Automated Text Classification

Hansi Koshila Cooray Wijayawarnasooriya
a1919301

The University of Adelaide

4533_COMP_SCI_7417_7717 Applied Natural Language Processing

Lecturer: Dr. Orvila Sarker

Table of Contents

1. Abstract

This project addresses one of NLP's key aspects: Text Classification, and focuses on developing an automated system to categorise Stack Overflow (SO) posts related to NLP questions faced by software engineers. The report details the processes of data collection, preprocessing, graphical representation and categorisation of NLP-tagged posts. A rule-based categorisation was implemented, classifying the posts into 25 distinct categories based on the keywords identified in the title and description of NLP-tagged posts. The system successfully categorised the entire database, providing a structural knowledge base derived from the developer discussion on Stack Overflow.

2. Introduction

Natural Language Processing (NLP) is a subset of Artificial Intelligence that uses Machine Learning techniques to enable computers to understand and interpret human language (1). NLP has become one of the most interesting and engaging fields for software developers as it empowers a range of applications like chatbots, machine learning translations, sentiment analysis and data extraction. However, software developers entering the field or handling new NLP tasks often find it challenging to navigate the complex NLP libraries, techniques and tools. Offline communities like Stack Overflow (SO) provide a valuable opportunity for developers to share such problems(6), discuss solutions and develop their knowledge. Although SO contains a massive NLP knowledge base (More than 21000 posts tagged as NLP), it can be difficult to quickly find helpful solutions to a particular NLP question.

This project addresses the need for a structured approach to efficiently find the best suitable answers for developers' NLP problems, by categorising them under meaningful entities. The main objective is to design and implement an automated system to process Stack Overflow NLP-tagged posts, clean and categorise them into predefined entities using a rule-based approach. These categories represent the common types of questions, popular libraries, specific tasks, and recurring problems encountered by the developers. The ultimate goal is to develop a well-structured, efficient NLP knowledge base system for developers who are seeking assistance to tackle NLP-related tasks.

To achieve this, the project followed the pipeline mentioned below.

- **Data Loading and Initial Cleaning** - Obtain the SO NLP-tagged posts using an API and perform preliminary cleaning (remove HTML tags and code segments of title and description).
- **Text Preprocessing -** Text preprocessing involves handling punctuation, converting to lowercase, tokenisation and removing stop words.
- **Data Visualisation** - Generating a word cloud using the wordcloud library to identify prominent words in the title and description for categorisation.
- **Rule-Based categorisation -** An automated rule-based approach was defined using regular expressions (re) to assign every post in the database to one of the 25 predefined categories. These categories are matched with the keywords identified in the title and description of posts.

3. Data Collection - NLP Posts

The foundation of this project is to generate a dataset comprising NLP-tagged posts from Stack Overflow. The original dataset contained 57000 NLP posts and 21083 accepted posts.

3.1. Methodology - Data Collection

The first step of data acquisition was the Stack Exchange API. Collecting a large dataset from SO without an API is challenging (7). Therefore, an API was created to extract posts efficiently based on specific criteria. However, the API did not provide 20000+ posts in one go. Hence, the data extraction was carried out using manual pagination. Below shows the simple steps of the data collection process :

- Fetch all posts tagged as NLP - Manual pagination - Check each page manually and collect and save data repeatedly. Then, clean the collected data to remove HTML tags.

