

Article

Effective Text Classification Through Supervised Rough Set-Based Term Weighting

Rasım Çekik 

Department of Computer Engineering, Faculty of Engineering, Şırnak University, 73000 Şırnak, Turkey; rasimcekik@srnak.edu.tr; Tel.: +90-486-216-40-08

Abstract: This research presents an innovative approach in text mining based on rough set theory. This study fundamentally utilizes the concept of symmetry from rough set theory to construct indiscernibility matrices and model uncertainties in data analysis, ensuring both methodological structure and solution processes remain symmetric. The effective management and analysis of large-scale textual data heavily relies on automated text classification technologies. In this context, term weighting plays a crucial role in determining classification performance. Particularly, supervised term weighting methods that utilize class information have emerged as the most effective approaches. However, the optimal representation of class–term relationships remains an area requiring further research. This study proposes the Rough Multivariate Weighting Scheme (RMWS) and presents its mathematical derivative, the Square Root Rough Multivariate Weighting Scheme (SRMWS). The RMWS model employs rough sets to identify information-carrying documents within the document–term–class space and adopts a computational methodology incorporating α , β , and γ coefficients. Moreover, the distribution of the term among classes is again effectively revealed. Comprehensive experimental studies were conducted on three different datasets featuring imbalanced-multiclass, balanced-multiclass, and imbalanced-binary class structures to evaluate the model’s effectiveness. The results show that RMWS and its derivative SRMWS methods outperform existing approaches by exhibiting superior performance on balanced and unbalanced datasets without being affected by class imbalance and number of classes. Furthermore, the SRMWS method is found to be the most effective for SVM and KNN classifiers, while the RMWS method achieves the best results for NB classifiers. These results show that the proposed methods significantly improve the text classification performance.



Academic Editor: Vasilis K. Oikonomou

Received: 16 December 2024

Revised: 4 January 2025

Accepted: 6 January 2025

Published: 9 January 2025

Citation: Çekik, R. Effective Text Classification Through Supervised Rough Set-Based Term Weighting. *Symmetry* **2025**, *17*, 90. <https://doi.org/10.3390/sym17010090>

Copyright: © 2025 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: text classification; term weighting; rough set; supervised learning; natural language processing

1. Introduction

The rapid advancement of technology has impacted every aspect of human life. Human–computer interaction, particularly with the development of web technologies, has reached new dimensions through information gathering and dissemination centers such as social media, online shopping sites, news, sports, and magazine platforms. This interaction has resulted in the generation of vast amounts of data. With the increasing volume of primarily textual data, including electronic documents, web pages, messages, etc., the organization, retrieval, and processing of these extensive textual data have become significant challenges. In this context, automatic text classification, often referred to as text categorization, stands out as one of the most widely-used technologies for addressing these

purposes. Text classification is the process of assigning predefined categories or classes to textual data [1]. This process is carried out using machine learning algorithms to analyze the content of textual data and make decisions based on this content. Text classification helps organize, analyze, and comprehend textual data, involving various operations and processes. At the forefront of these processes is term/feature weighting [2–4].

Term weighting is defined as the systematic process of increasing the value of a specific term or terms to give them more importance in analysis or computations. In the weighting process, the weight for each term in the documents is calculated using a term weighting algorithm. The primary objective is to highlight the difference between terms that provide distinct and specific information for classification and those that are commonly found across all documents, carrying no specific information. This way, a more effective and efficient classification process can be achieved. However, the challenges of high dimensionality and sparsity in text data hold a significant place in the term weighting stage. In the literature, representing document contents with multidimensional feature vectors is referred to as the vector space model (VSM) [5]. Considering terms in text data as features, leads to a highly sparse structure in the vector space model, resulting in a high-dimensional term space. One of the most significant research problems in text classification is to represent the relationship between extracted features and documents as effectively as possible in the vector space model, despite the sparsity structure. In this process, the term weighting operation comes into play, and numerous studies have been presented on this matter. For instance, Debole and Sebastiani [6] initially proposed the idea of supervised term weighting (STW), which involves weighting terms based on known categorical information in the training data. They introduced three STW schemes: TF-CHI, TF-IG, and TF-GR. These schemes replaced the global factor IDF in TF-IDF with feature selection functions like χ^2 statistic (CHI), information gain (IG), and gain ratio (GR). Similarly, Emmanuel et al. [7] expressed that the positive contribution of a feature to a category could be obtained by calculating its negative contributions to other categories and proposed the PIF (Positive Impact Factor) method for term weighting. Ren and Sohrab [8] have proposed two new term weighting schemes for text classification, named TF-IDF-ICF and TF-IDF-ICSDF. These schemes utilize, in addition to the TF-IDF information of terms, the inverse class frequency (ICF) and inverse class space density frequency (ICSDF) information, respectively, in the weighting process. The mentioned researchers emphasized that the TF-IDF-ICSDF term weighting scheme, particularly due to providing positive distinctiveness for both frequently and infrequently occurring terms, has shown promising results. Moreover, one of the most advanced algorithms in the recently published field of sentence extraction is the YAKE! model [9]. The model utilizes the concept of feature-based term weighting to measure the relevance of terms. The model also focuses on the position of stop words in the document to extract expressions.

Despite the successful term weighting schemes proposed in recent years, the ongoing introduction of new schemes in this field suggests that there is room for developing new schemes with weighting strategies that better reflect the discrimination potential of terms. Regardless of how up to date the proposed methods for term weighting are, there are instances where each of them may be insufficient in the weight calculation process, ignoring or failing to produce reasonable weights for terms in some extreme scenarios due to their weighting strategies. Therefore, this paper presents a new weighting model called Rough Multivariate Weighting Scheme (RMWS) and its mathematical derivative, Square Root Rough Multivariate Weighting Scheme (SRMWS). This study aims to provide the best representation by revealing hidden patterns in text data. For this purpose, the proposed scheme uses rough sets to reveal documents with special information. Firstly, with the help of rough sets, documents containing special information on a term basis are

identified. Then, the distribution relation of the term between the classes is revealed, and discrimination for the terms is determined based on these relations. Finally, term weight values are calculated depending on the distinctiveness of the terms. After determining the most distinctive features, classification algorithms are employed to evaluate the performance of the RMWS and SRMWS methods. In this study, for this purpose, Support Vector Machines (SVMs), K-Nearest Neighbors (KNNs), and Naive Bayes (NB) classifiers were utilized. Additionally, in experimental studies, the classification performance of the proposed method was compared with existing methods in the literature using evaluation criteria such as macro-F1 and micro-F1.

The flow of the rest of this study is as follows: Section 2 offers insights into related works and explores the background of the approaches used for comparison. In Section 3, a preliminary analysis of Rough Set is provided. Section 4 introduces the RMWS method, and Section 5 details the experimental work along with the obtained results. Finally, after including discussions about the study in Section 6, the study is concluded with Section 7.

2. Related Works

Term weighting processes are crucial tools to enhance the processes of extracting meaning and retrieving information from text documents. The selection of term weighting methods may vary depending on the purpose of use and the dataset under consideration. Depending on the dataset's utilization of class information, supervised or unsupervised methods are employed. Additionally, the preferred method may vary when dealing with a dataset that is binary or multiclass. Therefore, the literature includes weighting techniques with different working mechanisms. In this section, information about these techniques will be provided. Since there are common expressions and preliminaries in the weighting equations of these techniques, detailed information about these expressions and preliminaries is provided in Table 1 (for common expressions) and Table 2 (for preliminaries).

Table 1. Contingency table of the relationship between term and class [10].

<i>D</i> (Number of Documents)	Containing the Term	Not Containing the Term
<i>C</i> (belonging to class <i>C</i>)	t_x	$\overline{t_x}$
$\overline{C}$ (Not belonging to class <i>C</i>)	t_y	$\overline{t_y}$

Table 2. Preliminaries.

Notation	Description
t	refers to any term
d_k	refers to document k
$TF(t, d_k)$	specifies the frequency of term t in document d_k
$d(t)$	indicates the number of documents in which term t appears
D	represents the total number of documents
C	represents the number of classes in which the term appears
M	refers to the total number of classes
D_j	shows the total number of documents in class j
$d(t)_j$	represents the total number of documents where the term t occurs in class j .

The term frequency (TF) and term frequency–inverse document frequency (TF-IDF) methods, known as traditional term weighting schemes, are binary weighting methods that arise from information extraction [11]. Frequently occurring words in text data can increase computational costs. For example, the word “the” in English is commonly used, and its frequency is high; this means it is present in all documents. Such words often have low discriminative features in text classification processes and may, therefore, need to be excluded. To cope with this problem, the TF-IDF method is employed. This method

calculates a score by considering the frequency of a word in a particular document and its frequency across all documents. Using these scores, unique words representing important information in a specific document are identified and extracted from the text. Consequently, the IDF (inverse document frequency) value of a rarely-occurring word will be high, while the IDF value of a frequently-occurring word will be low. Therefore, in the TF-IDF method, by calculating the inverse document frequency values, low scores are assigned to common terms in the text collection, and high scores are assigned to rare terms. Mathematically expressed, TF-IDF [12] is calculated as follows:

$$W_{TF.IDF}(t) = TF(t, d_k) * \log\left(\frac{D}{d(t)}\right) \quad (1)$$

The supervised weighting method TF-PB, which utilizes class-internal and class-external possibility distributions, is an effective method for imbalanced datasets and binary classification [11]. This method calculates weighting as expressed in Equation (2) below.

$$W_{TF.IDF}(t) = TF(t, d_k) * \max_{i=1}^M \log\left(1 + \frac{t_x}{t_x} * \frac{t_x}{t_y}\right) \quad (2)$$

TF-RF is a weighting method used to measure the importance of terms in a document [13]. This method is a supervised learning technique, particularly employed in binary classification problems. It focuses on the frequency of occurrence of terms in positive and negative categories. Equation (3) below illustrates the TF-RF weighting formula.

$$W_{TF.RF}(t) = TF(t, d_k) * \max_{i=1}^M \left(\log\left(2 + \frac{t_x}{t_y}\right)\right) \quad (3)$$

TF-IDF-ICF is a supervised method that utilizes data related to the total number of documents and classes in which terms appear. Using this weighting method, the weight values of terms are obtained by multiplying the TF-IDF weight values for each term by ICF values [8]. Equation (4) represents the mathematical calculations.

$$W_{TF.IDF.ICF}(t) = TF(t, d_k) * \left(1 + \log\left(\frac{D}{d(t)}\right)\right) * \left(1 + \log\left(\frac{M}{C}\right)\right) \quad (4)$$

Weighting method TF-IDF-ICSDF is a supervised scheme that calculates the weight values of terms by multiplying the TF-IDF weight value of a term by ICSDF [8]. The key difference between the formula of this method and the previous one lies in considering not only the total number of documents in each class, where each term appears, but also the ratio of the number of documents in that class to the total number of documents in that class. Equation (5) includes the relevant weighting formula.

$$W_{TF.IDF.ICSDF}(t) = TF(t, d_k) * \left(1 + \log\left(\frac{D}{d(t)}\right)\right) * \left(1 + \log\left(\frac{M}{\sum_{j=1}^M \frac{d(t)_j}{D_j}}\right)\right) \quad (5)$$

TF-TRR is a supervised term weighting method that utilizes the distributions of positive and negative classes to accurately weigh terms for binary classification. In the TF-TRR weighting method, the TF value is used to determine how frequently a term appears in a document [14]. The TRR value is then used to assess the relevance of a term to the subject of the document. The mathematical representation is given by Equation (6).

$$W_{TF.TRR}(t) = \log(TF(t, d_k)) * \max_{i=1}^M \left(\log\left(2 + \frac{\frac{t_x}{t_x + t_x}}{\frac{t_y}{t_y + t_y}}\right)\right) \quad (6)$$

TF-IGM is a recently-proposed supervised weighting method for multiclass classification that provides weighting. Terms are calculated using the Inverse Geometric Moment (IGM) [15]. This method computes IGM by counting the number of documents in which each term occurs at least once for each class. These counts are then sorted in ascending order from the largest to the smallest. Equation (7) shows the mathematical formula used to calculate the IGM value for a term.

$$IGM(t) = \frac{f_1}{\sum_{r=1}^M f_r * r} \quad (7)$$

In this formula, $f_1 (r = 1, 2, \dots, M)$ represents the frequency of the term's class-based document frequency. In other words, f_1 indicates the number of text documents containing the term t in the r -th category, arranged in descending order. The TF-IGM weight of a term is calculated as shown in Equation (8).

$$W_{TF.IGM}(t) = TF(t, d_k) * (1 + \lambda * IGM(t)) \quad (8)$$

In this formula, λ is an adjustable constant typically defined in studies within the range of 5.0 to 9.0. Moreover, its default value is 7.0.

