Combined Term Weighting Scheme using FFNN, GA, MR, Sum, & Average for Text Classification

Mohamed Abdel Fattah[a,c], Mohammad Golam Sohrab[b,c,*]

Abstract— This work presents empirical studies on building a combinational process from different term weighting approaches to address a new Combined-Term-Weighting-Scheme (CTWS) in information access system, especially on automatic text classification (ATC). The CTWS including, TFCC, TFMI, TFOR, TFPB, TFRF, TFIDF, TFICF, TFICS$_\delta$F, TFIDFICF, and TFIDFICS$_\delta$F are used to generate the CTWS approach. Moreover, we introduce five different models to create global weight from a certain weighting scheme to assist the proposed approach. In this study, besides summation and average approaches, well-known mathematical regression (MR), genetic algorithm (GA), and Feed Forward Neural Network (FFNN) are incorporated for creating global weights from a certain weighting scheme. Experiment results show that the proposed combined term weighting schemes including, CTWS-Sum, CTWS-Avg., CTWS-FFNN, CTWS-GA, and CTWS-MR are very effective on the Reuter-21578, 20Newsgroups, and RCV1-v2/LYRL2004 datasets over the Centroid, Naive Bayes (NB) and Support Vector Machine (SVM) classifiers to enhance classification task.

Index Terms— Classifier, mathematical regression, genetic algorithm, feed forward neural network, term weighting, machine learning, Text classification.

1 INTRODUCTION

TEXT classification (TC) has been actively studied due to digital textual documents growing availability, to organize unstructured vast amount of documents to a set of classes, based on the textual content of the document. The majority of categorization tools analyse a text statistically and linguistically, specify significant document terms, then using these significant terms, generate a text to vector representation. Therefore, to enhance automatic text classification, without good text-to-vector representation, effective retrieval is difficult to accomplish [6], [14], [15], [21], [26].

In the statistical classification methods [5], [17], [20], [23], [24], [29], [30], based on document-indexing-based methods, many experiments have been conducted where term frequency incorporated with inverse document frequency (TF.IDF) is considered as the best term weighting criterion to address classification task. Besides, information element-based weighting approaches where four fundamental concepts for a certain term are used to enhance classification task. Most recently, Ren and Sohrab [21] proposed an approach which incorporated document- and class-indexing and reported the diversity of category information to generate more informative term for a specific category in the categorization task.

In recent years, besides statistical weighting approaches, many statistical classification approaches [34], [35], [36], [37] and machine learning approaches like support vector machines [12], [19], [34], probabilistic Bayesian models [1], [33], [38], decision trees [14], [35], Rocchio classifiers [15], [35], and multivariate regression models [34] are attempted to address ATC.

However, text-to-vector representations play a significant role in ATC. An information-rich weighting scheme is required to determine each term significance in differentiating certain document from others in order to judge a pair of documents similarity quantitatively. In terms of an individual weighting approach which contains partial information of a certain document that can not be fully trained by conventional term weighting methods.

Moreover, in the ML-based workbench, several works carried out with different effective weighting approaches. Most of the cases, not even a single weightng approache can significantly outperformed in compare to other approaches. In this work, we develop a combined-term-weighting-approach (CTWS) by sharing the information of different weighting approaches which are considered to be the most effective to enhance information retrieval or classification task. Any combination of influential weighting approaches can be incorporated to generate CTWS. In this work, we generate the CTWS by incorporating document-indexing-, class-indexing, and information-element-based approaches. The motivation of exploiting CTWS is to generate a more information-rich VSM which can be helpful as input to classification models. We therefore, introduce five different models to generate global weight to assist the proposed approach. This work provides several primary contributions based on the proposed CTWS to enhance ATC.

- The CTWS is very effective to help to enrich categorical performances which are performing low either in information-element- or document-indexing- or even in class-indexing-based weighing methods.
- The CTWS enriches every category performance of the Reuters-21578, 20Newsgroups, and RCV1-v2 datasets over the NB, Centroid, and SVM classifiers.
- The proposed approaches are very prominent, especially on NB and SVM classifiers.
- The proposed CTWS expands the existing information-element-, document-indexing-, and class-indexing-based methods in term weighting and generates more informa-

- [a] Department of Electronics Technology, FIE, Helwan University, Cairo, Egypt. E-mail: mohafi2003@helwan.edu.eg
- [b] National Institute of Advanced Industrial Science and Technology, Japan. E-mail: sohrab.mohammad@aist.go.jp
- [*] Corresponding author
- [c] Both authors contributed Equally to this work.

tive terms sharing the weight of different weighting approaches.

The rest of the manuscript is organized as follows. Sec. 2 presents the existence of different weighting schemes. Sec. 3 presents the proposed CTWS. In sec. 4, we elaborate on the Naïve Bayes, SVM and centroid classifiers. Sec. 5 gives experimental settings and results. Sec. 6 shows related work. Sec. 7 shows conclusion.

2 Term Weighting Schemes

Last few years, many experiments conducted based on different term weighting schemes [2], [4], [10], [11], [17], [21] to address the classification task as a statistical method. The most used term weighting methods in ATC are four fundamental information-element-based weighting and document-indexing approaches.

2.1 Document-indexing-based Approach

Document-indexing-based [9], [10], [22] a.k.a. inverse document frequency (IDF) incorporated with term frequency (TF) i.e., TF.IDF. For ATC, TF.IDF is the most popular traditional term weighting method. TF.IDF [29], [36] is calculated as:

$$W_{TF.IDF}(t_i,d_j)=tf_{(t_i,d_j)}\times(1+\log\frac{D}{d(t_i)}), \qquad (1)$$

where *d(ti)* is the collection number of documents in which term t_i occurs once at least, *D* is the collection total number of documents, $tf(t_i,d_j)$ is the term t_i number of occurrences in document d_j, $d(t_i)/D$ is the document frequency (DF), and $D/d(t_i)$ is the IDF of term t_i.

