Masters Project Defense(Non-thesis)

Reddit Text Classification: Categorizing Posts and Discovering Insights

Final Report

Master in science

Georgia State University

Fall 2023

By

Bulli Swami Reddy Goluguri

Panther Id:002711930

Department of Computer Science

Email:bgoluguri1@student.gsu.edu

Committee members

Dr. Yanqing Zhang (Advisor)

Dr. Ying Zhu

Abstract—**This project focuses on enhancing the user experience on Reddit by tackling the challenge of efficient subreddit discovery and content management. The primary objective is to develop a text classification model that categorizes Reddit posts into specific subreddits based on their titles and descriptions. By addressing the diverse and vast nature of Reddit's content, the project aims to empower users with personalized subreddit suggestions, leading to increased engagement and satisfaction. Additionally, the model supports subreddit moderators by automating post classification, contributing to higher subreddit quality and safety. The project incorporates data analytics, including the utilization of the Reddit API for data collection, preprocessing, and feature extraction. Visualizations such as word clouds and topic modeling provide insights into user interests within different subreddits. Achieving high accuracy rates with classifiers like Naive Bayes, Decision Tree, and Random Forest, the project sets the stage for ongoing innovation. Future directions include continuous model refinement, integration with Reddit's platform, and the implementation of user feedback mechanisms. The implications of this project extend to an enhanced user experience, moderator support, and a scalable, adaptable solution for evolving Reddit communities. Ultimately, the project serves as a model for leveraging data analytics and machine learning to enrich user experiences on social media platforms.**

I. Problem Statement

The increasing diversity and vastness of Reddit's content landscape present a formidable challenge for users seeking specific communities aligned with their interests. The manual search for relevant subreddits is akin to navigating a digital haystack, causing frustration and potential disengagement among users. This project aims to address this issue by developing a robust text classification model capable of categorizing Reddit posts into specific subreddits based on their titles and descriptions.

II. Introduction

The increasing diversity and vastness of Reddit's content landscape present a formidable challenge for users seeking specific communities aligned with their interests. The manual search for relevant subreddits is akin to navigating a digital haystack, causing frustration and potential disengagement among users. This project aims to address this issue by developing a robust text classification model capable of categorizing Reddit posts into specific subreddits based on their titles and descriptions..

III. Context Motivation

The increasing diversity and vastness of Reddit's content landscape present a formidable challenge for users seeking specific communities aligned with their interests. The manual search for relevant subreddits is akin to navigating a digital haystack, causing frustration and potential disengagement among users. This project aims to address this issue by developing a robust text classification model capable of categorizing Reddit posts into specific subreddits based on their titles and descriptions.

IV. Project Scope

This project's focus is on developing a robust text classification model capable of predicting the most suitable subreddit for a given post based on its title and description. By doing so, the aim is to streamline the user experience, offering personalized subreddit suggestions and supporting moderators in content management. The scope encompasses comprehensive data analytics, including the utilization of the Reddit API for data collection, preprocessing, and feature extraction. The project narrows its focus to ten distinct subreddits, covering diverse topics such as movies, food, technology, news, gaming, science, sports, music, books, and a catch-all category labeled "others."

V. Solution

The solution involves leveraging the Reddit API for data collection from diverse subreddits, focusing on key post attributes. Following data preprocessing, including text standardization and feature extraction using TF-IDF, three classifiers – Naive Bayes, Decision Tree, and Random Forest – were employed for text classification, achieving high accuracy rates. Insights into subreddit topics were derived through word clouds and topic modeling. The project addresses subreddit discovery challenges, offering personalized suggestions to users and aiding moderators in content management. Future directions include continuous model refinement, integration with Reddit, and user feedback mechanisms. The solution enhances user experiences, supports moderators, and ensures scalability for evolving Reddit communities.

