electronics

MDPI

Article

TF-TDA: A Novel Supervised Term Weighting Scheme for Sentiment Analysis

Arwa Alshehri * and Abdulmohsen Algarni

Department of Computer Science, College of Computer Science, King Khalid University, Abha 62529, Saudi Arabia; a.algarni@kku.edu.sa
* Correspondence: armalshehri@kku.edu.sa

Abstract: In text classification tasks, such as sentiment analysis (SA), feature representation and weighting schemes play a crucial role in classification performance. Traditional term weighting schemes depend on the term frequency within the entire document collection; therefore, they are called unsupervised term weighting (UTW) schemes. One of the most popular UTW schemes is term frequency–inverse document frequency (TF-IDF); however, this is not sufficient for SA tasks. Newer weighting schemes have been developed to take advantage of the membership of documents in their categories. These are called supervised term weighting (STW) schemes; however, most of them weigh the extracted features without considering the characteristics of some noisy features and data imbalances. Therefore, in this study, a novel STW approach was proposed, known as term frequency–term discrimination ability (TF-TDA). TF-TDA mainly presents the extracted features with different degrees of discrimination by categorizing them into several groups. Subsequently, each group is weighted based on its contribution. The proposed method was examined over four SA datasets using naive Bayes (NB) and support vector machine (SVM) models. The experimental results proved the superiority of TF-TDA over two baseline term weighting approaches, with improvements ranging from 0.52% to 3.99% in the F1 score. The statistical test results verified the significant improvement obtained by TF-TDA in most cases, where the P-value ranged from 0.0000597 to 0.0455.

Keywords: machine learning; text classification; sentiment analysis; feature extraction; supervised term weighting

Citation: Alshehri, A.; Algarni, A. TF-TDA: A Novel Supervised Term Weighting Scheme for Sentiment Analysis. *Electronics* **2023**, *12*, 1632. https://doi.org/10.3390/electronics12071632

Academic Editors: Edoardo Ragusa, Erik Cambria and Juan M. Corchado

Received: 15 February 2023
Revised: 28 March 2023
Accepted: 28 March 2023
Published: 30 March 2023

1. Introduction

Social networking sites have grown exponentially in recent years, encouraging people to express their opinions on different subjects online. Twitter has become one of the most important platforms that enables people to communicate. On Twitter, people post tweets that express their feelings. These tweets can be an excellent source for companies and institutions to monitor their services and improve themselves. However, the rapid spread of Twitter has created considerable unstructured data, making it difficult to limit and exploit these opinions. To address this issue, sentiment analysis (SA) has improved. SA addresses the problem of analyzing text concerning the opinions being expressed [1,2]. SA is a basic field of natural language processing (NLP) [3] and can be defined as "a process that automates the mining of attitudes, opinions, views, and emotions from text, speech, and database sources through NLP" [4]. Moreover, SA combines other academic disciplines, including data mining (DM) and text mining (TM). SA is of significant importance for businesses and organizations as it includes online commerce data for analysis [5].

Twitter sentiment analysis (TSA) is a classification process involving the classification of tweets into positive, negative, and neutral categories. To date, most sentiment analysis studies have classified sentiments in text into three categories [6]. TSA uses various approaches: machine learning (ML), lexicon-based, hybrid-based, and graph-based approaches [7]. For the ML approach, an automatic text classifier is created by learning the

characteristics of classes from a training set of previously labeled data. Consequently, the learning step is supervised by knowledge of the classes and training documents in these classes [8]. Before text classification, it is necessary to represent the text obtained in numeric form so the ML classifier can read it, which is known as the feature engineering phase. In text classification generally and SA specifically, significant attention is given to preprocessing operations and feature engineering methods to identify and extract representative features (terms) for natural language documents [9]. The extracted features are represented as feature vectors, where each term is associated with its own weight. The term weight must illustrate its importance among other terms. Therefore, the weighting scheme of the features used to represent the document affects the classification performance.

Feature extraction methods produce a massive number of features. However, a considerable number of these features are meaningless; thus, they do not contribute to identifying the document category due to their weak discriminatory power. Although many weighting schemes have been developed, most of them increase the importance of meaningless rather than deserving features. These features are mostly shared between categories (common features). Moreover, subsets of common features often occur equally or almost equally in different categories, which makes them neutral rather than useful (specific) features. In addition, another type of meaningless feature includes features found very rarely within the dataset, as they carry little meaning or impact. Therefore, weighting the entire feature collection similarly, without considering the characteristics of those features, can lead to several features losing their ability to discriminate.

Moreover, the data imbalance is mostly ignored by many weighting schemes, which can affect the classification process.

Accordingly, this study aims to improve a novel weighting scheme that is expected to distinguish the features more accurately and ensure that the features are weighted differently based on their discriminating ability.

The key contributions of this study are as follows:

- The proposal of a new term weighting scheme for effective text classification, especially for SA purposes. The proposed method relies on three aspects:
  - Adopting a method to distinguish between specific and neutral common terms by classifying them into several groups.
  - Revising the term weight differently for each term group so it presents its ability to discriminate.
  - Ensuring that the term weight reflects its actual presence in the dataset by handling data imbalance issues.
- The performance of the proposed scheme is validated by conducting a comparative study with two other weighting schemes. Moreover, the performance of the three term weighting schemes (including the proposed scheme) have been explored using three different local factors.
- The experiments are performed on four sentiment analysis datasets diversified in terms of language, size, and subject using two ML classification models.

The remainder of this paper is divided into different sections. The literature review is presented in Section 2 and the proposed scheme is discussed in Section 3. Section 4 mentions the data and experimentation; Section 5 contains the results and discussion. Finally, Section 6 contains the conclusion and directions for future work.

2. Related Work

Text mining (TM) is the process of discovering and extracting novel knowledge from unstructured data (textual data) [10]. Typical text mining tasks include text classification, text clustering, concept/entity extraction, and document summarization [11]. Many different text classification applications have been identified in different text mining fields, such as SA.

During the text classification process, the obtained dataset might contain noise and valueless data, which can harm classification accuracy. According to a previous study, the dataset contained approximately 40% noise [12]. On Twitter, users make many mistakes while writing, such as spelling and typographical errors, or they might use abbreviations or slang. Therefore, preprocessing methods can be applied to clean and normalize the dataset by stemming and removing noisy and valueless data, such as stop words, punctuation, and abbreviations.

The preprocessed data must be presented in a readable form for the ML model. Therefore, feature extraction methods are applied to convert the data into a set of features. Twitter data produce thousands of features, resulting in higher dimensionality. The high-dimensionality representation of data can decrease classifier performance and increase computational cost [13]. Therefore, feature selection techniques can be used to choose the features that provide meaningful information, which improves classification accuracy [14].

To achieve sentiment classification using supervised learning, classification algorithms are trained to predict the sentiment class of a specific document. Many possible classification algorithms can be used, such as naive Bayes (NB), support vector machine (SVM), k-nearest neighbor (*k*NN), and random forest (RF).

Finally, the performance of sentiment classification can be assessed using evaluation metrics.

2.1. Document Representation and Term Weighting

In the ML approach, feature extraction is a crucial step in which the right features can ease the difficulty of modeling and increase the quality of the results [15]. Each document should be transformed into a machine-readable format for classification.

In sentiment analysis, the usual approach to representing text documents in features is the use of the vector space model (VSM), where each document is converted to a numerical feature vector comprising the weights of terms taken from the text corpus in the following form:

$$d = \{w_1, w_2, \ldots, w_n\} \quad (1)$$

where d is a document consisting of n terms and w_i is the weight value of i-th term.

The most important aspect of document representation in VSM is that the term's weight can reflect its importance. Consequently, the categorization accuracy is directly influenced by the term weighting scheme chosen to represent documents. A term's weight generally consists of two factors: local and global.

2.2. Local Factor

In the local factor, the aim is to measure the contribution of each term within each document, regardless of others [16]. The most popular local scheme is known as term frequency (tf), which finds how many times the term occurs in a document [17].

Although studies mostly have focused on improving the collection frequency (global) factors, the performance of the weighting scheme is affected by the modification of local factors [1]. Other common local weighting methods are presented in Table 1.

2.3. Global Factor

In contrast to the local factor, the global factor considers the whole collection of documents to determine the term importance [16]. Table 2 shows the notations that are mostly used to formulate various global factors.

The weighting schemes of the global factor are grouped into two categories, which are unsupervised term weighting (UTW) and supervised term weighting (STW). More details about these factors are given in the following sections.

Table 1. Local weighting factors.

Notation	Formulation	Description
tf	tf	Raw term frequency, finds occurrences of term in the document.
ltf	$log(1+tf)$	Logarithmic term frequency.
$sqrt(tf)$	$\sqrt{tf}$	Square root of the term frequency.
tp	1 , if $tf > 0$ 0, otherwise	Term presence.
atf	$k + (1-k)\frac{tf}{max_i(tf)}$	Augmented term frequency [18].
btf	$\frac{(k_1+1)tf}{k_1\left((1-b)+b\frac{dl}{avg_{dl}}\right)+tf}$	BM25 tf [19].
ntf	$\frac{tf(t_i,d_j)}{max(tf(d_j))}$	Normalized tf [20].
ITF	$1-\frac{r}{r+tf}$	Inverse term frequency [21].

Table 2. Notations to formulate global factors.

Notation	Description
a	Positive document frequency; number of documents in positive class where the term t_i occurs at least once.
b	Number of training documents in the positive category where the term t_i does not occur.
c	Negative document frequency; number of documents in negative class where the term t_i occurs at least once.
d	Number of training documents in the negative category where the term t_i does not occur.
N	Total number of training documents in all classes; $N = a+b+c+d$.
N^+	Total number of training documents in positive class; $N^+ = a+b$.
N^-	Total number of training documents in negative class; $N^- = c+d$.
$\|C\|$	Number of classes in the collection.
cf	Class frequency—the number of classes that include the term t_i.

2.3.1. UTW Schemes

In UTW, the class information is ignored during the weighting process. In other words, the weighting schemes do not distinguish which class a term belongs to. Thus, the generated weight is influenced only by the term document frequency.