2.2 Class-indexing-based method

Ren and Sohrab [21] proposed two weighting methods TF.IDF.ICS$_\delta$F and TF.IDF.ICF for class-oriented indexing [21], [28], where inverse class space density frequency (ICS$_\delta$F), and inverse category frequency (ICF) as well as the IDF class-Indexing-based are combined with TF. We may create two term weighting schemes TF.ICS$_\delta$F and TF.ICF. These two representations are:

$$W_{TF.ICF}(t_i,d_j,c_k)=tf_{(t_i,d_j)}\times(1+\log\frac{C}{c(t_i)}), \qquad (2)$$

$$W_{TF.ICS_\delta F}(t_i,d_j,c_k)=tf_{(t_i,d_j)}\times(1+\log\frac{C}{CS_\delta(t_i)}), \qquad (3)$$

where $CS_\delta(t_i)/C$ is the class space density frequency (CS$_\delta$F) and $C/CS_\delta(t_i)$ is the ICS$_\delta$F of term t_i, $c(t_i)$ is the number of collection classes in which term t_i occurs once at least, *C* is the collection total number of predefined classes, $c(t_i)/c$ is the category frequency (CF), and $C/c(t_i)$ is the the term (t_i) ICF.

TF.IDF.ICF and TF.IDF.ICS$_\delta$F for specific term t_i in document d_j for the category c_k, are calculated as:

$$W_{TF.IDF.ICF}(t_i,d_j,c_k)=tf_{(t_i,d_j)}\times(1+\log\frac{D}{d(t_i)})\times(1+\log\frac{C}{c(t_i)}), \qquad (4)$$

$$W_{TF.IDF.ICS_\delta F}(t_i,d_j,c_k)=tf_{(t_i,d_j)}\times(1+\log\frac{D}{d(t_i)})\times(1+\log\frac{C}{CS_\delta(t_i)}), \qquad (5)$$

2.3 Information-element-based Approach

Recently, various term weighting methods alongside with document-indexing, including relevance frequency (RF) [13], [17], the probability based (PB) [17], mutual information (MI) [17], [26], odds ratio (OR) [17], [26], and correlation coefficient (CC) [17] have been reported the importance of these term weighting schemes for improving the ATC performance. Therefore, depending on four information elements, we implement these term weighting approaches to compare them with the proposed weighting approaches. The mathematical expressions of TF.CC, TF.MI, TF.OR, TF.PB, and TF.RF weighting schemes are defined as follows:

$$TF.CC=tf_{(t_i,d_j)}\times(\frac{\sqrt{N}(AF-BE)}{\sqrt{(A+E)(B+F)(A+B)(E+F)}}), \qquad (6)$$

$$TF.MI=tf_{(t_i,d_j)}\times\log(\frac{AN}{(A+B)(A+E)}), \qquad (7)$$

$$TF.OR=tf_{(t_i,d_j)}\times\log(\frac{AF}{BE}), \qquad (8)$$

$$TF.PB=tf_{(t_i,d_j)}\times\log(1+\frac{A}{B}\frac{A}{E}), \qquad (9)$$

$$TF.RF=tf_{(t_i,d_j)}\times\log(2+\frac{A}{E}), \qquad (10)$$

where from (6)-(10), *N* = total documents number. *A* = number of documents classified as class c_k for term t_i occurs at least once; *B* = number of documents not classified as class c_k for term t_i occurs at least once; *E* = number of documents classified as class c_k for term t_i does not occur; *F* = number of documents not classified as class c_k for term t_i does not occur.

3 Combined-Term-Weighting Schemes

The Combined-Term-Weighting-Scheme (CTWS) is another criterion of weighting a term, where combining all possible weighting approaches together and generate a new weighting scheme. In this approach, we take the summation of feature parameters associated with the document under consideration to calculate its score. Therefore, the CTWS score for a specific term in a certain document for a certain category is given as:

$$\begin{aligned}CTWS(t_i,d_j,c_k)&=w_1.TF.CC+w_2.TF.MI+w_3.TF.OR+\\&w_4.TF.PB+w_5.TF.RF+w_6.TF.IDF+w_7.TF.ICF+\\&w_8.TF.ICS_\delta F+w_9.TF.IDF.ICF+w_{10}.TF.IDF.ICS_\delta F,\end{aligned} \qquad (11)$$

In (11), for a certain term t_i, a weighted CTWS score function is exploited to integrate the ten feature score values; since w_i is the global weight of a respecting term weighting approach. The global weights *(*w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9,

w_{10}) are calculated from the vector space of TF.CC, TF.MI, TF.OR, TF.PB, TF.RF, TF.IDF, TF.ICF, TF.ICS$_\delta$F, TF.IDF.ICF, and TF.IDF.ICS$_\delta$F respectively. To generate global weight w_i from a respective VSM, we therefore introduce five different approaches, including CTWS with summation (CTWS-Sum), average (CTWS-Avg.), mathematical regression (CTWS-MR), genetic algorithm (CTWS-GA), and feed forward neural network (CTWS-FFNN).

3.1 CTWS-Sum Approach

We assume that, the output of global weights $(w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9, w_{10})$ are between 0 to 1, where 1 is the best and 0 is the worst score. Therefore, the weight between 0 to 1 for a certain global weight w_i $(i = 1, 2, ..., 10)$ is further incorporated with certain term weighting scheme respecting a certain dataset. As such, it must be bounded by,

$$0 \le w_i \le 1, \text{ where, } i = 1, 2, ..., 10$$

In CTWS-Sum approach, we assume that, the maximum weight values of $(w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9, w_{10}) = 1$. We apply (11) after using the defined weights from w_i. Therefore, the CTWS-Sum score is given as in (12):

$$\begin{aligned} CTWS\_Sum(t_i, d_j, c_k) &= w_1.TF.CC + w_2.TF.MI + w_3.TF.OR \\ &+ w_4.TF.PB + w_5.TF.RF + w_6.TF.IDF + w_7.TF.ICF + \\ & w_8.TF.ICS_\delta F + w_9.TF.IDF.ICF + w_{10}.TF.IDF.ICS_\delta F, \end{aligned} \qquad (12)$$

