Topic Modelling Bookmarker Extension

Lahari Anne
lanne2@illinois.edu
University of Illinois
Urbana-Champaign, Illinois, USA

Sanjay Raj Aerra
saerra2@illinois.edu
University of Illinois
Urbana-Champaign, Illinois, USA

Sasi Pavan Surapaneni
ss257@illinois.edu
University of Illinois
Urbana-Champaign, Illinois, USA

ABSTRACT

This proposal outlines the development of a Google Chrome extension aimed at enhancing the efficiency of bookmarking web pages by providing topic suggestions and sentiment analysis. The extension targets individuals who frequently bookmark web pages for reference or research purposes, offering them a streamlined approach to organizing online content. Leveraging natural language processing techniques and machine learning, the tool automates the process of suggesting topics for bookmarking and analyzing the sentiment of web pages. Through a collaborative effort, the project team plans to deliver a user-friendly and effective solution that improves content discovery and retrieval for users.

KEYWORDS

NLP, information retrieval, LDA, Google extension, bookmarker

1 INTRODUCTION

In an era where information overload is a common challenge, efficiently organizing and retrieving online content is crucial for individuals across various domains such as academia, research, and professional endeavors. This proposal presents a solution to address this challenge by developing a Google Chrome extension that assists users in categorizing and bookmarking web pages based on their content. By leveraging natural language processing (NLP) techniques, including Latent Dirichlet Allocation (LDA) for topic modeling, the extension offers advanced functionalities such as topic suggestions and sentiment analysis, enabling users to streamline their bookmarking process and enhance productivity.

This initiative aims to alleviate the burden of manual categorization and organization of bookmarked web pages, ultimately contributing to a more organized and efficient online experience for users. Through collaborative efforts and strategic planning, the project team endeavors to deliver a robust and user-centric solution that meets the diverse needs of individuals who rely on bookmarking for information management.

2 METHODOLOGY

This section outlines the process followed to acquire data for corpus creation, including collecting data from multiple sources to cover various generic topics for bookmarking, cleaning and preprocessing the data, training the model on this corpus, saving the model, and using this model to categorize the content of a webpage in real time.

2.1 Data Acquisition and Preprocessing

Data is collected from diverse sources to ensure coverage of a wide range of topics. This includes news articles, academic texts, and web content. The references to the data sources used is listed below.The various datasets used to generate the corpus are detailed below.

Huff Post New Category Dataset [7]: This dataset comprises approximately 210k news headlines spanning from 2012 to 2022, sourced from HuffPost, serving as a notable benchmark for computational linguistic tasks, with a majority of headlines collected before May 2018 due to changes in HuffPost's archive maintenance, featuring attributes like category, headline, authors, link, short description, and publication date.

State of the Union Corpus (1790 - 2018) [10]: This dataset encompasses the complete texts of State of the Union addresses delivered by U.S. Presidents from 1989 (Regan) to 2017 (Trump), providing an excellent resource for exploring Natural Language Processing techniques, including investigating the evolution of popular topics over time, discerning variations in tone among Presidents and across party lines, developing parsers to extract syntactic relationships between words, and attempting authorship identification of previously unseen addresses.

MachineHack News Category Dataset [4]: This dataset contains news articles along with their corresponding genres or categories, allowing Data Science and Machine Learning enthusiasts to utilize Natural Language Processing to predict the genre of a news article based on its content. It comprises 7,628 training records and 2,748 test records, with features including STORY (article content) and SECTION (genre/category), categorized into four sections: Politics, Technology, Entertainment, and Business.

UCI News Aggregator Dataset With Content [6] : This dataset provides information on news articles, including their ID, headline, URL, publisher, category (business, science/technology, entertainment, health), story ID, hostname, and publication timestamp (in Unix time). Additionally, it includes the article's main content and its length. It is sourced from the UCI Machine Learning Repository.

News Articles Classification Dataset for NLP and ML [11] : This dataset comprises news articles from various domains such as Business, Technology, Sports, Education, and Entertainment, sourced from "The Indian Express" news magazine. It exclusively focuses on events, developments, and topics related to India. With 10,000 rows and 5 columns, each category contains 2000 unique news contents. This dataset serves as a valuable resource for Natural Language Processing (NLP) and Machine Learning (ML) tasks like text classification, topic clustering, topic prediction, and Named Entity Recognition (NER).

