International Journal of Advanced Trends in Engineering and Management
(ISSN 2583-7052)

pp. 149-160, April

# An Efficient Gain Ratio–Driven Stemming Technique for Medical Text Preprocessing

Manikandan K[1], Mahalakshmi D[2]

[1]Assistant Professor, Department of Information Technology, A.V.C. College of Engineering, km.kathirai@gmail.com
[2]Assistant Professor, Department of Information Technology, A.V.C. College of Engineering, dmahaa1985@gmail.com

**[1]Corresponding Author E-mail: km.kathirai@gmail.com**

**ABSTRACT:** Pre-processing is a critical step in the extraction of medical information from unstructured text data. Medical text data, comprising of research articles, clinical notes, and patient records, frequently exhibit noise, variations in spelling and word forms, and other inconsistencies. Pre-processing methods are utilised to cleanse, standardise, and ready textual data for precise and significant retrieval of medical data. The goal of pre-processing is to enhance the quality of the text data by removing irrelevant elements, standardizing the representation of terms, and improving the understanding of the contextual information surrounding medical entities. By addressing these challenges, pre- processing lays the foundation for subsequent steps, such as entity recognition, relationship extraction, and data analysis, ultimately enabling the extraction of valuable insights from medical text data.

**Keywords:** Lemmatization, Stemming, Gain Ratio, ANN, SVM, RF

## 1. Introduction

The process of text pre-processing is a crucial component in the extraction of medical information from unstructured textual data through the application of text mining methodologies. Medical information extraction aims to uncover valuable insights from various sources such as medical literature, electronic health records, clinical notes, and patient forums. However, these sources often contain noisy and unstructured text, making it challenging to extract relevant information accurately. Text pre-processing addresses these challenges by transforming the raw text data into a clean and structured format suitable for analysis.

### 1.1 Steps in Text Pre-processing

Text pre-processing involves several steps that help transform raw text into a more structured and manageable format:

***a. Sentence Segmentation:*** Dividing the text into individual sentences is the first step. This allows for analysis at a more granular level and helps in understanding the context of medical information.

***b. Tokenization:*** Breaking down sentences into smaller units called tokens (typically words) is essential for further analysis. Tokenization enables the identification of relevant terms, entities, and relationships within the text.

***c. Stop Word Removal:*** Removing common stop words, such as "the," "is," or "and," helps reduce noise and improve the efficiency of subsequent analysis.

***d. Abbreviation Expansion:*** Medical text often includes abbreviations and acronyms that require expansion for accurate interpretation. Handling these abbreviations ensures that relevant medical terms are properly recognized.

***e. Lemmatization or Stemming:*** Reducing words to their base or root form using lemmatization or stemming techniques aids in reducing variations and ensuring consistent representation of terms. This improves the accuracy of analysis tasks such as information retrieval and entity recognition.

***f. Removal of Special Characters and Noise:*** Eliminating special characters, symbols, irrelevant noise, and non-alphanumeric characters helps in maintaining data cleanliness and reducing the complexity of analysis.

### 1.2 Purpose of Data Pre-processing Techniques

| Technique | Purpose |
|---|---|
| Data Cleaning | Improve data quality and reliability |
| Missing Value Imputation | Prevent data loss and model bias |
| Normalization (Min–Max Scaling) | Ensure equal feature contribution |
| Standardization (Z-score Scaling) | Handle features with different units/scales |
| Encoding Categorical Variables | Allow models to process categorical data |
| Feature Engineering | Improve model performance |
| Feature Selection | Reduce overfitting and training time |
| Dimensionality Reduction | Simplify models and remove redundancy |
| Outlier Detection & Removal | Improve robustness of models |
| Data Augmentation | Increase dataset size and diversity |
| Tokenization (Text Preprocessing) | Prepare text for NLP models |
| Stemming/Lemmatization | Reduce vocabulary size |
| Train–Test Split | Evaluate model generalization |
| Cross-Validation | Reliable performance estimation |

## 2. Overview of Stemming Techniques

The provision of word stemming is a crucial aspect that is facilitated by contemporary indexing and search systems. Indexing and searching are integral components of Text Mining applications, Natural Language Processing (NLP) systems, and Information Retrieval (IR) systems, as noted in reference [1]. The primary objective is to enhance the process of recollection through the automated manipulation of word suffixes, achieved by reducing the words to their respective word roots during the stages of indexing and searching. The retrieval of documents was enhanced in terms of recall while maintaining the precision of the retrieved documents. The process of stemming typically involves the elimination of affixes, both prefixes and suffixes, from index terms prior to their assignment to the index. The process of stemming in an information retrieval (IR) system results in an expansion of the retrieved documents due to the stem of a term representing a more general concept than the original term. The preprocessing stage of text clustering, categorization, and summarization involves the conversion of the text, which is a necessary step before any relevant algorithm can be applied [2][3].