The most popular UTW scheme is TF-IDF (term frequency–inverse document frequency), proposed by Jones [22]. It measures the importance of a term in a document according to the number of occurrences in the entire corpus. Mathematically, $TF-IDF$ is denoted as follows:

$$TF-IDF = TF(t_i,d_j) \times log\left(\frac{N}{df(t_i)}\right) \quad (2)$$

where $TF(t_i,d_j)$ is number of times term t_i appears in document d_j (raw term frequency), $N = |D_T|$ where D_T represents training set documents, and $df(t_i)$ is the number of documents that contain term t_i in the corpus. Based on Table 2, the df factor can be formulated as $df = (a+c)$.

TF-IDF has been widely adopted as a feature extraction scheme in many sentiment analysis studies, such as [23–27].

Various studies have proposed different variants of the *idf* factor. For example, the new scheme proposed by [28] replaced *idf* factor with probabilistic idf (pidf).

$$pidf = log\left(\frac{N-(a+c)}{(a+c)}\right) \tag{3}$$

Another modification presents the division of the term frequency by the sum of all the frequencies of that term in all the collection documents. This scheme is known as weighted idf (widf) [29].

$$widf = \left(\frac{1}{\sum_{d_x \in D_T} tf(t_i, d_x)}\right) \tag{4}$$

All previous schemes share two common characteristics. First, the term class distribution is ignored, since these schemes are UTW methods. Second, they provide higher weights for terms that rarely appear since they believe that a term that occurs more frequently in the corpora can discriminate less between documents.

2.3.2. STW Schemes

Knowledge about the category of training documents plays a crucial role in the training process. Therefore, text classification is considered a supervised task. This information is used by several term weighting techniques to control the term weighting process [16]. Many STW schemes have been proposed based on different insights, with the aim of solving specific classification problems. Different factors have been modified or proposed, as follows:

- IDF Variants:

Different studies exist in the literature that propose a supervised version of the TF-IDF scheme by modifying the *idf* factor. One of these variants is delta idf (didf), which separately computes the *idf* factor for positive and negative documents. Next, the variance between them is taken to boost the importance of words that are unevenly distributed between the positive and the negative classes to improve the classification accuracy. *didf* is formulated as follows:

$$didf = log\left(\frac{N^+}{a}\right) - log\left(\frac{N^-}{c}\right) = log\left(\frac{N^+c}{N^-a}\right) \tag{5}$$

The experimental results showed that *tf-didf* outperformed TF-IDF using the SVM model [30].

Moreover, different variants of the *didf* factor have been proposed by [31] to avoid errors caused by the case of a = 0 or c = 0, and more sophisticated methods originating from information retrieval (IR) have been used. These variants include delta smoothed idf (dsidf), delta smoothed prob idf (dspidf), and delta BM25 idf (dbidf). According to empirical evaluations, these smoothed delta variations outperformed the most accurate term weighting techniques for sentiment analysis. The formulation of these schemes is as follows:

$$dsidf = log\left(\frac{N^-a+0.5}{N^+c+0.5}\right) \tag{6}$$

$$dspidf = log\left(\frac{(N^- - c)(a+0.5)}{(N^+ - a)(c+0.5)}\right) \tag{7}$$

$$dbidf = log\left(\frac{(N^- - c+0.5)(a+0.5)}{(N^+ - a+0.5)(c+0.5)}\right) \tag{8}$$

- Feature Selection Metrics:

Another approach is to adopt feature selection methods to weight terms by replacing the *idf* factor with those metrics. These metrics include the information gain (ig), chi-square (chi), mutual information (mi), and gain ratio (gr). They present global factors as follows:

$$ig = \frac{a}{N} log \frac{aN}{(a+b)(a+c)} + \frac{b}{N} log \frac{bN}{(a+b)(b+d)} + \frac{c}{N} log \frac{cN}{(a+c)(c+d)} + \frac{d}{N} log \frac{dN}{(b+d)(c+a)} \quad (9)$$

$$gr = log \frac{ig}{-\frac{a+b}{N} log \frac{a+b}{N} - \frac{c+d}{N} log \frac{c+d}{N}} \quad (10)$$

$$chi = log \frac{N(ad-bc)^2}{(a+c)(b+d)(a+b)(c+d)} \quad (11)$$

$$mi = log \left(max \left(\frac{aN}{(a+c)N^+}, \frac{cN}{(a+c)N^-} \right) \right) \quad (12)$$

For example, the authors in [32] used the metrics *ig*, *chi*, and *gr*. The results obtained using SVM confirmed that *gr* exceeded the other schemes in most cases. However, the TF-IDF method performed better than these supervised approaches. In addition, several feature selection methods, including *ig*, *chi*, and *mi* were used by [33]. Experimental results using SVM showed that, compared with *bidf*, the *ig* scheme performed very poorly, whereas *mi* and *chi* produced better accuracy on two of the three datasets.

Moreover, a new STW scheme was developed using another feature selection method known as the distinguishing feature selector (DFS).

$$DFS = \sum_{j=1}^{|C|} \left(\frac{\frac{a_{ij}}{a_{ij}+c_{ij}}}{\frac{b_{ij}}{a_{ij}+b_{ij}} + \frac{c_{ij}}{c_{ij}+d_{ij}} + 1} \right) \quad (13)$$

The proposed scheme was evaluated using SVM and the Roccio classifier on two datasets. In most cases, the proposed scheme provides better results than other schemes [1].

- Relevance Frequency:

The idea behind relevance frequency (rf) is to deal only with documents containing the considered term. It is denoted as follows:

$$rf = log_2 \left(2 + \frac{a}{max(1,c)} \right) \quad (14)$$

Using this approach, the researchers in [34] proposed a STW scheme named *tf-rf*. The *rf* factor prefers terms that appear in a positive category rather than a negative category. In the case of multi-class classification, the classifier of a specific category c_k was built by considering c_k, a positive category, and combining all other categories into one negative category. Thus, the terms in positive samples are considered good discriminators. This scheme was investigated using SVM and *k*NN on three data corpora. The *tf-rf* performance was evaluated by comparing it with UTW schemes TF, TF-IDF, and *tp*, and STW schemes, such as *tf-ig* and *tf-chi*. The *tf-rf* provided the best performance in all experiments.

The authors in [35] provided a supervised variant of the IDF factor known as inverse document frequency of terms in classes (IDFC) and combined the new factor with *rf* to introduce a new STW, known as TF-IDFC-RF. TF-IDFC-RF mainly aims to consider intra- and inter-class distribution during the weighting process. SVM and NB are the classifiers used on four two-class datasets. Compared to TF-IDF, *tf-rf*, and the other seven weight-

ing schemes, TF-IDFC-RF outperforms all schemes with NB and SVM on two datasets. Mathematically, this is denoted by the following:

$$IDFC - RF = log_2\left(\frac{2 + max(a,c)}{max(2, min(a,c))} \times \sqrt{b+d}\right) \tag{15}$$

Similarly, a new global factor, named inverse document frequency excluding category-based (idfec-b), was proposed by [16]. The idea behind this factor is that words that appear in many documents belonging to the same category are not penalized, as in *idf*. This is inspired by *rf*, where both penalize the term weights based on the number of negative samples containing that term. However, the numerator in *idfec-b* also includes the number of negative samples in the corpora. The performance of the weighting scheme was verified using SVM and RF algorithms on four datasets. According to the experimental results, the proposed scheme achieved good effectiveness, especially with few features, while *tf-rf* performs better when the number of features increases.

$$idfec - b = log\left(2 + \frac{a+c+d}{max(1,c)}\right) \tag{16}$$

In text classification, the distribution of textual information is usually unbalanced. Hence, the authors in [20] aim to improve imbalanced text classification by producing a novel STW approach based on probability (prop-based). The main objective is to compute the term weight by combining a, b, and c factors to produce two relevance indicators, $\frac{a}{c}$ and $\frac{a}{b}$. These ratios are expected to provide the most valuable information reflecting the term's power in associating a category. The experiments were conducted on two imbalanced datasets using SVM and NB. The evaluation results confirmed the superiority of *prop-based* over TF-IDF and feature selection weighting schemes such as *ig* and *chi*.

$$prob - based = log\left(1 + \frac{a}{c} \times \frac{a}{b}\right) \tag{17}$$

- Class Frequency:

Based on the term class frequency, many researchers have widely used the inverse class frequency (ICF) factor to improve novel STW schemes. This is similar to the IDF factor, except that it concerns class frequency instead of document frequency. The authors in [36] have adopted ICF to improve two STW schemes. These are *tf.icf* and *icf-based*. For *tf.icf*, the entire weight of term t_i is simply the product of raw *tf* and *icf*. Conversely, *icf-based* depends on the combination of raw *tf* and *icf* as well as the *rf* factor. The two proposed methods were investigated using four classifiers: SVM with linear kernel, SVM with RBF kernel, *k*NN, and centroid-based. The two proposed methods achieved a better or similar performance, based on extensive experiments and comparisons, than the seven STW schemes and three UTW schemes. These factors are formulated as follows:

$$icf = log\left(\frac{|C|}{cf(t_i)}\right) \tag{18}$$

$$icf - based = log_2\left(2 + \frac{a}{max(1,c)} \times \frac{|C|}{cf(t_i)}\right) \tag{19}$$

According to this line of reasoning, the *idf* factor was composed using *ICF* to propose a new STW scheme named TF-IDF-ICF. The *ICF* factor mostly prefers terms that appear in a few categories in class space $C = c_1, c_2, \ldots c_k$ and considers them good discriminators. However, using the *ICF* factor, a higher weight is assigned to terms that rarely appear without a prior knowledge of the class space. Thus, this was improved by considering class space density (CS_δ) to prevent bias against frequent terms. Then, TF-IDF-ICS_δF was produced. Both schemes were examined using centroid-based, SVM, and NB classifiers on

three datasets. The results proved the superiority of TF-IDF-ICS$_\delta$F over other weighting schemes [37].

$$IDF - ICF = log\left(1 + \frac{N}{a+c}\right) \times log\left(1 + \frac{|C|}{cf(t_i)}\right) \quad (20)$$

$$IDF - ICS_\delta F = log\left(1 + \frac{N}{a+c}\right) \times log\left(1 + \frac{|C|}{CS_\delta(t_i)}\right) \quad (21)$$

Table 3 summarizes the studies related to SA and the improvement in term weighting schemes, where feature representation, datasets used, and the classification model are presented.