3.2 CTWS-Average Approach

In this approach, we estimate the global weight w_i using CTWS-Avg. model. In the VSM, each document d_j is considered as a term space vector. To calculate the global weights of a specific dataset, first we compute the document weight using (13) from document vector d_j,

$$d_j = \frac{1}{n} \sum_{k=1}^{n} t_k, \qquad (13)$$

since n is the number of terms $t_k (k=1, 2, 3, ..., n)$ in a document d_j. Next, calculate the global weight w_i as follows:

$$w_i = \frac{1}{m} \sum_{j=1}^{m} d_j, \qquad (14)$$

since m is the total number of documents in a dataset. Therefore, we compute the ten different global weight using ten different term weighting approaches for a certain dataset. We apply (11) after using the defined weights from $w_i = w_1, w_2, ..., w_{10}$. The numeric representation of combined term weighting scheme based on average weighting approach for a certain term, is represented as:

$$\begin{aligned} CTWS\_Avg.(t_i, d_j, c_k) &= w_1.TF.CC + w_2.TF.MI + w_3.TF.OR \\ &+ w_4.TF.PB + w_5.TF.RF + w_6.TF.IDF + w_7.TF.ICF + \\ & w_8.TF.ICS_\delta F + w_9.TF.IDF.ICF + w_{10}.TF.IDF.ICS_\delta F, \end{aligned} \qquad (15)$$

3.3 CTWS-MR Approach

In this approach, the global weights of TF.CC, TF.MI, TF.OR, TF.PB, TF.RF, TF.IDF, TF.ICF, TF.ICS$_\delta$F, TF.IDF.ICF, and TF.IDF.ICS$_\delta$F are calculated using Mathematical Regression (MR) model.

3.3.1 Mathematical Regression Model

The MR model is used to create a set of feature weights based on Reuters-21578, 20Newsgroups, and RCV1-v2 datasets. In this model a MR relates output to input. In matrix form, regression may be represented as:

$$\begin{bmatrix} Y_1 \\ Y_2 \\ . \\ . \\ . \\ Y_m \end{bmatrix} = \begin{bmatrix} X_0 1 & X_0 2 & X_0 3 & . & . & X_0 10 \\ X_1 1 & X_1 2 & X_1 3 & . & . & X_1 10 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ X_m 1 & X_m 2 & X_m 3 & . & . & X_m 10 \end{bmatrix} . \begin{bmatrix} w_1 \\ w_2 \\ . \\ . \\ . \\ w_{10} \end{bmatrix}, \qquad (16)$$

where X is the feature parameter input matrix, Y is the output vector, m is the training corpus total number of terms. w_i, the weights $w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9, w_{10}$ in (11) is the system linear statistical model. We apply (11) after exploiting MR weights after execution.

3.4 CTWS-GA Approach

In this approach, the global weights of TF.CC, TF.MI, TF.OR, TF.PB, TF.RF, TF.IDF, TF.ICF, TF.ICS$_\delta$F, TF.IDF.ICF, and TF.IDF.ICS$_\delta$F are calculated using genetic algorithm (GA) model.

3.4.1 Genetic Algorithm Model

The basic concept of genetic algorithms (GAs) is optimization. Since optimization problem arise frequently and GA performs outstanding in optimization where many of the real world problems involved finding optimal parameters. Therefore, the GA is used to create a set of feature weights based on Reuters-21578, 20Newsgroups, and RCV1-v2 datasets. Combination of all feature weights in the form of ($w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9, w_{10}$) is used to represent a chromosome. For each generation 1500 genomes are produced. Each generation involves selecting the best member, performing crossover and mutation and evaluate fitness of each genome. In this experiment, to obtain steady of feature weights, one hundred fifty generations are evaluated. We apply (11) after exploiting GA weights after execution. Therefore, the numeric representation of CTWS based on GA for a certain term is represented as:

$$\begin{aligned} CTWS\_GA(t_i, d_j, c_k) &= w_1.TF.CC + w_2.TF.MI + w_3.TF.OR \\ &+ w_4.TF.PB + w_5.TF.RF + w_6.TF.IDF + w_7.TF.ICF + \\ & w_8.TF.ICS_\delta F + w_9.TF.IDF.ICF + w_{10}.TF.IDF.ICS_\delta F, \end{aligned} \qquad (17)$$

3.5 CTWS-FFNN Approach

The FFNN is exploited to obtain an appropriate global weight using different weighting schemes based on Reuters-21578, 20Newsgroups, and RCV1-v2 datasets. The neural network layered structure that we use is shown in Fig. 1. We use 1 output unit; 10 hidden units and 10 input units to represent this network. The input unit represents the weight of a certain term using a certain weighting scheme as described in Section 2.

All the input features are represented by the feature vector *X*. The hidden layer output is given as:

$$O_j^{(1)} = f(\sum_{k=1}^{N} W_{jk}^{(1)} X_k), \tag{18}$$

where W_{jk} is the weight based on the line between the hidden unit *j* and the input unit *k*. A sigmoidal function *f* is calculated as:

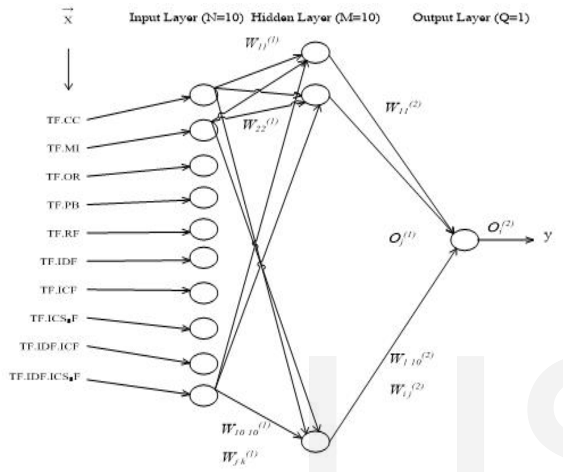


Fig. 1. The Feed Forward Neural Network Structure.

$$f(z) = \frac{1}{1+\exp(-z)}. \tag{19}$$

The output layer output is calculated as:

$$O_i^{(2)} = f(\sum_{k=1}^{M} W_{ij}^{(2)} O_j^{(1)}), \tag{20}$$

since W_{ij} is the weight based on the line between the output unit *i* and the hidden unit *j*. From (18) and (20),

$$O_i^{(2)} = f(\sum_{k=1}^{M} W_{ij}^{(2)} f(\sum_{k=1}^{N} W_{jk}^{(1)} X_k)), \tag{21}$$

The mean squared error “MSE” is calculated as:

$$E_{mse} = \frac{1}{2}\frac{1}{N}\sum_{i=1}^{N}\sum_{x \in N}\left[y_i(x) - O_i^{(2)}(x)\right]^2, \tag{22}$$

where *N* = training data size (cardinality), $y_i(x)$ is the required output value of the neural network while the input is *x*.