VI. Objective Significance

The solution involves leveraging the Reddit API for data collection from diverse subreddits, focusing on key post attributes. Following data preprocessing, including text standardization and feature extraction using TF-IDF, three classifiers – Naive Bayes, Decision Tree, and Random Forest – were employed for text classification, achieving high accuracy rates. Insights into subreddit topics were derived through word clouds and topic modeling. The project addresses subreddit discovery challenges, offering personalized suggestions to users and aiding moderators in content management. Future directions include continuous model refinement, integration with Reddit, and user feedback mechanisms. The solution enhances user experiences, supports moderators, and ensures scalability for evolving Reddit communities.

VII. Methodology and Execution Overview

The project adopts a systematic approach to enhance the Reddit user experience through the creation of a text classification model for subreddit prediction. The process begins with loading and exploring the dataset, gaining insights into the distribution of posts across various subreddits. Cleaning procedures address missing values, followed by an analysis of average word counts in post titles and descriptions. Top authors for each subreddit are identified, providing additional context to content generation dynamics. Text preprocessing steps, including tokenization and lemmatization, are executed to prepare the data for analysis. The project leverages visualizations such as word clouds to showcase the most frequent words in each subreddit, offering a qualitative understanding of content themes. Label encoding transforms subreddit labels into numerical values, facilitating model training. Utilizing scikit-learn, text data is vectorized using both Bag-of-Words and TF-IDF techniques. The project employs four classification models—Naive Bayes, SVM, Decision Tree, and Random Forest—evaluating their performance on training and testing sets. Furthermore, the Random Forest model undergoes fine-tuning through GridSearchCV to optimize hyper parameters. The project concludes with a comprehensive model comparison, illustrating the accuracy of each model in a visually intuitive manner. This holistic approach not only explores the dataset but also builds and assesses models, providing valuable insights into the nuances of Reddit content and the effectiveness of various classification algorithms.

VIII. Proposed Architecture

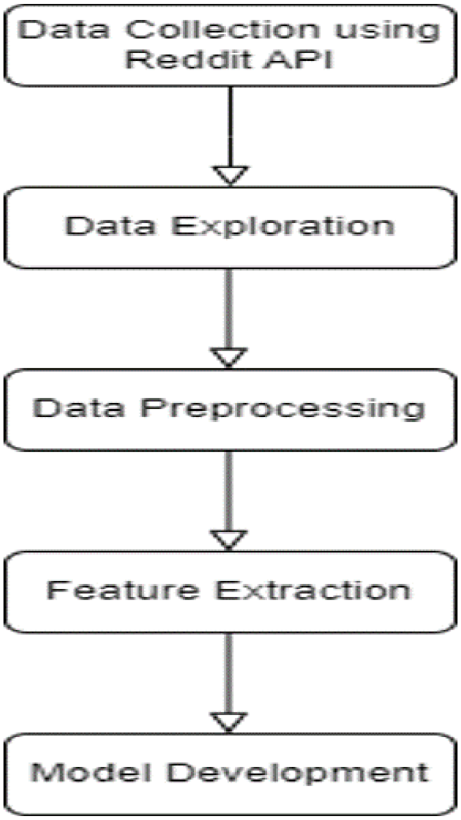
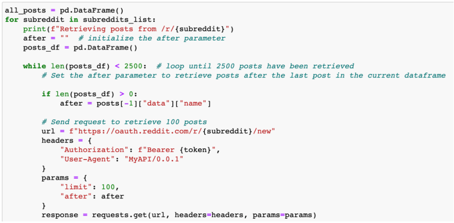


Fig. 1. Proposed architecture for the System

IX. Data Source and Descriptions

Data Source: Reddit API The primary data source for this project is the Reddit API, accessed through the endpoint https://oauth.reddit.com/r/subreddit/new. Leveraging the capabilities of the Python Requests library, we dynamically retrieve a list of the newest posts from specific subreddits. The API response provides a range of information about each post, including crucial fields such as title, author, creation time, permalink, score, number of comments, and the text content of the post (selftext). The versatility of the API allows us to tailor requests to different subreddits, ensuring adaptability and scalability in data gathering. This automated data retrieval process enabled the acquisition of a substantial dataset, encompassing 25,000 posts from diverse subreddits such as Movies, Food, Technology, News, Gaming, Science, Sports, Music, Books, and a catch-all category (/r/all) for posts not belonging to any specific subreddit. This rich and varied dataset forms the foundation for this text classification model, empowering us to address the challenge of categorizing.