Table 3. Related works summary.

Traditional Feature Engineering			
Ref.	**Feature Engineering**	**Dataset**	**Classification Model**
[23]	BoW and TF-IDF	Online-education-during-COVID-19	ETC, AdaBoost, GNB, *k*NN, SGD, RF, DT, SVM
[24]	BoW, TF-IDF, and hashing	Reviews regarding the calling apps	SVM, *k*NN, DT, LR, RF, LSTM, CNN, and GRU
[25]	TF, TF-IDF and word2vec	Twitter-airline-sentiment	Calibrated, SVM, AdaBoost, DT, Gaussian NB, ET, RF, LR, SGD, and GBM
[26]	TF-IDF, BoW, and hashing	Meeting app's reviews	SVM, DT, LR, *k*NN, and RF
Improved Weighting Schemes			
Ref.	**Global Factor**	**Dataset**	**Classification Model**
[30]	didf	Movie review	SVM
[31]	dsidf, dspidf, dbidf	Movie review, multi-domain sentiment dataset of Amazon products reviews, and BLOGS06	SVM
[32]	ig, gr, chi	Reuters-21578	Roccio, *k*NN, and SVM
[33]	ig, mi, chi	Cornell movie review, multi-domain sentiment dataset of Amazon products reviews, and Stanford large movie review	SVM
[1]	DFS	Reuters-21578 and 20 newsgroups	SVM and Roccio
[34]	rf	Reuters-21578, 20 newsgroups, and Ohsumed	*k*NN and SVM
[35]	IDFC-RF	Movie review, subjectivity, Amazon sarcasm, and polarity	NB, SVM
[16]	idfec-b	Reuters-21578, 20 newsgroups, movie review data, and multi-domain sentiment dataset of Amazon product reviews	RF, SVM
[20]	prob-based	MCV1 and Reuters-21578	NB, SVM
[36]	icf, icf-based	Reuters-21578, the balanced 20 newsgroups, and la12	*k*NN, SVM, and centroid
[37]	IDF-ICF, IDF-ICS$_\delta$F	Reuters-21578, 20 newsgroups, and RCV1-v2	Centroid-based, SVM, and NB

The UTW schemes such as the TF-IDF and the different unsupervised variants of idf factor share two limitations. Firstly, since they are unsupervised methods, they would ignore the distribution of terms among categories, meaning that the term weight is assigned based on the term frequencies within the entire dataset regardless of whether the term

is concentrated in one category or another; however, weighting the term based on its distribution among different categories could enhance the classification process. Secondly, these schemes provide a higher weight for terms with low frequencies in the corpus and consider terms with high frequencies to be weak discriminators and then introduce them with less weight. However, it is known that terms with very low occurrences are considered meaningless and noisy features [17,38]. Therefore, this study aims to improve a STW scheme to avoid the issues caused by the unsupervised approach, as well as ensuring that the terms gain the weight they deserve.

Most of the proposed variants of STW schemes are formulated based on the documented frequency of the term, meaning that the term is weighted based on the number of documents containing that term in each category. However, this is not sufficient when using an imbalanced dataset because the document frequency does not accurately show the term's presence in specific categories. Furthermore, the two STW schemes, namely tf-rf- and icf-based schemes, are not useful in binary classification due to the asymmetry of the rf factor. This means that the rf factor prefers and increases the weight of terms presented in the positive documents compared to the negative documents. For the icf factor, besides its ability to further reduce the weight of terms that exist in many categories compared to terms that exist in a few categories, there are neutral (meaningless) terms that are similarly distributed within each category. These terms cannot be extracted using only the icf factor. Considering the previous issues, the STW method proposed in this study aims to handle the imbalance issue, find neutral terms, group the rest of collection terms into different groups, and then revise the weight of each term group differently based on its distinguishing ability.

3. The Proposed STW Scheme

In sentiment analysis, text representation is considered challenging; therefore, the terms extracted and used to represent the document affect the classification performance. Accordingly, sufficient consideration has been given to handling terms in this study. The most popular term weighting scheme is TF-IDF, which assesses the importance of a term t_i in a document based on the entire corpus, as indicated in Equation (2). However, this is not completely useful for text classification and not always efficient for sentiment analysis problems [1,39]. Furthermore, TF-IDF is an unsupervised technique. Then, the term distribution among categories is ignored, treating all categories as one collection.

In this study, we believe that the importance of a term t_i may vary depending on the category of the tweet to which it belongs. Generally, the extracted term t_i can be classified into either the pure positive term group (terms that appeared only in the positive class T^+) or the pure negative terms group (terms that appeared only in negative class T^-). Moreover, both groups may share common terms (Com).

In text classification, a term that appears in fewer categories has a stronger distinguishing power than a term occurring in all or even most categories [39]. Consequently, in the case of binary classification, common terms will obtain the lowest ability to distinguish from pure groups (T^+, T^-). However, they form a considerable number of collection terms, which may mislead the classifier during the classification process. To enhance the classification process, this study aims to improve a novel STW scheme called term frequency–term discrimination ability (TF-TDA), which deals with the above-mentioned issues by grouping terms into more than the T^+, T^-, and Com groups. Then, each group is weighted based on its discrimination degree. Hence, the proposed method is performed in two stages: the term classification phase and the term weight revising phase.

3.1. Term Classification Approach

The collection terms are classified into three fundamental groups: T^+, T^-, and Com, as mentioned before. In addition to these groups, we believe that common terms (Com) can be better classified into three groups based on how closely they relate to specific classes such as the following:

- Frequently positive ($Freq^+$): common terms related to the positive class.

- Frequently negative ($Freq^-$): common terms related to the negative class.
- General (G): common terms related to both classes (neutral).

The different theories in the literature emphasize that terms with many occurrences in one class compared to the other can be considered a good discriminator for the first class [39–41]. In existing term weighting schemes, the weight is mostly assigned based on the document frequency of the term at the collection level. However, during the binary classification of an imbalanced dataset, document frequency cannot accurately show how the term is concentrated in a specific category. Therefore, we propose measuring a term's degree of belonging t_i to a specific category c_k using the percentage of the term's presence in that category to avoid the imbalance issue in the dataset.

Accordingly, each term t_i is associated with two factors: $P_E(t_i)$, which denotes the existence of t_i in the positive class, and $N_E(t_i)$, to denote the existence of t_i in the negative class. Based on the notations in Table 2, these factors are formulated as follows:

$$P_E(t_i) = \frac{a(t_i)}{N^+} \times 100 \tag{22}$$

$$N_E(t_i) = \frac{c(t_i)}{N^-} \times 100 \tag{23}$$

where a is the frequency of term t_i in positive documents and c is the frequency of term t_i in negative documents. Notably, N_E equals zero in the case of weighting T^+ terms and P_E is zero while weighting T^- terms.

After finding each term in both classes, the classification of common terms is performed using the variance between the P_E and N_E for each term, as follows:

$$Var(t_i) = P_E(t_i) - N_E(t_i) \tag{24}$$

Consequently, the common terms can then be classified into $Freq^+$ or $Freq^-$, or G by satisfying the constraints presented in Table 4.

Table 4. Common term classification constraints.

Term Group	Constraint
$Freq^+$	$Var \geq k$
$Freq^-$	$Var \leq -k$
G	$\lvert Var \rvert \leq k$

Where k works as a hyper-parameter, in which the classification of common terms can be controlled by changing the value of k. Thus, this represents the area that contains G terms.

As a result, the entire collection of terms can be classified into five groups (T^+, T^-, G, $Freq^+$, $Freq^-$), as illustrated in Figure 1.

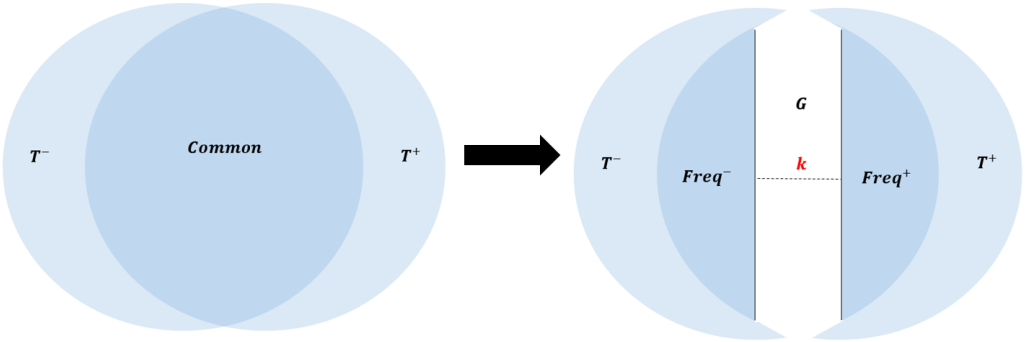


Figure 1. Common term classification.

3.2. Weight Revision Approach

The weight revision process will be applied to the global factor, where the local factor is the raw term frequency (TF). According to the terms' classification phase, it is obvious that the general group (G) has no discriminating ability; therefore, no global factor will be applied to the (G) group.

The weight revising manner is inspired by icf-based [36], which measures the distribution of term t_i using both relevance frequency (rf) [34] and inverse class frequency (icf), as presented in Equation (19).

In this study, the term classification stage helps to distinguish categories of terms and provides a basis for the term weight revising process. Accordingly, we propose making the term weight affected by its group at two levels: the distinguishing ability and the relevant class.