TF-IGM_{imp} attempts to address such weighting issues by incorporating $\log_{10} [D_{total}(t_{max}) / D_{t_{max}}]$ into the standard IGM formula, as shown in Equation (9). Terms are calculated using the Improved Inverse Geometric Moment (IGM) [16].

$$IGM(t) = \frac{f_1}{\sum_{r=1}^M f_r * r + \log_{10} [D_{total}(t_{max}) / D_{t_{max}}]} \quad (9)$$

$D_{total}(t_{max})$ is the number of total documents in the class in which t occurs most, and $D_{t_{max}}$ is the number of documents in the class in which t occurs most. $D_{t_{max}}$ also corresponds to f_1 in TF-IGM.

A variety of modifications and enhancements were discussed earlier to improve the performance of the TF-IDF scheme [17]. These changes are generally categorized as supervised and unsupervised methods. However, beyond these categories, there are also term weighting approaches with different working principles, known as vector-based term weighting [18]. Numerous term weighting schemes based on the vector concept have been proposed. Many of these models use n-grams to enhance algorithms in terms of understanding the semantics of a document [19–23]. However, in such approaches, as the word tree grows, the term space also expands, resulting in the widespread issue of high dimensionality. This, in turn, requires higher computational power and increases time complexity. Reducing dimensions in the high dimensionality problem makes the system lighter and more easily usable without excessive computational requirements. Consistently considering high dimensions may not be advantageous, as it can lead to uncertain results. For these reasons, the TF-IDF modification techniques mentioned earlier are often preferred. Indeed, the method proposed in this study is also a supervised TF-IDF multiclass approach.

3. Rough Set Theory

Rough set theory (RST) [24] is a mathematical approach for effective inference from incomplete and inconsistent data, uncovering hidden patterns without requiring additional information like membership functions. This makes it distinct from methods like Fuzzy Logic and Dempster–Shafer Theory. Widely applied in fields such as data mining, pattern recognition, and text mining, RST supports tasks like classification, rule generation, feature selection, and dimension reduction independently [25–27]. RST operates by organizing

uncertain data into rough sets and extracting approximate values for concepts. Its foundation lies in classifying relational databases to generate concepts and rules while identifying equivalence relations for further information discovery. Unlike Fuzzy Set Theory, which depends on membership functions with inherent uncertainty, RST uses precise boundary definitions to address uncertain problems. Key concepts of RST include the information system, which represents raw data collected from various fields. When the system includes decision attributes, it is termed a decision table; otherwise, it is an information table. Mathematically, let $S = (U, A, D)$ represent a decision table or information system, where $U = \{x_1, x_2, \dots, x_n\}$ denotes the universal set consisting of objects, $A = \{a_1, a_2, \dots, a_m\}$ denotes a conditional attribute set, and $D = \{d_1, d_2, \dots, d_k\}$ denotes a decision attribute set. If $D \neq \emptyset$, the system S is referred to as a decision table; otherwise, it is expressed as an information table. Table 3 provides an example of a decision table.

Table 3. An example of a decision table.

$x \in U$	a_1	a_2	a_3	a_4	d
x_1	1	2	1	2	1
x_2	0	2	1	1	1
x_3	1	1	2	2	3
x_4	2	2	2	3	3
x_5	2	2	1	3	3
x_6	1	1	0	2	2
x_7	0	1	1	1	1
x_8	2	2	0	3	2

The indiscernibility relation or discernibility relation determines the similarity or difference between objects in a knowledge system based on a subset of attributes. For any conditional attribute subset $T \subseteq A$, denoted as $IND(T)$, the T – indiscernibility relation is defined as follows:

$$IND(T) = \left\{ (x_i, x_j) \in U^2 \mid \forall a \in T, a(x_i) = a(x_j) \right\} \quad (10)$$

In the formula, the equivalence classes of the T – indiscernibility relation are represented as $[x]_T$.

Rough set theory introduces lower ($\underline{TX}$) and upper ($\overline{TX}$) approximations to analyze subsets $X \subseteq U$ using attributes $T \subseteq A$, as follows:

$$\begin{aligned} \underline{TX} &= \{x \mid [x]_T \subseteq X\} : \text{Objects definitively in } X. \\ \overline{TX} &= \{x \mid [x]_T \cap X \neq \emptyset\} :: \text{Objects possible in } X. \end{aligned} \quad (11)$$

These approximations help identify regions within rough sets, distinguishing certain membership ($x \in \underline{TX}$) from probabilistic membership ($x \in \overline{TX}$). For example, considering the decision table shown in Table 3, let $T = \{a_1, a_4\}$ and $X = \{x_1, x_2, x_5, x_7, x_8\}$. In this case, according to Equation (3), the pair $\langle \underline{TX}, \overline{TX} \rangle$ would be as follows:

- $\underline{TX} = \{x_2, x_7\}$: Certainmembers
- $\overline{TX} = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$: Possiblemembers

An accuracy measure of the set X for $T \subseteq A$ is defined as follows:

$$\mu_T(X) = \text{card}(\underline{TX}) / \text{card}(\overline{TX}) \quad (12)$$

It reflects the determinability of set X within U , ranging from 0 to 1. For the above example, $\mu_T(X) = \frac{2}{8} = 0.25$, indicating partial determinability.

4. Proposed Method: Rough Multivariate Weighting Scheme (RMWS)

In this section, a novel term weighting approach called Rough Multivariate Weighting Scheme (RMWS) is introduced, along with its mathematical derivative, the Square Root Rough Multivariate Weighting Scheme (SRMWS).

Text documents generally represent unstructured datasets. However, to enable processing by a classifier model, unstructured text data need to be transformed into a structured feature space. Creating a system that best represents the content of each document is crucial in this complex task. The vector space model is the most common method used for this purpose. The goal is to ensure that the vector space model effectively represents the dataset. Researchers are exploring various solutions to effectively represent document vectors, which is a significant challenge. In this context, when determining the relationships between the content of documents and terms, assigning appropriate weights to terms is a critical step. Therefore, there is a need for an effective term weighting scheme that assigns reasonable weights to terms based on their classification capabilities. While there are many term weighting schemes in the literature, it is challenging to claim that these schemes ideally reflect the true distinctive abilities of terms. For instance, the word “cat” may be more important in a text related to the “animal” category, while the conjunction “and” may not be as significant. Techniques that consider class information, such as “supervised” term weighting methods, can provide higher classification accuracy. These methods can better capture the importance of terms in different categories by assigning a separate weight for each term in each class. However, it can be argued that supervised weighting methods have not yet fully achieved effective representation of the class–term relationship, and ongoing research in this area indicates the need for further improvement. Therefore, effectively revealing the class–term relationship and identifying documents containing specific information related to this relationship form one of the most important foundations and motivations of this study. Thus, this study aims to provide the best representation by uncovering hidden patterns in text data, using proven rough set methods for successful extraction of hidden patterns from data. With the help of rough sets, documents containing specific information on a term-by-term basis have been identified. In text data, certain documents may contain indicative terms related to a person, field, topic, or object. The presence of these terms provides specific information. In this study, these specific patterns have been obtained using rough sets. The indiscernibility relation in rough sets has been utilized to select documents with distinctive frequencies for specific terms. The indiscernibility relation for documents related to a term is expressed as follows:

$$IND(T) = \{(d_i, d_j) \in D^2 | \forall t_r \in T, t_r(d_i) \neq t_r(d_j)\} \quad (13)$$

In the equation, D represents the document space, where d_i and d_j denote the i -th and j -th documents, respectively. Similarly, T represents the term space, and t_r represents the r -th term. Using the indiscernibility relation, equivalence classes denoted by $[d]_T$ are obtained, representing sets of documents. These equivalence classes include documents that contain specific terms in the document–term space, providing significant distinctive information. To determine how much information the equivalence classes offer, a subset approach is applied to the equivalence classes. For a document subset $D_k \subset D$, the document subset approximation $\underline{TD}_k$ is expressed as follows:

$$\underline{TD}_k = \{d \mid [d]_T \subseteq D_k\} \quad (14)$$

The information value provided by the determined subset $\underline{TD}_k$ depends on its ratio within the class. This ratio is referred to as Rough Rate (RR) in this study and is formulated as follows:

$$RR = \text{count}(\underline{TD}_k) / \text{count}(C_R) \quad (15)$$

In Equation (15), C_R represents the documents belonging to class R . After determining the document ratio providing specific information on a class basis over the equivalence classes, the distinctiveness of each term is determined. The distinctiveness of a term depends on some fundamental criteria:

A term that is frequently seen in a single class and not observed in other classes is considered distinctive.

A term that appears in some classes is relatively distinctive.

To reveal these pieces of information, the distribution among the classes of the term is examined. For this purpose, first, the possibility of the term within the class ($P(C_j|t_i)$) is calculated, and then the possibility of the term not occurring within the class ($P(\bar{t}_i|C_j)$) is computed.

Some coefficients are needed to fully reveal the distinctiveness of the term. Three constant coefficients were used in this study. Two of them are as follows:

β : It is used to balance the relationship between RR and $P(C_j|t_i)$. The calculated values of RR and $P(C_j|t_i)$ may differ significantly. This coefficient is employed to make the impact of RR more comprehensible.

λ : The value of $P(\bar{t}_i|C_j)$ may be zero. In such cases, this coefficient is used to enable the calculation.

In accordance with this information, the distinctiveness of a term is calculated using Equation (16) presented in this study.

$$RMWS(t_i) = \sum_{j=0}^M \frac{P(C_j|t_i) + \beta * RR}{P(\bar{t}_i|C_j) + \lambda} \quad (16)$$

After calculating the distinctiveness of the term, the weight calculation for the term is performed according to the proposed approach, as follows:

$$W_{RMWS}(t_i) = TF(t_i, d_k) * RMWS(t_i) * \alpha \quad (17)$$

Taking the square root of the $TF(t_i, d_k)$ value in Equation (17) yields the SRMWS value. Accordingly, SRMWS is calculated using Equation (18) below.

$$W_{SRMWS}(t_i) = \sqrt{(TF(t_i, d_k)) * RMWS(t_i) * \alpha} \quad (18)$$

In Equation (18), α represents the third coefficient.

1. α : This coefficient was used to pull the value into the appropriate range according to the classifier.

At this point, it is important to clarify that SRMWS is not an entirely new method; rather, it is a variation of the proposed approach, obtained by taking the square root of the term frequency (TF) values. This is analogous to the relationship between the TF-IGMimp [16] method and its derivative, SQRT_TF-IGMimp.

An Illustrative example:

The working principle of RMWS has been demonstrated through the application on a simple text document, represented by Table 4. The provided simple document collection in Table 4 illustrates the operational principle of RMWS.

Table 4. A simple collection of documents for RMWS.

Documents, Their Contents, and Classes		Pre-Processed Documents and Corresponding Term Frequency Vectors					
Documents	Content	Class/ Label	Pre-Processed Document Contents	Dimensions of Term Documents as Vectors			
				t_1 Apple	t_2 Cat	t_3 Dog	t_4 Mouse
D_1	"It's an apple"	C_1	<"apple">	1	0	0	0
D_2	"This is a cat. That one over there is a cat."	C_1	<"cat", "cat">	0	2	0	0
D_3	"This is a cat. That one over there is a dog."	C_2	<"cat", "dog">	0	1	1	0
D_4	"There is an apple here. There is also a dog. There is another dog."	C_2	<"apple", "dog", "dog">	1	0	2	0
D_5	"It's a mouse. That's a mouse over there."	C_3	<"mouse", "mouse">	0	0	0	2
D_6	"Here there is an apple, a dog, and a mouse."	C_3	<"apple", "dog", "mouse">	1	0	1	1

For the given collection of simple documents, the initial calculation involves determining the distinctiveness or importance of each term. The RMWS values for each term are computed using Equation (16) and, in this example, the values of all three coefficients (α , β , λ) have been set to 1.

$$RMWS(t_1) = \frac{\frac{1}{3} + 1 * \frac{0}{2}}{\frac{1}{2} + 1} + \frac{\frac{1}{3} + 1 * \frac{0}{2}}{\frac{1}{2} + 1} + \frac{\frac{1}{3} + 1 * \frac{0}{2}}{\frac{1}{2} + 1} = 0.6667$$

$$RMWS(t_2) = \frac{\frac{1}{2} + 1 * \frac{1}{2}}{\frac{1}{2} + 1} + \frac{\frac{1}{2} + 1 * \frac{1}{2}}{\frac{1}{2} + 1} + \frac{\frac{0}{2} + 1 * \frac{0}{2}}{\frac{2}{2} + 1} = 1.3334$$

$$RMWS(t_3) = \frac{\frac{0}{3} + 1 * \frac{0}{2}}{\frac{2}{2} + 1} + \frac{\frac{2}{3} + 1 * \frac{1}{2}}{\frac{0}{2} + 1} + \frac{\frac{1}{3} + 1 * \frac{0}{2}}{\frac{1}{2} + 1} = 1.3889$$

$$RMWS(t_4) = \frac{\frac{0}{2} + 1 * \frac{0}{2}}{\frac{2}{2} + 1} + \frac{\frac{0}{2} + 1 * \frac{0}{2}}{\frac{2}{2} + 1} + \frac{\frac{2}{2} + 1 * \frac{2}{2}}{\frac{0}{2} + 1} = 2.0000$$

Analyzing the distinctiveness of terms reveals that term t_4 has the highest distinctiveness, while the lowest value is assigned to t_1 . This distinction is due to t_4 being exclusively present in class C_3 and occurring with a specific majority, warranting a high score. Conversely, t_1 occurs in equal amounts across all three classes, leading to a lower score. Additionally, the distinctiveness of t_2 is only slightly less than that of t_3 . While t_2 occurs in classes C_1 and C_2 , t_3 appears in classes C_2 and C_3 . The difference lies in t_3 being more concentrated in one class (C_2) compared to t_2 , making it more distinctive and assigning it a higher score. Furthermore, intuitively, it is clearly evident that the distinctiveness of these terms aligns with the computation performed by RMWS. Table 5 provides a summary of these evaluations.