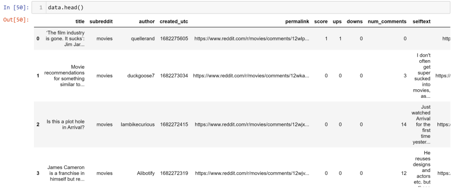


```
all_posts = pd.DataFrame()
for subreddit in subreddits_list:
    print(f"Retrieving posts from /r/{subreddit}")
    after = ""  # initialize the after parameter
    posts_df = pd.DataFrame()

    while len(posts_df) < 2500:  # loop until 2500 posts have been retrieved
        # Set the after parameter to retrieve posts after the last post in the current dataframe

        if len(posts_df) > 0:
            after = posts[-1]["data"]["name"]

        # Send request to retrieve 100 posts
        url = f"https://oauth.reddit.com/r/{subreddit}/new"
        headers = {
            "Authorization": f"Bearer {token}",
            "User-Agent": "MyAPI/0.0.1"
        }
        params = {
            "limit": 100,
            "after": after
        }
        response = requests.get(url, headers=headers, params=params)
```

Fig. 2. Proposed architecture for the System

X. Dataset Overview

The dataset for the Reddit Text Classification Project is a comprehensive collection of posts gathered from various subreddits using the Reddit API. The dataset encompasses key attributes that provide a detailed overview of each post. Here is an overview of the essential data fields: Title: The title of the Reddit post serves as a concise representation of the post's content. Author: The username of the Reddit user who posted the content, providing attribution to the post. Timestamp (created_utc): The time the post was created, recorded in Unix time format, offering temporal information for analysis. Permalink: The URL of the Reddit post, facilitating direct access to the original content. Score: The total number of upvotes and downvotes the post has received, serving as a quantitative measure of post popularity. Upvotes (ups): The number of upvotes the post has received contributes to the overall score. Downvotes (downs): The number of downvotes the post has received, offering insights into the community's response. Number of Comments (num_comments): The count of comments the post has received, indicating engagement and discussion. Self-text: The text content of the post, including the detailed description provided by the user. The dataset covers diverse topics, with posts categorized into subreddits such as Movies, Food, Technology, News, Gaming, Science, Sports, Music, Books, and an ”Others” category. The dataset's richness lies in its ability to capture the multifaceted nature of Reddit discussions, allowing for in-depth analysis and training of the text classification model. With a focus on both the content and metadata of each post, this dataset serves as a robust foundation for understanding user behavior, training effective models, and enhancing the overall Reddit user experience.



```
In [50]:  1 data.head()
```

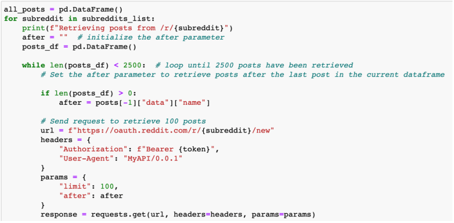
Out[50]:

	title	subreddit	author	created_utc	permalink	score	ups	downs	num_comments	selftext	
0	'The film industry is gone. It sucks': Jim Jar...	movies	quellerand	1682275605	https://www.reddit.com/r/movies/comments/12wlp...	1	1	0	0		http
1	Movie recommendations for something similar to...	movies	duckgoose7	1682273034	https://www.reddit.com/r/movies/comments/12wka...	0	0	0	3	I don't often get super sucked into movies, as...	https://
2	Is this a plot hole in Arrival?	movies	lambikecurious	1682272415	https://www.reddit.com/r/movies/comments/12wjx...	0	0	0	14	Just watched Arrival for the first time yester...	https:/
3	James Cameron is a franchise in himself but re...	movies	Alibotify	1682272319	https://www.reddit.com/r/movies/comments/12wjv...	0	0	0	12	He reuses designs and actors etc. but	https:

