Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 01, January-2024

Summarization of Unstructured Text Data Methodology and Pre-processing Approach

Harisha S
Research Scholar
Srinivas University
Mangaluru,
Karnataka, India

Dr. Subrahmanya Bhat
Professor and Dean
Institute of Computer and Information Science
Srinivas University, Mangaluru,
Karnataka, India

***Abstract*— Unstructured text data refers to textual information that lacks a predefined format or organizational structure, making it challenging to analyze directly with traditional methods. This type of data is often found in sources such as social media posts, news articles, research papers, and customer reviews. Preprocessing unstructured text is a critical step in preparing it for tasks like summarization, where the goal is to extract meaningful insights or produce concise representations of the content. The preprocessing process typically involves several key techniques, including tokenization (splitting the text into smaller units), normalization (such as lowercasing and removing stop words), stemming or lemmatization (reducing words to their base forms), and eliminating irrelevant or noisy data. This work concentrates on methodology of unstructured text summarization and its data preprocessing. These steps enhance the quality of the text, making it more suitable for summarization algorithms, which can then generate coherent, informative summaries while preserving the essence of the original content.**

***Keywords*—Unstructured Data, word cloud, Preprocessing, summarization.**

I. INTRODUCTION

Unstructured data summarization refers to the process of condensing large volumes of unstructured information, such as text, into shorter, more concise representations while preserving the essential meaning and key points. Unstructured data, which includes sources like emails, articles, social media posts, audio transcriptions, and more, lacks a predefined format or organization. Summarization of unstructured text can be achieved through two main approaches: extractive and abstractive. In extractive summarization, important sentences or phrases are directly extracted from the original text, while in abstractive summarization, the system generates new sentences to convey the gist of the document. Unstructured data summarization is critical for efficiently handling and analyzing vast amounts of text data, especially in domains like business, research, and customer service, where quick insights are needed from large volumes of information. Advances in Natural Language Processing (NLP), particularly through models like BERT and GPT, have significantly improved the quality and accuracy of automatic summarization, making it an essential tool for data-driven decision-making and information retrieval. The general approach for text summarization is as shown in figure 1.

Unstructured data summarization is an essential technique for handling the vast amounts of unstructured information that we encounter daily. Unstructured data, unlike structured data (such as spreadsheets or databases), lacks a predefined format, making it harder to analyze and interpret. It encompasses various types of content, such as text from documents, social media posts, news articles, emails, reviews, and more. The main challenge with unstructured data is its complexity and the difficulty in extracting meaningful insights quickly, which is where summarization comes into play [1].

Summarization techniques aim to reduce the volume of data while maintaining the key information, allowing users to quickly grasp the core concepts without reading the entire content [2, 3]. The two primary approaches are extractive summarization and abstractive summarization. In extractive summarization, the goal is to select and extract key sentences or phrases from the original text without altering the wording.

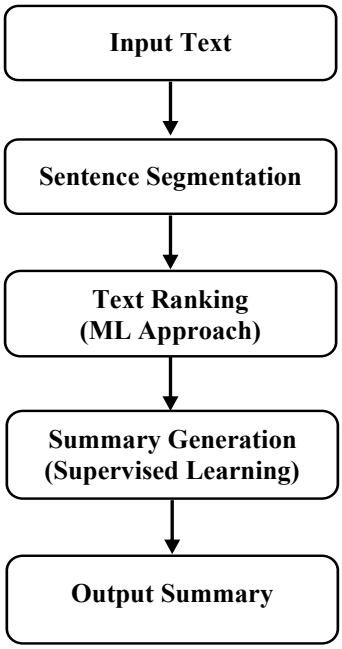

Figure 1 – The general approach for text summarization

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 01, January-2024

This method is straightforward but can result in summaries that may lack coherence or fluency. On the other hand, abstractive summarization involves generating new sentences that paraphrase and condense the original content. This method is more challenging but can lead to more natural and coherent summaries.