3.2.1. Distinguishing Ability

Based on the distinguishing power, there are two main groups of terms: pure terms (T^+,T^-) and common terms ($Freq^+$, $Freq^-$). As mentioned earlier, pure terms have more discriminatory power than common terms because they appear in fewer categories. Consequently, terms within high-distinguishing-ability groups will gain more weight than the other groups. This rule can already be satisfied using the class frequency factor, since it assigns 2 for terms within pure groups and 1 for terms within common groups. However, when using an imbalanced dataset, taking the class size of term t_i into account can provide a more accurate distinction. Thus, the distinguishing ability of the term t_i among categories is measured using a newly proposed factor known as class priority ($clsPrior$), formulated as follows:

$$clsPrior(c_k, t_i) = \frac{N^{c_k}}{N} \times \frac{|C|}{cf(t_i)} \tag{25}$$

3.2.2. Relevant Class

Based on the relevant class, the collection terms are mainly classified into positive groups (T^+, $Freq^+$) and negative groups (T^-, $Freq^-$). The term classification in the previous phase was performed based on term existence factors: P_E and N_E, as shown in Table 4. Accordingly, we propose to use these factors during the weight revising process in the form of high-discriminator (H_{dis}) and low-discriminator (L_{dis}) factors. H_{dis} represents term t_i's existence in the relevant class and L_{dis} represents the term existence in the irrelevant class. Both relevant and irrelevant classes are specified based on the term group. For example, if a term belongs to $Freq^+$, then the positive class is the relevant class, and the negative class is the irrelevant class.

The relationship between H_{dis} and L_{dis} is formulated based on the rf concept, as follows:

$$log_2\left(2 + \frac{H_{dis}}{max(1, L_{dis})}\right) \tag{26}$$

The use of this formulation could support the proposed scheme in two aspects:

- A term's discriminatory power increases with the increase in the difference between H_{dis} and L_{dis} of that term [41]. Thus, the use of the rf formulation will maintain this property. For example, consider the two common terms (t_1,t_2) and $k = 3$ with the data presented in Table 5.

As demonstrated in Table 5, although both terms have the same H_{dis}, the term t_1 obtained a higher weight than t_2 due to the large difference between relevant and irrelevant categories. In contrast, term t_2 is distributed in both categories with a relatively small difference.

- The denominator $max(1, L_{dis})$ will avoid division by zero while weighting pure term groups. Moreover, since TF-TDA deals with the percentage of the word's presence, it is possible to obtain a very low value (less than 1) for the L_{dis} factor. Then, the direct division by such small values will give the word more weight than it deserves. For example, assume that term t_i has $H_{dis} = 0.8$ and $L_{dis} = 0.1$. Then $\frac{0.8}{0.1} = 8$.

Using the $max(1, L_{dis})$ factor in such cases will divide by one and then preserve the actual weight of the word. Thus, $\frac{0.8}{max(1,0.1)} = 0.8$.

Table 5. Example 1: clarification of the advantage of using the rf factor.

Term	H_{dis}	L_{dis}	Var	Weight
t_1	30	10	20	$log_2(5) = 2.3$
t_2	30	21	9	$log_2(3.43) = 1.8$

Finally, the improved method (TF-TDA) is formulated as follows:

$$TF - TDA = TF(t_i, d_j) \times log_2\left(2 + \frac{H_{dis}}{max(1, L_{dis})} \times clsPrior\right) \quad (27)$$

In order to take advantage of the term's grouping phase, scheme factors such as H_{dis}, L_{dis}, and N^{c_k} are specified differently for each term group. Table 6 presents the weighting scheme for each group.

Table 6. Weight revising approach for each group.

Term Group	Weighting Scheme
T^+	$\mathrm{TF}(t_i,d_j) \times log_2\left(2 + \frac{P_E(t_i)}{max(1,N_E(t_i))} \times \left(\frac{N^+}{N} \times 2\right)\right)$
T^-	$\mathrm{TF}(t_i,d_j) \times log_2\left(2 + \frac{N_E(t_i)}{max(1,P_E(t_i))} \times \left(\frac{N^-}{N} \times 2\right)\right)$
G	$\mathrm{TF}(t_i,d_j)$
$Freq^+$	$\mathrm{TF}(t_i,d_j) \times log_2\left(2 + \frac{P_E(t_i)}{max(1,N_E(t_i))} \times \left(\frac{N^+}{N} \times 1\right)\right)$
$Freq^-$	$\mathrm{TF}(t_i,d_j) \times log_2\left(2 + \frac{N_E(t_i)}{max(1,P_E(t_i))} \times \left(\frac{N^-}{N} \times 1\right)\right)$

Figure 2 show the architecture of the proposed model. This clearly shows the steps that were followed to implement the TF-TDA model. In addition, the proposed model is publicly available at https://github.com/Arwa008/TF-TDA-code (accessed on 27 March 2023).

The time complexity of TF-TDA is mainly decided by the steps of classifying the *common* terms into three groups ($Freq^+, G, Freq^-$). Then, the revision process of the weight of four out of the five groups ($T^+, Freq^+, G, Freq^-, T^-$) is undertaken. Therefore, the time complexity of TF-TDA can be calculated as $O(mlog^m)$, where $m = |T|$.

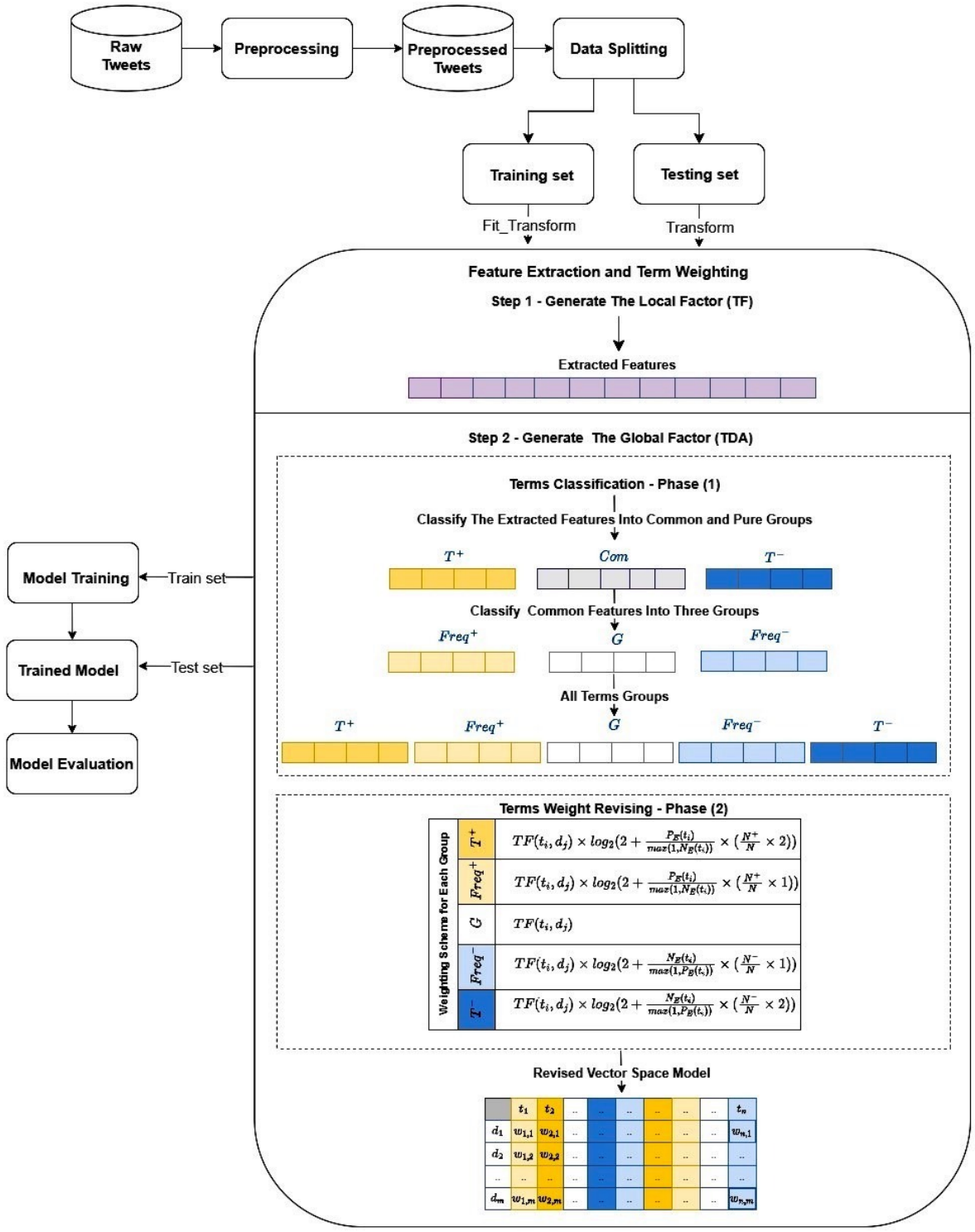


Figure 2. The architecture of the TF-TDA model.

4. Experimental Setup

An extensive experimental evaluation was performed to confirm the effectiveness of the proposed term weighting approach, among other schemes. In the following, the organization of these experiments is described in detail. Figure 2 shows the overall architecture of the proposed method.

4.1. Datasets

To verify the consistency of the proposed scheme, three Arabic datasets and one English dataset were presented. All the datasets were presented in different domains with different sizes. These are described below:

1. Multi-domain Arabic resources for sentiment analysis (MARSA): The largest annotated Gulf dataset was provided by the Arabic sentiment analysis research group at Imam Muhammad Ibn Saud Islamic University [42]. This includes several domains; however, the social and sport domains were used as two independent datasets in this study. For the sport domain, the tweets were collected using hashtags generated about football matches. The social dataset concentrated on issues affecting the Saudi society; therefore, the hashtags were created about social issues such as royal orders, Saudi budget, issues affecting the income of Saudi citizens, and others.
2. SenWave: The third dataset presents Arabic tweets concerns the COVID-19 domain and was published by [43] for SA purposes.
3. Twitter airline sentiment: Dataset made available by Kaggle, consisting of tweets in English that represent reviews of six United States (US) airline companies.