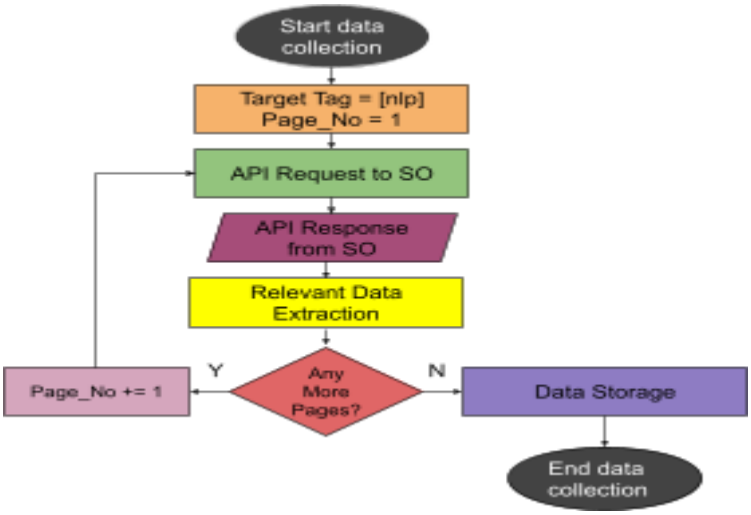

Figure 1. Workflow of retrieving NLP-related posts and issues from Stack Overflow.

- Fetch posts containing the accepted answers - After retrieving all NLP posts, the posts that contain at least one accepted answer were extracted in this stage.
- Save data to Excel tabs - The last step is to save the data in different tabs, original data and processed data, respectively.

3.2. Data Scope and Field

The target dataset for the assessment was over 20000 NLP-tagged posts to ensure a rich overview of NLP discussions. The dataset was collected with the aim of having the following key features.

- Question ID - A unique ID for each post.
- Title - Question title of each post posted by the developers.
- Description - The main body contains an explanation of the question.
- Tags - All tags associated with the post (This includes other tags with 'nlp')
- Accepted Answer - The answer that is marked with a green tick in the description body.

3.3. Data Storage

The collected and processed data were stored in an Excel sheet under above mentioned key features. The sheet contains four tabs :

- All_NLP_Post - All collected NLP posts.
- Accepted_Answer_Posts - Posts with one or more accepted answers
- Cleaned_Title_Descriptions - Preprocessed titles and descriptions with accepted answers.
- Categorized_Posts_simple - Final dataset after categorisation of the posts.

The link to the dataset can be obtained using the following link:-
https://docs.google.com/spreadsheets/d/1-RN7so7A9ArjXd4GO-5uHl7M1qKfaWgG/edit?usp=sharing&ouid=115701530022859255786&rtpof=true&sd=true

4. Text Preprocessing

The row text data obtained from the SO contained noise and irrelevant information for the NLP categorisation task. The posts contained code segments, HTML tags which needed to be removed to obtain the best expected results. Hence, a comprehensive text processing pipeline was applied to title, description and accepted answer columns in the dataset to clean and standardize data.

4.1. Preprocessing Pipeline

The following diagram illustrates the preprocessing pipeline of the project :

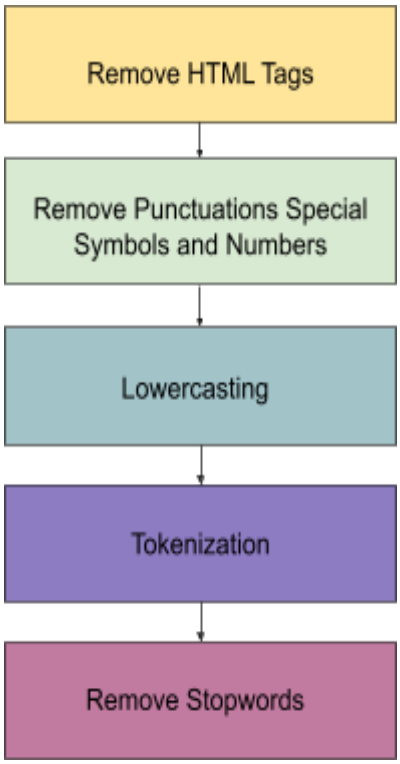

Figure 2. Pre-processing steps of Stack Overflow documents

The first step of the pipeline is to remove HTML tags from the content. The non-alphabetic characters cause inconsistency. Therefore, such characters are removed during the second stage. Next, the texts are converted to lowercase to ensure that variations in capitalisation do not affect the categorisation process. Tokenisation breaks the text string into individual words, providing a list of potential word units. This is helpful for the last stage of the pipeline: removing stopwords. Stopwords are frequently occurring words in a language which carry less semantic meaning for the analysis (2). At last, such words are removed from the corpus, keeping only alphabetic tokens.