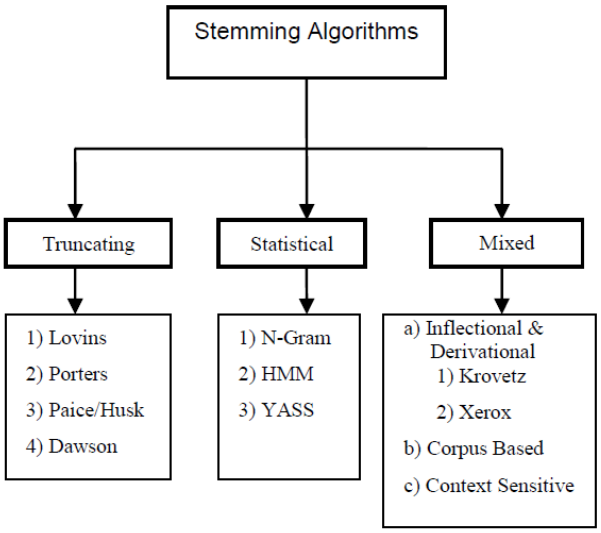
### 2.1 Working of a Stemmer

Empirical evidence suggests that in many cases, the morphological variations of words exhibit comparable semantic meanings and may be deemed interchangeable within the context of information retrieval (IR) applications. It is imperative to distinguish each word form from its base form, despite their shared meaning. Several stemming algorithms have been devised for this purpose. Every algorithm endeavours to map the morphological variations of a term, such as "introduction," "introducing," and "introduces," to the word "introduce." Certain algorithms have the capability to reduce the word to 'introduc'; however, this is permissible as long as all algorithms reduce the word to the same stem form [4]. Hence, stems are utilised to represent the essential terms of a query or document, rather than the original words. The concept involves minimising the overall quantity of unique expressions within a given document or query,

thereby resulting in a reduction of the computational duration of the ultimate outcome.

### 2.2 Classification of Stemming Algorithms

In a general sense, stemming algorithms can be categorised into three distinct groups: truncation-based methods, statistical methods, and hybrid methods. Each of these groups exhibits a characteristic method for identifying the stems of the word variations.




### 2.3 Lovins Stemming Algorithm

Lovins introduced the initial widely adopted and efficient stemming algorithm in 1968. The process entails conducting a search on a table consisting of 294 endings, 29 conditions, and 35 transformation rules that have been organised based on the principle of longest match, as previously documented [5]. The Lovins stemming algorithm is designed to eliminate the most extended suffix from a given word. Upon removal of the suffix, the term undergoes a process of recoding utilising an alternative table, which incorporates diverse modifications to transform these stems into legitimate words. The algorithm's single pass nature restricts it to removing only one suffix from a given word at a time [6].

This algorithm exhibits notable benefits, namely its high speed and proficiency in eliminating duplicate letters from words, such as transforming "getting" to "get". Additionally, it effectively manages numerous irregular plurals, such as "mouse" and "mice", "index" and "indices", among others.

One limitation of the Lovins approach is that it requires a significant investment of time and resources to collect and analyse the necessary data. Moreover, it is noteworthy that several suffixes are not included in the tabulated list of affixes. The process of word formation can be deemed unreliable at times, as it often encounters difficulties in generating words from their respective stems or in aligning the stems of words that share similar meanings. The author's utilisation of technical terminology is the cause for this phenomenon.

### 2.4 Porters Stemming Algorithm

The Porters stemming algorithm, which was introduced in 1980, is currently one of the most widely used stemming techniques. Numerous alterations and improvements have been implemented and proposed to the fundamental algorithm. The premise underlying this notion is that the suffixes present in the English language, numbering around 1200, are predominantly composed of amalgamations of smaller and less complex suffixes. The process comprises of five distinct stages, wherein specific regulations are implemented at each stage, until one of them satisfies the requisite criteria. Once a rule has been acknowledged, the corresponding suffix is subsequently eliminated, and the subsequent step is executed. The stem that is produced upon completion of the fifth step is subsequently returned. The aforementioned regulation is presented in the subsequent manner: The transformation of a given condition by replacing its suffix with a new suffix.

An illustrative instance of a linguistic rule, where m>0, is EED → EE, which denotes that in the event that a word terminates with EED and has at least one vowel and consonant, the ending should be modified to EE. The verb "agreed" is modified to its base form "agree", whereas the verb "feed" remains unaltered. The algorithm comprises

approximately 60 rules and is highly comprehensible.

Porter devised an intricate system of stemming that is commonly referred to as the 'Snowball' algorithm. The primary objective of the framework is to facilitate the creation of stemmers by programmers for diverse character sets or languages.

Paice arrived at the conclusion that the Porter stemmer exhibits a lower error rate than the Lovins stemmer, based on the observed stemming errors. It has been observed that Lovins stemmer is a more robust stemming algorithm that results in superior data reduction. The Lovins algorithm exhibits a larger size in comparison to the Porter algorithm, owing to its considerably comprehensive list of endings. However, one potential benefit of this approach is its increased speed. The algorithm in question has successfully exchanged spatial complexity for temporal efficiency. Its extensive set of suffixes enables it to execute the removal of a suffix in merely two primary stages, in contrast to the Porter algorithms which require five.