The output $O_i^{(2)}$ further represents global weight $w_i = \{w_1, w_2, ..., w_{10}\}$ using a certain weighting scheme. The global weights w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8, w_9, and w_{10} are incorporated with different term weighting approach of TF.CC, TF.MI, TF.OR, TF.PB, TF.RF, TF.IDF, TF.ICF, TF.ICS$_\delta$F, TF.IDF.ICF, and TF.IDF.ICS$_\delta$F respectively. Therefore, the numeric representation of CTWS based on FFNN for a certain term is represented as in (23):

$$\begin{aligned} &CTWS\_FFNN(t_i, d_j, c_k) = w_1.TF.CC + w_2.TF.MI + w_3.TF.OR \\ &+ w_4.TF.PB + w_5.TF.RF + w_6.TF.IDF + w_7.TF.ICF + \\ &w_8.TF.ICS_\delta F + w_9.TF.IDF.ICF + w_{10}.TF.IDF.ICS_\delta F, \end{aligned} \tag{23}$$

4 Classifiers

In the machine learning workbench, besides classic classifier for text classification like centroid and some other classifier like naïve bayes and support vector machines have ATC good performance. Therefore, to judge the effectiveness of different weighting methods, these three classifiers are taken into account.

4.1 Centroid Classifier

In this work, we implement the centroid model [8], [31], [32] to judge the proposed CTWS-based term weighting methods performance then compare it with other different traditional weighting approaches. A document d_j is represented as a term space vector. To find a specific class c_k centroid, add the document training data of vectors $d_j(j = 1, 2, ..., n)$ in the class $c_k(k = 1, 2, ..., m)$:

$$C_k^{sum} = \sum_{d \in c_k} d_j, \tag{24}$$

The normalized version of C_k^{sum} is given as:

$$C_k^{norm} = \frac{C_k^{sum}}{\left\| C_k^{sum} \right\|_2}, \tag{25}$$

where $\left\| C_k^{sum} \right\|_2$ is the 2-norm vector. Next, the similarity between each normalized centroid class vector and a query document is calculated based on inner-product as:

$$sim(d_j, c_k) = d_j.C_k^{norm}. \tag{26}$$

Thus, the test vector d_j is classified as class level c_k whose category prototype is the most similar to the query vector as follows.

$$L(C_k) = \arg\max_{c_k \in C}(d_j.C_k^{norm}). \tag{27}$$

4.2 Naive Bayes Classifier

The assumption in Bayesian model depends on a posterior and prior probability. The probability of a specific document $d_j \in C$ may be calculated using the observation t_i. Depending on Bayes' rule, the conditional probability $P(C|t_i)$ may be calculated as:

$$P(C \mid t_i) = \frac{P(t_i \mid C)P(C)}{P(t_i)}, \qquad (28)$$

We can therefore omit the probability *P(t$_i$)* since the denominator does not depend on the category. The probability *P(t$_i$|C)* may be estimated as:

$$P(t_i \mid C_k) = \prod_{i=1}^{m} P(t_i \mid C_k), \qquad (29)$$

Assume for a normal distribution, each term has a probability density function with standard deviation σ and mean μ in each category *c*. Then Eq. 29 may be typed as:

$$P(t_i \mid C) = \prod_{i=1}^{m} P(t_i; \mu_{i,c}, \sigma_{i,c})$$
$$= \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma_{i,c}^2}} \exp - \frac{(t_i - \mu_{i,c})^2}{2\sigma_{i,c}^2}. \qquad (30)$$

The probability logarithm for all m terms in corpus is elaborated in (31) based on this probability for a certain term:

$$\ln P(T_i \mid C_k) = \ln \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma_{i,c}^2}} \exp - \frac{(t_i - \mu_{i,c})^2}{2\sigma_{i,c}^2}$$

$$= \sum_{i}^{m} \ln \frac{1}{\sqrt{2\pi\sigma_{i,c}^2}} \exp - \frac{(t_i - \mu_{i,c})^2}{2\sigma_{i,c}^2}$$

$$= -\frac{m}{2}\ln(2\pi) - \frac{m}{2}\ln(\sigma_{i,c}^2) - \frac{m}{2}\sum_{i}^{m} \frac{(t_i - \mu_{i,c})^2}{\sigma_{i,c}^2}. \qquad (31)$$

Using a Laplacean prior, the prior probability of specific category is estimated as in (32):

$$P(C) = \frac{1 + N_{t,c}}{D + N_c}, \qquad (32)$$

where *D* is the total number of documents, *N$_c$* is the total number of categories, and *N$_{t,c}$* is the number of documents in a certain class *c$_k$*.

4.3 Support Vector machines

SVMs are the most accurate and robust models among all models [35]. Hence, SVM-Light is exploited in this experiment. Default values of SVM parameters are used.

5 Evaluations

Effectiveness of our CTWS over different term weighting approaches is provided in this section. In these experiments, we exploit 10-fold cross validation technique for the first two datasets. Reuters-21578 and 20Newsgroups data are randomly divided to 10-folds. For testing, one fold is used and the rest are used for training in each turn. We have kept the same data fold and experimental setup which is exploited in Ren & Sohrab [21]. For the third dataset, we divided the data to testing and training data that is explained in Section 5.1.3.

5.1 Experimental datasets

To investigate the effectiveness of the proposed CTWS including, CTWS-Sum, CTWS-Avg., CTWS-FFNN, CTWS-GA, and CTWS-MR with existing ten different baseline methods, we have conducted our experiments using Reuters-21578, 20Newsgroups, and RCV1-v2 / LYRL2004 [16].

5.1.1 Reuters-21578 dataset

This corpus contains 9,976 documents. We merged the testing and training documents together since the system is judged based on 10-fold cross validation.

5.1.2 20Newsgroups dataset

This dataset contains about 18,828 news articles in 20 newsgroups.