4.2. Application and Output

The text preprocessing function, which implements the above text preprocessing pipeline, was applied to the title, description and accepted_answer columns of the dataset. The generated text of the new columns was saved in a separate file tab. Moreover, these token lists were joined into single strings to create a new dataframe (df_processed). This provides clean, normalized text for visualization and categorisation.

5. Data Visualisation

To get a clear insight into the prominent characteristics and word patterns of Stack Overflow data, several visualizations were generated. WordCloud is one of the most significant among them (5). The word cloud was generated using the processed titles and descriptions of the posts. Significantly relevant features of those texts were used to categorise the NLP posts.

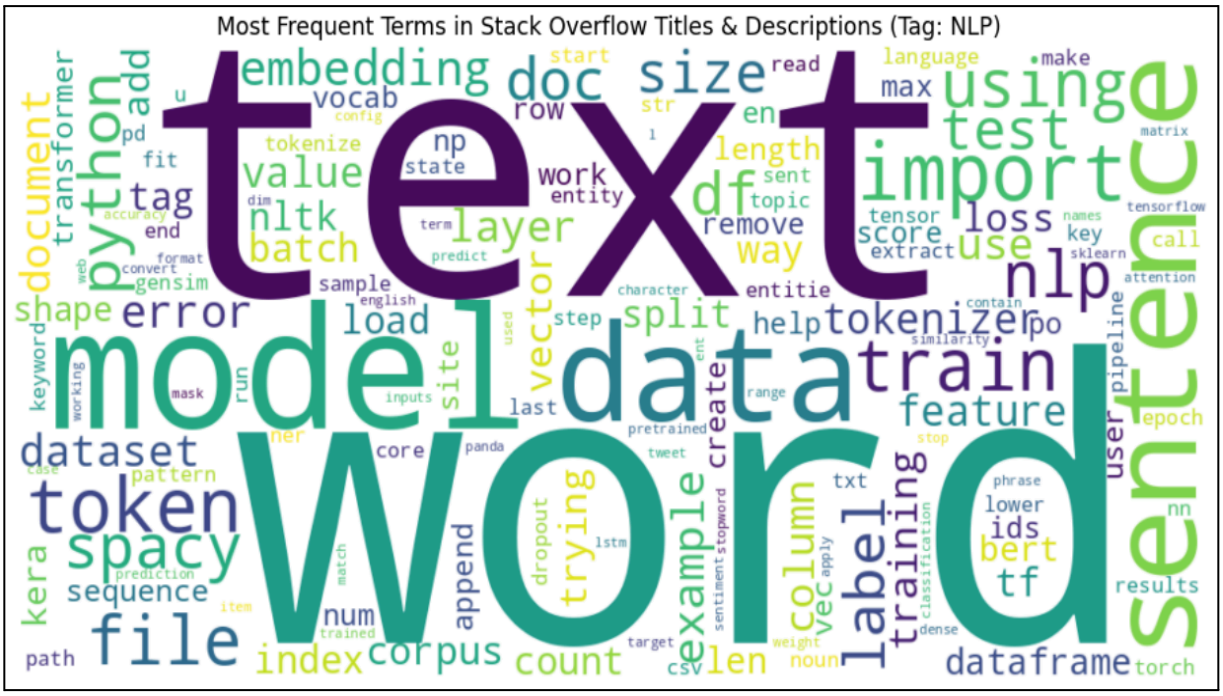

Figure 3. Wordcloud of the most frequent words

The wordcloud contains prominent words like: text, model, data, word, which are central to NLP tasks. Moreover, important concepts such as vector, feature, training, classification, tokenize, bert, transformer, etc., are visible in the wordcloud space. Keras, Spacy, Tensorflow, and Sklearn are some of the major tools of NLP. Action/problem-oriented words like error, train, load, test, fit, and predict represent the problem-solving nature of SO. Furthermore, words like data, file, CSV and corpus highlight the importance of data handling in CSV.