## 3. Related Works

Parlar, Tuba, Selma Ozel, and Fei Song [7] This study aimed to evaluate the impact of various data pre-processing techniques on sentiment analysis. The effectiveness of these techniques, both individually and in combination, was assessed for both English and Turkish, the latter being an agglutinative language. The researchers endeavoured to address the research inquiry regarding potential distinctions in pre-processing techniques for sentiment analysis between agglutinative and non-agglutinative languages.

Maheswari, S., and K. Arthi [8] A stemmer that follows a rule-based approach and employs truncation has been suggested. The implementation of the novel stemming mechanism has resulted in numerous alterations in morphology. The recently developed morphological variation removable stemming algorithm, based on rules, has demonstrated superior performance compared to other established algorithms, including New Porter, Paice/Lovins, and Lancaster stemming algorithms.

Eler, Danilo Medeiros, et al [9] A methodology was proposed to systematically vary distinct combinations of pre-processing steps and analyse their impact on precision, in order to identify the pre-processing combination that yields the highest level of accuracy. To demonstrate various pre-processing techniques, a series of experiments were conducted to compare different combinations, including stemming, term weighting, low-frequency term elimination, and stop word removal. The aforementioned combinations were utilised in tasks related to text and opinion mining. The resulting correct classification rates were computed to emphasise the significant influence of the pre-processing combinations.

Oussous, Ahmed, Ayoub Ait Lahcen, and Samir Belfkih [10] The study conducted an evaluation of the effects of the pre-processing phase on Arabic Sentiment Analysis, with a focus on metrics such as accuracy, precision, and recall. Various stemming techniques (Khoja, ISRI, Tashaphyne, Light10, and MOTAZ), n-gram models, and stop word removal were employed in our experimental investigations. The secondary objective is to investigate the effects of amalgamating various classifiers on the task of Arabic sentiment analysis. The utilisation and assessment of the vote algorithm, in tandem with three classifiers (Naive Bayes, Support Vector Machine, and Maximum Entropy), have been conducted through the implementation of k-fold cross-validation. This approach has been adopted for the aforementioned purpose.

Nhlabano, V. V., and P. E. N. Lutu [11] The scope of Sentiment Analysis can be broadened to encompass the identification of emotional states, such as sadness, anger, and happiness. Sentiment Analysis is a computational technique used to

forecast the polarity of a given textual unit with the aim of ascertaining whether the writer is conveying a positive, negative, or neutral viewpoint regarding a particular subject matter. The research topic of Sentiment Analysis has garnered significant attention over time due to its diverse practical applications, potential commercial advantages, and the intriguing challenges and research problems it poses to the academic community. Text pre-processing is a critical step in sentiment analysis, as it involves the classification of a text's orientation as either positive or negative. This task is considered challenging in the realm of text classification. The researchers examined the impact of text pre-processing techniques employed for social media data. The findings of the experiment indicate that the utilisation of text pre-processing techniques enhances the predictive precision of the resultant models for sentiment categorization.

Troussas, Christos, Akrivi Krouska, and Maria Virvou [12] The objective of this study is to establish a set of guidelines for selecting the most effective pre-processing techniques and classifiers for conducting sentiment analysis on Twitter data. The present study utilised three widely recognised Twitter datasets (OMD, HCR, and STS-Gold) within a series of experimental procedures. The present study delves into an in-depth analysis of sentiment polarity classification methods for Twitter text, with a particular focus on extended comparison. Additionally, the study examines the significance of text pre-processing in sentiment analysis. Subsequently, an evaluation has been conducted on four established classifiers that rely on learning algorithms, namely Naive Bayes, Support Vector Machine, k-Nearest Neighbours, and C4.5. The evaluation was based on the analysis of confusion matrices. Thirdly, an analysis is conducted on the prevalent ensemble methods, namely Bagging, Boosting, Stacking, and Voting, and their outcomes are compared to those of the underlying classifiers. Ultimately, a case study is presented that pertains to the utilisation of Twitter sentiment analysis within the realm of e-learning.

Mustafa, Arazo M., and Tarik A. Rashid [13] A proposition has been put forth for an active approach to stem Kurdish Sorani texts, with the aim of reducing word variations to singular terms or stems. The findings of this study indicate that the reduction of feature vector dimensionality in documents can enhance retrieval efficacy in the presence of stemming. The methodology utilised for Kurdish Sorani has the potential to be modified and implemented for Kurdish Kurmanji, resulting in improved efficiency and efficacy in the realm of digital text classification and related applications.