Table 5. A summary of the calculated values for the simple example given in Table 4.

Terms	# of Class/ # of Doc	Doc Frequencies	Class Name	Sorted According to Distinctiveness (Intuitively)	RMWS(t_i) Values
t_1	3/6	{1, 0, 0, 1, 0, 1}	{C ₁ , C ₁ , C ₂ , C ₂ , C ₃ , C ₃ }	$t_1 < t_2 < t_3 < t_4$	0.6667
t_2		{0, 2, 1, 0, 0, 0}			1.3334
t_3		{0, 0, 1, 2, 0, 1}			1.3889
t_4		{0, 0, 0, 0, 2, 1}			2.0000

Now, the weight set for each term can be calculated. For this purpose, the calculated term weights are used to update the value set for the terms, according to Equation (17).

$$W_{RMWS}(t_1) = \begin{bmatrix} 0.6667 \\ 0.0000 \\ 0.0000 \\ 0.6667 \\ 0.0000 \\ 0.6667 \end{bmatrix}$$

$$W_{RMWS}(t_2) = \begin{bmatrix} 0.0000 \\ 2.6667 \\ 1.3333 \\ 0.6667 \\ 0.0000 \\ 0.0000 \end{bmatrix}$$

$$W_{RMWS}(t_3) = \begin{bmatrix} 0.0000 \\ 0.0000 \\ 1.3889 \\ 2.7778 \\ 0.0000 \\ 1.3889 \end{bmatrix}$$

$$W_{RMWS}(t_4) = \begin{bmatrix} 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ 4.0000 \\ 2.0000 \end{bmatrix}$$

After the weighting of the data is completed, the remaining step is to observe the impact of these weightings on the classifier. Finally, the illustrative representation of the working principle of the proposed approach related to RMWS is provided in Figure 1.

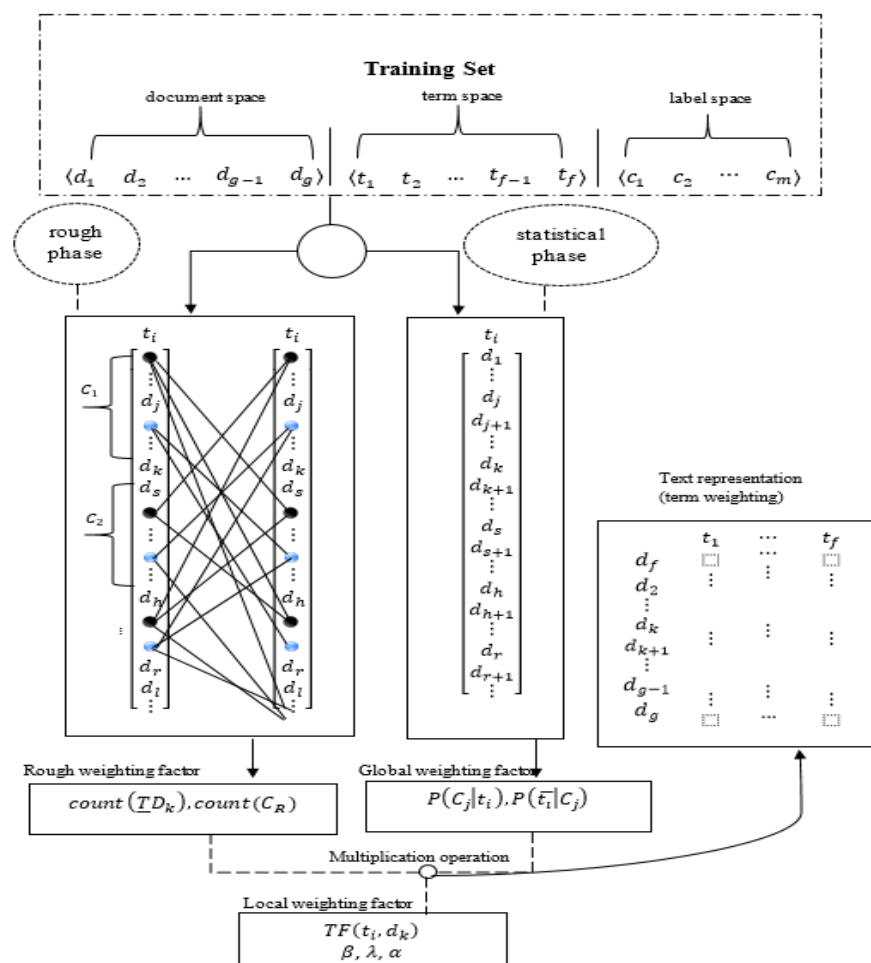


Figure 1. Representative illustration of the working mechanism of RMWS.

5. Experimental Works

In this section, we present the outcomes of our experimental endeavors. Initially, we provide concise details about the employed datasets, followed by an elucidation of the chosen success metrics and used classifiers. Then, various tests have been conducted to determine suitable values for constant coefficients. Subsequently, we assess the impact of terms weighted by proposed methods on the efficacy of classifiers. For this purpose, we compare the performance of proposed methods with the term weighting methods outlined in the preceding sections. The performance of classifiers using terms weighted by their corresponding term weighting algorithms is discussed in this section. Lastly, a set of statistical analyses is incorporated into the conclusion of this section to ascertain whether the performance enhancement achieved by the proposed methods is statistically significant in comparison to other methods.

5.1. Datasets

Within the scope of experimental studies, three different reference datasets have been used for text classification: Reuters-21578, 10 Mini Newsgroups, and Enron1. These collections have been preferred due to their characteristics, such as balanced and unbalanced or multiple and binary classes. In other words, the advantage of this diversity has been utilized to make a fair comparison among the proposed term weighting methods.

Reuters-21578 is a collection of documents published on the Reuters news channel in 1987. The documents have been compiled and indexed into categories. Additionally, the dataset includes the first ten classes of the well-known Reuters ModApte split [28], widely

used in many text classification research studies. This dataset is termed unbalanced because it contains a different number of documents in each class and is multiclass. In the context of this study, experiments were conducted on the training and test splits of Reuters-21578. During the feature extraction process, multi-labeled documents were removed from the Reuters-21578 data, and subsequently, the two classes named 'wheat' and 'corn' were deleted because these two classes became empty. Further details about the Reuters-21578 dataset are presented in Table 6.

Table 6. Reuters_21578 dataset.

Id	Category/Label	Training Samples	Testing Samples
1	earn	2840	1083
2	acq	1596	696
3	money-fx	206	87
4	grain	41	10
5	crude	253	121
6	trade	251	117
7	interest	190	75
8	ship	108	36

The 20 Newsgroups dataset [28] is divided into two subsets, totaling approximately 18,000 newsgroups and covering 20 different topics: one for training or development and the other for testing or performance evaluation. The division between the training and test sets is based on messages sent before and after a specific date. The 20 Mini Newsgroups dataset used in this study is a mini subset of the popular text collection 20 Newsgroups, containing ten different classes. This dataset has a balanced structure, meaning the number of documents in each class is equal, and it is multiclass. In the experiments, the dataset was manually divided into training (70%) and test sections (30%). Detailed information about the 10 Mini Newsgroups dataset is provided in Table 7.

Table 7. The 10 Mini Newsgroups dataset.

Id	Category/Label	Training Samples	Testing Samples
1	alt.atheism	700	300
2	comp.graphics	700	300
3	comp.os.ms-windows.misc	700	300
4	comp.sys.ibm.pc.hardware	700	300
5	comp.sys.mac.hardware	700	300
6	comp.windows.x	700	300
7	misc.forsale	700	300
8	rec.autos	700	300
9	rec.motorcycles	700	300
10	rec.sport.baseball	700	300

The Enron–Spam dataset, a source described in the publication ‘Spam Filtering with Naive Bayes—Which Naive Bayes?’ by V. Metsis, I. Androutsopoulos, and G. Paliouras [29], was collected by the mentioned authors. The dataset contains a total of 17,171 spam and 16,545 non-spam email messages (a total of 33,716 emails). In this study, a subset of the Enron–Spam dataset, named Enron1, was used. This dataset is imbalanced as it contains a different number of documents in each class. Additionally, it is used for binary classification as it consists of only two classes. Content information related to Enron1 is presented in Table 8.

Table 8. Enron1 dataset.

Id	Category/Label	Training Samples	Testing Samples
1	spam	1000	500
2	legitimate	2448	1224

5.2. Assessment of Performance

This study employed micro-F1 and macro-F1 scores as key performance metrics to assess the efficacy of feature selection methods. The F1 score, incorporating both precision and recall, was utilized in the evaluation. In the macro-averaging approach, the F1 score is individually calculated for each class, and subsequently, the mean across all classes is determined [30]. The computation of the macro-F1 score is illustrated below in Equation (19).

$$Macro - F1 = \frac{\sum_{i=1}^M F_i}{M}, F_i = 2 * \frac{p_i r_i}{p_i + r_i} \quad (19)$$

In this equation, p_i and r_i represent the precision and recall scores of class j , respectively.

Conversely, the F1 score is computed in micro-averaging without considering class-specific information. Therefore, all classification decisions are taken into account across all corpora. In the evaluation of imbalanced datasets, the micro-averaging approach may result in the dominance of large classes over small ones. However, this scenario might not be applicable to balanced datasets, where the number of documents in each class is equal, and the feature counts are similar. The calculation of the micro-F1 score is illustrated below in Equation (20).

$$Macro - F1 = 2 * \frac{p * r}{p + r} \quad (20)$$

where p and r represent precision and recall values across all classes. The micro-F1 score, influenced by the prevalence of larger classes with more documents, might not ensure a fair assessment in all scenarios. Consequently, to achieve a more unbiased evaluation, the micro-F1 score is preferred for balanced datasets, whereas the macro-F1 score is employed for imbalanced datasets. In this study, a diverse range of datasets, including both balanced and imbalanced ones, were employed. Thus, the experiments utilized both micro-F1 scores and macro-F1 criteria to ensure comprehensive evaluation.

5.3. Classifiers

The proposed method, RMWS, is not dependent on the learning model as it is a term weighting technique. Therefore, to explore the impact of the features incorporated into the ultimate feature set on classification accuracy, four distinct classifiers were utilized in the experimental phase of the study. Concise explanations of the classifiers employed are outlined in Table 9.

Table 9. Classifiers used for investigating the performance of the proposed feature selection method.

Classifier	Description
Support Vector Machines (SVMs)	The SVM is widely recognized as a potent classifier in the literature, founded on the principle of margin maximization. Additionally, it offers both linear and non-linear variations, contingent on the type of kernel applied.
K-Nearest Neighbors (KNNs)	The KNN is a non-parametric method that retains all existing cases and assigns classifications to new cases based on similarity measures. This approach has found extensive use in statistical estimation and pattern recognition.
Naïve Bayes (NB)	The NB method relies on Bayes' theorem, making independence assumptions among predictors.

5.4. Coefficient Analysis

This study utilizes three adjustable coefficients, akin to a potentiometer, to achieve specific balances. To set these coefficients to appropriate values and observe their impact on the results, micro-F1 and macro-F1 results for SVM, KNN, and NB classifiers were obtained on the Reuters_21578 dataset with term sizes of 500, 1000, and 2000. The results obtained are presented in Tables 10–12. Furthermore, the best-performing results presented in the tables are emphasized using bold font to facilitate the reader’s focus on the most significant outcomes.

Table 10. On the Reuters_21578 dataset, the macro-F1 and micro-F1 results of classifiers for varying α values while keeping λ and β constant.