6. Data Categorisation

Data categorisation is the most crucial part of this study. Following the pre-processing stage, the main goal was to implement an automated text categorisation system to classify the Stack Overflow data into meaningful groups depicting types of questions discussed by the developers in the NLP domain.

6.1. Methodology - Data Categorisation

An automated rule-based process was implemented to categorise the titles and descriptions of the NLP-tagged posts using pre-defined regular expressions. The pipeline consists of input cleaned data(processed title and description) to the system, category definition, followed by keyword and pattern matching: searching the text for keywords and regular expression patterns associated with predefined categories. Basically, there are 25 predefined categories.

- ***Library-Specific***: Identifying posts focused on popular libraries.
  Library - spaCy, Library - NLTK, Library - Transformers/HF, Library - Gensim, Library - Scikit-learn, Library - Keras/TF/PyTorch

- ***Task-Specific***: Identifying posts related to common NLP tasks.
  Task - Classification, Task - NER, Task - Embeddings/Vectorisation, Task - Tokenisation, Task - Preprocessing/Cleaning

- ***Problem Typ***e: Grouping posts by the nature of the query.
  Implementation Issues, Conceptual Understanding, Debugging/Errors

- ***Resource Focus:*** Identifying posts related to data, setup, or environments.
  Data/Resources/Setup

- ***Fallback Category:*** A meaningful default category, General NLP Query, was established.

These categories are meaningful as they directly reflect the common areas that the developers discuss on SO. The priority-based rule system was introduced, for example, library-related posts were given more priority than task-related or implementation posts, to minimise overlapping.

6.2. Implementation and Output

The data categorisation pipeline was applied to preprocessed and combined text (title + description) to generate the desired output, following these steps :

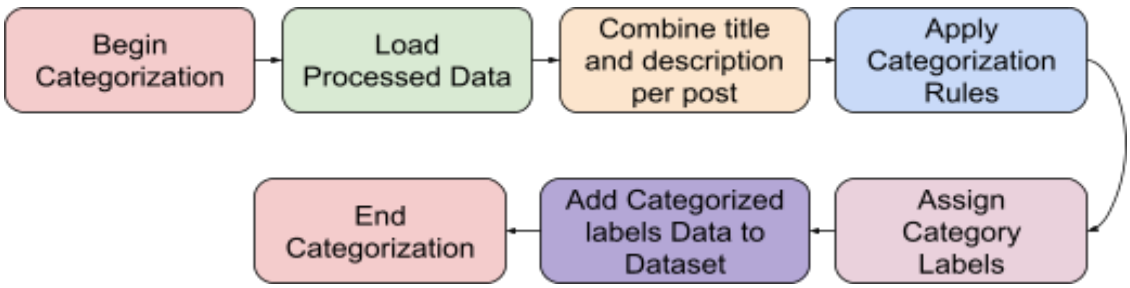

Figure 4. Implementation of Data Categorisation

6.3. Results and Analysis

The application results of the automated categorisation process are demonstrated in the following table.

- Library Categories - 6
- Task Categories - 14
- Specific Check Categories - 4
- Fallback/General Category -1