5.1.3 RCV1-v2 dataset/LYRL2004

From four parent topics, this dataset contains 103 categories in 804,414 documents. We extracted about 23,000 documents out of 804,414 as single-labeled documents. For single-label classification task, we have created larger corpus by extracting all the documents that are labelled with at least two classes a parent with child category. For each document, to produce single-label classification, the parent category is removed and child category is assigned. Now we have a total of 219,667 documents which falls into 54 different categories. 196,518 documents are used for testing and 23,149 documents are used for training.

5.2 Performance measurement

For performance measurement, we used recall, precision, and F$_1$-measure [5], [37]. For a target category *c$_k$*, F$_1$-measure is defined as:

$$F_1(C_k) = \frac{2.TP(C_k)}{2TN(C_k) + FP(C_k) + FN(C_k)} = \frac{2.PR}{P + R}, \qquad (33)$$

where TN(C$_k$) is the set of test documents correctly rejected, *FN(C$_k$)* is the set of test documents wrongly rejected, *FP(C$_k$)* is the set of test documents incorrectly classified to the category, and *TP(C$_k$)* is the set of test documents correctly classified to the category *c$_k$*. *P* is precision and *R* is recall.

The effectiveness across a set of categories is measured using Macro-average. The macro-average F$_1$-measure (F_1^M) is defined as:

$$F_1^M = \frac{1}{m}\sum_{k=1}^{m} F_1(C_k), \qquad (34)$$

where *m* is the number of classes in a certain dataset. We also measure the micro-average (F_1^μ) which computes the effectiveness based on per-category contingency tables sum.

$$F_1^{\mu} = \frac{2.P^{\mu}.R^{\mu}}{P^{\mu}+R^{\mu}} \qquad (35)$$

where P^{μ} is the micro-average of precision, i. e.,

$$P^{\mu} = \frac{\sum_{k=1}^{m} TP(C_k)}{\sum_{k=1}^{m} TP(C_k) + \sum_{k=1}^{m} FP(C_k)} \qquad (36)$$

and R^{μ} is the micro-average of recall, i. e.,

$$R^{\mu} = \frac{\sum_{k=1}^{m} TP(C_k)}{\sum_{k=1}^{m} TP(C_k) + \sum_{k=1}^{m} FN(C_k)} \qquad (37)$$

5.3 Results

We present results on different weighting approaches over the centroid, NB, and SVM classifiers. To compare the categorical and overall performances, we use three benchmark datasets to judge the effectiveness in ATC.

5.3.1 Categorical Performance Comparison

Figures 2(a), 3(a), and 4(a) for the Reuters-21578 and Figs. 5(a), 6(a), and 7(a) for the 20Newsgroups, show the categorical performance based on F_1-measure of information-element-, document-indexing-, and class-indexing-based approaches over the centroid, NB, and SVM classifier respectively. As shown in the figures some categories are performing very low over different weighting approaches. In Figs. 2(b), 3(b), and 4(b) for the Reuters-21578 and Figures 5(b), 6(b), and 7(b) for the 20Newsgroups, show that applying CTWS including CTWS-Sum, CTWS-Avg., CTWS-FFNN, CTWS-GA, and CTWS-MR over the different classifiers, it enriches each and every categories performance over the NB, Centroid, and SVM classifiers.

The above results show that the combinational process including CTWS-Sum, CTWS-Avg., CTWS-FFNN, CTWS-GA, and CTWS-MR approaches are very effective to help to enrich categorical performance which are performing low either in information-element- or document-indexing- and class-indexing-based weighing methods.

5.3.2 Overall Performance Comparison of Efficient Weighting Approaches

In this paper, we compare the CTWS with ten different weighting approaches. First, we select the most efficient weighting approaches among the ten different weighting approaches in different classifiers and compare the most effective weighting approaches with the CTWS. Ren and Sohrab [21] introduces the class-indexing-based where the scores of the TF.IDF.ICF and TF.IDF.ICS$_{\delta}$F are taken into account to judge the most effective weighting methods in different classifiers and datasets.

Tables 1, 2, and 3 explain the performance comparison with $F_1{}^M$ and $F_1{}^{\mu}$ on different term weighting approaches based on Reuters-21578, 20Newsgroups, and RCV1-v2 datasets using

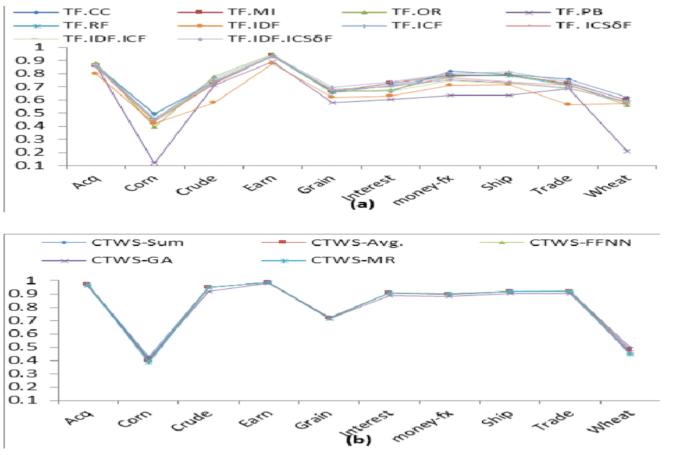


Fig. 2. Categorical performance based on F_1-measure in the Reuters-21578 dataset over the centroid classifier (a) Information-element-based, document- and class-indexing-based (b) CTWS.

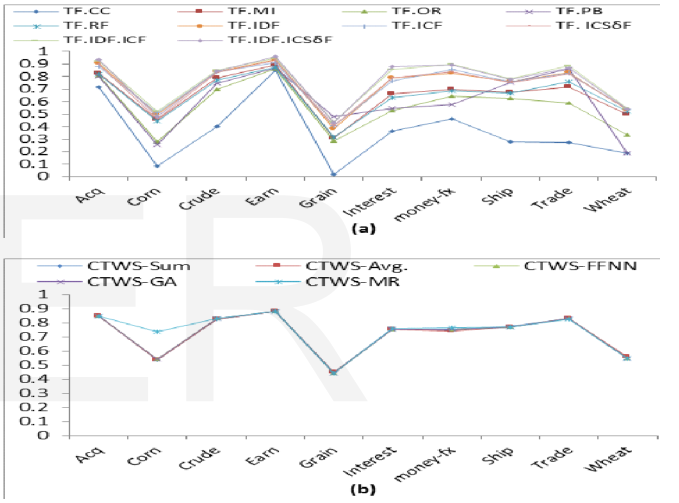


Fig. 3. Categorical performance based on F_1-measure in the Reuters-21578 dataset over the NB classifier: (a) Information-element-based, document and class-indexing-based (b) CTWS.