(SVM)	α	β	λ	Micro-F1			Macro-F1		
				500	1000	2000	500	1000	2000
RMWS	1	1	1	90.9023	90.6015	91.0150	81.5121	79.8833	81.1063
RMWS	0.8	1	1	90.8271	90.9774	90.9398	81.4307	80.6318	80.7550
RMWS	1.7	1	1	90.6015	90.3383	90.7895	81.0764	79.5086	80.9575
RMWS	3	1	1	90.3759	90.1128	90.5263	80.9442	79.1551	80.5139
SRMWS	1	1	1	91.6165	91.5038	91.9549	82.6334	81.6160	82.8792
SRMWS	0.8	1	1	91.5414	91.5414	92.0301	82.5663	81.3795	82.9334
SRMWS	1.7	1	1	91.1654	90.9398	91.7669	81.9519	81.1528	82.6218
SRMWS	3	1	1	90.6391	90.6391	91.4662	81.4506	80.5691	82.0570
(KNN)	α	β	λ	Micro-F1			Macro-F1		
				500	1000	2000	500	1000	2000
RMWS	1	1	1	89.4361	89.3985	88.6466	79.0139	78.9318	77.9083
RMWS	0.8	1	1	89.4361	89.3985	88.6466	79.0139	78.9318	77.9083
RMWS	1.7	1	1	89.4361	89.3985	88.6466	79.0139	78.9318	77.9083
RMWS	3	1	1	89.4361	89.3985	88.6466	79.0139	78.9318	77.9083
SRMWS	1	1	1	91.1654	91.0902	89.9624	81.0113	81.0030	79.6915
SRMWS	0.8	1	1	91.1654	91.0902	89.9624	81.0113	81.0030	79.6915
SRMWS	1.7	1	1	91.1654	91.0902	89.9624	81.0113	81.0030	79.6915
SRMWS	3	1	1	91.1654	91.0902	89.9624	81.0113	81.0030	79.6915
(NB)	α	β	λ	Micro-F1			Macro-F1		
				500	1000	2000	500	1000	2000
RMWS	1	1	1	87.4436	86.6917	85.6391	78.4903	77.5530	76.5597
RMWS	0.8	1	1	83.3083	78.4586	75.8647	74.7645	69.5383	66.5792
RMWS	1.7	1	1	87.7820	86.8797	86.0150	78.9820	77.7272	77.1018
RMWS	3	1	1	87.2932	85.0752	82.9699	77.1535	75.4056	72.8317
SRMWS	1	1	1	72.1429	64.6992	62.7820	65.2112	59.0703	56.7011
SRMWS	0.8	1	1	57.5188	50.1880	47.5940	53.6716	48.6339	45.0508
SRMWS	1.7	1	1	87.7820	86.8797	86.0150	78.9820	77.7272	77.1018
SRMWS	3	1	1	87.2932	85.0752	82.9699	77.1535	75.4056	72.8317

When examining the tables, Table 10 has been created to analyze the effect of the α coefficient. In this table, β and λ values are kept constant while the α value is varied. The α value is tested for random values greater and less than 1, $\alpha = \{0.8, 1, 1.7, 3\}$. Accordingly, the α coefficient has negligible impact on the KNN classifier, whereas it exerts a significant influence on the NB classifier. For the SVM classifier, this coefficient has a relatively modest effect. The performance of the NB classifier is notably enhanced when the value of this coefficient exceeds 1. For instance, with $\alpha = 0.8$ and 500 terms, the macro-F1 value is 57.5188%, while for $\alpha = 1.7$, it increases to 87.7820%. This implies a positive enhancement in the performance of the NB classifier when the α coefficient is greater than 1. However, a careful examination of Table 2 reveals a decline in results when this value is 3. Therefore, it

is necessary for the α value to be within a certain range. The results show that $\alpha = 1.7$ is a reasonable value.

Table 11. On the Reuters_21578 dataset, the macro-F1 and micro-F1 results of classifiers for varying β values while keeping α and λ constant.

(SVM)	α	β	λ	Micro-F1			Macro-F1		
				500	1000	2000	500	1000	2000
RMWS	1	1	1	90.9023	90.6015	91.0150	81.5121	79.8833	81.1063
RMWS	1	0.8	1	90.9398	90.5639	91.0150	81.5558	79.7980	81.1063
RMWS	1	1.5	1	90.9398	90.6015	90.9774	81.6045	79.8297	81.1115
RMWS	1	2.5	1	90.9398	90.6015	91.0526	81.6533	79.8132	81.2780
SRMWS	1	1	1	91.5414	91.5038	91.9549	82.6334	81.6160	82.8792
SRMWS	1	0.8	1	91.5414	91.4662	91.9549	82.6334	81.5750	82.8792
SRMWS	1	1.5	1	91.6165	91.5038	91.9549	82.7803	81.6160	82.8792
SRMWS	1	2.5	1	91.6541	91.5038	91.9925	82.8957	81.6160	83.1633

(KNN)	α	β	λ	Micro-F1			Macro-F1		
				500	1000	2000	500	1000	2000
RMWS	1	1	1	89.4361	89.3985	88.6466	79.0139	78.9318	77.9083
RMWS	1	0.8	1	89.3233	89.3609	88.6090	78.9281	78.8774	77.8208
RMWS	1	1.5	1	89.3985	89.4361	88.6842	78.9990	78.9531	78.0595
RMWS	1	2.5	1	89.4361	89.3609	88.6842	79.1744	78.9561	77.9125
SRMWS	1	1	1	91.1654	91.0902	89.9624	81.0113	81.0030	79.6915
SRMWS	1	0.8	1	91.1278	91.0150	89.9624	80.9669	80.8545	79.6715
SRMWS	1	1.5	1	91.2406	91.0902	90.0000	81.1954	80.9031	79.7053
SRMWS	1	2.5	1	91.2406	91.1278	90.1504	81.2205	81.0138	79.6607

(NB)	α	β	λ	Micro-F1			Macro-F1		
				500	1000	2000	500	1000	2000
RMWS	1	1	1	87.4436	86.6917	85.6391	78.4903	77.5530	76.5597
RMWS	1	0.8	1	87.4436	86.6917	85.6391	78.4903	77.5530	76.5597
RMWS	1	1.5	1	87.4060	86.6917	85.6391	78.4638	77.5944	76.6018
RMWS	1	2.5	1	87.4060	86.6917	85.6391	78.4638	77.5944	76.6018
SRMWS	1	1	1	72.1429	64.6992	62.7820	65.2112	59.0703	56.7011
SRMWS	1	0.8	1	72.0677	64.6241	62.5188	65.1612	58.9988	56.5718
SRMWS	1	1.5	1	72.0677	64.8496	62.8571	65.1473	59.3228	56.9782
SRMWS	1	2.5	1	73.5714	65.8647	63.6842	66.1758	59.8109	57.3950

Table 11 was created to analyze the β value. In this table, other coefficients are fixed, and the β variable has random values: $\beta = \{0.8, 1, 1.5, 2.5\}$. The β coefficient generally appears to positively influence results when its value is high for this dataset. This coefficient is a constant used within the method to adjust its contribution to uncovering hidden patterns within the dataset. A high value signifies the importance of rough clustering in revealing specific information. In this analytical study, the value was set at a maximum of 2.5, and optimal results were obtained for all classifiers at this value. It indicates that taking values slightly higher than 2.5 for the β parameter, and in proportion to it, will improve the results.

Table 12 was created to show the effect of the λ coefficient. The $\{0.1, 0.4, 0.8, 1\}$ random values were taken for λ . The λ value is a constant given to facilitate calculations in cases where the denominator in the method equation is zero. Therefore, values less than 1 have been assigned. This coefficient is observed to vary on a class-by-class basis.

As a result, the coefficients predicted to be ideal for the Features dataset are provided in Table 13 below. The fixed values mentioned in the table were used in the experimental section of this study.

Table 12. On the Reuters_21578 dataset, the macro-F1 and micro-F1 results of classifiers for varying λ values while keeping α and β constant.

(SVM)	α	β	λ	Micro-F1			Macro-F1		
				500	1000	2000	500	1000	2000
RMWS	1	1	1	90.9023	90.6015	91.0150	81.5121	79.8833	81.1063
RMWS	1	1	0.8	90.9774	90.6015	91.0902	81.6651	79.7989	81.2997
RMWS	1	1	0.4	90.8271	90.5263	90.9774	81.4297	79.8698	81.1589
RMWS	1	1	0.1	90.4887	90.3008	90.6767	81.0101	79.4333	80.4257
SRMWS	1	1	1	91.5414	91.5038	91.9549	82.6334	81.6160	82.8792
SRMWS	1	1	0.8	91.5414	91.5789	92.0301	82.5784	81.5823	83.0536
SRMWS	1	1	0.4	91.2030	91.3534	91.9173	82.0896	81.7809	82.7391
SRMWS	1	1	0.1	90.9398	90.8647	91.6165	81.7736	81.0307	82.1836

(KNN)	α	β	λ	Micro-F1			Macro-F1		
				500	1000	2000	500	1000	2000
RMWS	1	1	1	89.4361	89.3985	88.6466	79.0139	78.9318	77.9083
RMWS	1	1	0.8	89.3233	89.3609	88.4586	78.9246	78.8684	77.8730
RMWS	1	1	0.4	89.3985	89.3233	88.1955	79.3376	78.6851	77.4002
RMWS	1	1	0.1	89.0602	88.1579	87.2556	79.0722	77.4602	76.8887
SRMWS	1	1	1	91.1654	91.0902	89.9624	81.0113	81.0030	79.6915
SRMWS	1	1	0.8	91.0526	90.8647	90.0376	81.0238	80.6670	79.6909
SRMWS	1	1	0.4	90.7519	90.3008	90.2256	80.6432	79.3284	79.3950
SRMWS	1	1	0.1	89.0602	89.4361	89.0226	79.0722	77.7635	77.2477

(NB)	α	β	λ	Micro-F1			Macro-F1		
				500	1000	2000	500	1000	2000
RMWS	1	1	1	87.4436	86.6917	85.6391	78.4903	77.5530	76.5597
RMWS	1	1	0.8	87.4060	86.6917	85.8271	78.4638	77.6872	77.0840
RMWS	1	1	0.4	87.8571	86.8797	85.9774	79.0259	77.9154	77.0256
RMWS	1	1	0.1	87.7444	86.7669	85.9774	78.1979	77.0973	76.5829
SRMWS	1	1	1	72.1429	64.6992	62.7820	65.2112	59.0703	56.7011
SRMWS	1	1	0.8	82.5940	76.2782	73.0451	73.9820	67.3905	64.7270
SRMWS	1	1	0.4	87.8571	86.8797	85.9774	79.0259	77.9154	77.0256
SRMWS	1	1	0.1	87.7444	86.7669	85.9774	78.1979	77.0973	76.5829

Table 13. Appropriate coefficient values calculated as a result of the analysis of the Reuters_21578 dataset by classifier.

Classifiers	Coefficients		
	α	β	λ
SVM	1	2.5<	0.8
KNN	1	2.5<	1
NB	1.7	2.5<	0.4

Again, when the tables created for coefficient analyses and Table 13 are examined, the relationship between coefficients can be analyzed on a classifier basis. To perform this analysis, the binary interaction effects of coefficients were examined on a classifier basis. Accordingly, binary coefficient relationship graphs for each classifier are given in order in Figures 2–4.

In order to see whether the values given in Table 13 would have an effect on the results as stated, some more tests were performed. In these tests, the coefficient values given for each classifier in Table 13 were compared with the best results for the same classifier in Tables 10–12. The results obtained are given in Table 14 and the best results are highlighted in bold within the table.

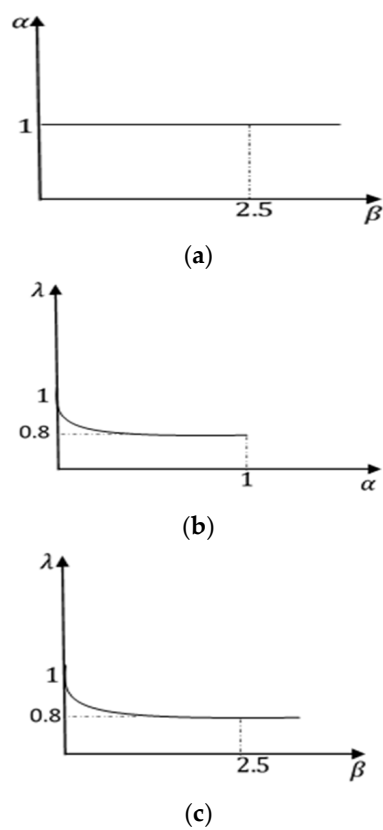


Figure 2. Correlation between coefficients for SVM classifier: (a) α and β , (b) α and λ , and (c) λ and β .

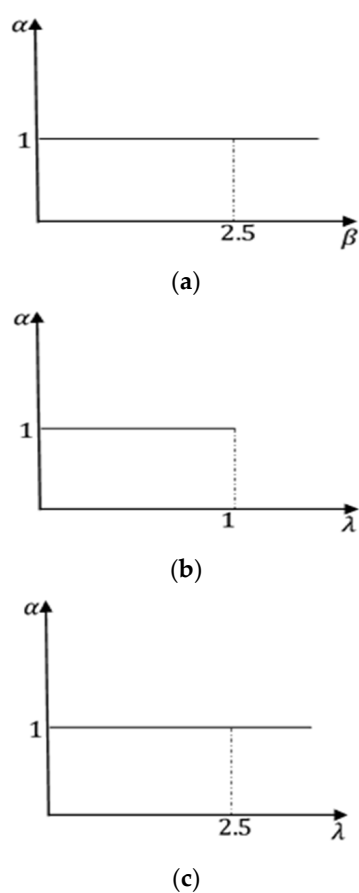


Figure 3. Correlation between coefficients for KNN classifier: (a) α and β , (b) α and λ , and (c) λ and β .