Maylawati, Dian Sa’adillah, et al [14] The author has put forth two cutting-edge factors. Firstly, the utilisation of both an Indonesian dictionary and an Indonesian Slang dictionary has been suggested to anticipate the re-stemming of an infinitive. Secondly, the incorporation of natural languages and slangs has been proposed to enhance the stemming process on word particles. Two scenarios were analysed to evaluate the algorithm. The first scenario involved 379 words and the second scenario involved 20 text data. Additionally, the authors conducted a comparison of the algorithm with the Porter, Nazief&Adriani, and Lucene stemmer algorithms. The findings indicate that the algorithm proposed was able to effectively perform the stemming process on Indonesian slang, achieving an accuracy rate of approximately 88.65%. The algorithm's accuracy level was observed to have improved. However, its memory usage was found to be comparable to that of other algorithms. Notably, the processing time did not exhibit a significant difference.

Gayatri N. Varekar; Nikita Chavan [15] The paper evaluates domainspecific models like BioBERT and ClinicalBERT, provides a thorough synthesis of recent NLP frameworks, and identifies issues with data heterogeneity, annotation scarcity, and model transparency. It ends by identifying future research objectives that center on massive

language models, multimodal data fusion, and explainable AI to support more reliable and scalable biomedical information systems.

Luis B. Elvas, Ana Almeida [16] This literature review aimed to analyze recent NLP approaches for medical text processing, examining techniques, performance metrics, and advancements across different languages and healthcare contexts.

Ashwini Tuppad, S. Patil [17] This paper signifies the need for data pre-processing and explains the data pre-processing pipeline with various underlying stages constituting it. It also presents a comparative analysis of various data pre-processing techniques for handling missing values and outliers in a dataset.

Arati K Kale, Dr. Dev Ras Pandey [18] This research shows that the pre-processing techniques chosen have a considerable positive impact on the model's performance when comparing the model's efficiency with and without pre-processed data.

## 4. Proposed Pre-Processing Approach

In this pre-processing method, Gain Ratio based Stemming method which uses content sensitivity for stemming processing. The steps like Sentence Segmentation, Word tokenization, Gain Ratio based Stemming method, and Lemmatization is used for pre- processing the review dataset.[19]

### 4.1 Sentence Segmentation

Sentence segmentation refers to the identification of larger units of processing that may comprise one or more words. The objective of this task is to discern the boundaries of sentences that separate words in distinct sentences. The paragraph within this review dataset has been segmented into individual sentences through the utilisation of punctuation marks such as full stops and exclamation points, which serve as the boundaries for sentence selection.

### 4.2 Word Tokenization

Tokenization refers to the procedure of segmenting the character sequence in a given text by identifying the boundaries of individual words, which are the demarcation points indicating the conclusion of one word and the commencement of another. Tokens are commonly referred to as the words identified for the purpose of computational linguistics.

### 4.3 Proposed Gain Ratio based Stemming Method

This proposed Gain Ratio based Stemming method is used the Information Gain to calculate the content sensitivity based on the three factors. The sensitivity of the word is computed using word sensitivity graph. The sensitivity entropy has been proposed as a means of estimating word sensitivity, taking into account three distinct factors. The computation of Shannon's entropy for a discrete random variable X can be achieved through the utilisation of Equation (1).

$$H(x)=\sum_{i=1}^{n} p(x_i)I(x_i)=-\sum_{i=1}^{n} p(x_i)\log p(x_i) \quad (1)$$

The probability mass function of state xi in a system with n distinct states is denoted by p(xi) in equation (1). The sensitivity of a particular node in the word sensitivity graph is determined in the proposed model by evaluating its impact on the system subsequent to its removal from the graph. The computation of the impact of node removal from the sensitivity graph of a word is accomplished through the utilisation of the information gain formula, as presented in Equation 2. The calculation of Information Gain involves subtracting the entropy of the original system from the entropy that is obtained after the removal of the node.

$$IG(X,a) = H(x) - H(x|a) \quad (2)$$

The equation (2) denotes that *IG(X ,a)* signifies the information gain of the system subsequent to the elimination of node a from the word sensitivity graph. *H(X)* represents the initial entropy of the system, while *H(X|a)* represents the entropy of the system after the removal of node a. In the event that the elimination of a node results in network disconnection while computing entropy, the calculation of entropy is performed

using the most extensive connected sub-graph. The following factors are presented:

**Word Usage Detection:** Probability Mass Function (PMF) for word usage in the data sensitivity graph (DSG) is calculated through number of times a word is used by the single customer, and the total number of times a word can be used by all other customers.

$$p(R_i) = \frac{\sum_{i=1}^{m} C_{v_i,D_j}}{\sum_{j=1}^{k}\sum_{i=1}^{m} C_{v_i,D_j}} \quad (3)$$

The equation (3) denotes that signifies the frequency of word $D_j$ being accessed by the customers, while ∑(j=1) represents the summation of all the words accessed by the customers.The expression $\sum_{i=1}^{m} C_{U_i,} D_j$ denotes the frequency of access of all the remaining words by the entire customer base. The entropy of usage is calculated for each node that corresponds to requests (Di ) in the word sensitivity graph.