The text data from all the above data sources is cleaned to remove noise and irrelevant information. The NLTK [5] library is used for tokenization, removing stopwords, lemmatization and stemming [1]. Gensim's corpora.Dictionary class is employed to create a dictionary mapping of words to numerical IDs for efficient processing. The distribution of words counts across all documents in our corpus can be seen in Figure 1. From all the five data sources listed above, we obtained around 800 thousand documents. From the figure, we observe that most of the documents have less than 200 words in them.

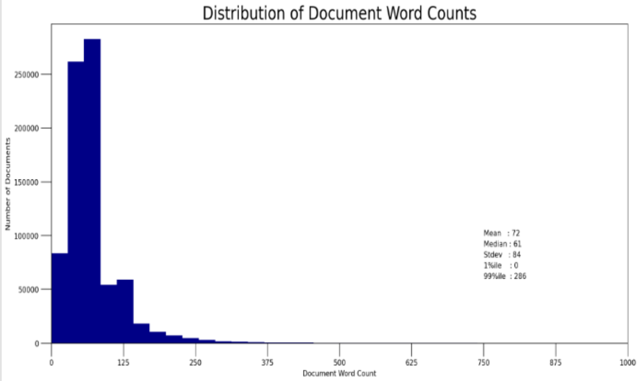


Figure 1: Word counts distribution

2.2 LDA Model Training

The cleaned and preprocessed corpus is used to train the Latent Dirichlet Allocation (LDA) model [2]. LDA is a generative probabilistic model for assigning topics to documents and words to topics. It assumes that documents are produced from a mixture of topics, where each topic is characterized by a distribution of words. In training, LDA learns the topic-word and document-topic distributions from the input corpus. These distributions are then used to infer the topics present in new, unseen documents.

Topics are represented as probability distributions over words, and documents are represented as probability distributions over topics. Each word in a document is assumed to be generated by one of the topics, with a certain probability based on the document's distribution over topics and the topic's distribution over words.

Gensim's LdaModel class is utilized for training, which implements the LDA algorithm internally. The model is trained with a specified number of topics and iterations to uncover the latent topics present in the corpus.

We then clustered the obtained topics into three main bookmarker topics. These topics include Business & Politics, Entertainment & lifestyle (including social media and philosophy) and Technology & Science (including Education & Literature). Figure 2 represents a t-SNE clustering plot. t-Distributed Stochastic Neighbor Embedding (t-SNE) is a machine learning algorithm for dimensionality reduction that is particularly well suited for the visualization of high-dimensional datasets.

Each color cluster represents documents that are closely related in terms of thematic content. Clear boundaries between clusters suggest good model separation ability. The clear segregation of most topics indicates effective learning and representation by the LDA model. Overlapping regions could be explored further to refine topic definitions or adjust model parameters for clearer distinctions. This t-SNE visualization assisted in understanding how well the LDA model has performed in segregating different thematic contents into discernible and coherent topics.

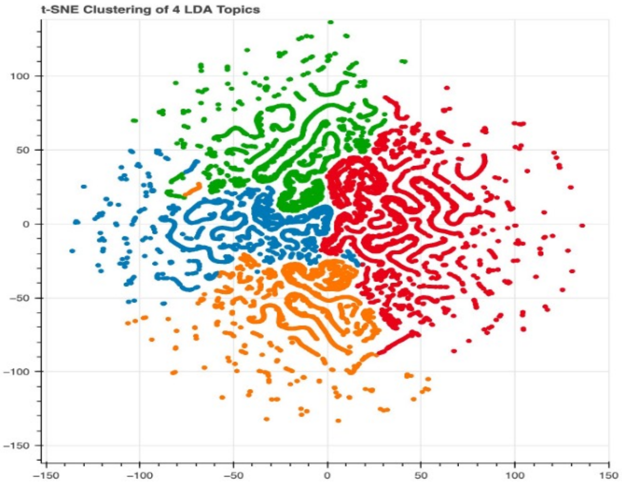


Figure 2: t-SNE clustering of first 4 topics from LDA model

2.3 LDA Model Evaluation

Since LDA is an unsupervised learning algorithm, there is no traditional evaluation metric like accuracy. Instead, the trained model's effectiveness is often assessed by inspecting the generated topics and their coherence. We performed tests on 100 sample websites and compared the performance of the trained LDA model with that of human perception and categorization of these webpages. The results of this evaluation are presented in the Results section below.

