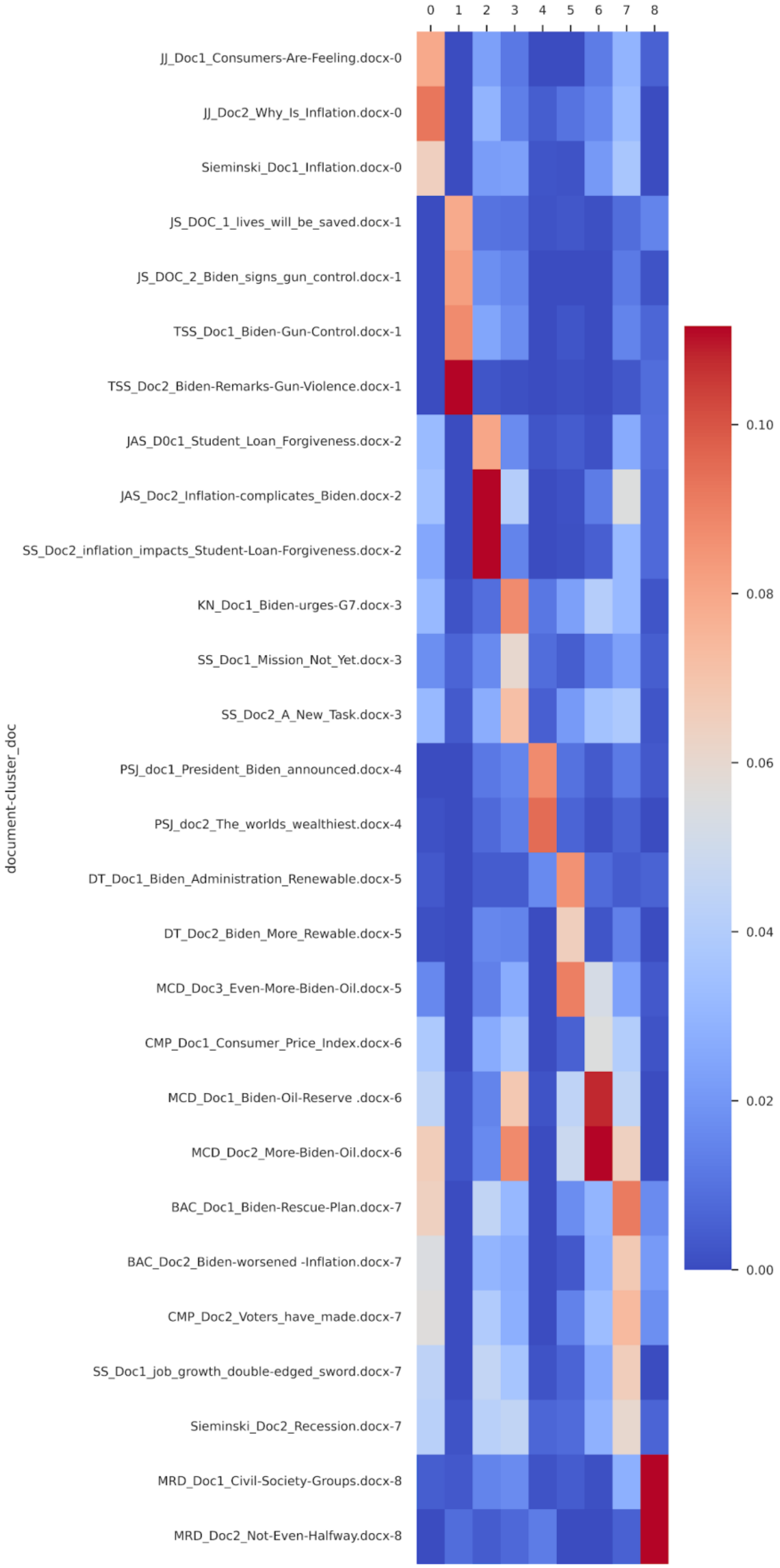


Figure 27. K-means Clustering with TF-IDF (K=9) - Heatmap

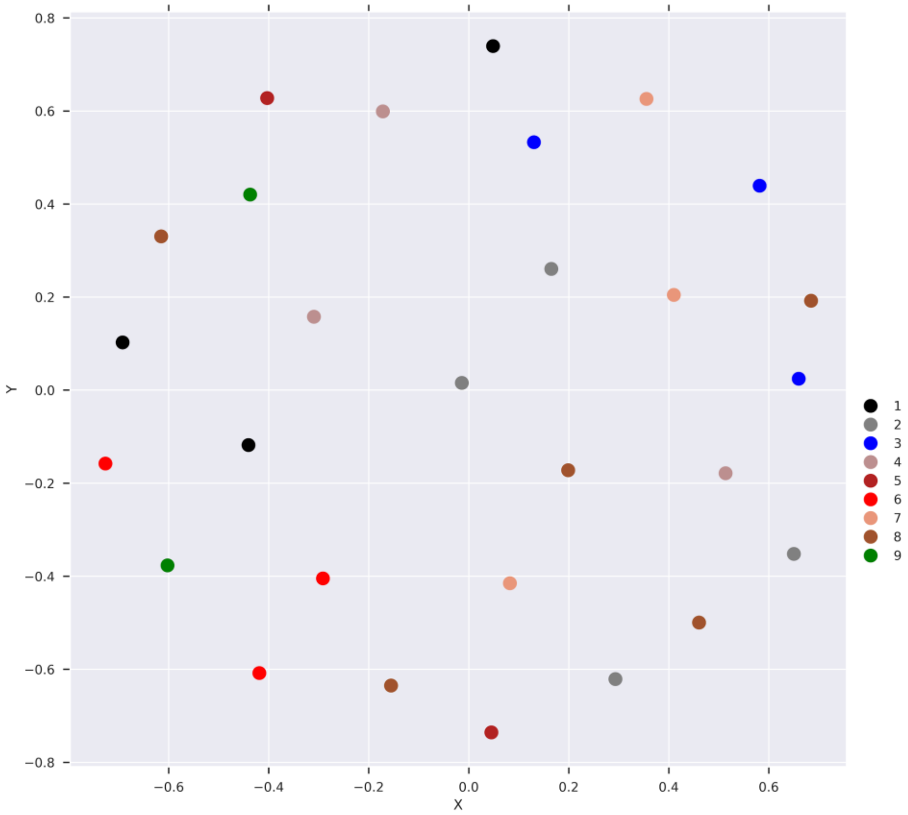