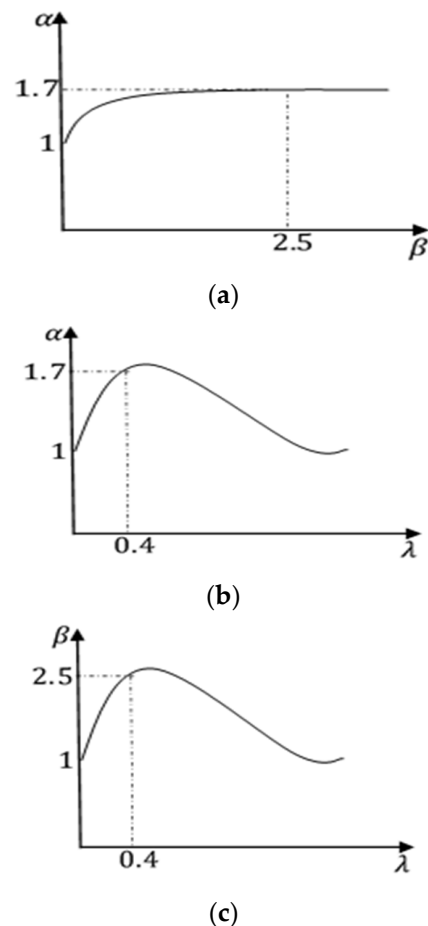


Figure 4. Correlation between coefficients for NB classifier: (a) α and β , (b) α and λ , and (c) λ and β .

Table 14 shows that the proposed methods give better results at the coefficient values determined on a classifier basis. This situation indicates both the necessity and importance of the analyses performed.

Note: Similar procedures have been applied to other datasets in this study. The determined coefficient values have yielded comparable results in these datasets as well. Therefore, the values in Table 13 have been used for all datasets and accepted as default values for the proposed methods.

5.5. Accuracy Analysis

This section presents a comprehensive comparison of the proposed term weighting methods with established approaches, namely TF-IDF, ICF, ICSDf, TRR, IGM, SIGM, IGMimp, and SIGMimp. The performance evaluation is conducted on various term dimensions: 750, 1500, 2500, 3750, 4500, 5750, 6500, and 7750. The Distinguishing Feature Selector (DFS) [31] algorithm is utilized for term selection within the dimension selection process. Term selection methods are frequently used to address the high dimensionality of term spaces and to assess the effectiveness of proposed techniques in text classification. The DFS approach is preferred in this study because it provides effective results. This study investigates the performance of SVM, KNN, and NB classifiers in terms of macro-F1 and micro-F1 criteria on the weighted term dimensions. The obtained results are presented in separate graphs for each dataset.

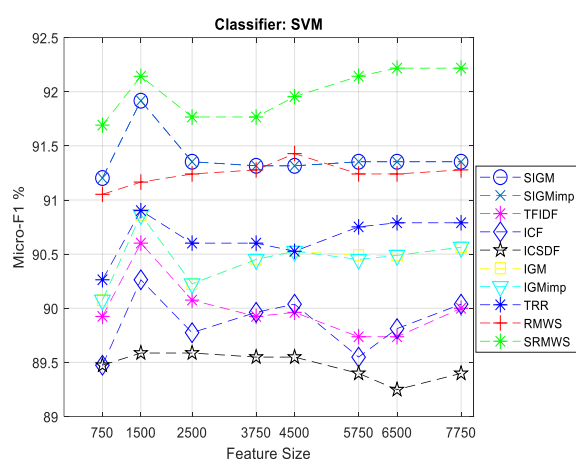
Figures 5–7: Macro-F1 and micro-F1 results for the Reuters_21578 dataset.

Table 14. Comparison of the coefficient values given for each classifier in Table 13 with the best results for the same classifier in Tables 10–12.

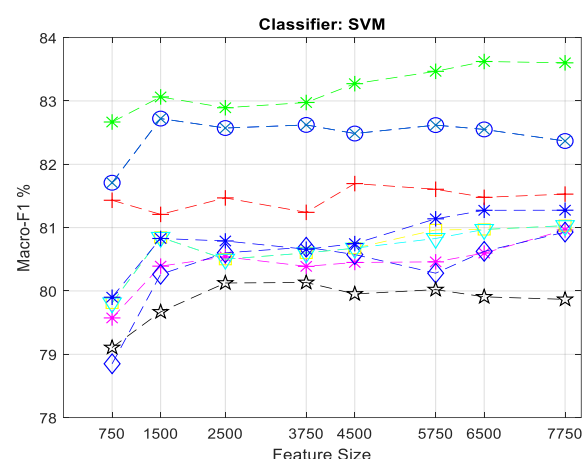
(SVM)	α	β	λ	Micro-F1			Macro-F1		
				500	1000	2000	500	1000	2000
RMWS	1	1	0.8	90.9774	90.6015	91.0902	81.6651	79.7989	81.2997
RMWS	1	2.5	1	90.9398	90.6015	91.0526	81.6533	79.8132	81.2780
RMWS	1	1	1	90.9023	90.6015	91.0150	81.5121	79.8833	81.1063
RMWS	1	2.5	0.8	90.9774	90.6767	91.1278	81.7298	79.8901	81.2650
RRMWS	1	1	0.8	91.5414	91.5789	92.0301	82.5784	81.5823	83.0536
RRMWS	1	2.5	1	91.6541	91.5038	91.9925	82.8957	81.6160	83.1633
RRMWS	0.8	1	1	91.6165	91.5414	92.0301	82.5663	81.3795	82.9334
RRMWS	1	2.5	0.8	91.7669	91.6917	92.0301	82.8982	81.8484	82.9315

(KNN)	α	β	λ	Micro-F1			Macro-F1		
				500	1000	2000	500	1000	2000
RMWS	1	1	1	89.4361	89.3985	88.6466	79.0139	78.9318	77.9083
RMWS	1	2.5	1	89.4361	89.3609	88.6842	79.1744	78.9561	77.9125
RMWS	1	1	1	89.4361	89.3985	88.6466	79.0139	78.9318	77.9083
RMWS	1	2.5	1	89.4361	89.3609	88.6842	79.1744	78.9561	77.9125
RRMWS	1	1	1	91.1654	91.0902	89.9624	81.0113	81.0030	79.6915
RRMWS	1	2.5	1	91.2406	91.1278	90.1504	81.2205	81.0138	79.6607
RRMWS	1	1	1	91.1654	91.0902	89.9624	81.0113	81.0030	79.6915
RRMWS	1	2.5	1	91.2406	91.1278	90.1504	81.2205	81.0138	79.6607

(NB)	α	β	λ	Micro-F1			Macro-F1		
				500	1000	2000	500	1000	2000
RMWS	1	1	0.4	87.8571	86.8797	85.9774	79.0259	77.9154	77.0256
RMWS	1	2.5	1	87.4060	86.6917	85.6391	78.4638	77.5944	76.6018
RMWS	1.7	1	1	87.7820	86.8797	86.0150	78.9820	77.7272	77.1018
RMWS	1.7	2.5	0.4	87.2932	85.0752	82.9699	77.153	75.4056	72.8317
RMWS	1.7	2.5	1	87.8947	86.9549	86.1278	79.2004	77.8838	77.3484
RRMWS	1	1	0.4	87.8571	86.8797	85.9774	79.0259	77.9154	77.0256
RRMWS	1	2.5	1	73.5714	65.8647	63.6842	66.1758	59.8109	57.3950
RRMWS	1.7	1	1	87.7820	86.8797	86.0150	78.9820	77.7272	77.1018
RRMWS	1.7	2.5	0.4	87.2932	85.0752	82.9699	77.1535	75.4056	72.8317
RRMWS	1.7	2.5	1	87.8947	86.9549	86.1278	79.2004	77.8838	77.3484



(a)



(b)

Figure 5. (a) Micro-F1 and (b) macro-F1 metric results for multiclass text classification using the SVM classifier and ten different term weighting schemes on the Reuters-21578 corpus, with different feature sizes.

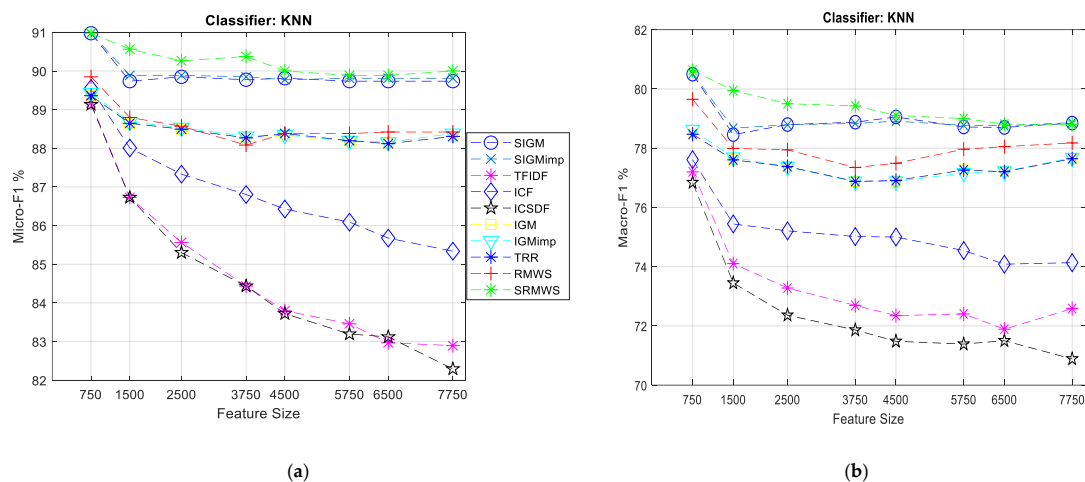


Figure 6. (a) Micro-F1 and (b) macro-F1 metric results for multiclass text classification using the KNN ($k = 5$) classifier and ten different term weighting schemes on the Reuters-21578 corpus, with different feature sizes.

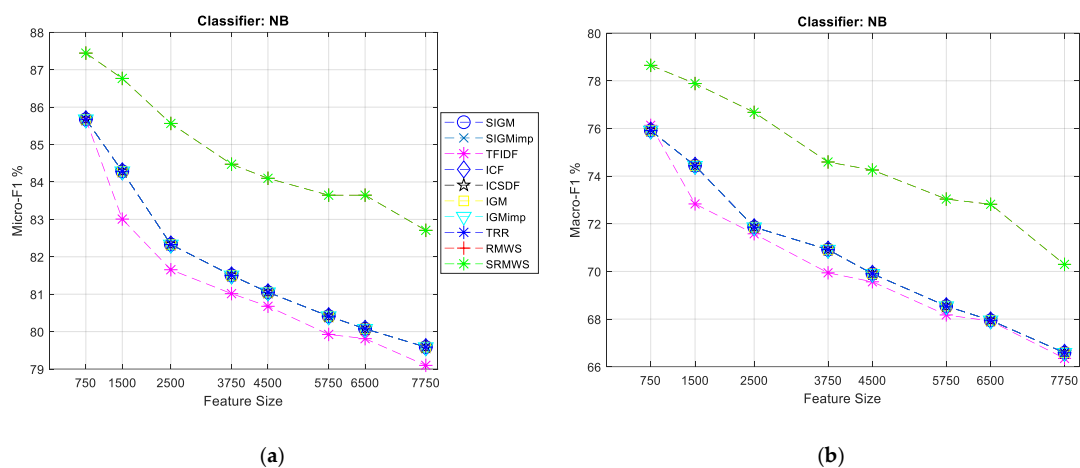


Figure 7. (a) Micro-F1 and (b) macro-F1 metric results for multiclass text classification using the NB classifier and ten different term weighting schemes on the Reuters-21578 corpus, with different feature sizes.

Figures 8–10: Macro-F1 and micro-F1 results for the 10 Mini Newsgroups dataset.

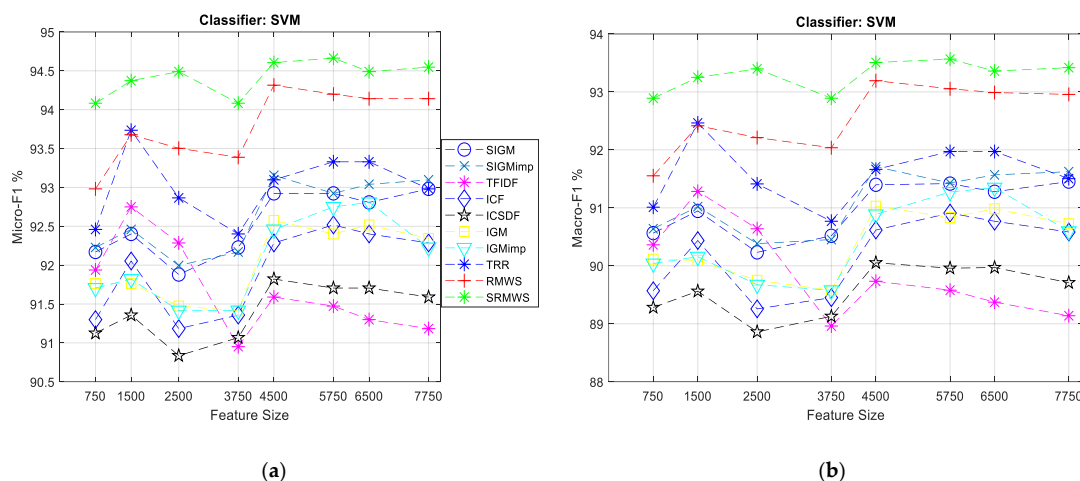


Figure 8. (a) Micro-F1 and (b) macro-F1 metric results for binary class text classification using the SVM classifier and ten different term weighting schemes on the Enron1 corpus, with different feature sizes.