Advancements in Natural Language Processing (NLP) and the development of powerful pre-trained models like BERT, GPT, and T5 have greatly enhanced the accuracy and efficiency of unstructured data summarization. These models can understand the context, semantics, and relationships within text, making them capable of producing highly accurate summaries for complex unstructured data. In real-world applications, unstructured data summarization is used in a variety of fields, including customer service, where chat logs and support tickets are summarized to identify common issues, or in research [4], where large volumes of academic papers can be summarized to extract relevant findings. Additionally, abstractive summarization plays a vital role in applications like news aggregation, where concise and readable summaries are required from multiple sources.

II RELATED WORK

A. Traditional ML Techniques for text summerization

Traditional machine learning techniques for text summarization focus primarily on extractive methods, where the goal is to identify and extract the most important sentences or phrases from a document. These approaches rely heavily on statistical and feature-based methods to analyze the text structure and content. Key features such as term frequency-inverse document frequency (TF-IDF), word frequency, sentence position, and n-grams are used to assess the relevance and importance of sentences. Algorithms like Naive Bayes, Support Vector Machines (SVM), Random Forests, and K-Means clustering are commonly employed to classify or rank sentences based on these features. These methods are computationally efficient and relatively simple to implement, making them suitable for smaller datasets or scenarios with limited computational resources. However, they often struggle with understanding the deeper semantic meaning of text and tend to focus solely on surface-level patterns, leading to summaries that may lack coherence and contextual relevance. While traditional ML techniques have been effective for extractive summarization tasks, their limitations have paved the way for more advanced, deep learning-based methods for handling complex and unstructured text data [5].

Traditional machine learning (ML) techniques for text summarization rely on extractive methods to condense information by identifying and selecting the most relevant sentences or phrases directly from the source text. These methods do not involve generating new sentences but instead focus on determining which parts of the text best represent its overall content. A variety of statistical and linguistic features are used to evaluate sentence importance, including term frequency-inverse document frequency (TF-IDF), sentence length, word frequency, position of sentences within the text (e.g., introductory sentences often being more relevant), and the presence of keywords or named entities. These features are extracted and fed into machine learning algorithms such as Naive Bayes, Support Vector Machines (SVM), Decision Trees, and Random Forests, which then rank or classify sentences based on their relevance.

Clustering algorithms like K-Means are also used to group sentences into clusters, identifying central sentences as representatives of the document's main themes. Graph-based algorithms, such as TextRank or LexRank, leverage sentence similarity measures (e.g., cosine similarity) to create a graph where nodes represent sentences, and edges signify their semantic relationships [9, 10]. Centrality measures, such as PageRank, are then applied to select the most important sentences.

While these methods are computationally efficient and well-suited for relatively small datasets or structured text, they often lack the ability to understand semantic nuances or context. For instance, they may overlook implicit relationships between ideas or fail to maintain the logical flow of the text. Additionally, these methods typically depend on manually engineered features and predefined rules, which limit their adaptability to diverse domains or languages. Despite their simplicity and interpretability, traditional ML techniques often produce summaries that are less coherent or informative when compared to more advanced deep learning-based models [6]. However, these methods remain valuable for scenarios requiring lightweight, interpretable, and resource-efficient summarization.

B. Deep Learning approach for text summerization

Deep learning approaches for text summarization represent a significant advancement over traditional methods, as they are capable of capturing deeper semantic relationships and generating more coherent and contextually relevant summaries. These methods include both extractive and

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 01, January-2024

abstractive techniques, with a strong emphasis on neural network architectures like Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM), Gated Recurrent Units (GRUs), and, more recently, transformer models. In extractive summarization, deep learning models use hierarchical attention mechanisms or embeddings like Word2Vec, GloVe, or contextualized representations such as BERT to rank sentences based on their importance. For abstractive summarization, sequence-to-sequence (Seq2Seq) models with attention mechanisms have been widely adopted, allowing the generation of new sentences that paraphrase the input text while retaining its meaning. Transformer-based models, such as BERT, GPT, T5, and BART, have set state-of-the-art benchmarks by leveraging pre-trained language models fine-tuned for summarization tasks [7].