Data Pre-Processing

A critical and time-consuming procedure is data pre-processing. Some Python tools were used to prepare the text documents for mining tasks. In all datasets, only positive and negative tweets were used; the rest have been removed. Data pre-processing includes the following tasks:

- Remove stop words: Tweets typically contain valueless words known as stop words, including pronouns, prepositions, and other words [44]. Accordingly, two lists of stop words were defined based on the Arabic and English stop word corpus contained in the NLTK package. Then, all tweets were filtered from the list of words.
- Normalization: The data must be normalized by converting all word forms into a common form. For the English dataset, all words are converted to lower case. For the Arabic datasets, the nature of the Arabic language may require additional steps, such as the following:
  - Remove Arabic diacritical marks (taskeel), for example: اَلْعَرَبِيَة (Arabic) is converted to العربية;
  - Remove 'tatweel' character 'ـ', for example: جميـــــل (Beautiful) is converted to جميل;
  - Replace elongation characters with a single character, for example: [أأإآ] is replaced with ا and [ؤئ] with ء; also replace the final letter ة with ه and ى with ي. Moreover, some special characters are normalized, for example: گ is normalized to ك.
- Emoji handling: Currently, emojis are widely used by people on social media to express their opinions, and can provide significant information about a text, especially in SA. Therefore, the emojis were handled by replacing them with their text format. Moreover, corresponding texts consisting of several words were combined into a single word. For example, the text format "broken heart" was converted to "brokenheart". As words such as "heart" can appear in emotions with opposite meanings, such as "red heart" and "broken heart", weighting the word "heart" as an individual feature will make it a meaningless (neutral) term. Moreover, combining corresponding text into a single word can reduce feature dimensionality.
- Stemming: Stemming is the process of reducing a word to its stem, base, or root [44]. In this study, two stemmers included in NLTK were used: ISRIStemmer for the Arabic datasets and PorterStemmer for the English dataset.

Some general pre-processing steps were also applied, such as removing numbers, repeated characters, punctuation, and new line marks. In addition, the researchers performed procedures such as removing words with fewer than three characters, tweets consisting of two words or less, URLs, hashtags, and duplicated tweets. Figure 3 provides an overview of the datasets and numbers of positive and negative samples that were used. After performing the pre-processing steps, all datasets were split into 80% for training and 20% for testing. Table 7 shows the distribution of samples in both training and test sets for each dataset.

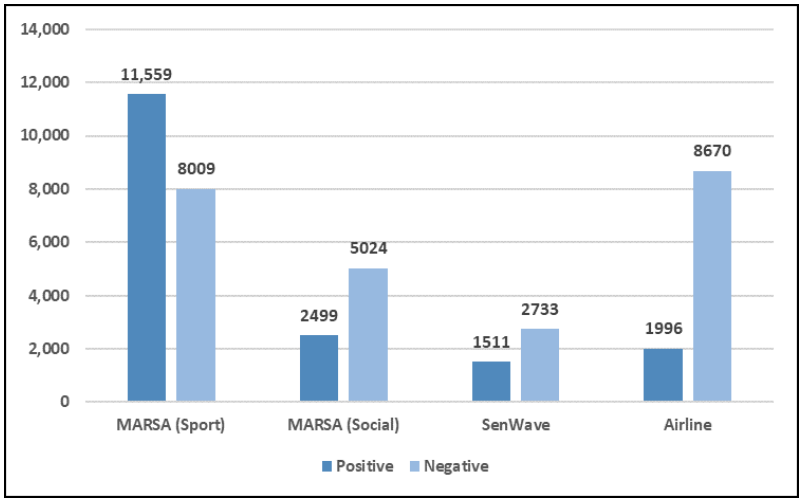


Figure 3. The number of tweets in each class for all the datasets used.

Table 7. Distribution of samples among training and test sets.

Dataset	Class Label	# Training Samples	# Testing Samples	# Total Samples
MARSA (Sport)	Positive Negative	9276 6378	2283 1631	19,568
MARSA (Social)	Positive Negative	1992 4026	507 998	7523
SenWave	Positive Negative	1206 2189	305 544	4244
Airline	Positive Negative	1580 7090	416 1752	1083

4.2. Feature Representation and Term Weighting

A vector space model (VSM) was used for term representation, using words as features. The weighting process was performed in two phases, generating local and global factors. As mentioned in Section 3, the adopted local factor is (TF). Second, the global factor was created from scratch to implement the proposed scheme.

This study emphasizes a novel STW scheme to enhance the classification task. Therefore, no feature selection techniques, such as *ig*, *mi*, and *chi*, were adopted as feature selection criteria. Instead, the term classification phase in TF-TDA divided all the extracted features into groups, which were weighted differently according to their importance, meaning that all extracted features were used. Table 8 presents the number of extracted features for each dataset; additionally, the number of terms in pure and common groups were specified.

Table 8. Number of unique features (including common and pure features) for each dataset.

Dataset	# *Com*	# T^+	# T^-	# All Features
MARSA (Sport)	3228	2375	2230	7833
MARSA (Social)	2249	3211	862	6322
SenWave	1771	817	2507	5095
Airline	1775	767	3968	6510

4.3. Classification Models

This study employed supervised learning for document classification. Based on our literature review, many supervised learning algorithms were used for text classification. SVM and NB were found to be the most successful solutions [45,46], and have been used in many similar studies, as shown in Table 3. Furthermore, several Arabic and English sentiment analysis studies have employed both classifiers, proving their consistency with both languages, for example [23,24,47–51].

For NB, the multinomial NB (M-NB) was adopted due to its sufficient text classification performance. This was used when the data were represented based on the feature frequency.

For SVM, among different types of kernels, the linear SVM was used in this study because the data were linearly separable; this was also preferred when there were many features. The rest of the parameters were specified using default values such as gamma: scale, C: 1.0.

4.4. Criteria for Performance Evaluation

In this study, the experimental results are presented to ensure the efficiency of the proposed scheme. Popular performance metrics for text classification tasks include precision and recall. However, when considered individually, neither precision nor recall make sense. Therefore, the F1-score metric was usually used since it combines both precision and recall [34]. These metrics are defined as follows:

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \tag{28}$$

where,

$$Precision = \frac{TP}{TP + FP} \tag{29}$$

and

$$Recall = \frac{TP}{TP + FN} \tag{30}$$

When a true positive (TP) occurs, the resulting tweet is positive and is predicted to be positive. In a false negative (FN), the resulting tweet is positive but is predicted to be negative. In a false positive (FP), the resulting tweet is negative, but is predicted to be positive. In a true negative (TN), the resulting tweet is negative and is predicted to be negative.

The weighted average of the F1 score was used to assign the weight w_{c_k} to each class c_k. The weight is usually given based on the number of instances in a class. Then, the metrics for each class were calculated and the weighted average of these metrics were computed.

5. Experimental Results

This section presents the results of experiments conducted on different datasets using different classifiers. The performance of the proposed method (TF-TDA) was investigated by comparison with both standard TF-IDF and the scheme from which our scheme was inspired (icf-based).

Table 9 depicts the results of each classifier with the four datasets used. Moreover, the improvements obtained by TF-TDA compared to other schemes are presented, along with the improvement average.

Table 9. F1 score for all datasets.

Model	Dataset	TF-IDF	icf-Based	TF-TDA	TF-IDF*	icf-Based *	Average
M-NB	MARSA (Sport)	87.70%	86.93%	89.00%	1.30%	2.07%	1.69%
	MARSA (Social)	82.31%	83.25%	85.52%	3.21%	2.27%	2.74%
	Senwave	78.62%	78.19%	80.96%	2.34%	2.77%	2.56%
	Airline	88.10%	90.20%	90.72%	2.62%	0.52%	1.57%
SVM	MARSA (Sport)	85.62%	88.04%	88.64%	3.02%	0.60%	1.81%
	MARSA (Social)	79.23%	81.55%	83.22%	3.99%	1.67%	2.83%
	Senwave	75.29%	77.04%	76.94%	1.65%	−0.10%	0.77%
	Airline	85.93%	88.76%	89.76%	3.83%	1.00%	2.42%

* The improvement percentage obtained by TF-TDA compared to other schemes.

As revealed in Table 9, notably, TF-TDA mostly outperformed TF-IDF and icf-based across the different datasets and classification algorithms. The proposed scheme achieved the highest performance in terms of the F1 score in 23 out of 24 experiments. The enhancement appears more clearly in the MARSA (social) dataset, since this achieved the highest enhancement average for both M-NB and SVM.

When considering the classification algorithms, one can note that the superiority of TF-TDA over other schemes is more evident with the M-NB classifier. The F1-score difference between the two algorithms was more than 4% for the SenWave dataset, 2% for the MARSA (Social) dataset, 0.96% for Airline dataset, and up to 0.36% for the MARSA (Sport) dataset.

STW methods are naturally expected to be superior to UTW methods because they consider document distribution [34]. However, the results revealed that the UTW scheme (TF-IDF) is almost better than the STW scheme (icf-based) with M-NB on the SenWave and MARSA (Sport) datasets, with improvements of up to 0.43% and 0.77%, respectively. This observation was also stated in [34].

TF-IDF did not achieve the best results considering all four datasets with the SVM classifier.

However, the TF-TDA presented results better than TF-IDF in all experiments among both classifiers. Moreover, the results provide compelling evidence that TF-TDA achieves better results than the icf-based scheme on all four datasets with M-NB. When considering the SVM classifier, the TF-TDA outperformed the icf-based scheme on three datasets, MARSA (Social), MARSA (Sport), and Airline, and performed slightly worse than icf-based schemes with a 0.10% difference to the SenWave dataset.

The observed superiority of TF-TDA over icf-based schemes can be attributed to the revision of the term weight based on the percentage of term existence instead of the term frequency, which clearly addresses the imbalance issue. Moreover, classifying terms into groups and making a term's weight affected by its group increased terms' distinguishing power. Weighting each term group differently can avoid asymmetry in the rf factor.

Several factors could explain the superiority of TF-TDA over the TF-IDF scheme. Firstly, the TF-IDF scheme provides a higher weight for terms with low occurrences in the corpus and considers terms with high occurrences as weak discriminators; these are then

introduced with lower weights, as mentioned in Section 2.3.1. However, TF-TDA weighs the term according to the percentage of its existence; therefore, a term with few occurrences was assumed to gain less weight. Secondly, terms with high frequencies in the dataset are often the common terms (Com) due to their presence in both categories. In TF-TDA, they are divided and weighted based on their association with these categories. TF-IDF reduces their weights without any knowledge of their relationship to the categories, which may affect the results.

To evaluate the difference of performances between the proposed weighting scheme and the other schemes, the McNemar's statistical test is employed [52], where the significance level is 0.05. As presented in Table 10, the significance test results proved the significant improvement achieved by TF-TDA over other schemes in most cases. The p-value obtained ranged from 0.0000597 to 0.0455. Moreover, some p-values showed insignificant performance differences between TF-TDA and TF-IDF such as in the case of using M-NB with MARSA (Sport) dataset, since the p-value is 0.846; in this case, the null hypothesis cannot be rejected.