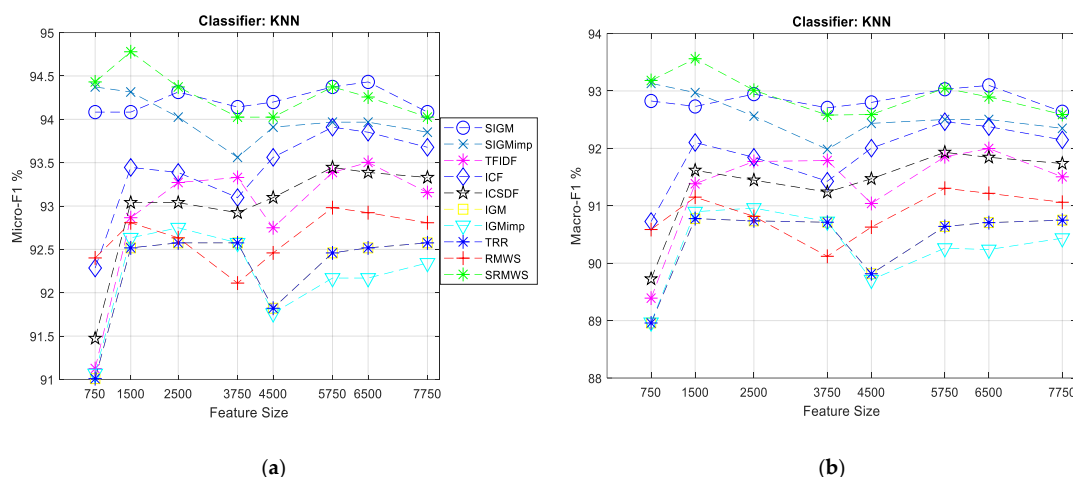


Figure 9. (a) Micro-F1 and (b) macro-F1 metric results for binary class text classification using the KNN ($k = 5$) classifier and ten different term weighting schemes on the Enron1 corpus, with different feature sizes.

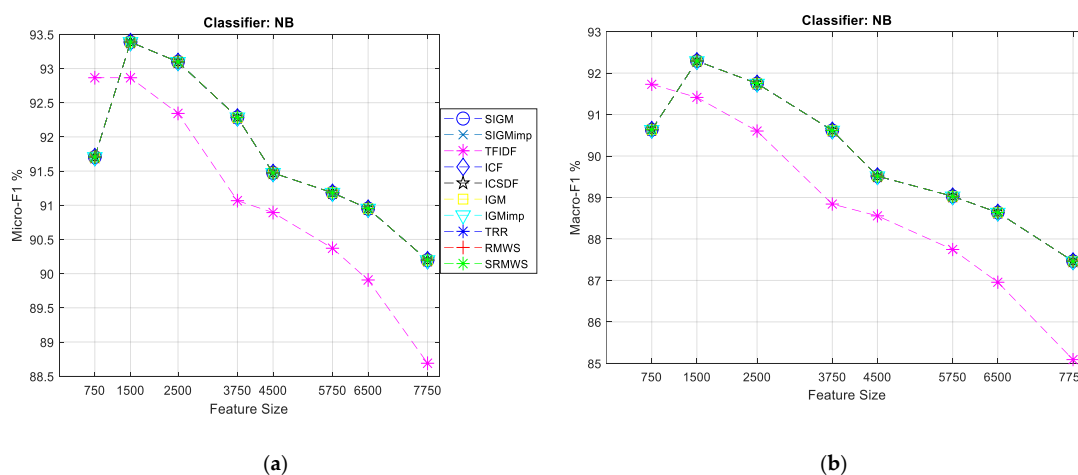


Figure 10. (a) Micro-F1 and (b) macro-F1 metric results for binary class text classification using the NB classifier and ten different term weighting schemes on the Enron1 corpus, with different feature sizes.

Figures 11–13: Macro-F1 and micro-F1 results for the Enron1 dataset.

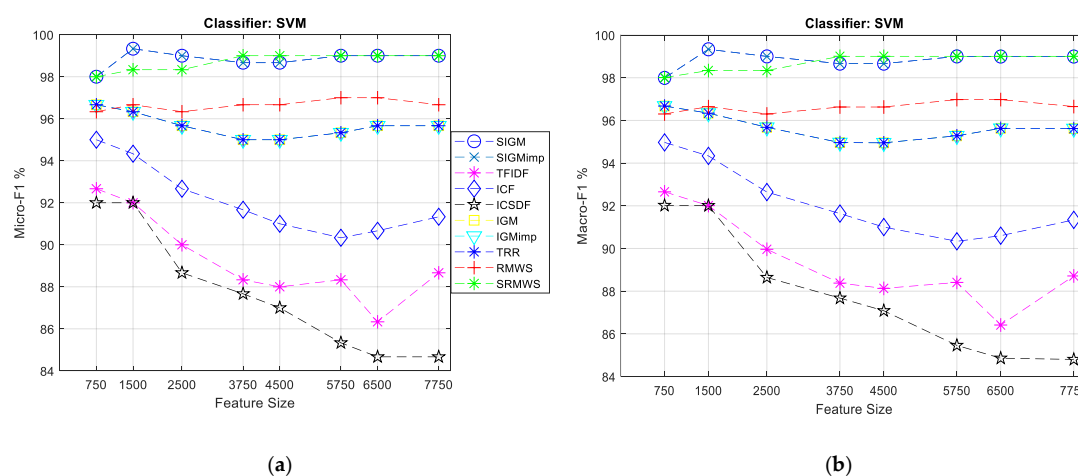


Figure 11. (a) Micro-F1 and (b) macro-F1 metric results for multiclass text classification using the SVM classifier and ten different term weighting schemes on the 10 Mini Newsgroups corpus, with different feature sizes.

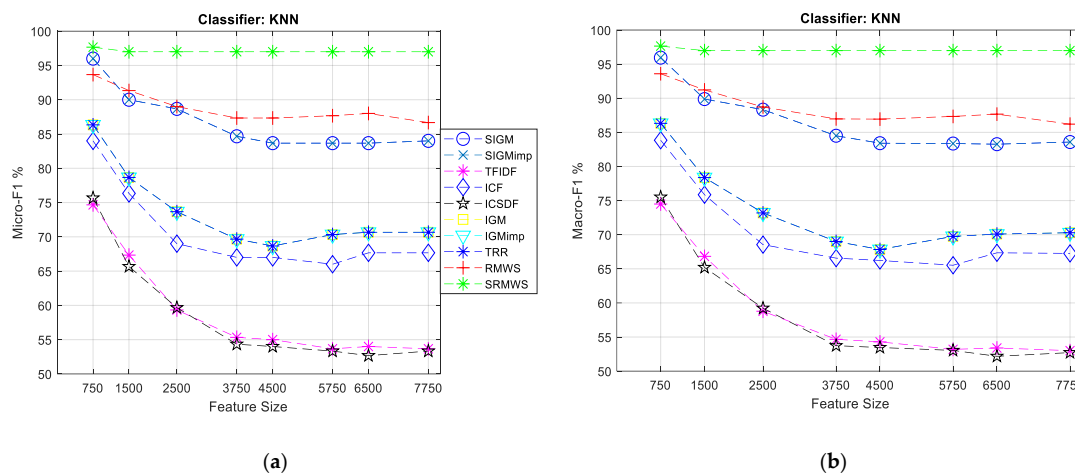


Figure 12. (a) Micro-F1 and (b) macro-F1 metric results for multiclass text classification using the KNN ($k = 5$) classifier and ten different term weighting schemes on the 10 Mini Newsgroups corpus, with different feature sizes.

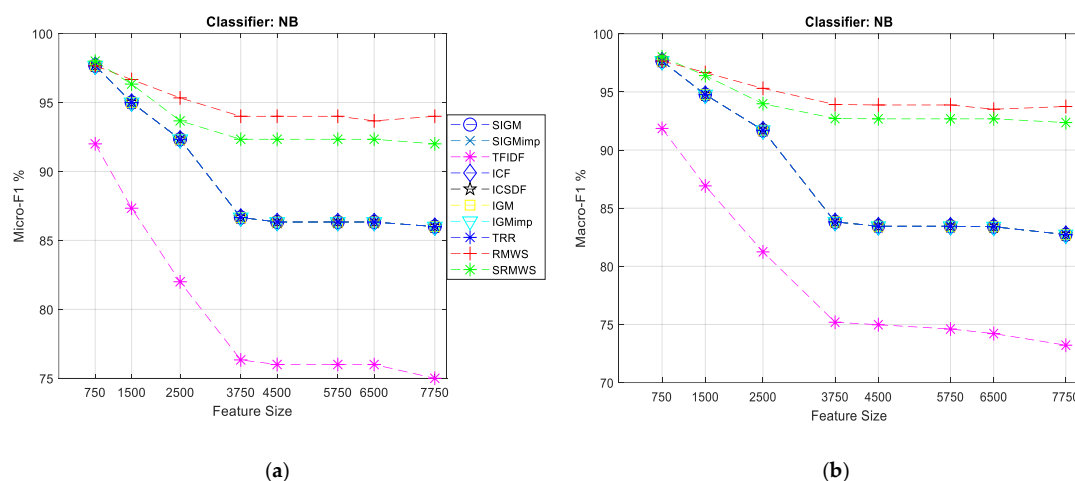


Figure 13. (a) Micro-F1 and (b) macro-F1 metric results for multiclass text classification using the NB classifier and ten different term weighting schemes on the 10 Mini Newsgroups corpus, with different feature sizes.

Figure 5 presents a comparative analysis of the term weighting methods alongside RMWS, SRMWS, and eight additional approaches from the literature, based on micro-F1 and macro-F1 criteria, using the SVM classifier on the Reuters_21578 dataset. As observed in the figure, the SRMWS method outperforms all other methods in terms of both micro-F1 and macro-F1 scores across all term dimensions. Similar to RMWS and SRMWS, SIGM and SIGMimp represent the square root variants of IGM and IGMimp approaches, respectively. Another key observation from Figure 5 is that RMWS emerges as the second-best approach after SIGM and SIGMimp. Notably, RMWS even surpasses these approaches in the 4500-dimensional setting, according to the micro-F1 criterion. RMWS has demonstrated a remarkable superiority over non-variant methods, showcasing its potential to make a significant mark in the field of term weighting. These findings can contribute to the development of text classification algorithms and lay a crucial foundation for future research. This finding demonstrates that the proposed term weighting model offers a more effective weighting strategy compared to existing models in the literature. The weight scores assigned to terms also serve as a measure of their discriminative power.

Figure 6 presents the results obtained using the KNN classifier. It can be observed that the SRMWS method achieves the same result as SIGM and SIGMimp for micro-F1

at the 750th dimension, while outperforming them in all other dimensions. Similarly, for macro-F1, it achieves the best result in all dimensions, except for the 7750th dimension, where it is equal to SIGM and SIGMimp. Among the non-variant methods, RMWS shows the highest performance for macro-F1. For micro-F1, it shows the highest performance in all dimensions except for 3750 and 4500.

Figure 7 presents the performance of term weighting algorithms for the NB classifier. It can be observed that the RMWS and SRMWS methods yield the same results across all term sizes and demonstrate superior performance compared to other approaches. Additionally, it is noted that algorithms derived from the square root of TF values (variants) also produce similar results to their parent algorithms.

Figure 8 shows that the SRMWS method outperforms all other methods in all dimensions for both micro-F1 and macro-F1 scores when classifying text documents with the SVM classifier. This suggests that the term weights assigned by SRMWS are more effective and discriminative. Furthermore, RMWS closely follows SRMWS in terms of performance, demonstrating the potential benefits of incorporating variants in the term weighting process for SVM classification.

Figure 9 shows that the KNN classifier achieves the best micro-F1 and macro-F1 results with SRMWS at 1500 and 2500 dimensions. At 5750 dimensions, SRMWS performs as well as SIGM. These findings suggest that SRMWS is a flexible option for dimension selection and can perform well with the KNN classifier.