**Word Connectivity Detection:** A lexical unit has the potential to function as a linking element within the corpus of reviews. The DSG employs a method for determining connectivity detection whereby the total number of arcs or paths incident to all nodes is summed to calculate the number of arcs of path to a given node.

$$p(R_i) = \frac{\sum_{j,m,n=1,1,1}^{j,m,n=a,b,c} d_{R_{i,j},R_{m,n}}}{\sum_{i,j,m,n=1,1,1}^{i,j,m,n=d,a,b,c} d_{R_{i,j},R_{m,n}}} \quad (4)$$

A dataset for review analysis can potentially function as a linkage point among various terminologies. Data items, akin to lexical units in other linguistic contexts, possess a degree of sensitivity as they enable access to data items that may be considered sensitive in other evaluations. The equation (4) denotes that $\sum_{j,m,n=1,1,1}^{j,m,n=a,b,c} d_{Ri,j,R_{m,n}}$ signifies the count of arcs of paths that are linked to the node $R_i$. Additionally, $\sum_{i.j,m,n=1,1,1,1}^{i,j,m,n=d,a,b,c} d_{Ri,j,R_{m,n}}$summation of all arcs or paths that are linked to every node $R_i$.

**Word Quality Detection:** The computation of the Probability Mass Function (PMF) is contingent upon the term "quality". The sensitivity of data increases with the quality of the words used.

$$p(R_i) = \frac{\sum s_r\, co(R_j)}{s_r} \Big/ \sum_{i=1}^{n} \frac{\sum s_r\, co(R_i)}{s_r} \quad (5)$$

The equation (5) represents the measurement of word quality, where $co(R_j)$ denotes the count of current entries of all data items in $R_i$ , $S_i$ refers to the total number of entries of all data items in R_i. The ratio of correct data in a single review is represented by $\frac{\sum S_r co(R_j)}{S_r}$. The summation of the correct word entries in all reviews is represented by $\sum_{i=1}^{n} \frac{\sum S_i co(R_j)}{S_i}$.

**Data Sensitivity Computation using Gain Ratio:** The measure of combined entropy is obtained by computing the Gain Ratio of all three entropy measures, namely the entropy of Word Usage detection, the entropy of word connection detection, and the entropy of word quality detection.

The measure of combined entropy is denoted in the following manner:

$$H(x_i) = H(C_{u,d_i}).H(d_{r_i}).\ H(N_{r_i}) \quad (6)$$

The equation (6) presented above utilizes $H(xi)$ to represent the aggregate entropy measurement of a review. The entropy measure computed by data usage is represented as $D_i$, $H(Cu,_{di})$ and is expressed in equation (7) below. The entropy measure, denoted as $H(d_{ri})$, is computed based on data similarity and is represented by equation (8). The entropy measure, denoted by $H(N_{ri})$, is determined based on data quality and can be computed using equation (9).

$$H(C_{u,d_i}) = -p(R_i)logp(R_i) \quad (7)$$

$$H(d_{r_i}) = -p(R_i)logp(R_i) \quad (8)$$

$$H(N_{r_i}) = -p(R_i)logp(R_i) \quad (9)$$

**Sensitivity Score:** The determination of a word's sensitivity score is based on the impact of its removal from the sensitivity graph of the review. The aforementioned phenomenon can be

expressed as the disparity between the summation of the collective entropy metric of all lexical units and the entropy metric of a single lexical unit.

$$C(x_i) = \sum_{i=1}^{n} H(x_i) - H(x_i) \qquad (10)$$

The equation (10) presented above represents the sensitivity of a review, denoted by *C(xi)*. The entropy scores of a review are denoted as $H(x_i)$. The mathematical expression i, $\sum_{i=1}^{n} H(x_i)$ represents the summation of entropies of n reviews.

The computation of the adjusted sensitivity measure is accomplished through the utilisation of equation (11).

$$C_{adj}(x) = \alpha . C(x) \qquad (11)$$

The variable denoted by in equation (11) signifies a numerical value bestowed upon by a specialist in the relevant field. The review is assigned a weight that denotes its level of sensitivity. A review that has a high C(x) score, yet is not considered to be particularly sensitive by the domain expert, will be assigned a low α value. The score is measured on a scale of 0 to 1.

## 5. Result and Discussion

The recently introduced PubMed 200k RCT dataset is derived from PubMed and is intended for the purpose of sequential sentence classification. The corpus encompasses roughly 200,000 abstracts pertaining to randomised controlled trials, comprising a total of 2.3 million sentences. The abstracts' sentences are categorised into five distinct classes, namely background, objective, method, result, or conclusion, as per the reference [11]. The present study employed Table 1 to assess the performance of the Proposed Pre-Processing method and other existing stemming algorithms, utilising various performance metrics.