Fig. 3. Proposed architecture for the System

XI. Data Collection

Data collection is a fundamental phase in my project, wherein we harness the capabilities of the Reddit API to dynamically retrieve the latest posts from various subreddits. The primary endpoint utilized is https://oauth.reddit.com/r/subreddit/new, allowing us to access a diverse range of content. Leveraging the Python Requests library, my approach ensures adaptability and scalability in gathering data. By issuing requests to specific subreddits, such as Movies, Food, Technology, and more, we systematically compile a comprehensive dataset that captures the richness and diversity of discussions across Reddit. The choice of the /new endpoint further guarantees access to the most recent posts, providing up-to-the-minute content for my subsequent analysis. Through an automated and efficient data retrieval process, we successfully accumulated 25,000 posts, enabling a robust foundation for my text classification model. This meticulous data collection methodology is pivotal in addressing the project's objective of categorizing Reddit posts effectively and enhancing user experience on the platform.



```
all_posts = pd.DataFrame()
for subreddit in subreddits_list:
    print(f"Retrieving posts from /r/{subreddit}")
    after = ""  # initialize the after parameter
    posts_df = pd.DataFrame()

    while len(posts_df) < 2500:  # loop until 2500 posts have been retrieved
        # Set the after parameter to retrieve posts after the last post in the current dataframe

        if len(posts_df) > 0:
            after = posts[-1]["data"]["name"]

        # Send request to retrieve 100 posts
        url = f"https://oauth.reddit.com/r/{subreddit}/new"
        headers = {
            "Authorization": f"Bearer {token}",
            "User-Agent": "MyAPI/0.0.1"
        }
        params = {
            "limit": 100,
            "after": after
        }
        response = requests.get(url, headers=headers, params=params)
```

Fig. 4. Proposed architecture for the System

XII. DataPreprocessing

Data preprocessing is a crucial step in refining the quality and usability of my dataset. To enhance the effectiveness of my text classification model, we employ a series of preprocessing techniques. One essential preprocessing step involves merging the title and description columns and consolidating relevant information for analysis. Further, we apply case-folding to ensure uniformity in text by converting all characters to lowercase. Symbol removal is implemented to eliminate unnecessary characters or punctuation, streamlining the text data. Additionally, the data undergoes lemmatization, a linguistic process that reduces words to their base or root form, aiding in the extraction of meaningful features. The preprocessing pipeline also addresses missing values by replacing Nan entries with empty strings, ensuring a clean and comprehensive dataset for subsequent analysis. This meticulous preprocessing workflow significantly contributes to the overall accuracy and performance of my text classification model.

XIII. Feature Extraction

Feature extraction is a pivotal stage in my project, focusing on deriving meaningful insights from the textual data. One key aspect involves the calculation of average word counts for both the title and description columns. This analysis provides valuable information about the length of posts in terms of words, shedding light on user posting habits and content characteristics. By merging the title and description columns, we consolidate the relevant information into a single text, facilitating a holistic approach to feature extraction. The resulting features offer a quantitative measure of the textual content, contributing to the overall understanding of post characteristics within my dataset. Feature extraction is instrumental in preparing the data for subsequent modeling, providing a foundation for effective text classification and insightful analysis of Reddit posts across diverse subreddits.