Table 10. Statistical test results.

Dataset	Model	TF-TDA vs. TF-IDF	TF-TDA vs. icf-Based
		***p*-Value**	***p*-Value**
MARSA (Sport)	M-NB	0.846	0.0000597
	SVM	0.0006	0.0455
MARSA (Social)	M-NB	0.00052	0.0000296
	SVM	0.000233	0.000039
SenWave	M-NB	0.021	0.000021
	SVM	0.38	0.92
Airline	M-NB	0.076	0.0034
	SVM	0.000028	0.00005

5.1. Discussion

This section discusses the results in more detail by showing the impact of using different local factors on the weighting schemes, discussing the followed approach to find the optimal k, and measuring the effectiveness of the term grouping concept.

5.1.1. The Effect of Using Different Local Factors

As mentioned in Section 2.2, the performance of the weighting scheme could be affected by changing the local factor. Therefore, the TF-TDA was investigated using two other local factors: logarithmic term frequency (ltf) and term presence (tp). Moreover, TF-IDF and icf-based schemes were also evaluated using both local factors. Figures 4–6 present the results of the three weighting schemes using both local factors. As presented in the results, the F1 score varied with the change in the local factor in most of the experiments. Furthermore, the variation in results showed different degrees for each dataset, classification model, and weighting scheme.

5.1.2. Find the Optimal k Value

In this study, the value of k was assumed to vary based on the dataset used due to the variation in the sample number and the terms' distribution among classes. It may also be affected by the classification model, with each model treating weighted terms in a specific mechanism.

Accordingly, different values of k were evaluated. The values in the range [0.1, 0.2, . . . , 4.9, 5] are considered and all these values were experimented on for all datasets, using both M-NB and SVM separately. Figures 7–10 present the F1 score of all considered k values for all datasets.

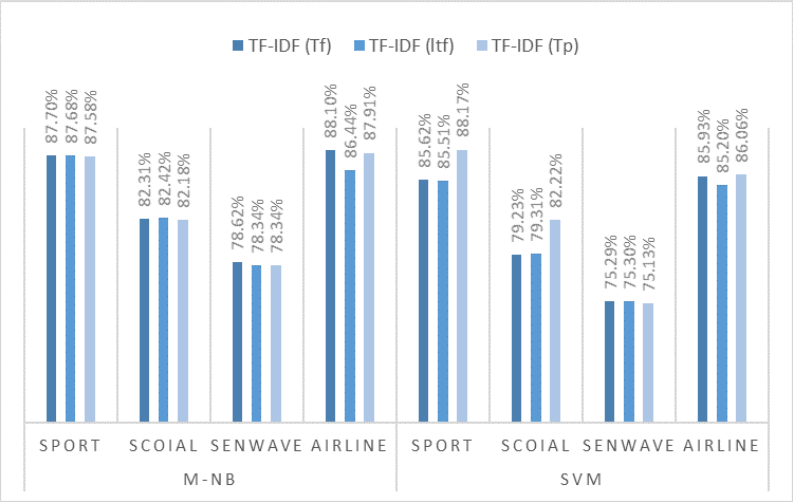


Figure 4. TF-IDF using different local factors.

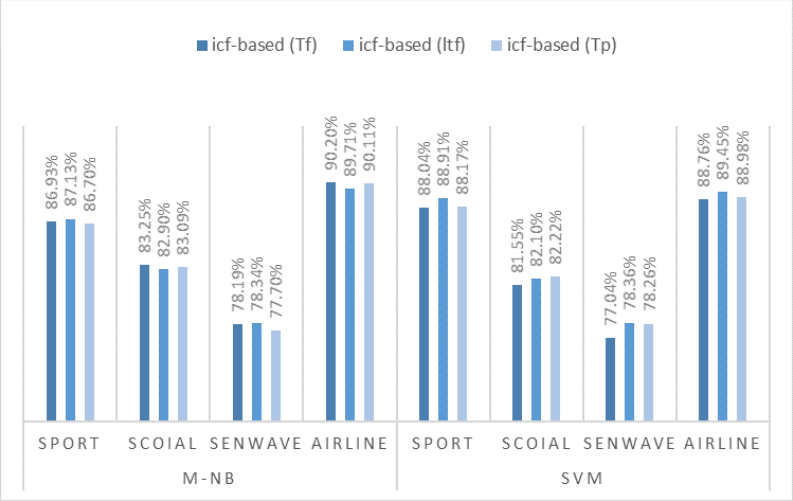


Figure 5. icf-based using different local factors.

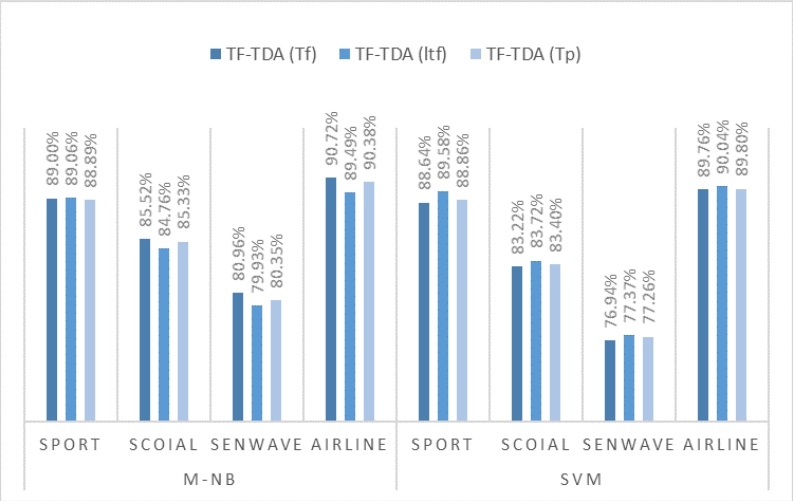
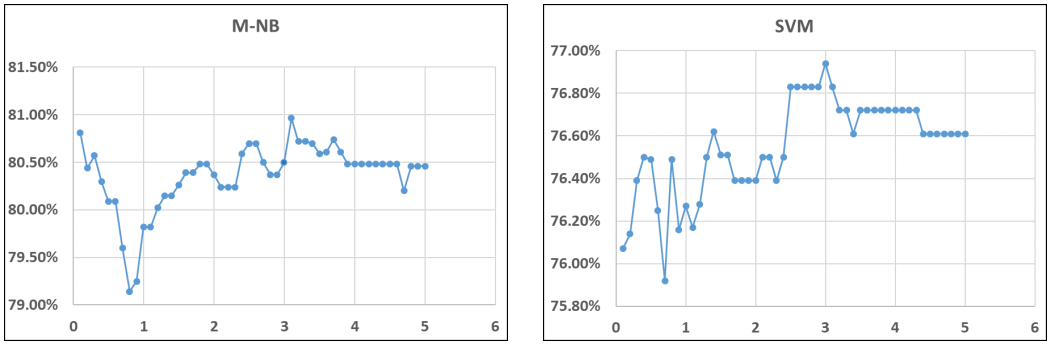
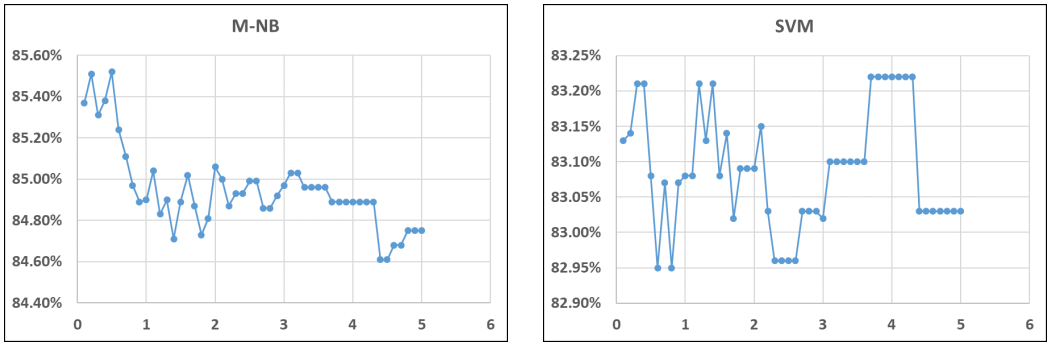


Figure 6. TF-TDA using different local factors.



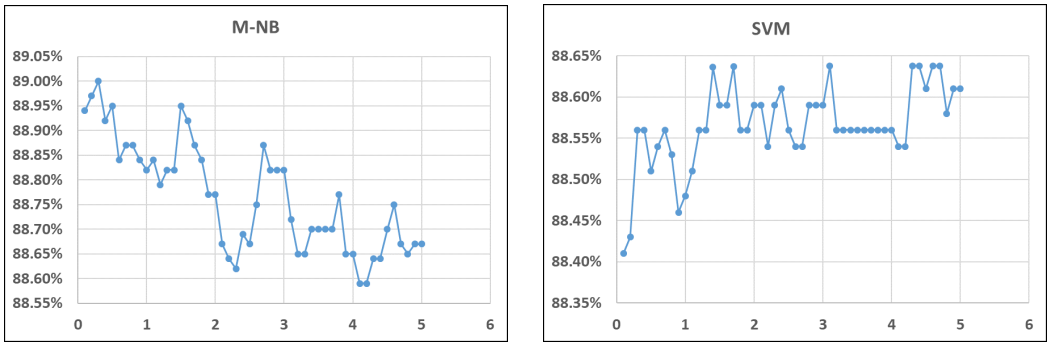
(**a**) F1 score for different *k* values using M-NB (**b**) F1 score for different *k* values using SVM

Figure 7. Results of all considered *k* values for SenWave dataset.