2.4 Sentiment Analysis

In addition to topic modelling, we also include a feature that performs sentiment analysis of the content of the webpage. Sentiment analysis helps users gauge the relevance of the webpage content to their interests. Positive sentiments may indicate informative or enjoyable content, while negative sentiments may suggest irrelevant or objectionable material. By analyzing the sentiment of webpage content, users can filter out irrelevant or negative content, enhancing their browsing experience. This ensures that bookmarked pages align with the user's preferences and interests. Sentiment analysis can aid in categorizing bookmarked webpages based on their emotional tone. Positive sentiment may correspond to categories like "inspirational" or "motivational," while negative sentiment may indicate "critical reviews" or "warnings."

The sentiment analysis is performed using the VADER (Valence Aware Dictionary and sEntiment Reasoner) [3] sentiment analysis tool. VADER is a lexicon and rule-based sentiment analysis tool specifically designed for social media text, but it is also widely used for general text sentiment analysis due to its simplicity and effectiveness.

The SentimentIntensityAnalyzer() class from the NLTK (Natural Language Toolkit) [5] library is instantiated to perform sentiment intensity analysis on the text data. This analyzer assigns sentiment scores to the text based on positive, negative, and neutral sentiments, as well as an overall compound score that represents the aggregated sentiment. The polarity_scores method of the sentiment analyzer is used to obtain sentiment scores for the text data extracted from the webpage. This method returns a dictionary containing the scores for positive, negative, neutral, and compound sentiments. These scores provide insights into the sentiment expressed in the webpage content.

2.5 Google Extension

To create the required Google extension, we adopted a methodology that involves integrating the LDA topic modeling and sentiment analysis functionalities into a user-friendly interface accessible through a Google Chrome extension. The manifest.json file provides important metadata about the extension, including its name, version, and permissions. Permissions are declared for accessing bookmarks, tabs, and notifications, along with host permissions to allow interaction with web content. The popup includes buttons for categorizing and bookmarking web pages (categorizeBtn) and for analyzing sentiment (sentimentBtn). Results of sentiment analysis are displayed in a designated area (sentimentResult).

The background bookmarking script serves as the intermediary between the extension's UI and the functionality provided by the Flask server. Event listeners are added to the categorize and sentiment buttons. When clicked, these buttons trigger functions to send messages to the Chrome runtime. The messages contain information about the desired action (categorizeAndBookmark or analyzeSentiment) along with any necessary data, such as the URL of the currently active tab. The Flask server hosts the backend functionality for categorizing web pages and performing sentiment analysis.

This methodology leverages a combination of frontend and backend technologies to create a seamless user experience within the Google Chrome browser. Users can easily categorize and bookmark web pages or analyze their sentiment with just a few clicks, thanks to the integration of advanced NLP techniques and a user-friendly interface provided by the extension.

3 MODEL WORKFLOW

A user initiates the bookmarking process by utilizing the Topic Modelling Bookmarker extension while browsing the web. Upon bookmarking a webpage using our Google extension, the tool automatically processes the content of the webpage in the background using web scraping technique. The Gensim-powered Latent Dirichlet Allocation (LDA) model is employed to extract the main topic or theme of the webpage based on its textual content. Concurrently, the tool performs sentiment analysis on the webpage content to determine its emotional tone and sentiment. This analysis provides insights into whether the content conveys a positive, negative, or neutral sentiment, enriching the user's understanding. The results of both the topic extraction and sentiment analysis are presented back to the user in a user-friendly format, within the Google extension. Users can view the identified topic and sentiment of the bookmarked webpage, aiding in their categorization and organization of bookmarks for future reference. The webpage is automatically bookmarked in the appropriate folder in the bookmarks. This workflow is visualized in Figure 3.

The user interface of the Google Extension is as shown in Figure 4. Clicking the Categorize and Bookmark button makes a call to the server, which internally first calls the web scraping function to acquire the content of the web page. This data is then cleaned, tokenized and passed as input to our custom trained LDA model. The output from the model is then mapped to the three main clustered bookmark categories. Based on the topic probability returned by the model, most appropriate category is returned and webpage is bookmarked into that respective category folder.