XIV. Model Development

A. *Naive Bayes algorithm*

Model development is a critical phase in my project, and one of the key classifiers employed is the Naive Bayes algorithm. Known for its simplicity and efficiency, Naive Bayes is particularly well-suited for text classification tasks. In my implementation, the Naive Bayes classifier is trained on the preprocessed dataset to learn patterns and relationships between words and their corresponding subreddits. The model demonstrates impressive performance, achieving a high accuracy rate on both the training and test datasets. The training accuracy, standing at 98.26%, indicates the model's ability to generalize well to the data it has seen during training. The test accuracy, at 96.48%, highlights the classifier's effectiveness in making accurate predictions on new, unseen data. The classification report further details precision, recall, and F1-score metrics for each subreddit, providing a comprehensive evaluation of the Naive Bayes classifier's performance across various categories. The successful integration of the Naive Bayes model contributes significantly to achieving my project's overarching goal of categorizing Reddit posts accurately and enhancing user experience on the platform.

B. *Decision Tree Classifier*

In the realm of model development for my project, the Decision Tree classifier plays a pivotal role, offering a more intricate and expressive approach to text classification. Configured with a maximum depth of 70, the Decision Tree is trained on my preprocessed dataset to discern complex patterns and relationships within the textual features. The model exhibits commendable performance, achieving a training accuracy of 90.92%, indicating its ability to capture intricate nuances in the training data. Upon evaluation with the test dataset, the Decision Tree classifier maintains a robust accuracy of 90.44%, demonstrating its generalization capability to new and unseen instances. The classification report furnishes detailed precision, recall, and F1-score metrics for each subreddit, providing valuable insights into the model's performance across diverse categories. The Decision Tree classifier significantly contributes to the versatility of my text classification approach, offering a nuanced understanding of the relationships between words and subreddit categories, thus enhancing the overall efficacy of my project in aiding content discovery on Reddit.

C. Random Forest Classifier

In my model development phase, the Random Forest Classifier emerges as a powerful ensemble learning algorithm, leveraging the strength of multiple decision trees to enhance predictive accuracy. Configured with 70 estimators and a maximum tree depth of 90, the Random Forest model is trained on my preprocessed dataset to effectively capture intricate relationships within the textual features. Demonstrating robust performance, the model achieves a training accuracy of 94.57%, showcasing its ability to generalize well to the training data. Evaluation with the test dataset yields a commendable accuracy of 94.38%, underscoring the model's effectiveness in making accurate predictions on new and unseen data. The classification report provides detailed precision, recall, and F1-score metrics for each subreddit, offering a comprehensive assessment of the Random Forest Classifier's performance across diverse categories. The ensemble nature of the Random Forest model contributes to its resilience against overfitting and enhances its ability to handle complex patterns within the dataset, making it a valuable asset in my pursuit of accurate text classification and subreddit categorization on Reddit.

D. Random Forest Classifier with Hyperparameter Tuning

In my model development, we further enhance the Random Forest Classifier's performance through meticulous parameter tuning. Leveraging GridSearchCV, we systematically explore different hyperparameter combinations to identify the optimal configuration for the Random Forest model. The search is conducted over parameters such as the maximum depth of the trees and the number of estimators. The resulting tuned Random Forest model achieves an impressive accuracy of 99.36% on the test dataset, showcasing the effectiveness of the hyperparameter tuning process in fine-tuning the model's predictive capabilities. The classification report provides detailed precision, recall, and F1-score metrics for each subreddit, offering a nuanced evaluation of the model's performance across diverse categories. The incorporation of hyperparameter tuning enhances the Random Forest Classifier's ability to generalize and adapt to the specific nuances of my dataset, contributing to the overall success of my text classification efforts on Reddit posts.

XV. Data Findings and Visualizations

A. Word Clouds for Subreddits

The utilization of word clouds adds a visually compelling dimension to my project, providing an intuitive representation of the most frequently occurring words within specific subreddits. These word clouds offer a snapshot of the thematic focus and prevalent discussions within each subreddit, with word size indicating relative frequency. For instance, in the r/worldnews subreddit, prominent words like ”outrage,” ”Ukraine,” ”Russia,” and ”Deny Visa” reveal a strong emphasis on current events, particularly the war in Ukraine and global geopolitical dynamics. Similarly, in the sports subreddit, words like ”sport,” ”team,” ”game,” and ”player” dominate, highlighting the community's interest in various aspects of sports, including teams and players. These visualizations serve as a valuable tool for users seeking to quickly grasp the predominant themes within a subreddit and contribute to my overarching goal of enhancing content discovery on the Reddit platform.