**Table 1:** *Performance Metrics*

| Performance Metric | Formula |
|---|---|
| Accuracy (%) | $\frac{TP + TN}{TP + TN + FP + FN} \times 100$ |
| Recall (%) | $\frac{TP}{TP + FN} \times 100$ |
| Precision (%) | $\frac{TP}{TP + FP} \times 100$ |
| F-Measure (F1-Score) | $\frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$ |
| Miss Rate | $1 - \text{Recall} = \frac{FN}{TP + FN}$ |
| False Discovery Rate | $1 - \text{Precision} = \frac{FP}{TP + FP}$ |

The performance of the proposed Gain Ratio based Stemming (GRS) Method is evaluated with the existing stemming algorithms like Porter Stemmer, and Lovins Stemmer using three different classifiers like Support Vector Machine (SVM), Artificial Neural Network (ANN) and Random Forest (RF). The results obtained in this section is without using any feature extraction and feature selection techniques.

**Table 2:** *Classification Accuracy (in %) obtained by the Proposed and Existing Stemming Methods using SVM, ANN and RF*

| Pre-Processing with Stemming | Classification Accuracy (in %) | | |
|---|---|---|---|
| | ANN | SVM | RF |
| **Porter Stemmer** | 49.82 | 48.25 | 45.54 |
| **Lovins Stemmer** | 48.13 | 46.32 | 44.18 |
| **Proposed GR Based Stemmer** | 58.97 | 56.64 | 55.75 |

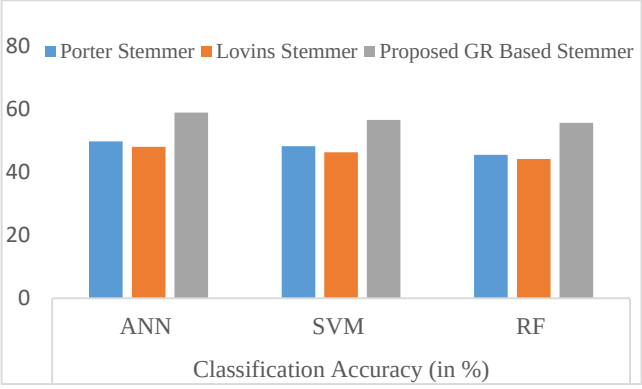



**Figure 1:** *Graphical representation of the Classification Accuracy (in %)*

The results presented in Table 2 and Figure 1 demonstrate that the implementation of the proposed stemming algorithm in conjunction with

classifiers yields superior accuracy when compared to alternative stemming algorithms.

**Table 3:** *Precision (in %) obtained by the Proposed and Existing Stemming Methods using SVM, ANN and RF*

| Pre-Processing with Stemming | Precision (in %) | | |
|---|---|---|---|
| | ANN | SVM | RF |
| **Porter Stemmer** | 48.91 | 47.34 | 44.63 |
| **Lovins Stemmer** | 47.24 | 45.23 | 43.27 |
| **Proposed GR Based Stemmer** | 57.24 | 55.79 | 53.37 |

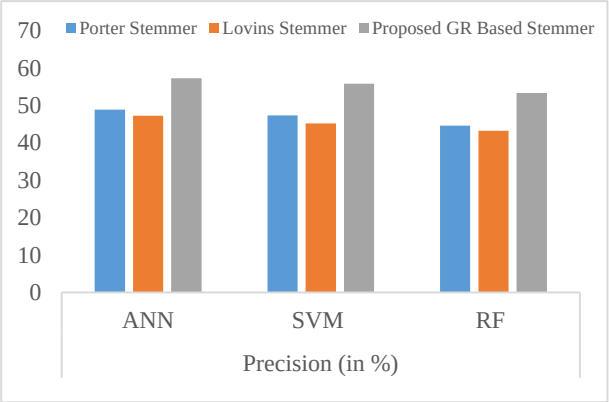



**Figure 2:** *Graphical representation of the Precision (in %)*

The findings from Table 3 and Figure 2 demonstrate that the utilisation of classifiers in conjunction with the proposed stemming algorithm yields higher precision rates compared to alternative stemming algorithms.

**Table 4:** *Recall (in %) obtained by the Proposed and Existing Stemming Methods using SVM, ANN and RF*

| Pre-Processing with Stemming | Recall (in %) | | |
|---|---|---|---|
| | ANN | SVM | RF |
| **Porter Stemmer** | 47.82 | 46.43 | 43.54 |
| **Lovins Stemmer** | 46.15 | 44.15 | 41.36 |
| **Proposed GR Based Stemmer** | 59.73 | 57.16 | 54.39 |

The findings presented in Table 4 and Figure 3 demonstrate that the utilisation of classifiers in conjunction with the proposed stemming algorithm results in superior recall performance compared to alternative stemming algorithms.

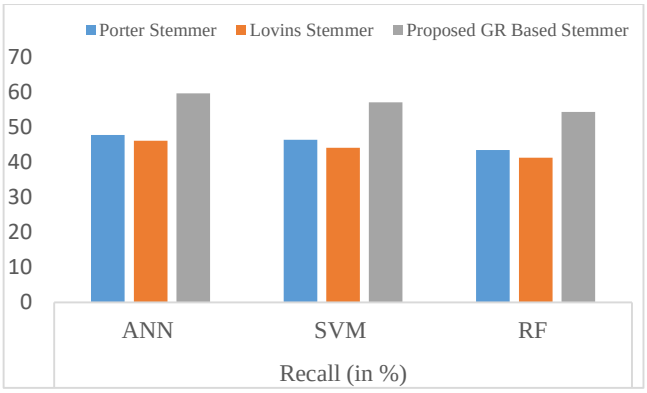



**Figure 3:** *Graphical representation of the Recall (in %)*

The results presented in Table 5 and Figure 4 indicate that the utilisation of classifiers in conjunction with the proposed stemming algorithm yields superior F-Measure outcomes in comparison to alternative stemming algorithms.