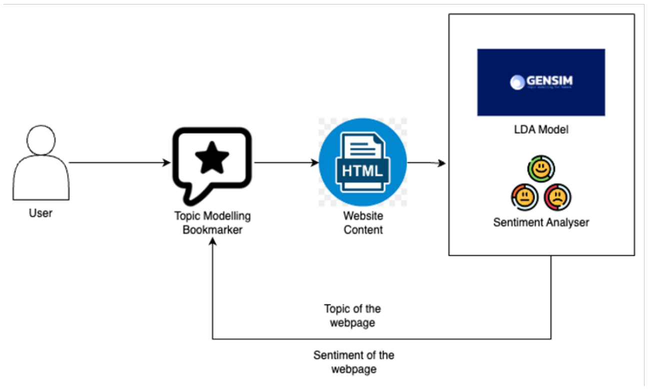


Figure 3: Proposed model workflow

The sentiment Analyser button calls the web scraper function which based on the content of the webpage determines the emotional tone of the content. This is done using the VADER tool as explained in the methodology section.

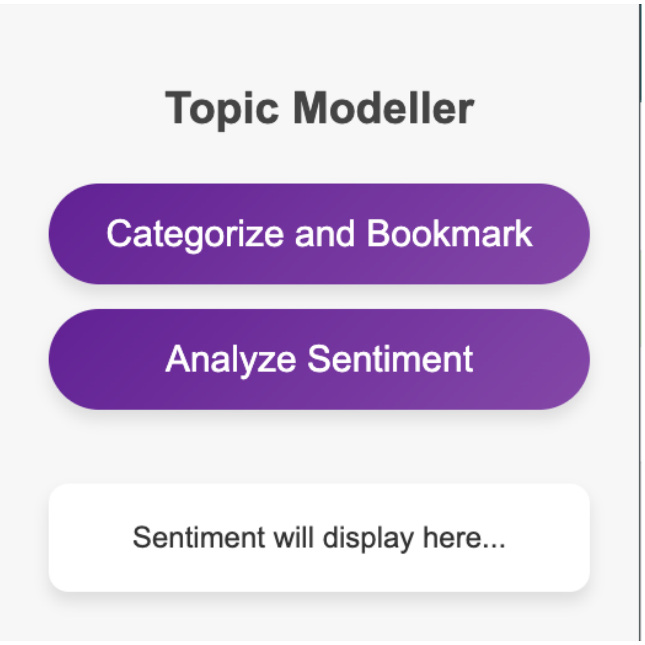


Figure 4: Developed Google Extension user interface

3.1 Pseudo Code

The Pseudo code for the workflow is as below. Each step is performed synchronously.

Step 1 : Scrape the webpage content.

–When user clicks on the Categorize and Bookmark button or the sentiment analysis button, make an api call to the backend server with the content from the webpage.

Step 2 : If Categorize and Bookmark is selected:

– Clean the scraped data and pass it to pre-trained custom LDA model.

– Get the resultant topic the webpage needs to be categorized into.

– Return the topic to the front-end server.

– Save the webpage as a bookmark in the respective topic folder.

Else if Sentiment Analysis is selected:

– Clean the scraped data and pass it to pre-trained Sentiment Analyser model.

– Get the resultant emotional quotient of the content in terms of Compound score.

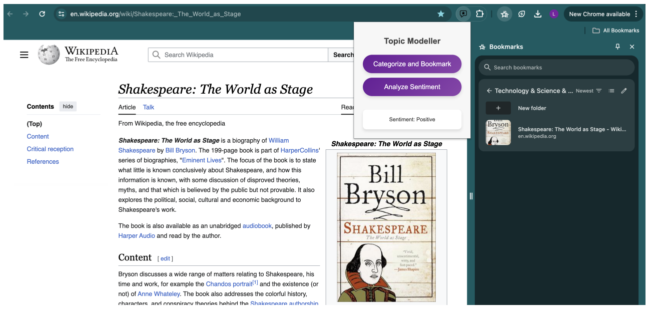
– Return the compound score of the sentiment to the front end server.

– Display the emotion on the extension UI itself.

Step 3 : Reset the Extension state for next webpage or modifying the current webpage bookmarking.