These models use self-attention mechanisms to process entire sequences of text in parallel, enabling them to capture long-range dependencies and generate high-quality summaries even for lengthy and complex documents. Deep learning approaches are particularly effective in handling unstructured text data, domain-specific jargon, and multiple languages, making them highly versatile. However, they require large labeled datasets for training, substantial computational resources, and careful tuning to ensure fluency, coherence, and factual accuracy in generated summaries. Despite these challenges, the ability of deep learning models to produce summaries that resemble human-generated text has made them the preferred choice for modern summarization tasks in various industries [8].

III METHODOLOGY

The methodology for applying machine learning (ML) approaches to unstructured text data classification typically involves several key steps as shown in the figure 2.

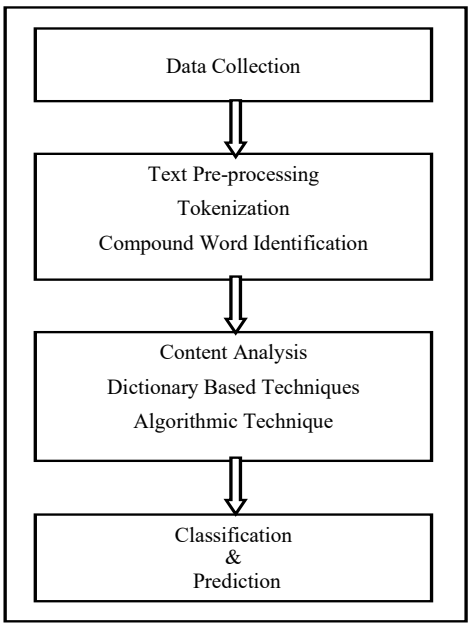

Figure 2 – Process flow of text processing

Data collection is the first and crucial step in natural language processing (NLP) tasks. It involves gathering raw data, typically in the form of text, from various sources like websites, social media, books, and articles. This data can be unstructured, meaning it lacks a predefined format, making it challenging to process. In many NLP applications, data is collected to serve as input for tasks like sentiment analysis, text classification, or named entity recognition. The quality and relevance of the data are essential for the success of the task, as noisy or irrelevant data can lead to poor model performance. Data collection methods may vary depending on the specific use case, including web scraping, APIs, or open datasets.

Once the data is collected, it undergoes text pre-processing, which is a series of steps to clean and standardize the text for further analysis. Pre-processing is essential to remove noise and ensure the text is in a format that machines can understand. Common steps in text pre-processing include tokenization (splitting the text into words or sentences), lowercasing (to standardize the text), removing stop words (common words like “the,” “is,” and “and” that don’t contribute significant meaning), stemming (reducing words to their root form), and lemmatization (converting words to their base or dictionary form). Pre-processing is crucial because it reduces the complexity of the text and enhances the performance of NLP models. Compound word identification is the process of detecting and handling compound words, which are formed by combining two or more words to create a new meaning. In some languages, compound words are common and need to be processed carefully, especially in languages like German or Finnish, where compound words are lengthy and complex. The challenge is that these compound words may not be recognized as a single word by standard tokenizers or models. Identifying compound words can be performed using techniques like dictionary lookup, morphological analysis, or statistical models. This step ensures that the compound word is treated as a single entity, improving the accuracy of tasks like text classification or machine translation.

Dictionary-based techniques play a significant role in many NLP tasks. These techniques rely on predefined dictionaries or lexicons to identify words, entities, or phrases of interest in the text. For example, a sentiment analysis task may use a sentiment lexicon to classify words as positive, negative, or neutral. In other tasks, such as named entity recognition, a dictionary of known names, places, or organizations may be used to identify and classify entities in the text. Dictionary-based approaches are efficient and interpretable but may suffer from coverage limitations, as they can only recognize words or entities that are already included in the dictionary. To address this, dictionaries can be expanded dynamically, or hybrid methods can combine dictionary-based techniques with machine learning models.

Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 01, January-2024

Finally, classification is the process of categorizing text into predefined classes or labels. It is one of the most common tasks in NLP and involves training a machine learning model on labeled data, where the correct category for each piece of text is known. Popular classification algorithms include decision trees, support vector machines (SVM), and neural networks. The goal is to train the model to identify patterns in the data so that it can correctly assign labels to unseen text. The success of a classification model depends on various factors, such as feature selection, the quality of labeled data, and the choice of algorithm. Classification tasks in NLP include sentiment analysis, spam detection, and topic categorization.

IV RESULTS

A. Visualizing Data Before Preprocessing

The graphical representation of textual data where in individual words are visually depicted with varying magnitudes corresponding to their frequency or significance within a particular corpus. The prominence of a word, represented by its size, correlates directly with its occurrence or relevance in the dataset, allowing for the immediate identification of dominant lexemes. Word clouds are frequently employed in textual analysis, sentiment assessment, and topic modeling to elucidate the principal themes or conceptual elements within a body of text. They serve as an aesthetically compelling and cognitively accessible method for distilling extensive textual information, often utilized in presentations or analytical reports to succinctly convey high-level insights as shown in figure 3. However, word clouds may obscure intricate semantic relationships or contextual nuances between terms, thereby making them more suitable for preliminary exploration rather than in-depth, interpretive analysis.

Figure 3 - Word cloud before text pre-processing

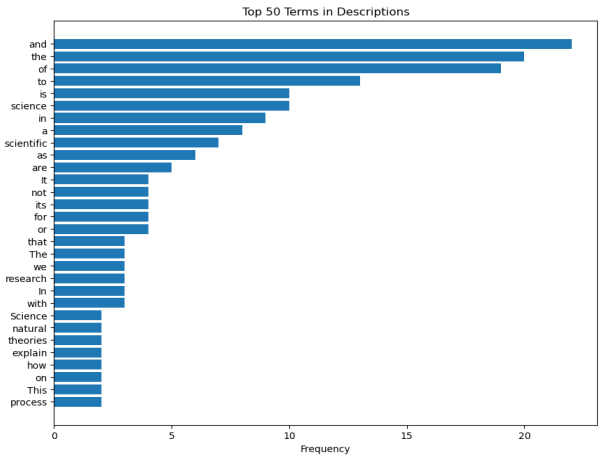

Figure 4 - Top 50 most frequent terms before text pre-processing

A Top Terms graph is a visualization technique used to represent the most frequent or significant terms in a text corpus or dataset. This graph typically displays keywords or terms along with their frequency or importance, offering a clear overview of the dominant themes within the text. It is commonly used in natural language processing (NLP) for tasks such as topic modeling or content analysis. The graph can take various forms, such as bar charts or word clouds, where each term is sized according to its frequency or weighted relevance. Top Terms graphs are useful for quickly identifying the key concepts in a collection of documents, aiding in tasks like information retrieval, text summarization, and content classification as shown in figure 4. The next step involved is cleaning and preparing textual data to make it suitable for analysis or model training. One common preprocessing step is removing URLs, which eliminates links that might not contribute meaningful information to the analysis. Similarly, removing special characters such as punctuation, symbols, or emojis is often necessary to ensure that the text contains only relevant words. This step helps standardize the text and prevents unwanted noise from affecting the results. Another essential technique is removing stop words, which are commonly used words like "the," "is," "and," or "of" that typically do not carry significant meaning. By excluding stop words, we can focus on the more meaningful parts of the text, such as nouns, verbs, and adjectives, which are more relevant for most NLP tasks. Combined with other steps like tokenization, lemmatization, or stemming, these preprocessing techniques help reduce the complexity of text data, making it easier for algorithms to extract valuable insights or patterns.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 01, January-2024