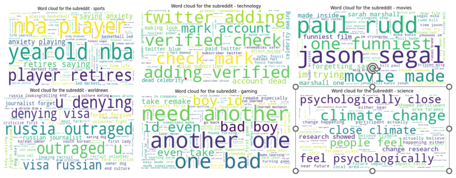


Fig. 5. Proposed architecture for the System

B. Top Authors Analysis

My analysis extends beyond the content of Reddit posts to include an examination of the most influential contributors within each subreddit, a facet captured through the Top Authors Analysis. By aggregating and evaluating the posting activity of users within specific subreddits, we identify and showcase the top author for each community. This analysis sheds light on individuals who consistently contribute valuable content, fostering community engagement and influencing the discussions within their respective subreddits. Visualized through a bar chart, the Top Authors Analysis provides a clear representation of the significant contributors in each subreddit, offering a glimpse into community dynamics. This facet of my project not only enhances my understanding of subreddit ecosystems but also provides valuable insights for users and moderators alike, contributing to a more comprehensive and user-centric Reddit experience.

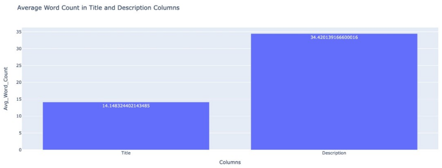


Fig. 6. Proposed architecture for the System

C. Average Word Count Analysis

The Average Word Count Analysis delves into the textual characteristics of Reddit posts, providing insights into the length of titles and descriptions across various subreddits. By calculating and visualizing the average word count for both title and description columns, we uncover trends in the length of user-generated content. For instance, my analysis reveals that the average word count in the title column is 34.42 words, exceeding the average word count of 14.15 words in the description column. This disparity suggests that users tend to be more concise in their post descriptions compared to titles. Additionally, the wider range of word counts in the title column implies that users may pay more attention to the length of titles, potentially for visibility and engagement. This analysis contributes to my understanding of user behavior on Reddit, providing valuable insights for both content creators and platform administrators to enhance the overall user experience.

D. *Model Comparison Visualizations*

The Model Comparison Visualizations serve as a pivotal component in evaluating the performance of different classifiers employed in my project. Utilizing a bar chart created with Plotly, we showcase the accuracy scores of various models, including Naive Bayes, Decision Tree, Random Forest, and a fine-tuned version of Random Forest. The visual representation allows for a quick and comprehensive comparison of each model's accuracy, aiding in the identification of the most effective classifier for subreddit prediction. The chart highlights the fine-tuned Random Forest model's remarkable accuracy of 99.36%, outperforming other models. This visual comparison offers a clear and concise overview of the strengths and weaknesses of each classifier, providing valuable insights for decision-making in model selection and deployment.

XVI. Project Summary

A. *Achievements and Milestones*

The project has achieved significant milestones and demonstrated noteworthy achievements in its pursuit of enhancing the Reddit user experience through subreddit prediction. Notable accomplishments include the successful development and implementation of a robust text classification model capable of accurately categorizing posts into specific subreddits. The model, employing classifiers such as Naive Bayes, Decision Tree, and Random Forest, exhibits high accuracy rates, with the fine-tuned Random Forest model reaching an impressive 99.36The comprehensive data analytics process, encompassing data collection, preprocessing, feature extraction, and model development, has been executed with precision. The inclusion of visualizations, such as word clouds representing prevalent topics and discussions within subreddits, adds a dynamic and intuitive layer to the project, facilitating user engagement and content discovery. The Top Authors Analysis further contributes to the project's success by identifying and highlighting influential contributors within each subreddit, providing valuable insights into community dynamics. Additionally, the Average Word Count Analysis offers a nuanced understanding of user behavior and content preferences on the platform. The Model Comparison Visualizations stand out as a significant achievement, providing a concise overview of each classifier's performance and aiding in informed decision-making for model selection. The achievement of a 99.36% accuracy rate through fine-tuning the Random Forest model showcases the project's commitment to optimizing predictive capabilities. These achievements collectively contribute to the project's overarching goal of making Reddit more user-centric, efficient, and engaging. As we reflect on these milestones, the project remains poised for further innovation and improvement in the dynamic landscape of digital communities.