3.2 Implementation Samples

The Google extension implementation and samples are represented in Figures 5 and 6. In Figure 5, we can see that our tool has categorized a Wikipedia page on Shakespeare to be of a positive sentiment and bookmarked it into Technology and Education folder. In Figure 6, a website containing bad movie reviews in accurately categorized into Entertainment and lifestyle folder. Since the webpage is about bad movies, it is expected to have negative emotion to an extent. Our sentiment analyser accurately and efficiently identifies the hidden emotion and classifies the page as negative.



Shakespeare: The World as Stage is a biography of William Shakespeare by Bill Bryson. The 199-page book is part of HarperCollins' series of biographies, "Eminent Lives". The focus of the book is to state what little is known conclusively about Shakespeare, and how this information is known, with some discussion of disproved theories, myths, and that which is believed by the public but not provable. It also explores the political, social, cultural and economic background to Shakespeare's work.

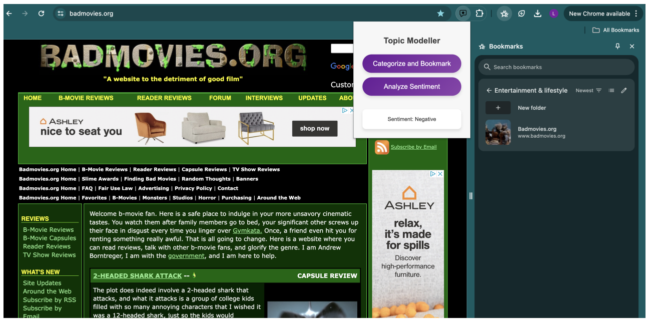
The book is also available as an unabridged audiobook, published by Harper Audio and read by the author.

Bryson discusses a wide range of matters relating to Shakespeare, his time and work, for example the Chandos portrait[1] and the existence (or not) of Anne Whateley. The book also addresses the colorful history, characters, and conspiracy theories behind the Shakespeare authorship

Figure 5: Using the developed tool on Wikipedia webpage about Shakespeare.

4 EVALUATION AND RESULTS

The results of our performance evaluation test are reported in Table 1. The model achieved a high overall accuracy in correctly categorizing web pages based on their content. 94% overall accuracy across all tested web pages. Looking at each main topic separately we observe that each topic has a minimum 90% match rate. Business & Politics: 94.34% accuracy. Out of 53 web pages, 50 were correctly matched to the topic. Entertainment & Lifestyle has 95.24% accuracy. Out of 21 web pages, 20 were correctly matched. Technology & Science: 92.31% accuracy. Out of 26 web pages, 24 were correctly matched. These accuracy metrics are visualized in Figure 7



Welcome b-movie fan. Here is a safe place to indulge in your more unsavory cinematic tastes. You watch them after family members go to bed, your significant other screws up their face in disgust every time you linger over Gymkata. Once, a friend even hit you for renting something really awful. That is all going to change. Here is a website where you can read reviews, talk with other b-movie fans, and glorify the genre. I am Andrew Borntreger, I am with the government, and I am here to help.

The plot does indeed involve a 2-headed shark that attacks, and what it attacks is a group of college kids filled with so many annoying characters that I wished it was a 12-headed shark, just so the kids would

Figure 6: Using the developed tool on Bad movie reviews website.

High accuracy rates indicate robust topic modeling capabilities, particularly in differentiating content types effectively. Areas for Improvement: Minor mismatches suggest potential refinement in model training or adjustment in topic definitions might further enhance accuracy.

Topic	Match	Number of Webpages	Total Number of Webpages
Business & Politics	Match	50	53
	Not Match	3	
Entertainment & lifestyle	Match	20	21
	Not Match	1	
Technology & Science	Match	24	26
	Not Match	2	
Total	Match	94	100
	Not Match	6	

Table 1: Topic Modelling Bookmarker performance on 100 sample websites

5 CONCLUSION AND FUTURE SCOPE

By integrating the LDA model and sentiment analysis, our tool goes beyond traditional bookmarking, enabling users to categorize bookmarks based on both content and emotional tone. This innovative approach enhances users' ability to organize and retrieve information efficiently, addressing the challenges posed by vast amounts of data in the digital age. Our system prioritizes user interaction, offering immediate feedback on the topic and sentiment of bookmarked pages. This user-centric design ensures a seamless and intuitive experience, empowering users to manage large volumes of information with ease and confidence.