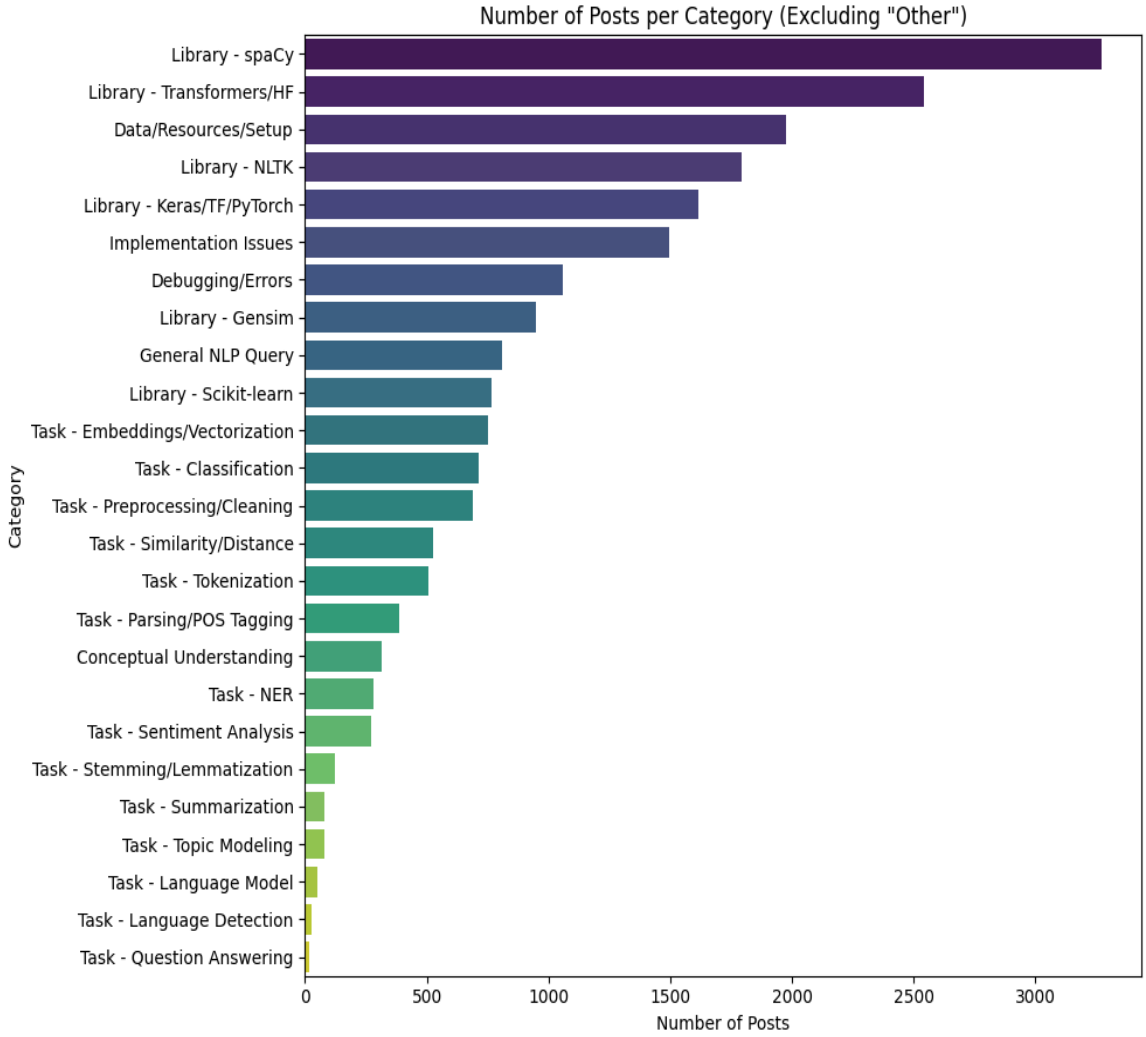

Figure 5. Tags vs the number of posts per Tag

Table 1: Results From Automated Implementation

No	Category	Number of Posts Per Tag	An Example Title Categorised by the System and Linked to SO
1	Library-SpaCy	3271	How can I share a complex spaCy NLP model across multiple Python processes to minimize memory usage? https://stackoverflow.com/questions/79159805/how-can-i-share-a-complex-spacy-nlp-model-across-multiple-python-processes-to-mi
2	Library- Transformers/HF	2544	How to convert character indices to BERT token indices? https://stackoverflow.com/questions/79173053/how-to-convert-character-indices-to-bert-token-indices
3	Library - NLTK	1795	Finding the nouns in a sentence given the context in Python https://stackoverflow.com/questions/77455738/finding-the-nouns-in-a-sentence-given-the-context-in-python
4	Library - Keras/TF/Pytorch	1615	Laptop stopped when training a pytorch lstm model, while tensorflow counterpart works https://stackoverflow.com/questions/77382923/laptop-stopped-when-training-a-pytorch-lstm-model-while-tensorflow-counterpart
5	Library - Gensim	949	Gensim's Doc2Vec with documents in multiple languages https://stackoverflow.com/questions/78262529/gensims-doc2vec-with-documents-in-multiple-languages
6	Library - Scikit-Learn	766	Error while peforming Tf-idfvectorizer() on the training values https://stackoverflow.com/questions/76752935/error-while-peforming-tf-idfvectorizer-on-the-training-values
7	Implementation Issues	1494	Why is the text in the files I am concatenating in Powershell coming out altered? https://stackoverflow.com/questions/76717272/why-is-the-text-in-the-files-i-am-concatenating-in-powershell-coming-out-altered
8	Debugging/Errors	1057	ModuleNotFoundError: No module named 'pycaret.nlp' https://stackoverflow.com/questions/76686270/modulenotfounderror-no-module-named-pycaret-nlp
9	Conceptual Understanding	315	If an FST transition is based on a given context, how can it be called as 'non deterministic'? https://stackoverflow.com/questions/76633913/if-an-fst-transition-is-based-on-a-given-context-how-can-it-be-called

			-as-non-d
10	Data/Resource/Setup	1974	How to create a dataset for a model like Falcon-7b/40b? https://stackoverflow.com/questions/76612874/how-to-create-a-dataset-for-a-model-like-falcon-7b-40b
11	Task - Embeddings/Vectorisation	750	How embedding lookup works in Word2Vec https://stackoverflow.com/questions/76472136/how-embedding-lookup-works-in-word2vec
12	Task - Classification	714	LIME gives this error "classifier models without probability scores" in python https://stackoverflow.com/questions/77085879/lime-gives-this-error-classifier-models-without-probability-scores-in-python