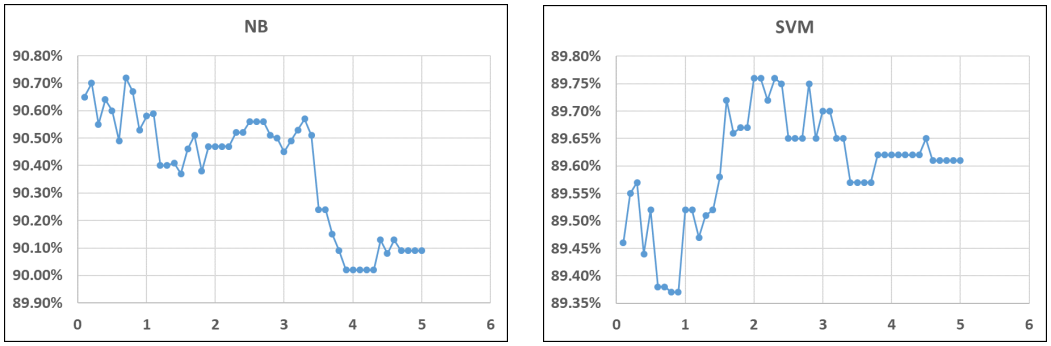
(**a**) F1 score for different *k* values using M-NB (**b**) F1 score for different *k* values using SVM

Figure 8. Results of all considered *k* values for MARSA (Social) dataset.



(**a**) F1 score for different *k* values using M-NB (**b**) F1 score for different *k* values using SVM

Figure 9. Results of all considered *k* values for MARSA (Sport) dataset.



(**a**) F1 score for different *k* values using M-NB (**b**) F1 score for different *k* values using SVM

Figure 10. Results of all considered *k* values for Twitter airline sentiment dataset.

The charts reveal that there are distinct k values (either consecutive or non-consecutive) that obtained the same F1 score. Based on extensive experiments, there could be two reasons for this observation:

1. The same distribution being produced among term groups: Some consecutive k values produce the same distribution among term groups, which immediately produces the same result. For example, the values from 3.7 to 4.3 in Figure 8b obtained the same terms' distribution and the same F1 score. Table 11 presents the number of terms in each group and the F1 score that was obtained.
2. The effect of using certain terms together: A term's presence or absence within a group of terms can significantly affect the result. For example, when using the SVM with MARSA (Sport), consider the results of values 4.4, 4.5, and 4.6, as presented in Table 12.

Although these values produce different term distributions, the non-consecutive values 4.4 and 4.6 obtained the same F1 score. In contrast, the value located between them (4.5) obtained a lower result, as shown in Figure 9b. This is due to the effect of some terms interacting. To provide a solid understanding, Table 13 presents the terms that were included in the $Freq^+$ and $Freq^-$ groups for those values.

As can be seen from Table 13, both 4.4 and 4.5 share the same $Freq^+$ terms. However, in terms of the $Freq^-$ group, value 4.5 missed t_8. Conversely, after comparing the values 4.5 and 4.6, the same $Freq^-$ terms were included in both. However, in terms of the $Freq^+$ group, the value 4.6 missed t_7, which impacts the result.

In summary, the lowest result (88.61%) was produced when $k = 4.5$, due to the presence of t_7 in $Freq^+$ and the absence of t_8 in $Freq^-$. Nonetheless, the presence or absence of both terms, as presented in 4.4 and 4.6, achieved a better result with 88.64%, proving that the obtained result is not penalized according to the number of terms in each group. Instead, it is affected by the co-occurrence of some terms.

Table 11. Term distribution of [3.7–4.3] k for MARSA (Social) using SVM.

k	# G	# $Freq^+$	# $Freq^-$	F1 Score
[3.7–4.3]	2235	12	2	83.22%

Table 12. Term distribution of [4.4, 4.5, 4.6] k for MARSA (Sport) using SVM.

k	# G	# $Freq^+$	# $Freq^-$	F1 Score
4.4	3211	7	10	88.64%
4.5	3212	7	9	88.61%
4.6	3213	6	9	88.64%

The optimal k is specified based on two conditions:

- The value that provides the highest weighted F1 score was chosen because the F1 score incorporates both precision and recall scores.
- Among the set of values that provided the same highest F1 score, the value that provided the lowest number of G terms was chosen.

The results of the proposed method presented in Table 9 and Figures 4–6 were obtained using diverse k values. Table 14 depicts these values for all datasets using both M-NB and SVM. Moreover, the number of terms in all groups is presented.

Table 13. Explanation of the reason that the same F1 score is produced with different k values.

Terms in $Freq^+$					Terms in $Freq^-$				
Term ID	**Term**	$k = 4.4$	$k = 4.5$	$k = 4.6$	**Term ID**	**Term**	$k = 4.4$	$k = 4.5$	$k = 4.6$
t_1	برك	*	*	*	t_1	هلل	*	*	*
t_2	الف	*	*	*	t_2	الي	*	*	*
t_3	فوز	*	*	*	t_3	حكم	*	*	*
t_4	زعم	*	*	*	t_4	دونيس	*	*	*
t_5	بطل	*	*	*	t_5	فتح	*	*	*
t_6	جمل	*	*	*	t_6	فرج	*	*	*
t_7	شكر	*	*	-	t_7	حسب	*	*	*
					t_8	تسل	*	-	-
					t_9	ظلم	*	*	*
					t_{10}	ردس	*	*	*
Total =		7	7	6	**Total =**		10	9	9

The symbol (*) denotes that term t_i is found on the corresponding k value

Table 14. Optimal k value and common term distribution for all datasets.

Model	**Dataset**	k	**# G**	**# $Freq^+$**	**# $Freq^-$**
M-NB	MARSA (Sport)	0.3	2806	156	266
	MARSA (Social)	0.5	1995	111	143
	SenWave	3.1	1751	16	4
	Airline	0.7	1594	57	124
SVM	MARSA (Sport)	3.1	3200	13	15
	MARSA (Social)	3.7	2235	12	2
	SenWave	3	1749	17	5
	Airline	2.3	1737	12	26

5.1.3. Evaluate the Effectiveness of Common Term Grouping

To verify the beneficial effects of classifying common terms into more than one group, the proposed scheme was assessed using two approaches, as follows:

- Approach 1: all common terms were considered general (G); hence, there are only three groups of terms (G, T^+, and T^-).
- Approach 2: the common terms were classified based on the optimal k presented in Table 14. Thus, there are five term groups (G, $Freq^+$, $Freq^-$, T^+, and T^-).

Table 15 presents the results of both approaches and the improvement achieved by grouping the common terms. As can be seen from Table 15, Approach 2 outperformed Approach 1 in seven out of eight experiments. This confirms that a subset of common terms may be useful for enhancing the classification task; thus, it may belong to a specific class and should not be ignored.

Table 15. Evaluation of the term classification phase.

Model	Dataset	Approach 1	Approach 2	Improvement
M-NB	MARSA (Sport)	89.35%	89.00%	−0.35%
	MARSA (Social)	84.93%	85.52%	0.59%
	SenWave	80.37%	80.96%	0.59%
	Airline	89.67%	90.72%	1.05%
SVM	MARSA (Sport)	88.61%	88.64%	0.03%
	MARSA (Social)	83.02%	83.21%	0.19%
	SenWave	76.83%	76.94%	0.11%
	Airline	89.56%	89.76%	0.20%

6. Conclusions and Future Directions

In this study, a novel term weighting approach (TF-TDA) was proposed to effectively enhance text classification tasks, particularly SA. First, a method was used to differentiate between useful and meaningless common terms. Then, the extracted terms were weighted differently to show each term's contribution to classifying a document. Moreover, the effectiveness of the proposed scheme was investigated with other term weighting approaches using M-NB and SVM classifiers to address the SA task. The experiments were conducted using the MARSA (Sport), MARSA (Social), SenWave, and Twitter airline sentiment datasets as benchmark collections.

The experiment results show that the TF-TDA method outperformed two other term weighting approaches in most cases, since the F1 score obtained ranged from 0.52% to 3.99%. In particular, TF-TDA consistently outperformed other term weighting approaches using M-NB. In addition, the term's classification approach improved classification performance. Moreover, the proposed method handled the problem of data imbalances using the term's existence percentage instead of the term's frequency. To overcome the limitations of this study, more research could be conducted using the proposed model in multi-label classes. Moreover, the proposed method could be combined with statistical feature selection methods to investigate the proposed method's performance with different features sizes in future work.

Author Contributions: Methodology, A.A. (Arwa Alshehri); Formal analysis, A.A. (Arwa Alshehri) and A.A. (Abdulmohsen Algarni); Investigation, A.A. (Arwa Alshehri); Resources, A.A. (Abdulmohsen Algarni); Writing—original draft, A.A. (Arwa Alshehri); Writing—review and editing, A.A. (Abdulmohsen Algarni); Supervision, A.A. (Abdulmohsen Algarni); Funding acquisition, A.A. (Abdulmohsen Algarni). All authors have read and agreed to the published version of the manuscript.

Funding: This research was financially supported by the Deanship of Scientific Research at King Khalid University under research grant number (R.G.P.1/188/41).