Figure 10 presents a noteworthy finding regarding the NB classifier. All approaches, except for the TF-IDF method, yielded identical results in all dimensions for both micro-F1 and macro-F1 criteria. The binary class structure of the dataset plays a pivotal role in obtaining these results. In binary class datasets, the performance of different term weighting methods may exhibit minimal variations. This stems from the fact that only a few key terms are sufficient to discriminate between the two classes. In this case, frequency-based methods such as TF-IDF cannot provide a significant advantage compared to other methods.

Figure 11 presents a comparison of the performance of various weighting approaches when employing the SVM classifier. Excluding dimensions 1500 and 2500, the SIGM, SIGMimp, and SRMWM schemes attain the highest values in both micro-F1 and macro-F1 metrics. However, following these approaches, RMWS yields the best results in all dimensions, except for the 750th dimension. The results obtained with the KNN classifier are presented in Figure 12. As can be observed, SRMWS clearly achieves the highest values based on both criteria in all dimensions. Again, the KNN classifier achieves the best results after SRMWS in all dimensions except for the 750th dimension with RMWS. The best NB results are obtained with RWMS in micro-F1 and macro-F1, achieving the highest score, except for dimension 750. In the 750th dimension, the highest score is obtained with SRMWS. These cases are illustrated in Figure 13. In summary, for this dataset, results regarding term weighting schema are as follows: (a) In SVM and KNN classifiers, the SRMWS method stands out as the most effective approach. (b) The SIGM, SIGMimp, and SRMWM methods also exhibit high performance in the SVM classifier. (c) For the NB classifier, the RWMS method yields the optimal outcomes.

As a result, in this study, two novel methods, RMWS and SRMWS, have been proposed and compared with existing approaches. The comparison was conducted on imbalanced-multiclass, imbalanced-binary class, and balanced-multiclass datasets. The obtained results clearly demonstrate that the RMWS and SRMWS methods outperform existing approaches on imbalanced-multiclass, imbalanced-binary class, and balanced-multiclass datasets. It was observed that RMWS and SRMWS methods consistently exhibit superior performance without being significantly affected by class imbalances and the number of classes in the

dataset. Furthermore, it was determined that the SRMWS method outperforms the RMWS method on balanced-multiclass datasets.

5.6. Statistical Analysis

This section presents a comprehensive statistical analysis to assess whether the RMWS and SRMWS approaches yield meaningful results. The initial test examines the average performance of term weighting approaches across all datasets, specifically focusing on results obtained with the same term dimension. This allows us to assess whether an approach delivers consistent results regardless of the specific dataset. Moreover, certain approaches may perform well on specific datasets but exhibit poor performance on others, indicating a lack of consistency. While an approach is not expected to yield the best results on every dataset, it should not produce excessively poor results either. Consistency is crucial for evaluating an approach's reliability and overall performance. To facilitate this analysis, Tables 15–17 have been devised for each classifier, specifically SVM, KNN, and NB, respectively. Furthermore, the notation 'fs' within the tables denotes the term dimensions and the best results are highlighted in bold within the table.

Table 15. The average results for the specified term dimensions across all datasets for the SVM classifier.

fs	SIGM	SIGMimp	TFIDF	ICF	Micro-F1		IGMimp	TRR	RMWS	SRMWS
					ICSDF	IGM				
750	93.7908	93.8101	91.5096	91.9243	90.8663	92.8350	92.8157	93.1297	93.4557	94.5917
1500	94.5506	94.5700	91.7836	92.2166	90.9812	92.9871	93.0064	93.6570	93.8365	94.9499
2500	94.0776	94.1162	90.7868	91.2081	89.6961	92.4552	92.4358	93.0445	93.6924	94.8632
3750	94.0699	94.0506	89.7364	90.9954	89.4276	92.2888	92.2888	92.6676	93.6663	94.8390
4500	94.3019	94.3794	89.8505	91.1076	89.4567	92.7005	92.6619	92.8745	93.9147	95.0757
5750	94.4256	94.4256	89.8478	90.7998	88.8123	92.7411	92.8446	93.1382	93.9244	95.2688
6500	94.3869	94.4642	89.1231	90.9600	88.5400	92.8909	92.9876	93.2619	93.9051	95.2358
7750	94.4449	94.4836	89.9500	91.2187	88.5515	92.8580	92.8194	93.1458	93.9176	95.2552

fs	SIGM	SIGMimp	TFIDF	ICF	Macro-F1		IGMimp	TRR	RMWS	SRMWS
					ICSDF	IGM				
750	90.0886	90.1137	87.5302	87.7968	86.8023	88.8667	88.8416	89.1954	89.7622	91.1895
1500	91.0005	91.0252	87.8949	88.3457	87.0791	89.0933	89.1147	89.8781	90.0900	91.5508
2500	90.6013	90.6515	87.0463	87.5009	85.8759	88.6428	88.6176	89.2934	89.9944	91.5428
3750	90.5999	90.5782	85.9090	87.2587	85.6457	88.3795	88.3795	88.7963	89.8597	91.5128
4500	90.8480	90.9518	86.1045	87.3985	85.6961	88.8846	88.8379	89.1207	90.2889	91.8173
5750	91.0122	91.0155	86.1495	87.1757	85.1462	89.0259	89.1276	89.4637	90.3204	92.0130
6500	90.9406	91.0377	85.4617	87.3310	84.9126	89.1947	89.3162	89.6248	90.2561	91.9960
7750	90.9382	90.9952	86.2719	87.6178	84.7928	89.1252	89.0820	89.4657	90.2624	92.0085

A close examination of Tables 15 and 16 reveals that SRMWS clearly produces the most successful results. This finding indicates that SVM and KNN classifiers exhibit more effective performances when coupled with SRMWS, rendering this method preferable for these classifiers. Moreover, SRMWS consistently outperforms other methods across all datasets and term dimensions, establishing a statistically significant difference.

An examination of Table 17 reveals that RMWS is the most successful method when used with the NB classifier. This finding indicates that the NB classifier exhibits more effective performance when coupled with RMWS, rendering this method preferable for this classifier. While other methods achieve similar results to RMWS in some term dimensions, RMWS generally emerges as the best-performing method. This demonstrates that RMWS consistently outperforms other methods when used with the NB classifier, establishing a statistically significant difference.

Table 16. The average results for the specified term dimensions across all datasets for the KNN classifier.

fs	SIGM	SIGMimp	TFIDF	ICF	Micro-F1		IGMimp	TRR	RMWS	SRMWS
					ICSDF	IGM				
750	93.6869	93.7836	84.9883	88.6114	85.4251	88.9011	88.9455	88.9011	91.9725	94.3198
1500	91.2734	91.4009	82.3093	85.9287	81.8118	86.6102	86.6614	86.6102	90.9985	94.1338
2500	90.9439	90.8598	79.3895	83.2394	79.3356	84.9127	84.9833	84.9127	90.0489	93.8789
3750	89.5275	89.3592	77.6996	82.3006	77.2309	83.5042	83.5168	83.5042	89.2144	93.7424
4500	89.2260	89.1168	77.1821	82.3300	76.9397	82.9321	82.9253	82.9571	89.4114	93.6751
5750	89.2590	89.1487	76.8376	81.9999	76.6581	83.6627	83.5660	83.6627	89.7352	93.7535
6500	89.2783	89.1487	76.8244	82.3983	76.3915	83.7681	83.6646	83.7681	89.8201	93.7149
7750	89.2734	89.2211	76.5723	82.2275	76.3186	83.8501	83.7978	83.8501	89.3177	93.6558
fs	SIGM	SIGMimp	TFIDF	ICF	Macro-F1		IGMimp	TRR	RMWS	SRMWS
					ICSDF	IGM				
750	89.7579	89.8651	80.3617	84.0608	80.6923	84.5843	84.6377	84.5843	87.9419	90.4358
1500	87.0213	87.1750	77.4361	81.1339	76.7565	82.2521	82.3192	82.2521	86.8133	90.1866
2500	86.6899	86.5600	74.6089	78.5310	74.3454	80.4362	80.5078	80.4362	85.7977	89.8306
3750	85.3622	85.1074	73.0482	77.6690	72.2779	78.8663	78.8675	78.8663	84.8655	89.5935
4500	85.0886	84.9194	72.5591	77.7429	72.1370	78.1780	78.1390	78.1859	85.0414	89.5590
5750	85.0351	84.8785	72.4834	77.5101	72.1184	79.2184	79.0589	79.2184	85.6055	89.6769
6500	85.0207	84.8416	72.4263	77.9355	71.8399	79.3338	79.1812	79.3338	85.6941	89.5639
7750	85.0225	84.9316	72.3641	77.8472	71.7800	79.5606	79.4660	79.5606	85.1701	89.4328

Table 17. The average results for the specified term dimensions across all datasets for the NB classifier.

fs	SIGM	SIGMimp	TFIDF	ICF	Micro-F1		IGMimp	TRR	RMWS	SRMWS
					ICSDF	IGM				
750	91.6829	91.6829	90.1807	91.6829	91.6829	91.6829	91.6829	91.6829	92.2718	92.3829
1500	90.8910	90.8910	87.7354	90.8910	90.8910	90.8910	90.8910	90.8910	92.2737	92.1625
2500	89.2538	89.2538	85.3325	89.2538	89.2538	89.2538	89.2538	89.2538	91.3315	90.7760
3750	86.8186	86.8186	82.8052	86.8186	86.8186	86.8186	86.8186	86.8186	90.2530	89.6974
4500	86.2864	86.2864	82.5233	86.2864	86.2864	86.2864	86.2864	86.2864	89.8570	89.3014
5750	85.9767	85.9767	82.0986	85.9767	85.9767	85.9767	85.9767	85.9767	89.6099	89.0544
6500	85.7866	85.7866	81.9064	85.7866	85.7866	85.7866	85.7866	85.7866	89.4215	88.9770
7750	85.2612	85.2612	80.9289	85.2612	85.2612	85.2612	85.2612	85.2612	88.9680	88.3013
fs	SIGM	SIGMimp	TFIDF	ICF	Macro-F1		IGMimp	TRR	RMWS	SRMWS
					ICSDF	IGM				
750	88.0697	88.0697	86.5737	88.0697	88.0697	88.0697	88.0697	88.0697	88.9820	89.0977
1500	87.1653	87.1653	83.7215	87.1653	87.1653	87.1653	87.1653	87.1653	88.9478	88.8595
2500	85.1040	85.1040	81.1420	85.1040	85.1040	85.1040	85.1040	85.1040	87.9102	87.4688
3750	81.7852	81.7852	77.9908	81.7852	81.7852	81.7852	81.7852	81.7852	86.3751	85.9738
4500	80.9556	80.9556	77.6994	80.9556	80.9556	80.9556	80.9556	80.9556	85.8826	85.4848
5750	80.3406	80.3406	76.8429	80.3406	80.3406	80.3406	80.3406	80.3406	85.3140	84.9162
6500	79.9976	79.9976	76.3631	79.9976	79.9976	79.9976	79.9976	79.9976	84.9889	84.7129
7750	78.9263	78.9263	74.8852	78.9263	78.9263	78.9263	78.9263	78.9263	83.8351	83.3747

A *t*-test was also used as a statistical analysis to demonstrate the validity of the proposed best-performing RMWS and SRMWS. For this purpose, Table 18 is constructed for RMWS and Table 19 for SRMWS. The tables show the results of one-sided, paired *t*-tests for the obtained *p*-values. If the *p*-value is below 0.05, the obtained results are deemed statistically significant. In particular, a *p*-value below 0.05 signifies statistical significance at a confidence level of 95%. Furthermore, if the *p*-value is less than 0.01, the results achieve statistical significance at an even higher confidence level of 99%.

Table 18. Statistical significance of performance improvements achieved from the proposed best-performing RMWS for all datasets.

		SIGM/ RMWS	SIGMimp/ RMWS	TFIDF/ RMWS	ICF/ RMWS	ICSDF/ RMWS	IGM/ RMWS	IGMimp/ RMWS	TRR/ RMWS
SVM	Micro-F1	0.0056 *	0.0040 *	0.0001 *	0.0034 *	0.0050 *	0.0022 *	0.0030 *	0.06904 **
	Macro-F1	0.0055 *	0.0032 *	0.0008 *	0.0006 *	0.0023 *	0.0004 *	0.0008 *	0.0090 *
KNN	Micro-F1	0.1959	0.2669	0.0099 *	0.0026 *	0.0015 *	0.0141 **	0.0180 **	0.0138 **
	Macro-F1	0.1997	0.3046	0.0044 *	0.0011 *	0.0071 *	0.0124 **	0.0175 **	0.0123 **
NB	Micro-F1	0.0019 *	0.0019 *	0.0026 *	0.0019 *	0.0019 *	0.0019 *	0.0019 *	0.0019 *
	Macro-F1	0.0018 *	0.0018 *	0.0023 *	0.0018 *	0.0018 *	0.0018 *	0.0018 *	0.0018 *

* Significance at 99%. ** Significance at 95%.

Table 19. Statistical significance of performance improvements achieved from the proposed best-performing SRMWS for all datasets.