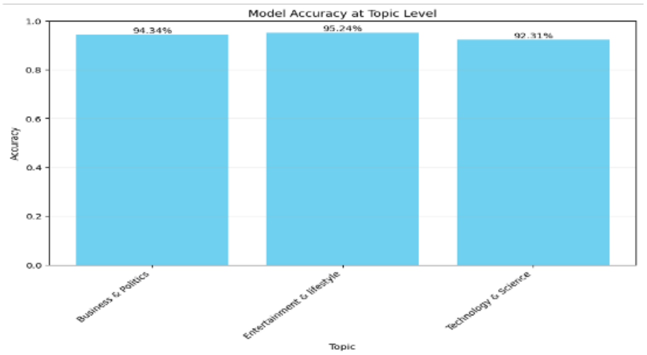


Figure 7: Model Accuracy across the three major bookmark categories

As part of our commitment to continuous improvement, we aim to refine the LDA model and sentiment analysis features further. Leveraging advanced language models (LLMs, openai, Gemini, etc), we seek to enhance the accuracy and granularity of topic generation, ensuring that our bookmarking tool remains at the forefront of efficiency and functionality in information management. A comparative analysis of the efficiency of these models for this use case, and potential feasibility to combine the models for adding advanced features to this tool is a promising developmental scope. We aim to continue our efforts in this direction, incorporating more features to our tool, to further enhance the user experience and tool usability.

6 CONTRIBUTIONS

Lahari Anne: Led the frontend development of the Chrome extension, focusing on creating an intuitive user interface design. Implemented the topic modeling functionality, utilizing Gensim-powered LDA algorithms for content analysis. Ensured seamless integration of frontend components with backend functionalities for a cohesive user experience.

Sanjay Raj Aerra: Spearheaded the backend development efforts, responsible for implementing sentiment analysis algorithms. Orchestrated the setup of the backend server infrastructure, ensuring robustness and scalability. Integrated advanced sentiment analysis techniques into the system to provide users with insights into the emotional tone of bookmarked content.

Sasi Pavan: Played a pivotal role in integrating frontend and backend components, ensuring smooth communication and data exchange. Led comprehensive testing and debugging efforts to identify and resolve any issues or inconsistencies. Coordinated user testing sessions to gather feedback and insights for further optimization, contributing to the refinement of the final product.

7 APPENDICES

Similar to the t-SNE clustering distribution shown in Figure 2, below Figure 8 is a t-SNE clustering we performed for the first 10 topics. This exercise is performed to ensure that the model trained can clearly distinguish between the topics. This ability of the model to distinguish between topics in turn points to the efficiency of the model in accurately categorizing the the content of the web page.