13	Task - Preprocessing/Cleaning	690	How can I train a model for specific information extraction https://stackoverflow.com/questions/76961114/how-can-i-train-a-model-for-specific-information-extraction
14	Task - Similarity/Distance	524	In R Str_count: Counting occurrences of words at a certain distance e.g. 1 to 30 words apart https://stackoverflow.com/questions/76822491/in-r-str-count-counting-occurrences-of-words-at-a-certain-distance-e-g-1-to-30
15	Task - Tokenization	507	'BpeTrainer' object cannot be converted to 'Sequence' when training Bpetokenizer https://stackoverflow.com/questions/76753611/bpetrainer-object-cannot-be-converted-to-sequence-when-training-bpetokenizer
16	Task - Parsing/ POS Tagging	1589	how to do a web scraping getting all the data https://stackoverflow.com/questions/76758016/how-to-do-a-web-scraping-getting-all-the-data
17	Task - NER	279	How to extract the output from an NLP model to a dataframe? https://stackoverflow.com/questions/74629744/how-to-extract-the-output-from-an-nlp-model-to-a-dataframe
18	Task - Sentiment Analysis	270	Unsure how to resolve language error message from Google's natural language api: "The language sq is not supported for document_sentiment analysis." https://stackoverflow.com/questions/74925744/unsure-how-to-resolve-language-error-message-from-googles-natural-language-api
19	Task - Stemming/Lemmatisation	121	NLP stemming with JavaScript and PHP pages in the browser https://stackoverflow.com/questions/74376193/nlp-stemming-with-javascript-and-php-pages-in-the-browser
20	Task - Summarization	81	How to get an output from ONEAI NLP API? https://stackoverflow.com/questions/74390115/how-to-get-an-output-from-oneai-nlp-api

7. Discussion

A rule-based automated approach to categorise NLP tagged posts is easy to implement and understand (3). The rules are easy to follow and precise. That is the major benefit of this rule-based approach. However, this approach has a downside as well. The regular expressions check for the exact keywords. This might miss some of the important and relevant posts if they use different wordings and keywords, as it can't understand deeper meanings (4). The 25 pre-defined categories are well situated for the classification as they directly aim at general topics in NLP that the developers discuss in the Stack Overflow community. The system includes 6 library categories, 14 task categories, 4 specific check categories and 1 general category. To prevent overlapping, a priority-based system was introduced during categorisation. This system classifies the posts based on priority, where library categories are given the highest priority and the general category is given the least.