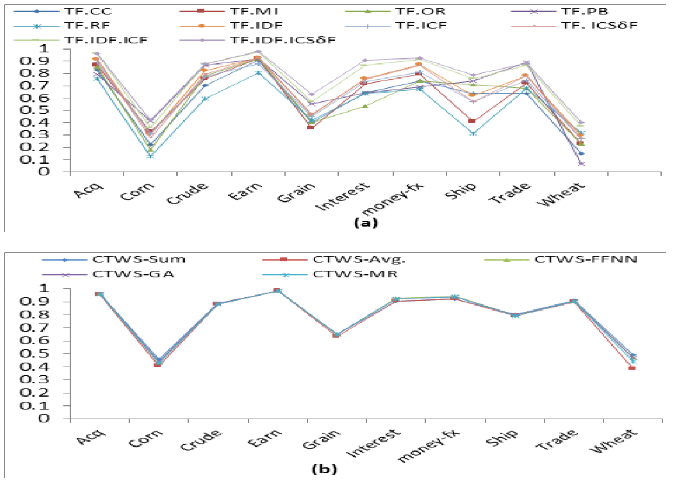


Fig. 4. Categorical performance based on F_1-measure in the Reuters-21578 dataset over the SVM classifier: (a) Information-element-based, document and class-indexing-based (b) CTWS.

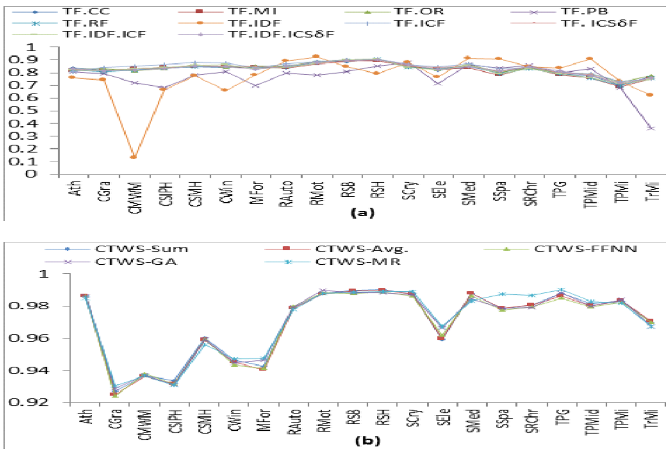


Fig. 5. Categorical performance based on F1-measure in the 20Newsgroups dataset over the centroid classifier (a) Information-element-based, document- and class-indexing-based (b) CTWS.

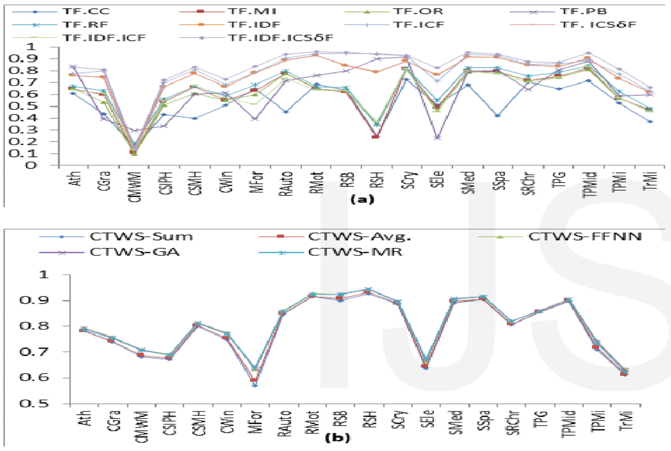


Fig. 6. Categorical performance based on F1-measure in the 20Newsgroups dataset over the NB classifier (a) Information-element-based, document- and class-indexing-based (b) CTWS.

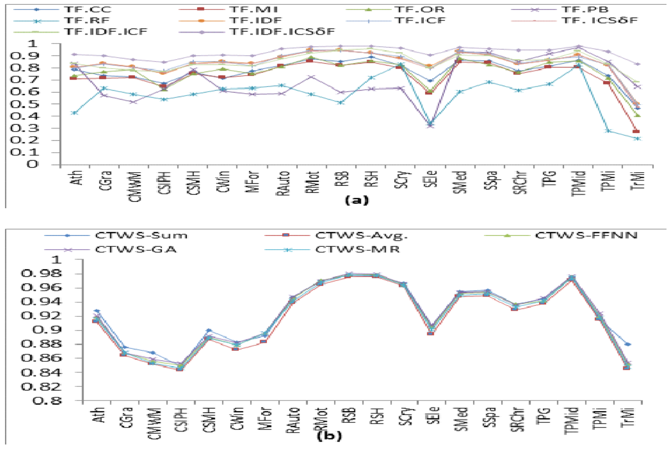


Fig. 7. Categorical performance based on F1-measure in the 20Newsgroups dataset over the SVM classifier (a) Information-element-based, document- and class-indexing-based (b) CTWS.

NB, centroid, and SVM classifiers respectively. In Table 1, the TF.IDF.ICS$_\delta$F shows its superiority both in F_1^M and F_1^μ using SVM however it has low performances in NB where TF.ICS$_\delta$F shows its own superiority. It is also noticeable in centroid classifier, where the TF.IDF.ICF and TF.IDF.ICS$_\delta$F outperformed in F_1^M and F_1^μ respectively.

In Table 2 the TF.IDF.ICS$_\delta$F enriches in every cases over the SVM, centroid, and NB classifier. The TF.IDF.ICS$_\delta$F in Table 3 outperformed in NB and SVM classifiers where TF.ICS$_\delta$F and TF.IDF.ICF shows its superiority in F_1^M and F_1^μ respectively over the centroid classifier.