Figure 28. K-means Clustering with TF-IDF (K=9)

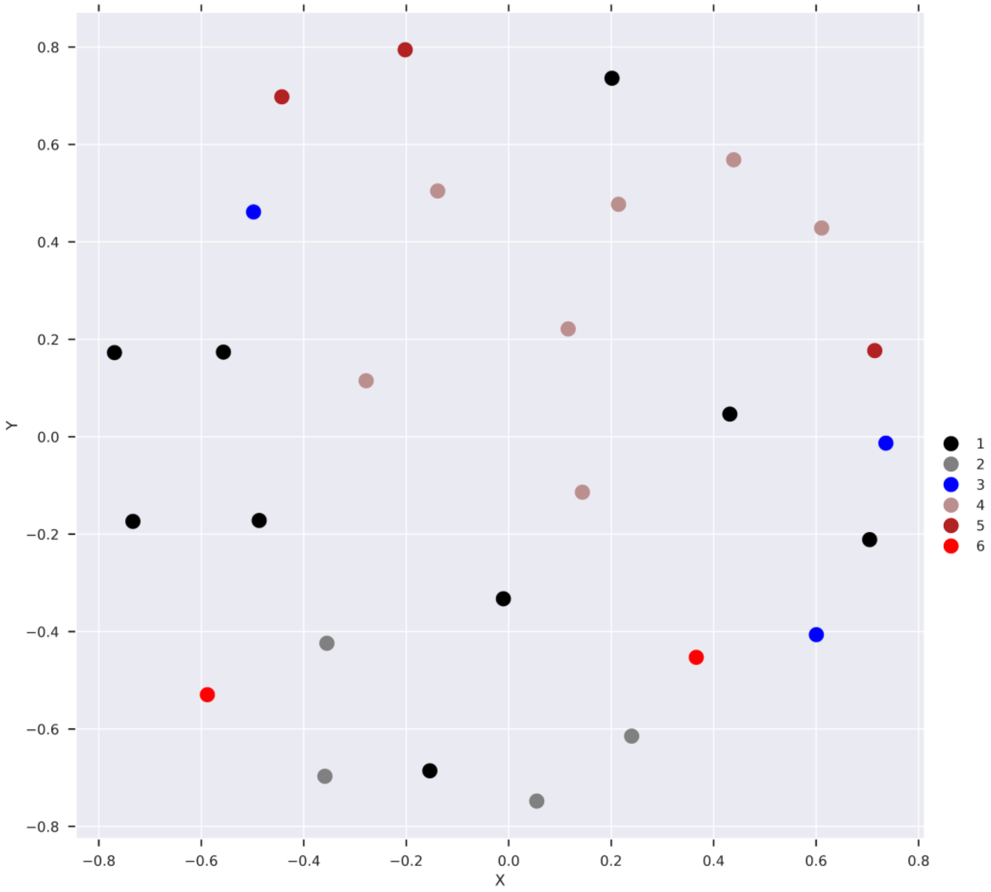


Figure 29. K-means Clustering with Doc2Vec (K=6)