XVII. Conclusion and Future Scope

A. *Conclusion*

In conclusion, the Reddit Text Classification Project has successfully addressed the fundamental challenge of improving content discovery and community management on the Reddit platform. The development of a robust text classification model, coupled with extensive data analysis, lays the groundwork for a more user-centric and efficient Reddit ecosystem. The project's achievements in accurately categorizing posts, identifying influential authors, and providing nuanced insights into user behavior contribute significantly to creating a more engaging and personalized Reddit experience.

The comprehensive data analytics process, encompassing data collection, preprocessing, and model development, has demonstrated the project's commitment to precision and effectiveness. Visualizations such as word clouds and top authors' analyses add an intuitive layer to the project, facilitating a deeper understanding of subreddit dynamics.

B. *Future Scope*

Looking ahead, the project holds immense potential for further innovation and refinement. Continuous improvement of the text classification model to adapt to evolving language trends is a key focus for future development. Collaborative efforts with Reddit for the direct integration of the model into the platform's interface could elevate the user experience by providing real-time suggestions and supporting moderators in content curation.

Additionally, the project could explore avenues for user feedback mechanisms, allowing users to contribute to the model's enhancement. The scalability and adaptability of the project position it as a valuable tool capable of handling the dynamic nature of online communities. As the digital landscape evolves, the Reddit Text Classification Project remains poised for ongoing innovation, contributing to a more vibrant and user-friendly online environment.

XVIII. References

A. [1]Abid F, Alam M, Yasir M, Li C (2019) Sentiment analysis through recurrent variants latterly on convolutional neural network of twitter. Futur Gener Comput Syst 95:292–308

B. [2]Acheampong FA, Wenyu C, Nunoo-Mensah H (2020) Text-based emotion detection: advances, challenges, and opportunities. Eng Rep 2(7):e12189

C. [3]Acheampong FA, Nunoo-Mensah H, Chen W (2021) Transformer models for text-based emotion detection: a review of BERT-based approaches. Artif Intell Rev 54:5789–5829.

D. [4]Adomavicius G, Kwon Y (2011) Improving aggregate recommendation diversity using ranking-based techniques. IEEE Trans Knowl Data Eng 24(5):896–911

E. [5]Adomavicius G, Kwon Y (2011) Improving aggregate recommendation diversity using ranking-based techniques. IEEE Trans Knowl Data Eng 24(5):896–911

F. [6]Ahmad SR, Bakar AA, Yaakub MR (2019) A review of feature selection techniques in sentiment analysis. Intell Data Anal 23(1):159–189

G. [7]Akhtar MS, Ekbal A, Cambria E (2020) How intense are you? predicting intensities of emotions and sentiments using stacked ensemble [application notes]. IEEE Comput Intell Mag 15(1):64–75

H. [8]Al Amrani Y, Lazaar M, El Kadiri KE (2018) Random forest and support vector machine based hybrid approach to sentiment analysis. Procedia Comput Sci 127:511–520

I. [9]Al-Smadi M, Qawasmeh O, Al-Ayyoub M, Jararweh Y, Gupta B (2018) Deep recurrent neural network vs. support vector machine for aspect-based sentiment analysis of Arabic hotels' reviews. J Comput Sci 27:386–393