**Table 5:** *F-Measure (in %) obtained by the Proposed and Existing Stemming Methods using SVM, ANN and RF*

| Pre-Processing with Stemming | F-Measure (in %) | | |
|---|---|---|---|
| | ANN | SVM | RF |
| **Porter Stemmer** | 48.35 | 46.88 | 44.07 |
| **Lovins Stemmer** | 46.68 | 44.68 | 42.29 |
| **Proposed GR Based Stemmer** | 58.45 | 56.46 | 53.87 |

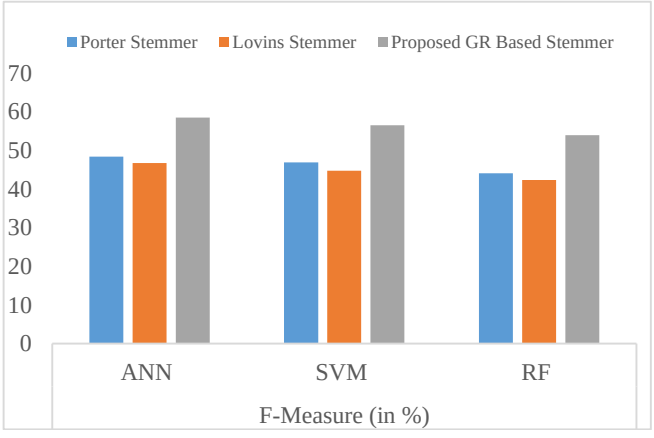



**Figure 4:** *Graphical representation of the F-Measure (in %)*

The findings derived from Table 6 and Figure 5 demonstrate that the implementation of the proposed stemming algorithm in conjunction with

classifiers results in a decreased miss rate when compared to alternative stemming algorithms.

**Table 6:** *Miss Rate (in %) obtained by the Proposed and Existing Stemming Methods using SVM, ANN and RF*

| Pre-Processing with Stemming | Miss Rate (in %) | | |
|---|---|---|---|
| | ANN | SVM | RF |
| **Porter Stemmer** | 52.18 | 53.57 | 56.46 |
| **Lovins Stemmer** | 53.85 | 55.85 | 58.64 |
| **Proposed GR Based Stemmer** | 40.27 | 42.84 | 45.61 |


**Figure 5:** *Graphical representation of the Miss Rate (in %)*

**Table 7:** *False Discovery Rate (in %) obtained by the Proposed and Existing Stemming Methods using SVM, ANN and RF*

| Pre-Processing with Stemming | False Discovery Rate (in %) | | |
|---|---|---|---|
| | ANN | SVM | RF |
| **Porter Stemmer** | 51.09 | 52.66 | 55.37 |
| **Lovins Stemmer** | 52.76 | 54.77 | 56.73 |
| **Proposed GR Based Stemmer** | 42.76 | 44.21 | 46.63 |

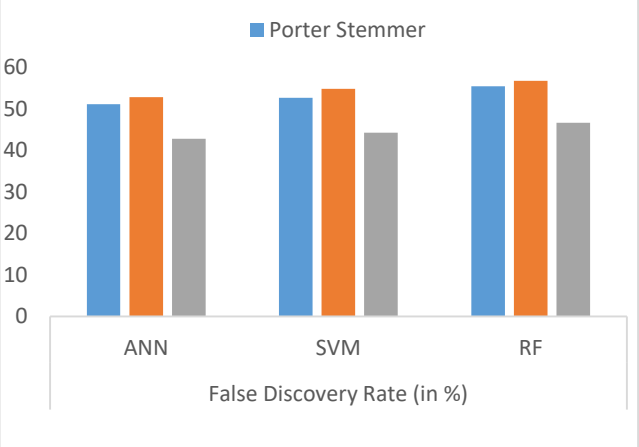



**Figure 6:** *Graphical representation of the False Discovery Rate (in %)*

The findings presented in Table 7 and Figure 6 demonstrate that the implementation of the proposed stemming algorithm in conjunction with classifiers results in a decreased false discovery rate (FDR) in comparison to alternative stemming algorithms.

## 6. Conclusion

In this paper, an enhanced stemming method is proposed using Sentence Segmentation, work tokenizer and Gain Ratio. This stemming is proposed to improve the accuracy of finding the sentiments for extraction of valuable insights from medical text data. The performance of the proposed stemming method is compared with existing stemming methods like Poster Stemmer and Lovins Stemmer with various evaluation metrics like Accuracy, Recall, Precision, F-Measure and error rates like False Discovery Rate, and Miss rate. From the results obtained, it is shown that the proposed pre-processing methods gives better result than the other stemming for the considered pudmed dataset.