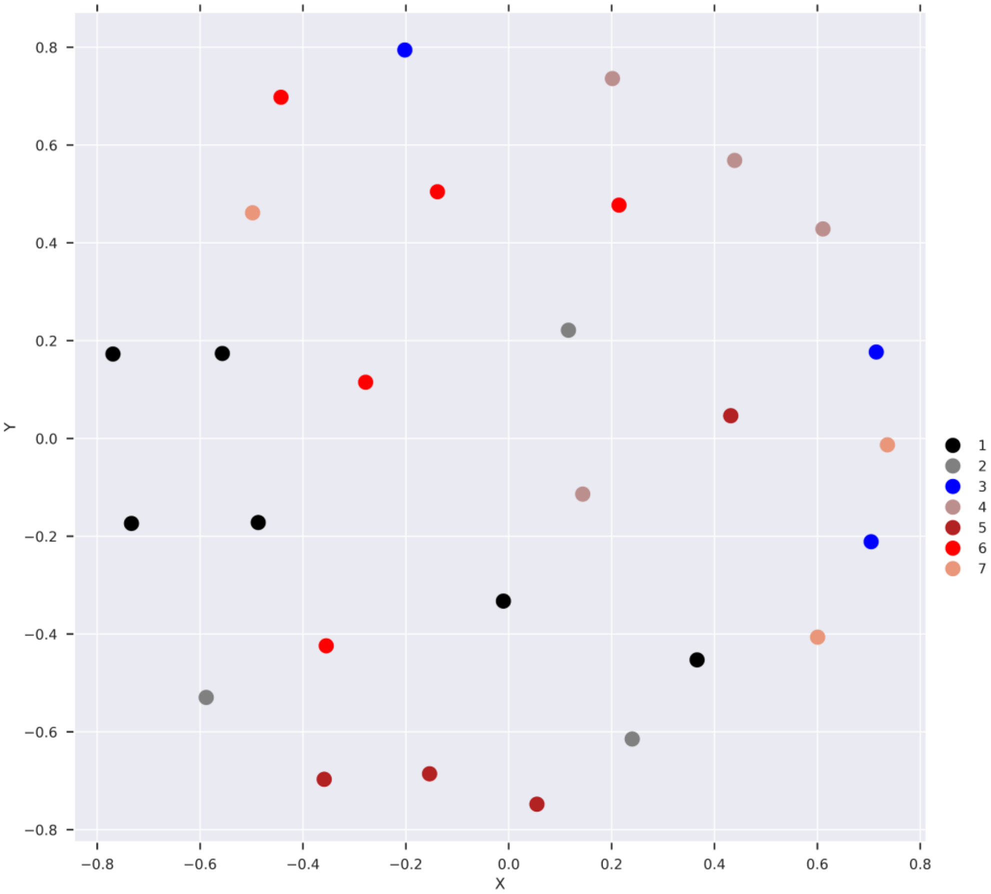


Figure 30. K-means Clustering with Doc2Vec (K=7)