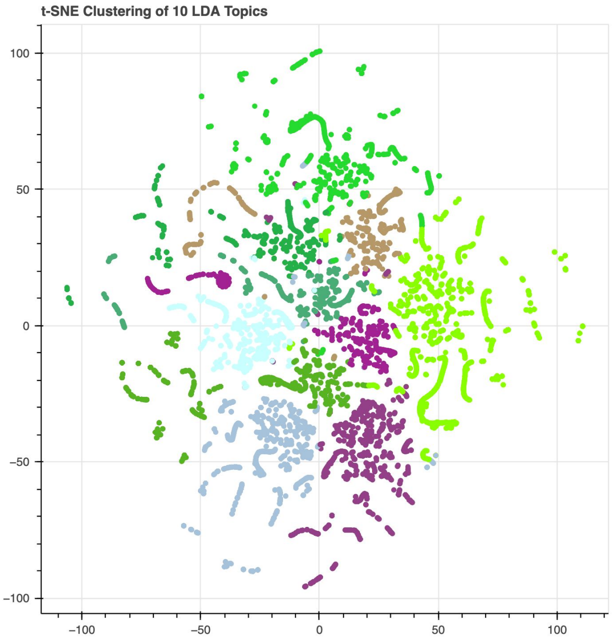


Figure 8: t-SNE clustering plot for first 10 topics of the trained LDA model

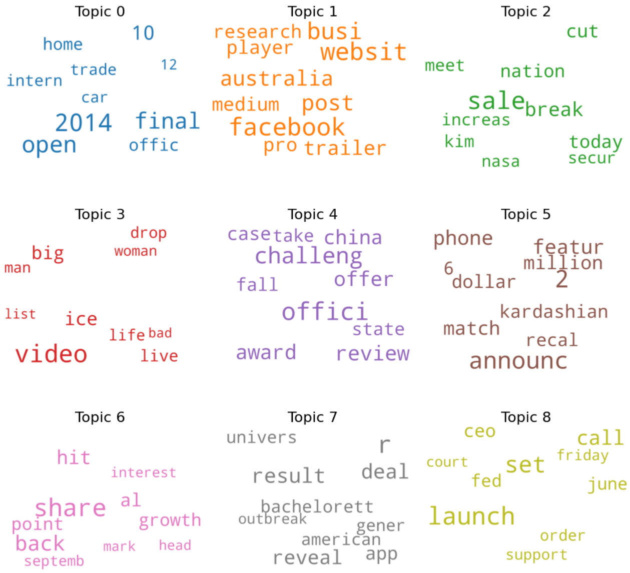


Figure 9: Topic clouds for first 10 topics from the LDA model

The trained LDA model was configured to output 20 topics, each with 10 words. Figure 9 represents the topic cloud for the first 10 topics resulted from the trained LDA model. Each of these clouds represent the 10 keywords that have a high probability in the respective topics.

The source code to this Google extension tool can be found at GitHub repository [8].

ACKNOWLEDGMENTS

We express our sincere gratitude to our CS 510 course staff for providing us with the invaluable opportunity to embark on this project. Their guidance, encouragement, and expertise have been instrumental in shaping our understanding and approach throughout the course and the development of this tool. We are also very grateful for the opportunity to present our development work in the demo conference session held, which gave us exposure to other developmental efforts in this domain. This conference also opened doors for future collaboration with fellow enthusiasts.

8 SOURCE CODE AND PRESENTATION VIDEO

The source code can be found at Reference [8] and the corresponding link is below.

https://github.com/laharianne/topic_modelling_bookmarker
Hyperlink : Github link

The video presentation is uploaded in the Google Drive which is publicly available. The link to this is below which is also referred in Reference [9].

https://drive.google.com/file/d/1ay-p4yosUsFsk5UCWsPLhQ 6kXbdPIA y/view?usp=sharing

Hyperlink : Google Drive Link

REFERENCES

[1] Vimala Balakrishnan and Ethel Lloyd-Yemoh. 2014. Stemming and lemmatization: A comparison of retrieval performances. (2014).

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

[3] Shihab Elbagir and Jing Yang. 2019. Twitter sentiment analysis using natural language toolkit and VADER sentiment. In *Proceedings of the international multiconference of engineers and computer scientists*, Vol. 122. sn.

[4] Akash Gupta. [n. d.]. *MachineHack News Category Dataset.* https://www.kaggle.com/datasets/akash14/news-category-dataset

[5] Nitin Hardeniya, Jacob Perkins, Deepti Chopra, Nisheeth Joshi, and Iti Mathur. 2016. *Natural language processing: python and NLTK.* Packt Publishing Ltd.

[6] LouisKitLungLaw. [n. d.]. *UCI News Aggregator Dataset With Content.* https://www.kaggle.com/datasets/louislung/uci-news-aggregator-dataset-with-content

[7] Rishabh Misra. [n. d.]. *HuffPost News Category Dataset.* https://www.kaggle.com/datasets/rmisra/news-category-dataset

[8] Lahari Anne; Sanjay Raj; Sasi Pavan. 2014. *Topic Modelling Bookmarker source github repository.* https://github.com/laharianne/topic_modelling_bookmarker

[9] Lahari Anne; Sanjay Raj; Sasi Pavan. 2024. *Topic Modelling Bookmarker video presentation.* https://drive.google.com/file/d/1ay-p4yosUsFsk5UCWsPLhQ6kXbdPIA-y/view?usp=sharing

[10] Rachael Tatman. [n. d.]. *State of the Union Corpus (1790 - 2018).* https://www.kaggle.com/datasets/rtatman/state-of-the-union-corpus-1989-2017

[11] Banuprakash V. [n. d.]. *News Articles Classification Dataset for NLP and ML.* https://www.kaggle.com/datasets/banuprakashv/news-articles-classification-dataset-for-nlp-and-ml