Based on the category distribution in Figure 05, the results can be interpreted as follows:

- The high frequency under the library categories like SpaCy, Transformers, NLTK and Keras confirms that these tools are actively used and the developers are encountering issues with them in the NLP domain
- A significant number of posts under data/resource/setup and debugging/errors highlight that there are issues related not only to specific libraries and tools, but also issues related to data handling, environmental configurations and troubleshooting as well.
- Most importantly, the relatively small group of the general category, almost 4%, depicts that the other 24 categories are effective and meaningful.

8. Conclusion

Natural Language Processing (NLP) has become most prominent in modern computer science. As a novel technology, it has a lot of new tools, libraries and concepts for developers to explore and utilize. The software developers who have newly entered the NLP domain and those who tackle new projects in NLP often discuss their emerging questions and solutions in the Stack Overflow (SO) community. Although the SO community consists of massive NLP-related discussions, it is quite difficult to quickly find an answer to a specific question within this domain. As a solution to this matter, this study has focused on developing an automated classification system, where the NLP-related posts were tagged into 25 meaningful categories. This was done by using a rule-based system with the help of regular expression patterns and a predefined set of meaningful tags. Over 20000 posts from the SO were obtained, preprocessed, categorised, labelled and stored during the course. The system shows an excellent performance in classifying those posts into meaningful categories. However, it is to be mentioned that there are some drawbacks of using a rule-based approach as well. The system merely restricts and obeys the pre-defined rule set instead of understanding the deeper semantic meaning of the texts. Nevertheless, the knowledge base developed by the rule-based automated classification system would help the developers to search for answers to their questions efficiently and effectively.

9. Data Availability

- Data set - https://docs.google.com/spreadsheets/d/1-RN7so7A9ArjXd4GO-5uHl7M1qKfaWgG/edit?usp=sharing&ouid=115701530022859255786&rtpof=true&sd=true

- Code Implementation - https://github.com/hansi959/NLP--Text-Classification

10. References

[1] Khurana, D., Koli, A., Khatter, K. and Singh, S. (2022). Natural language processing: state of the art, current trends and challenges. Multimedia Tools and Applications, 82, pp.3713–3744. DOI: 10.1007/s11042-022-13428-4.

[2] Silva, C. (2003). The importance of stop word removal on recall values in text categorization. In: Proceedings of the International Joint Conference on Neural Networks (IJCNN), Vol. 3, pp. 1661–1666. DOI: 10.1109/IJCNN.2003.1223656.

[3] Liu, H., Gegov, A. and Stahl, F. (2014). Categorization and construction of rule-based systems. In: Communications in Computer and Information Science, Vol. 459. Springer. DOI: 10.1007/978-3-319-11071-4_18.

[4] BotPenguin.(n.d.). Rule-Based System. Available at: https://botpenguin.com/glossary/rule-based-system [Accessed 18 Apr. 2025]

[5] Bao, C. and Wang, Y. (2021). A survey of word cloud visualization. Journal of Computer-Aided Design & Computer Graphics, 33, pp.532–544. DOI: 10.3724/SP.J.1089.2021.18811.

[6] Weinstein, J. (2020) 'The best and worst ways to use Stack Overflow', The Startup, 29 July. Available at: https://medium.com/swlh/the-best-and-worst-ways-to-use-stack-overflow-711a077f2892 (Accessed: 18 April 2025).

[7] Stack Exchange (2020) 'API: What are some good strategies to limit hitting database on query-based search?', Software Engineering Stack Exchange, 10 August. Available at: https://softwareengineering.stackexchange.com/questions/448546/api-what-are-some-good-strategies-to-limit-hitting-database-on-query-based-sea (Accessed: 19 April 2025).