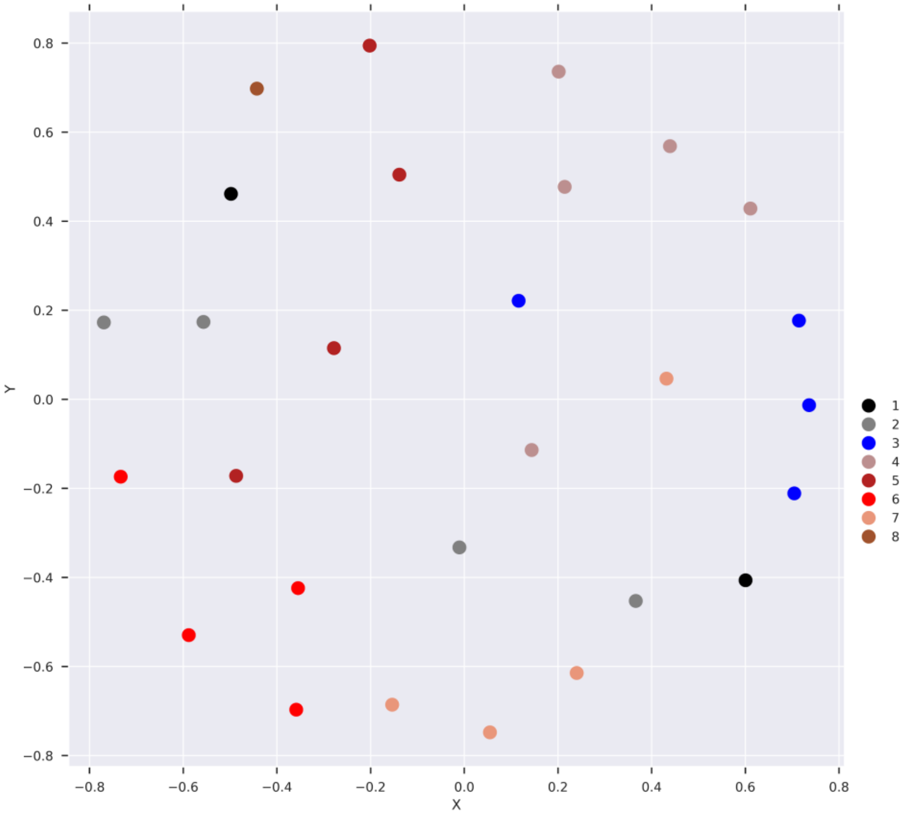


Figure 31. K-means Clustering with Doc2Vec (K=8)

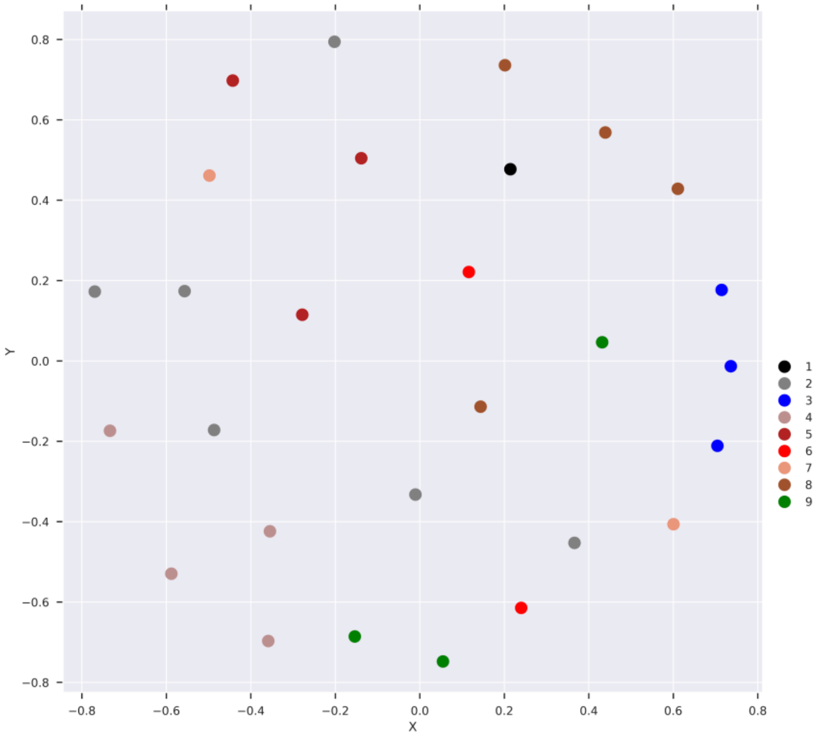


Figure 32. K-means Clustering with Doc2Vec (K=9)