		SIGM/ SRMWS	SIGMimp/ SRMWS	TFIDF/ SRMWS	ICF/ SRMWS	ICSDF/ SRMWS	IGM/ SRMWS	IGMimp/ SRMWS	TRR/ SRMWS
SVM	Micro-F1	0.0000 *	0.0001 *	0.0001 *	0.0012 *	0.0006 *	0.0003 *	0.0009 *	0.0001 *
	Macro-F1	0.0000 *	0.0005 *	0.0003 *	0.0085 *	0.0006 *	0.0002 *	0.0000 *	0.0000 *
KNN	Micro-F1	0.0076 *	0.0001 *	0.0008 *	0.0014 *	0.0012 *	0.0091 *	0.0010 *	0.0090 *
	Macro-F1	0.0058 *	0.0082 *	0.0035 *	0.0067 *	0.0054 *	0.0070 *	0.0085 *	0.0073 *
NB	Micro-F1	0.0015 *	0.0015 *	0.0017 *	0.0015 *	0.0015 *	0.0015 *	0.0015 *	0.0015 *
	Macro-F1	0.0016 *	0.0016 *	0.0017 *	0.0016 *	0.0016 *	0.0016 *	0.0016 *	0.0016 *

* Significance at 99%.

The results demonstrate that the performance gains achieved with the proposed RMWS weighting scheme compared to other schemes are statistically significant, with a very high confidence level of 99% for NB. Additionally, this confidence level is also at 99% for all values except one p -value for the SVM classifier. As seen in Table 18, for the KNN classifier, almost all p -values are at 95% and 99% confidence levels, except for a few cases. When Table 19 is examined, it is observed that SRMWS provides results with a very high confidence level of 99% for all classifiers. The obtained p -values indicate that the RMWS and SRMWS schemes perform significantly better than other schemes. This implies that the possibility of random coincidence is low, and the findings are reliable.

As a result, these findings clearly validate the superiority of the proposed RMWS and SRMWS weighting schemes compared to other schemes. Both schemes provide a statistically significant performance increase when used with NB, SVM, and KNN classifiers.

6. Discussion

The findings from our study indicate that the proposed methods, RMWS and SRMWS, significantly enhance the performance of text classification tasks. By exploring the class-term relationship using rough sets, these methods provide a novel approach to term weighting that outperforms both traditional and contemporary methods. The RMWS approach utilizes rough sets to identify terms that offer specific and distinctive information relevant to different classes. This capability allows for the identification of patterns within documents that may otherwise be overlooked by other term weighting schemes. By incorporating the coefficients α , β , and γ , RMWS can adjust the influence of these terms more accurately, leading to superior classification results. The SRMWS further refines this process by taking the square root of RMWS values, making the term weights more balanced and discriminative.

Experimental results revealed that RMWS and SRMWS consistently outperform existing term weighting schemes, regardless of dataset structure. This includes imbalanced-multiclass, balanced-multiclass, and imbalanced-binary class datasets, demonstrating the

robustness of the proposed methods. Notably, SRMWS showed the highest classification performance across most scenarios, suggesting that the square root transformation adds significant value in enhancing term distinctiveness.

One critical finding is the statistical significance of the performance improvements offered by RMWS and SRMWS. The p -values from our t -tests indicate that the differences in performance are not due to random chance. This statistical validation underscores the efficacy of the proposed methods and their potential utility in practical applications.

When comparing classifiers, the NB classifier showed the greatest improvement with RMWS, while the SVM and KNN classifiers benefited more from SRMWS. This suggests that the choice between RMWS and SRMWS may depend on the specific classifier used and the nature of the dataset. Future studies could explore this relationship further, examining how different classifiers interact with these term weighting schemes under various conditions.

This study's results also highlight the importance of proper coefficient selection for optimizing the performance of RMWS and SRMWS. The α , β , and γ coefficients have been shown to significantly impact classification outcomes, necessitating careful calibration based on the dataset and classifier used. This aspect presents an opportunity for future research to develop automated techniques for coefficient tuning, potentially using machine learning algorithms.

Overall, the incorporation of rough set theory into term weighting presents a promising direction for improving text classification. By focusing on revealing hidden patterns and specific class-term relationships, RMWS and SRMWS offer a more nuanced and effective approach to term weighting. The findings from this study contribute to the field by presenting robust, statistically validated methods that outperform existing approaches, paving the way for future advancements in automated text classification.

7. Conclusions

This study introduces a novel supervised term weighting scheme for text classification: the Rough Multivariate Weighting Scheme (RMWS), along with its derivative, the Root (Sqrt) Rough Multivariate Weighting Scheme (SRMWS). The proposed schema leverages the information extraction capabilities of rough sets to analyze and extract discriminative features from the document-term-class space. This approach surpasses traditional methods by incorporating a broader and deeper understanding of the relationships between terms, documents, and classes, enabling a more precise determination of term importance for classification tasks. Comprehensive experiments were performed on datasets with varying characteristics, including imbalanced and balanced structures, as well as multiclass and binary class scenarios. The results revealed the effectiveness of RMWS and its derivative SRMWS in enhancing classification accuracy and F1 scores, compared to traditional and other supervised term weighting methods. Notably, SRMWS consistently achieved the best performance across all dataset types, showcasing its robustness in modeling complex relationships within the data. Additionally, RMWS outperformed its foundational approaches, such as IGM and IGMimp, highlighting its superior capability in extracting and utilizing discriminative term information. In summary, this study contributes a single main term weighting model, RMWS, and demonstrates the potential benefits of its derivative, SRMWS, as a complementary variation. Both approaches underline the importance of leveraging rough set theory for effective feature weighting and classification in text mining. Future research could explore the integration of RMWS with other advanced rough set techniques or artificial intelligence methods, aiming to further enhance its performance and applicability to diverse text classification problems.

Funding: This research received no external funding.

Data Availability Statement: The Reuters-21578 and 10 Mini Newsgroups datasets, which support this study’s findings, are freely available on Machine Learning Repository-UCI (Reference [28]). The Enron1 dataset is available at <https://www.kaggle.com/datasets/wcukierski/enron-email-dataset> (accessed on 7 May 2015).

Conflicts of Interest: The author declares that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Lan, M.; Sung, S.-Y.; Low, H.-B.; Tan, C.-L. A Comparative Study on Term Weighting Schemes for Text Categorization. In Proceedings of the 2005 IEEE International Joint Conference on Neural Networks, Montreal, QC, Canada, 31 July–4 August 2005; IEEE: Piscataway, NJ, USA, 2005; Volume 1, pp. 546–551.
2. Cekik, R.; Uysal, A.K. A New Metric for Feature Selection on Short Text Datasets. *Concurr. Comput.* **2022**, *34*, e6909. [\[CrossRef\]](#)
3. Parlak, B.; Uysal, A.K. A Novel Filter Feature Selection Method for Text Classification: Extensive Feature Selector. *J. Inf. Sci.* **2023**, *49*, 59–78. [\[CrossRef\]](#)
4. Cekik, R.; Uysal, A.K. A Novel Filter Feature Selection Method Using Rough Set for Short Text Data. *Expert Syst. Appl.* **2020**, *160*, 113691. [\[CrossRef\]](#)
5. Manning, C.D. *Introduction to Information Retrieval*; Syngress Publishing: Rockland, MA, USA, 2008.
6. Debole, F.; Sebastiani, F. Supervised Term Weighting for Automated Text Categorization. In Proceedings of the 2003 ACM Symposium on Applied Computing, Melbourne, FL, USA, 9–12 March 2003; pp. 784–788.
7. Emmanuel, M.; Khatri, S.M.; Babu, D.R.R. A Novel Scheme for Term Weighting in Text Categorization: Positive Impact Factor. In Proceedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics, Manchester, UK, 13–16 October 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 2292–2297.
8. Ren, F.; Sohrab, M.G. Class-Indexing-Based Term Weighting for Automatic Text Classification. *Inf. Sci.* **2013**, *236*, 109–125. [\[CrossRef\]](#)
9. Campos, R.; Mangaravite, V.; Pasquali, A.; Jorge, A.; Nunes, C.; Jatowt, A. YAKE! Keyword Extraction from Single Documents Using Multiple Local Features. *Inf. Sci.* **2020**, *509*, 257–289. [\[CrossRef\]](#)
10. Doğan, T. Metin Sınıflandırma İçin Terim Ağırlıklandırma. Ph.D. Thesis, Eskişehir Technical University, Eskişehir, Turkey, August 2019.
11. Liu, Y.; Loh, H.T.; Sun, A. Imbalanced Text Classification: A Term Weighting Approach. *Expert Syst. Appl.* **2009**, *36*, 690–701. [\[CrossRef\]](#)
12. Spärck Jones, K. A Statistical Interpretation of Term Specificity and Its Application in Retrieval. *J. Doc.* **2004**, *60*, 493–502. [\[CrossRef\]](#)
13. Lan, M.; Tan, C.L.; Su, J.; Lu, Y. Supervised and Traditional Term Weighting Methods for Automatic Text Categorization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *31*, 721–735. [\[CrossRef\]](#) [\[PubMed\]](#)
14. Ko, Y. A New Term-weighting Scheme for Text Classification Using the Odds of Positive and Negative Class Probabilities. *J. Assoc. Inf. Sci. Technol.* **2015**, *66*, 2553–2565. [\[CrossRef\]](#)
15. Chen, K.; Zhang, Z.; Long, J.; Zhang, H. Turning from TF-IDF to TF-IGM for Term Weighting in Text Classification. *Expert Syst. Appl.* **2016**, *66*, 245–260. [\[CrossRef\]](#)
16. Dogan, T.; Uysal, A.K. Improved Inverse Gravity Moment Term Weighting for Text Classification. *Expert Syst. Appl.* **2019**, *130*, 45–59. [\[CrossRef\]](#)
17. Okkalioglu, M. TF-IGM Revisited: Imbalance Text Classification with Relative Imbalance Ratio. *Expert Syst. Appl.* **2023**, *217*, 119578. [\[CrossRef\]](#)
18. Rath, R.N.; Mustafi, A. The Importance of Term Weighting in Semantic Understanding of Text: A Review of Techniques. *Multimed. Tools Appl.* **2023**, *82*, 9761–9783. [\[CrossRef\]](#) [\[PubMed\]](#)
19. Dai, Z.; Callan, J. Context-Aware Term Weighting for First Stage Passage Retrieval. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual, 25–30 July 2020; pp. 1533–1536.
20. Bojanowski, P.; Grave, E.; Joulin, A.; Mikolov, T. Enriching Word Vectors with Subword Information. *Trans. Assoc. Comput. Linguist.* **2017**, *5*, 135–146. [\[CrossRef\]](#)
21. Peters, M.E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. Deep Contextualized Word Representations. *arXiv* **2018**, arXiv:1802.05365.
22. Zhang, D.; Xu, H.; Su, Z.; Xu, Y. Chinese Comments Sentiment Classification Based on Word2vec and SVMperf. *Expert Syst. Appl.* **2015**, *42*, 1857–1863. [\[CrossRef\]](#)

23. Yang, K.; Cai, Y.; Chen, Z.; Leung, H.; Lau, R. Exploring Topic Discriminating Power of Words in Latent Dirichlet Allocation. In Proceedings of the COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, Osaka, Japan, 11–16 December 2016; pp. 2238–2247.
24. Pawlak, Z. Rough Set Theory and Its Applications to Data Analysis. *Cybern. Syst.* **1998**, *29*, 661–688. [[CrossRef](#)]
25. Cekik, R.; Telceken, S. A New Classification Method Based on Rough Sets Theory. *Soft Comput.* **2018**, *22*, 1881–1889. [[CrossRef](#)]
26. Zhao, H.; Wang, P.; Hu, Q. Cost-Sensitive Feature Selection Based on Adaptive Neighborhood Granularity with Multi-Level Confidence. *Inf. Sci.* **2016**, *366*, 134–149. [[CrossRef](#)]
27. Zheng, L.; Diao, R.; Shen, Q. Self-Adjusting Harmony Search-Based Feature Selection. *Soft Comput.* **2015**, *19*, 1567–1579. [[CrossRef](#)]
28. Asuncion, A.; Newman, D. UCI Machine Learning Repository 2007. Available online: <https://archive.ics.uci.edu/dataset/137/reuters+21578+text+categorization+collection> (accessed on 25 September 2024).
29. Metsis, V.; Androutsopoulos, I.; Paliouras, G. Spam Filtering with Naive Bayes-Which Naive Bayes? In Proceedings of the CEAS, Mountain View, CA, USA, 27–28 July 2006; Volume 17, pp. 28–69.
30. ÇEKİK, R.; Mahmut, K. A New Feature Selection Metric Based on Rough Sets and Information Gain in Text Classification. *Gazi Univ. J. Sci. Part A Eng. Innov.* **2023**, *10*, 472–486. [[CrossRef](#)]
31. Uysal, A.K.; Gunal, S. A Novel Probabilistic Feature Selection Method for Text Classification. *Knowl. Based Syst.* **2012**, *36*, 226–235. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.