5.3.3 Oveall Performance Comparison of CTWS

In this task, we compare the CTWS including CTWS-Sum, CTWS-Avg., CTWS-FFNN, CTWS-GA, and CTWS-MR with the most efficient weighting methods including TF.IDF.ICS$_\delta$F, TF.IDF.ICF, and TF.ICS$_\delta$F which are outperformed in Table 1, 2, and 3. In Table 4 and 5 with Reuters-21578 dataset, the F_1^M and F_1^μ of CTWS-Sum, CTWS-Avg., CTWS-FFNN, CTWS-GA, and CTWS-MR show significant improvement over the TF.IDF.ICS$_\delta$F, TF.IDF.ICF, and TF.ICS$_\delta$F using SVM and NB classifiers. We may also notice that the CTWS shows a drop in centroid classifier. In NB, the CTWS shows an improvement over (8-11)% both F_1^M and F_1^μ in NB classifier.

In the 20Newsgroups dataset where Table 6 and 7, the F_1^M and F_1^μ of CTWS enriches system performances not only from the NB and SVM classifier but also some cases in centroid classifier. The CTSW shows an improvement over (10-14)% both F_1^M and F_1^μ in NB and a significant improvement in SVM classifier.

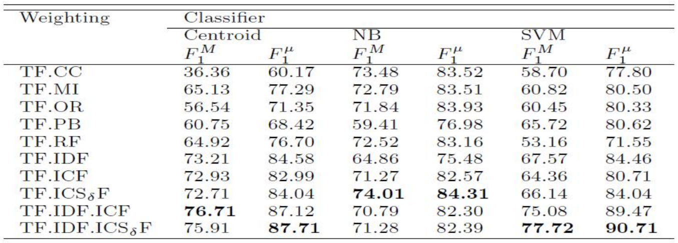
Finally, in RCV1-v2 dataset where Tables 8 and 9, the CTWS-Sum, CTWS-Avg., CTWS-FFNN, CTWS-GA, and CTWS-MR shows its superiority over the NB and SVM classifiers. In centroid classifier the CTWS shows a drop from the TF.IDF.ICS$_\delta$F, TF.IDF.ICF, and TF.ICS$_\delta$F. In Tables 4, 5, 6, 7, 8, and 9, where results in parentheses denote the performance decrease or increase from the TF.IDF.ICS$_\delta$F, TF.IDF.ICF, and F.ICS$_\delta$F respectively.

5.3.4 Discussions

In this paper, we use three different datasets where the categories domain of Reuters-21578 and RCV1-v2 datasets are unbalanced and the categories domain of 20Newsgroups are uniformly distributed. In these datasets, Tables 4, 5, 6, 7, 8, and 9 show that the performances of F_1^μ are very effective in unbalanced or even in balanced datasets. This indicate that the CTWS-Sum, CTWS-Avg., CTWS-FFNN, CTWS-GA, and CTWS-MR weighting approaches are not biased with larger categories domain where most of the cases information element-based weighting approaches are unable to predict a test document from a comparatively smaller categories domain.

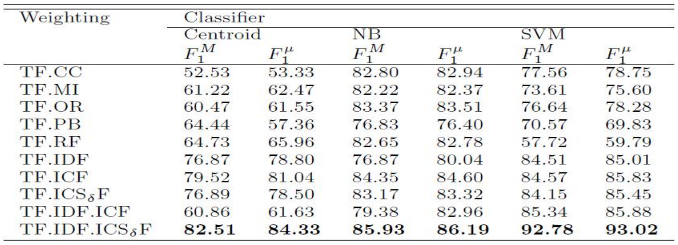
It is also noticeable that the CTWS performs relatively low in centroid classifier. Since the CTWS is a combinational approach where the information element-based approaches in Tables 1, 2 and 3 are performing relatively very low in compare to other weighting approaches including TF.IDF, TD.ICF, TF.ICS$_\delta$F, TF.IDF.ICF, and TF.IDF.ICS$_\delta$F. Thus, if we select the most efficient weighting approaches among all, we believe that the performances will not only improve on the centroid classifier but also in NB as well as in SVM classifiers.

TABLE 1
Performance Comparison of Efficient Weighting Approaches using F_1^M and F_1^μ in the Reuters-21578



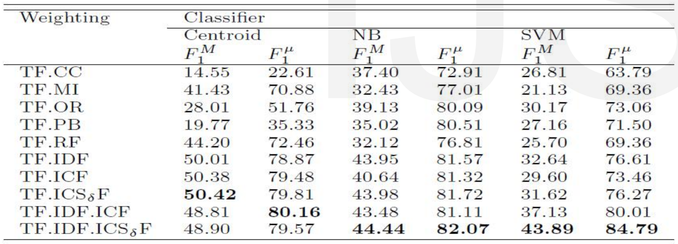
Weighting	Classifier					
	Centroid		NB		SVM	
	F_1^M	F_1^μ	F_1^M	F_1^μ	F_1^M	F_1^μ
TF.CC	36.36	60.17	73.48	83.52	58.70	77.80
TF.MI	65.13	77.29	72.79	83.51	60.82	80.50
TF.OR	56.54	71.35	71.84	83.93	60.45	80.33
TF.PB	60.75	68.42	59.41	76.98	65.72	80.62
TF.RF	64.92	76.70	72.52	83.16	53.16	71.55
TF.IDF	73.21	84.58	64.86	75.48	67.57	84.46
TF.ICF	72.93	82.99	71.27	82.57	64.36	80.71
TF.ICS$_\delta$F	72.71	84.04	**74.01**	**84.31**	66.14	84.04
TF.IDF.ICF	**76.71**	87.12	70.79	82.30	75.08	89.47
TF.IDF.ICS$_\delta$F	75.91	**87.71**	71.28	82.39	**77.72**	**90.71**

TABLE 2
Performance Comparison of Efficient Weighting Approaches using F_1^M and F_1^μ in the 20Newsgroups