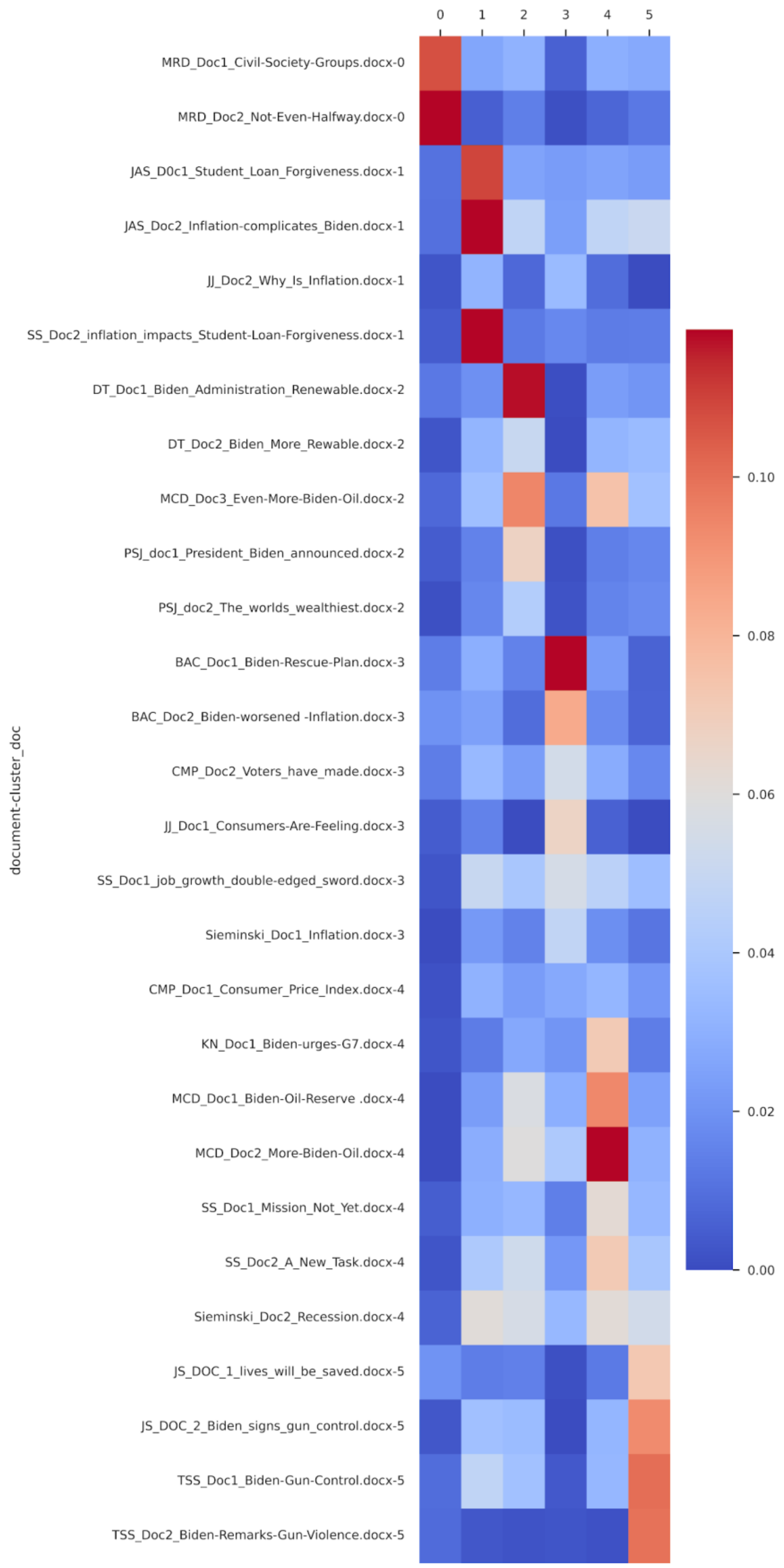


Figure 33. K-means Clustering TF-IDF with Equivalence Classes (K=6) - Heatmap

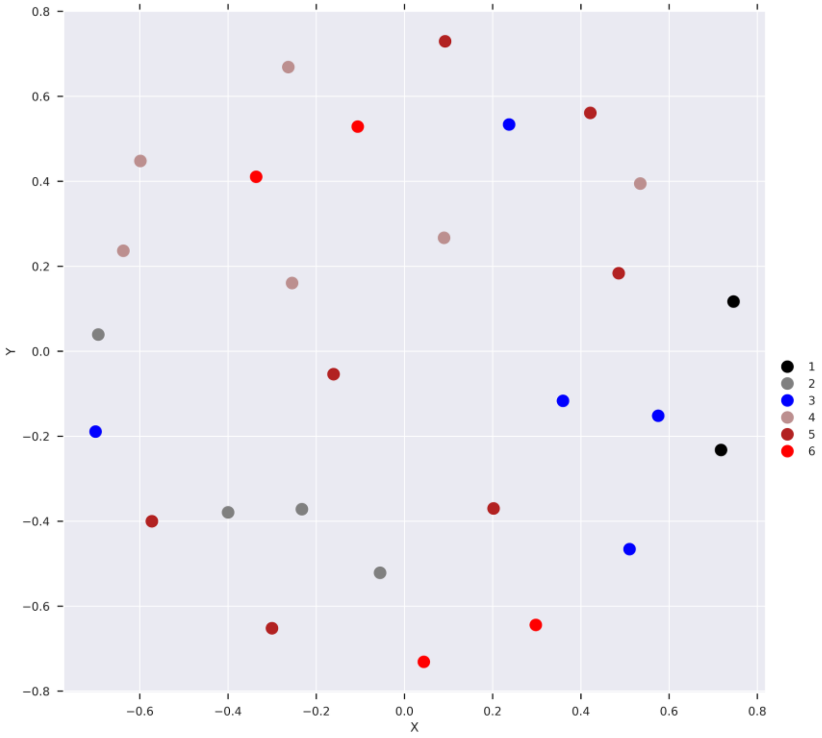


Figure 34. K-means Clustering TF-IDF with Equivalence Classes (K=6)