## References

1. Aditya Wiha Pradana; Mardhiya Hayaty, Year: 2019, “The effect of stemming and removal of stopwords on the accuracy of sentiment analysis on indonesian-language texts”, Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control, pp. 375-380.

2. Samar Al-Saqqa; Arafat Awajan; Said Ghoul, Year: 2019, “Stemming effects on sentiment analysis using large arabic multi-domain resources”, 2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS). IEEE, pp. 211-216.

3. H. A. Shehu; Sezai Tokat, Year: 2019, “A hybrid approach for the sentiment analysis of Turkish Twitter data”, The International Conference on Artificial

Intelligence and Applied Mathematics in Engineering. Springer, Cham, pp. 182-190.

4. Hunaida Awwad; Adil Alpkocak, Year: 2017, "Using hybrid-stemming approach to enhance lexicon-based sentiment analysis in arabic", 2017 International Conference on New Trends in Computing Sciences (ICTCS). IEEE, pp. 229-235.

5. Anvitha Hegde; Savitha K. Shetty, Year: 2015, "A Study on Stemming Algorithms", Int. J. Emerg.Trends Sci. Technol, Vol: 2, No: 5, pp. 2301-2364.

6. Marsel Widjaja; Seng Hansun, Year: 2015, "Implementation of Porter's Modified Stemming Algorithm in an Indonesian Word Error Detection Plugin Application", International Journal of Technology, Vol: 6, pp. 139-150.

7. Tuba Parlar; Selma Ozel; Fei Song, Year: 2019, "Analysis of data pre-processing methods for sentiment analysis of reviews", Computer Science, Vol: 20.

8. S. Maheswari; K. Arthi, Year: 2019, "Rule Based Morphological Variation Removable Stemming Algorithm", International Journal of Recent Technology and Engineering (IJRTE), pp. 2277-3878.

9. Danilo Medeiros Eler; Denilson Grosa; Ives Pola; Rogério Garcia; Ronaldo Correia; Jaqueline Teixeira, Year: 2018, "Analysis of document pre-processing effects in text and opinion mining", Information, Vol: 9, No: 4, pp. 100.

10. Ahmed Oussous; Ayoub Ait Lahcen; Samir Belfkih, Year: 2019, "Impact of text pre-processing and ensemble learning on Arabic sentiment analysis", Proceedings of the 2nd International conference on networking, information systems & security, pp. 1-9.

11. V. V.Nhlabano; P. E. N. Lutu, Year: 2018, "Impact of text pre-processing on the performance of sentiment analysis models for social media data", 2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD). IEEE, pp. 1-6.

12. Christos Troussas; Akrivi Krouska; Maria Virvou, Year: 2019, "Trends on sentiment analysis over social networks: pre-processing ramifications, stand-alone classifiers and ensemble averaging", Machine Learning Paradigms. Springer, Cham, pp. 161-186.

13. Mustafa, Arazo M., and Tarik A. Rashid. "Kurdish stemmer pre-processing steps for improving information retrieval", Journal of Information Science 44.1 (2018): 15-27.

14. Dian Sa'adillah Maylawati; Wildan Budiawan Zulfikar; Cepy Slamet; Muhammad Ali Ramdhani; Yana Aditia Gerhana, Year: 2018, "An improved of stemming algorithm for mining indonesian text with slang on social media", 2018 6th International Conference on Cyber and IT Service Management (CITSM). IEEE, pp. 1-6.

15. Gayatri N. Varekar; Nikita Chavan; Manisha Bharti, Year: 2025, "Natural Language Processing Approaches in Biomedical Literature Mining: A Review of Techniques and Clinical Impact", 2025 IEEE Pune Section International Conference (PuneCon). IEEE, pp. 1-5.

16. Luis B Elvas; Ana Almeida; João C., Year: 2025, "Natural language processing in medical text processing: A scoping literature review", International Journal of Medical Informatics, Vol: 204, pp. 106049.

17. Ashwini Tuppad; Shantala Devi Patil, Year: 2023, "Data Pre-processing Issues in Medical Data Classification", International Conference on Network, Multimedia and Information Technology (NMITCON), pp. 1-6.

18. Arati K. Kale; Dev Ras Pandey, Year: 2024, "Data Pre-Processing Technique for Enhancing Healthcare Data Quality Using Artificial Intelligence", International Journal of Scientific Research in Science

and Technology (IJSRST), Vol: 11, No: 1, pp. 299-309.

19. S. Karthikeyan; P. Srivaramangai, Year: 2020, "Information Gain Based Stemming and Optimizations Based Feature Selection Methods for Sentiment Analysis on Amazon Product Review Dataset", International Journal of Management (IJM), Vol. 11, No.12, pp. 3832-3841,

20. https://www.kaggle.com/code/mpwolke/medical-abstracts/data