Data Availability Statement: The datasets have been taken from these links: https://arbml.github.io/masader/ (accessed on 12 October 2022), https://www.kaggle.com/datasets/crowdflower/twitter-airline-sentiment (accessed on 28 February 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dogan, T.; Uysal, A.K. On term frequency factor in supervised term weighting schemes for text classification. *Arab. J. Sci. Eng.* **2019**, *44*, 9545–9560. [CrossRef]
2. Giachanou, A.; Crestani, F. Like it or not: A survey of twitter sentiment analysis methods. *ACM Comput. Surv. (CSUR)* **2016**, *49*, 1–41. [CrossRef]

3. Dogra, V.; Alharithi, F.S.; Álvarez, R.M.; Singh, A.; Qahtani, A.M. NLP-Based Application for Analyzing Private and Public Banks Stocks Reaction to News Events in the Indian Stock Exchange. *Systems* **2022**, *10*, 233. [CrossRef]
4. Kharde, V.; Sonawane, P. Sentiment analysis of twitter data: A survey of techniques. *arXiv* **2016**, arXiv:1601.06971.
5. Narayanaswamy, G.R. Exploiting BERT and RoBERTa to Improve Performance for Aspect Based Sentiment Analysis. Master's Thesis, Technological University Dublin, Dublin, Ireland, 2021.
6. Alruily, M. Classification of arabic tweets: A review. *Electronics* **2021**, *10*, 1143. [CrossRef]
7. Adwan, O.; Al-Tawil, M.; Huneiti, A.; Shahin, R.; Zayed, A.A.; Al-Dibsi, R. Twitter sentiment analysis approaches: A survey. *Int. J. Emerg. Technol. Learn. (iJET)* **2020**, *15*, 79–93. [CrossRef]
8. Aggarwal, C.C. *Machine Learning for Text*; Springer: Berlin/Heidelberg, Germany, 2018; Volume 848. [CrossRef]
9. Shanavas, N. Graph-Theoretic Approaches to Text Classification. Ph.D. Thesis, Ulster University, Ulster, Ireland, 2020.
10. Kumar, A.; Dabas, V.; Hooda, P. Text classification algorithms for mining unstructured data: A SWOT analysis. *Int. J. Inf. Technol.* **2020**, *12*, 1159–1169. [CrossRef]
11. Ezzat, S.; El Gayar, N.; Ghanem, M.M. Sentiment analysis of call centre audio conversations using text classification. *Int. J. Comput. Inf. Syst. Ind. Manag. Appl.* **2012**, *4*, 619–627.
12. Fayyad, U.M.; Piatetsky-Shapiro, G.; Uthurusamy, R. Summary from the KDD-03 panel: Data mining: The next 10 years. *ACM Sigkdd Explor. Newsl.* **2003**, *5*, 191–196. [CrossRef]
13. Prusa, J.D.; Khoshgoftaar, T.M.; Dittman, D.J. Impact of feature selection techniques for tweet sentiment classification. In Proceedings of the Twenty-Eighth International Flairs Conference, Hollywood, FL, USA, 18–20 May 2015.
14. Parlar, T.; Özel, S.A. An Investigation of Term Weighting and Feature Selection Methods for Sentiment Analysis. *Majlesi J. Electr. Eng.* **2018**, *12*, 63–68.
15. Zheng, A.; Casari, A. *Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2018.
16. Domeniconi, G.; Moro, G.; Pasolini, R.; Sartori, C. A comparison of term weighting schemes for text classification and sentiment analysis with a supervised variant of tf. idf. In Proceedings of the International Conference on Data Management Technologies and Applications, Colmar, France, 20–22 July 2015; pp. 39–58. [CrossRef]
17. Wu, H.; Gu, X.; Gu, Y. Balancing between over-weighting and under-weighting in supervised term weighting. *Inf. Process. Manag.* **2017**, *53*, 547–557. [CrossRef]
18. Salton, G.; Buckley, C. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manag.* **1988**, *24*, 513–523. [CrossRef]
19. Jones, D. Group nepotism and human kinship. *Curr. Anthropol.* **2000**, *41*, 779–809. [CrossRef]
20. Liu, Y.; Loh, H.T.; Sun, A. Imbalanced text classification: A term weighting approach. *Expert Syst. Appl.* **2009**, *36*, 690–701. [CrossRef]
21. Leopold, E.; Kindermann, J. Text categorization with support vector machines. How to represent texts in input space? *Mach. Learn.* **2002**, *46*, 423–444. [CrossRef]
22. Jones, K.S. A statistical interpretation of term specificity and its application in retrieval. *J. Doc.* **1972**, eb026526. [CrossRef]
23. Mujahid, M.; Lee, E.; Rustam, F.; Washington, P.B.; Ullah, S.; Reshi, A.A.; Ashraf, I. Sentiment analysis and topic modeling on tweets about online education during COVID-19. *Appl. Sci.* **2021**, *11*, 8438. [CrossRef]
24. Aslam, N.; Xia, K.; Rustam, F.; Hameed, A.; Ashraf, I. Using Aspect-Level Sentiments for Calling App Recommendation with Hybrid Deep-Learning Models. *Appl. Sci.* **2022**, *12*, 8522. [CrossRef]
25. Rustam, F.; Ashraf, I.; Mehmood, A.; Ullah, S.; Choi, G.S. Tweets classification on the base of sentiments for US airline companies. *Entropy* **2019**, *21*, 1078. [CrossRef]
26. Aslam, N.; Xia, K.; Rustam, F.; Lee, E.; Ashraf, I. Self voting classification model for online meeting app review sentiment analysis and topic modeling. *PeerJ Comput. Sci.* **2022**, *8*, e1141. [CrossRef]
27. Altawaier, M.; Tiun, S. Comparison of machine learning approaches on arabic twitter sentiment analysis. *Int. J. Adv. Sci. Eng. Inf. Technol.* **2016**, *6*, 1067–1073. [CrossRef]
28. Wu, H.; Salton, G. A comparison of search term weighting: Term relevance vs. inverse document frequency. In Proceedings of the 4th Annual International ACM SIGIR Conference on Information Storage and Retrieval: Theoretical Issues in Information Retrieval, Oakland, CA, USA, 31 May–2 June 1981; pp. 30–39. [CrossRef]
29. Tokunaga, T.; Iwayama, M. *Text Categorization Based on Weighted Inverse Document Frequency*; Information Processing Society of Japan: Tokyo, Japan, 1994.
30. Martineau, J.; Finin, T. Delta tfidf: An improved feature space for sentiment analysis. In Proceedings of the International AAAI Conference on Web and Social Media, San Jose, CA, USA, 17–20 May 2009; Volume 3, pp. 258–261. [CrossRef]
31. Paltoglou, G.; Thelwall, M. A Study of Information Retrieval Weighting Schemes for Sentiment Analysis. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden, 11–16 July 2010; Association for Computational Linguistics: Uppsala, Sweden, 2010; pp. 1386–1395.
32. 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. [CrossRef]
33. Deng, Z.H.; Luo, K.H.; Yu, H.L. A study of supervised term weighting scheme for sentiment analysis. *Expert Syst. Appl.* **2014**, *41*, 3506–3513. [CrossRef]

34. 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]
35. Carvalho, F.; Guedes, G.P. TF-IDFC-RF: A novel supervised term weighting scheme. *arXiv* **2020**, arXiv:2003.07193.
36. Wang, D.; Zhang, H. Inverse-category-frequency based supervised term weighting scheme for text categorization. *arXiv* **2010**, arXiv:1012.2609. [CrossRef]
37. Ren, F.; Sohrab, M.G. Class-indexing-based term weighting for automatic text classification. *Inf. Sci.* **2013**, *236*, 109–125. [CrossRef]
38. Jiang, Z.; Gao, B.; He, Y.; Han, Y.; Doyle, P.; Zhu, Q. Text classification using novel term weighting scheme-based improved tf-idf for internet media reports. *Math. Probl. Eng.* **2021**, *2021*, 1–30. [CrossRef]
39. 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]
40. Ghosh, S.; Desarkar, M.S. Class specific TF-IDF boosting for short-text classification: Application to short-texts generated during disasters. In Proceedings of the Web Conference 2018, Lyon, France, 23–27 April 2018; pp. 1629–1637. [CrossRef]
41. Roul, R.K.; Sahoo, J.K.; Arora, K. Modified TF-IDF term weighting strategies for text categorization. In Proceedings of the 2017 14th IEEE India Council International Conference (INDICON), Roorkee, India, 15–17 December 2017; pp. 1–6. [CrossRef]
42. Alowisheq, A.; Al-Twairesh, N.; Altuwaijri, M.; Almoammar, A.; Alsuwailem, A.; Albuhairi, T.; Alahaideb, W.; Alhumoud, S. MARSA: Multi-domain Arabic resources for sentiment analysis. *IEEE Access* **2021**, *9*, 142718–142728. [CrossRef]
43. Yang, Q.; Alamro, H.; Albaradei, S.; Salhi, A.; Lv, X.; Ma, C.; Alshehri, M.; Jaber, I.; Tifratene, F.; Wang, W.; et al. Senwave: Monitoring the global sentiments under the COVID-19 pandemic. *arXiv* **2020**, arXiv:2006.10842.
44. Oussous, A.; Benjelloun, F.Z.; Lahcen, A.A.; Belfkih, S. ASA: A framework for Arabic sentiment analysis. *J. Inf. Sci.* **2020**, *46*, 544–559. [CrossRef]
45. Wu, X.; Kumar, V.; Ross Quinlan, J.; Ghosh, J.; Yang, Q.; Motoda, H.; McLachlan, G.J.; Ng, A.; Liu, B.; Yu, P.S.; et al. Top 10 algorithms in data mining. *Knowl. Inf. Syst.* **2008**, *14*, 1–37. [CrossRef]
46. Sabbah, T.; Selamat, A.; Selamat, M.H.; Al-Anzi, F.S.; Viedma, E.H.; Krejcar, O.; Fujita, H. Modified frequency-based term weighting schemes for text classification. *Appl. Soft Comput.* **2017**, *58*, 193–206. [CrossRef]
47. Abdelaal, H.M.; Elmahdy, A.N.; Halawa, A.A.; Youness, H.A. Improve the automatic classification accuracy for Arabic tweets using ensemble methods. *J. Electr. Syst. Inf. Technol.* **2018**, *5*, 363–370. [CrossRef]
48. Duwairi, R.M.; Qarqaz, I. A framework for Arabic sentiment analysis using supervised classification. *Int. J. Data Mining Model. Manag.* **2016**, *8*, 369–381. [CrossRef]
49. AlSalman, H. An improved approach for sentiment analysis of arabic tweets in twitter social media. In Proceedings of the 2020 3rd International Conference on Computer Applications & Information Security (ICCAIS), Riyadh, Saudi Arabia, 19–21 March 2020; pp. 1–4. [CrossRef]
50. Aljabri, M.; Chrouf, S.M.B.; Alzahrani, N.A.; Alghamdi, L.; Alfehaid, R.; Alqarawi, R.; Alhuthayfi, J.; Alduhailan, N. Sentiment analysis of Arabic tweets regarding distance learning in Saudi Arabia during the COVID-19 pandemic. *Sensors* **2021**, *21*, 5431. [CrossRef]
51. Duwairi, R.M.; Marji, R.; Sha'ban, N.; Rushaidat, S. Sentiment analysis in arabic tweets. In Proceedings of the 2014 5th International Conference on Information and Communication Systems (ICICS), Irbid, Jordan, 1–3 April 2014; pp. 1–6.
52. Dietterich, T.G. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Comput.* **1998**, *10*, 1895–1923. [CrossRef] [PubMed]