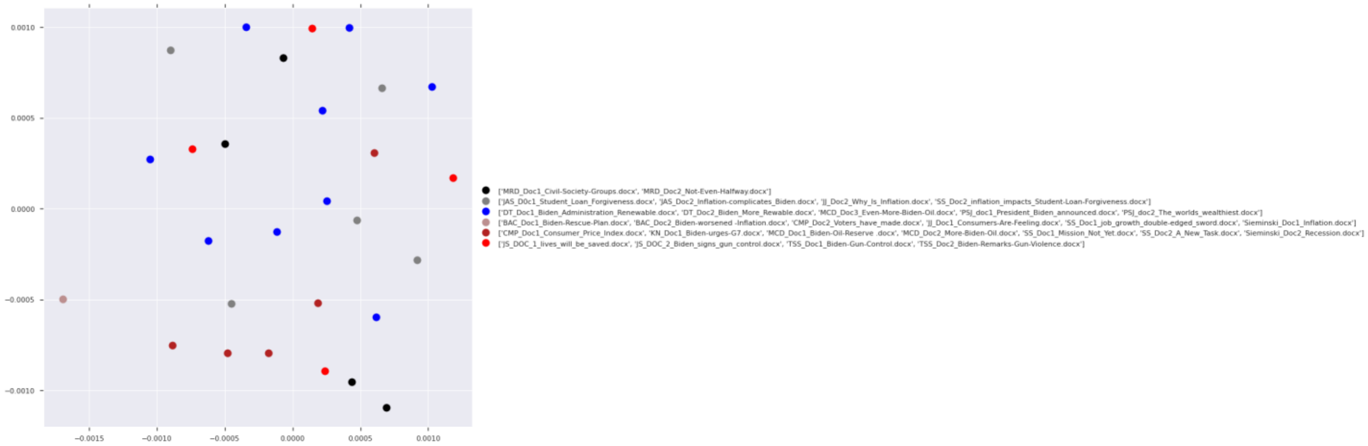


Figure 35. K-means Clustering TF-IDF with Equivalence Classes (K=6)