V CONCLUSION

In conclusion, the methodology for summarizing unstructured text data relies heavily on effective preprocessing to transform raw, disorganized content into a structured format that can be easily analyzed and summarized. Through techniques like tokenization, stop word removal, stemming, and lemmatization, preprocessing addresses the inherent challenges of unstructured data, reducing complexity and improving the quality of the input for summarization models. By refining the text in this way, it becomes possible to extract meaningful information, identify key themes, and generate coherent, concise summaries that preserve the essence of the original text. While the preprocessing phase is vital to ensure accuracy and relevance, the choice of summarization technique—whether extractive or abstractive—further determines the efficiency and quality of the output. As NLP technologies continue to evolve, advancements in preprocessing and summarization methodologies will play an increasingly crucial role in handling large-scale unstructured data and enabling more effective content analysis.

REFERENCES

[1] Z. Li, X. Yu, T. Wei and J. Qian, "Unstructured Big Data Threat Intelligence Parallel Mining Algorithm," in Big Data Mining and Analytics, vol. 7, no. 2, pp. 531-546, June 2024, doi: 10.26599/BDMA.2023.9020032.

[2] S. Liu et al., "Multimodal Data Matters: Language Model Pre-Training Over Structured and Unstructured Electronic Health Records," in IEEE Journal of Biomedical and Health Informatics, vol. 27, no. 1, pp. 504-514, Jan. 2023, doi: 10.1109/JBHI.2022.3217810.

[3] Y. Seo, J. Park, G. Oh, H. Kim, J. Hu and J. So, "Text Classification Modeling Approach on Imbalanced-Unstructured Traffic Accident Descriptions Data," in IEEE Open Journal of Intelligent Transportation Systems, vol. 4, pp. 955-965, 2023, doi: 10.1109/OJITS.2023.3335817.

[4] Gowda, Madhu Belur Gopala, Naveen Kumar Boraiah, Varun Eshappa, and Gopala Krishna Chandra Shekara. "Classification of Epileptic EEG Signals Using Improved Atomic Search Optimization Algorithm." International Journal of Intelligent Engineering & Systems 16, no. 6 (2023).

[5] K. Adnan, R. Akbar and K. S. Wang, "Towards Improved Data Analytics Through Usability Enhancement of Unstructured Big Data," 2021 International Conference on Computer & Information Sciences (ICCOINS), Kuching, Malaysia, 2021, pp. 1-6, doi: 10.1109/ICCOINS49721.2021.9497187.

[6] Y. Asim, A. K. Malik, B. Raza, A. R. Shahid and N. Qamar, "Predicting Influential Blogger's by a Novel, Hybrid and Optimized Case Based Reasoning Approach With Balanced Random Forest Using Imbalanced Data," in IEEE Access, vol. 9, pp. 6836-6854, 2021, doi: 10.1109/ACCESS.2020.3048610.

[7] M. Elsayed, A. Abdelwahab and H. Ahdelkader, "A Proposed Framework for Improving Analysis of Big Unstructured Data in Social Media," 2019 14th International Conference on Computer Engineering and Systems (ICCES), Cairo, Egypt, 2019, pp. 61-65, doi:10.1109/ICCES48960.2019.9068154.

[8] A. Moldagulova and R. B. Sulaiman, "Document Classification Based on KNN Algorithm by Term Vector Space Reduction," 2018 18th International Conference on Control, Automation and Systems (ICCAS), PyeongChang, Korea (South), 2018, pp. 387-391.

[9] E, V., & Pushpa Ravikumar, D. (2018). Attribute Selection for Telecommunication Churn Prediction. International Journal of Engineering & Technology, 7(4.39), 506-509. https://doi.org/10.14419/ijet.v7i4.39.24364

[10] E. Varun and P. Ravikumar, "Community Mining in Multi-relational and Heterogeneous Telecom Network," 2016 IEEE 6th International Conference on Advanced Computing (IACC), Bhimavaram, India, 2016, pp. 25-30, doi: 10.1109/IACC.2016.15.