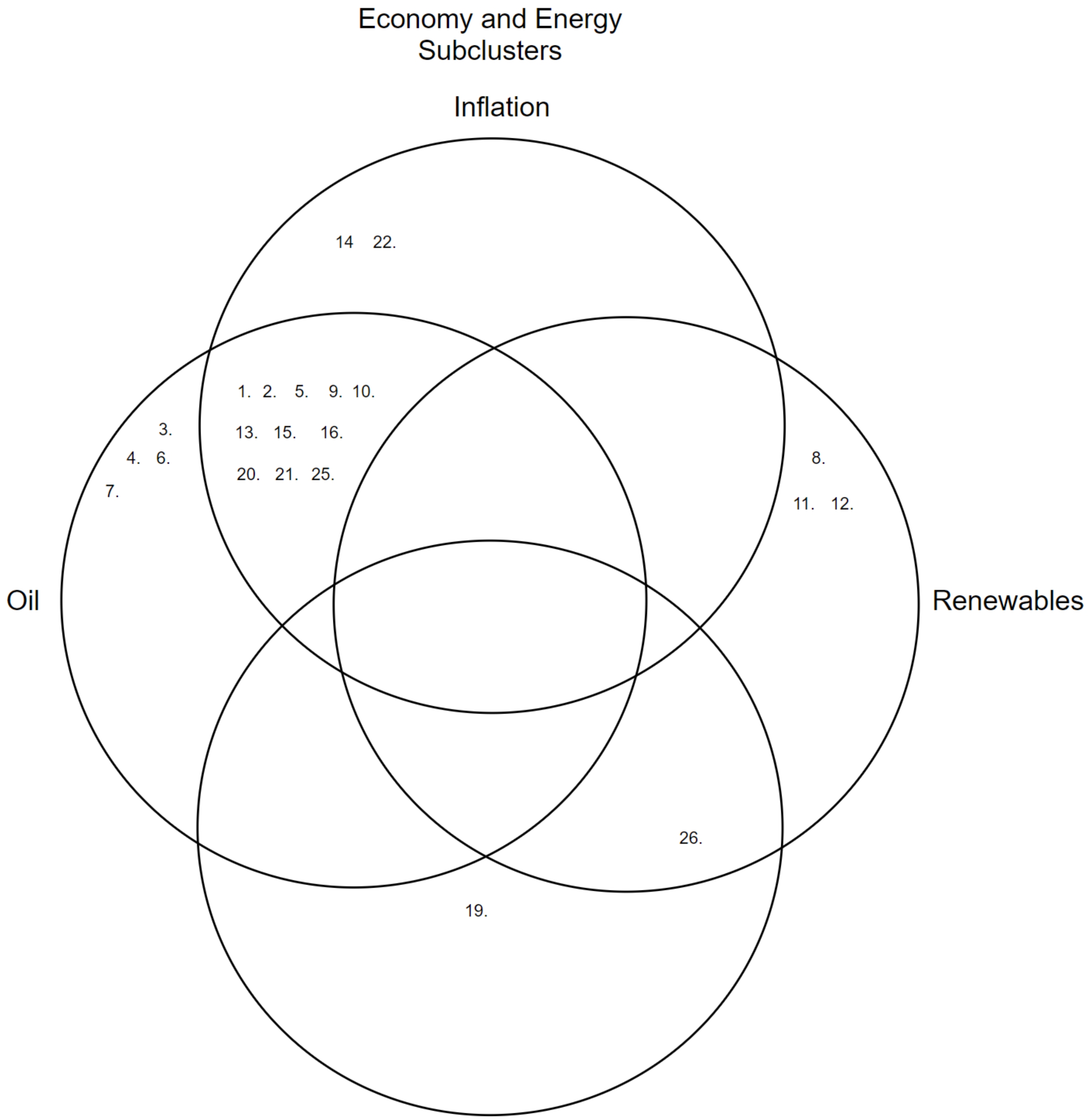


Figure 36. Sub-Cluster Level Ontology: Economy and Energy

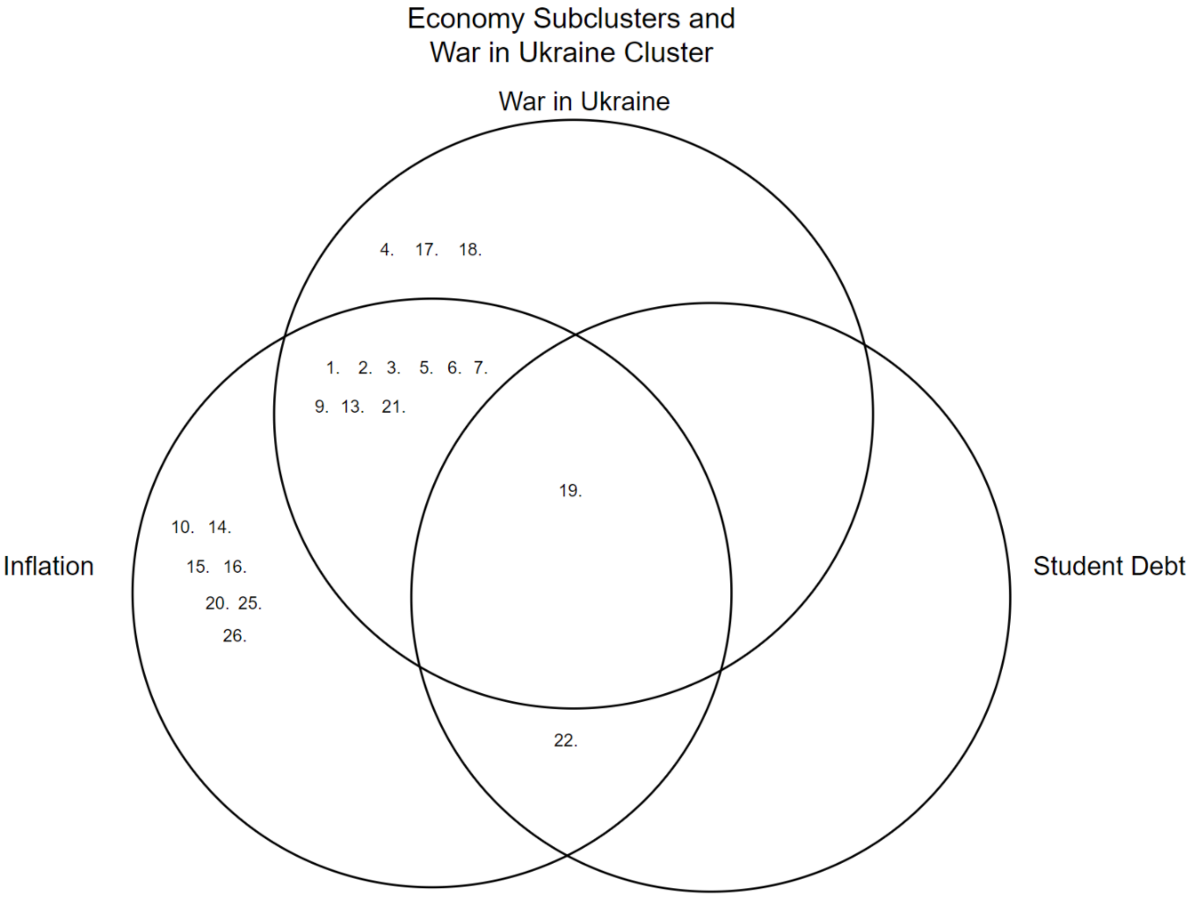


Figure 37. Sub-Cluster Level Ontology: Economy and the War in Ukraine

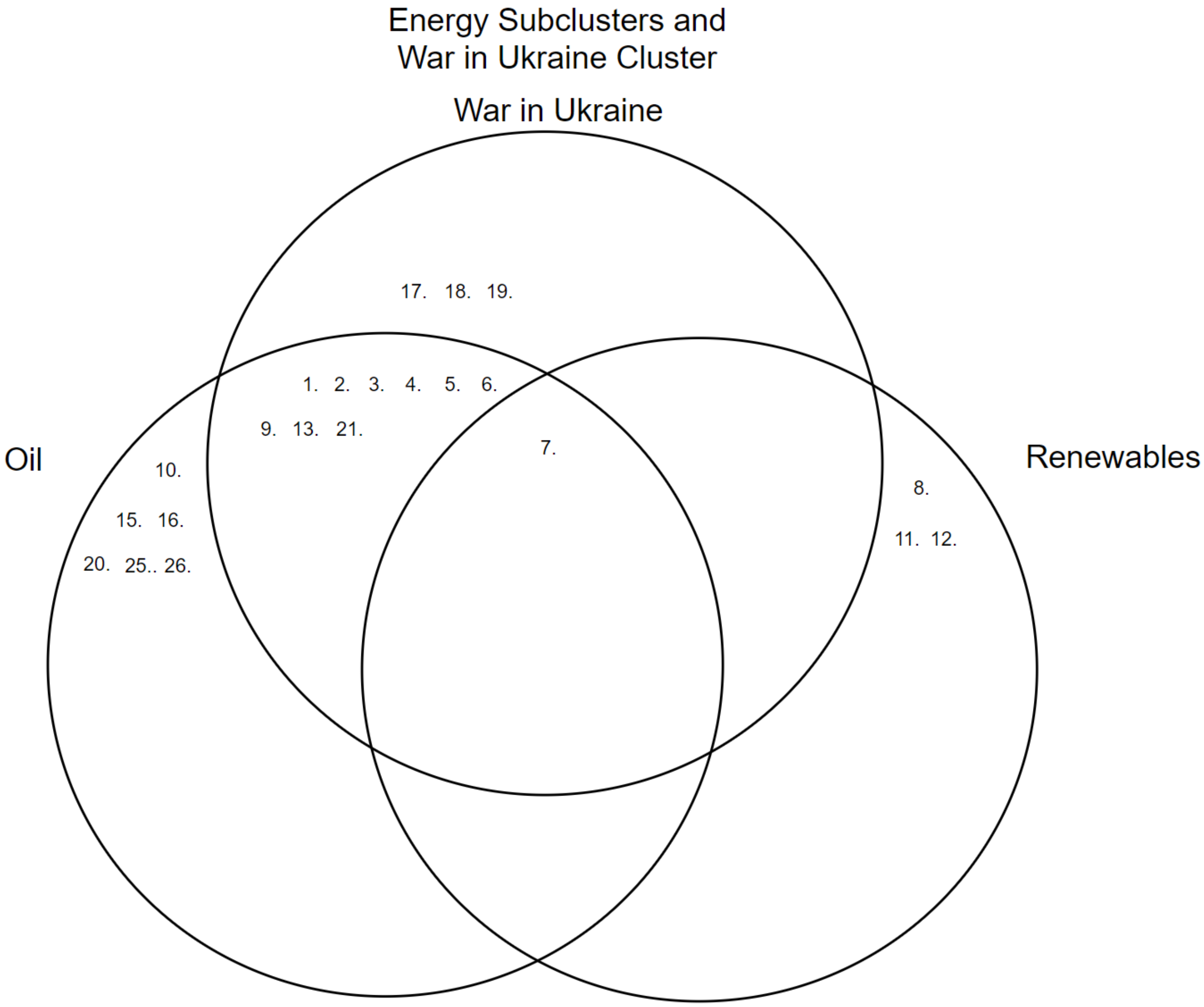


Figure 38. Sub-Cluster Level Ontology: Energy and the War in Ukraine