Weighting	Classifier					
	Centroid		NB		SVM	
	F_1^M	F_1^μ	F_1^M	F_1^μ	F_1^M	F_1^μ
TF.CC	52.53	53.33	82.80	82.94	77.56	78.75
TF.MI	61.22	62.47	82.22	82.37	73.61	75.60
TF.OR	60.47	61.55	83.37	83.51	76.64	78.28
TF.PB	64.44	57.36	76.83	76.40	70.57	69.83
TF.RF	64.73	65.96	82.65	82.78	57.72	59.79
TF.IDF	76.87	78.80	76.87	80.04	84.51	85.01
TF.ICF	79.52	81.04	84.35	84.60	84.57	85.83
TF.ICS$_\delta$F	76.89	78.50	83.17	83.32	84.15	85.45
TF.IDF.ICF	60.86	61.63	79.38	82.96	85.34	85.88
TF.IDF.ICS$_\delta$F	**82.51**	**84.33**	**85.93**	**86.19**	**92.78**	**93.02**

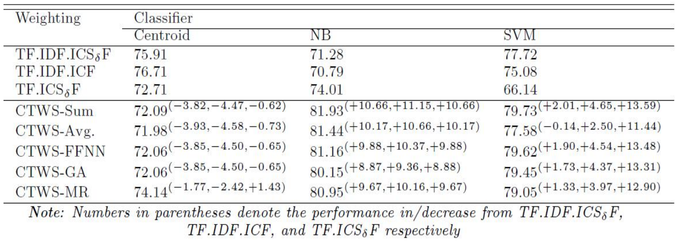
TABLE 3
Performance Comparison of Efficient Weighting Approaches using F_1^M and F_1^μ in the RCV1-v2



Weighting	Classifier					
	Centroid		NB		SVM	
	F_1^M	F_1^μ	F_1^M	F_1^μ	F_1^M	F_1^μ
TF.CC	14.55	22.61	37.40	72.91	26.81	63.79
TF.MI	41.43	70.88	32.43	77.01	21.13	69.36
TF.OR	28.01	51.76	39.13	80.09	30.17	73.06
TF.PB	19.77	35.33	35.02	80.51	27.16	71.50
TF.RF	44.20	72.46	32.12	76.81	25.70	69.36
TF.IDF	50.01	78.87	43.95	81.57	32.64	76.61
TF.ICF	50.38	79.48	40.64	81.32	29.60	73.46
TF.ICS$_\delta$F	**50.42**	79.81	43.98	81.72	31.62	76.27
TF.IDF.ICF	48.81	**80.16**	43.48	81.11	37.13	80.01
TF.IDF.ICS$_\delta$F	48.90	79.57	**44.44**	**82.07**	**43.89**	**84.79**

The results of above experiments also show that the proposed CTWS-Sum, CTWS-Avg., CTWS-FFNN, CTWS-GA, and CTWS-MR term weighting approaches consistently performs higher and very effective to enrich categorical and over all performances; especially in NB and SVM classifiers to enhance classification task. Our results show that the CTWS approaches are very effective with SVM method in three different datasets. Moreover, the result shows that FFNN, GA, and MR models are very good models to improve the weights and enhance categorical and overall performances. Thus, the CTWS approaches are useful for ATC enhancement.

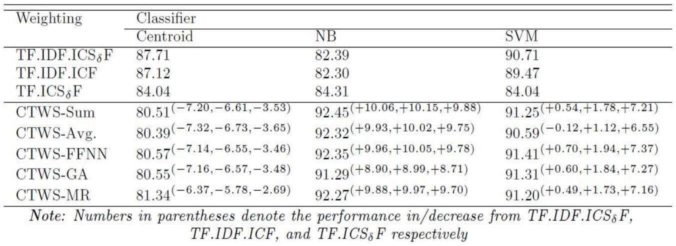
TABLE 4
Performance Comparison of CTWS with Efficient Weihting Approaches using F_1^M in the Reuters-21578



Weighting	Classifier		
	Centroid	NB	SVM
TF.IDF.ICS$_\delta$F	75.91	71.28	77.72
TF.IDF.ICF	76.71	70.79	75.08
TF.ICS$_\delta$F	72.71	74.01	66.14
CTWS-Sum	$72.09^{(-3.82,-4.47,-0.62)}$	$81.93^{(+10.66,+11.15,+10.66)}$	$79.73^{(+2.01,+4.65,+13.59)}$
CTWS-Avg.	$71.98^{(-3.93,-4.58,-0.73)}$	$81.44^{(+10.17,+10.66,+10.17)}$	$77.58^{(-0.14,+2.50,+11.44)}$
CTWS-FFNN	$72.06^{(-3.85,-4.50,-0.65)}$	$81.16^{(+9.88,+10.37,+9.88)}$	$79.62^{(+1.90,+4.54,+13.48)}$
CTWS-GA	$72.06^{(-3.85,-4.50,-0.65)}$	$80.15^{(+8.87,+9.36,+8.88)}$	$79.45^{(+1.73,+4.37,+13.31)}$
CTWS-MR	$74.14^{(-1.77,-2.42,+1.43)}$	$80.95^{(+9.67,+10.16,+9.67)}$	$79.05^{(+1.33,+3.97,+12.90)}$

Note: Numbers in parentheses denote the performance in/decrease from TF.IDF.ICS$_\delta$F, TF.IDF.ICF, and TF.ICS$_\delta$F respectively

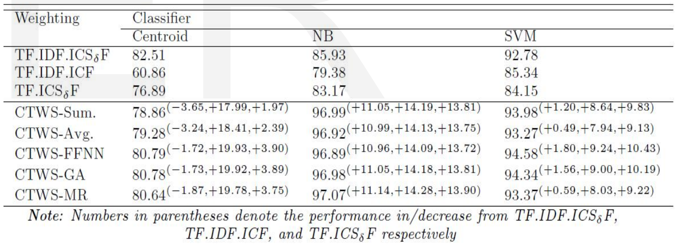
TABLE 5
Performance Comparison of CTWS with Efficient Weihting Approaches using F_1^μ in the Reuters-21578



Weighting	Classifier		
	Centroid	NB	SVM
TF.IDF.ICS$_\delta$F	87.71	82.39	90.71
TF.IDF.ICF	87.12	82.30	89.47
TF.ICS$_\delta$F	84.04	84.31	84.04
CTWS-Sum	$80.51^{(-7.20,-6.61,-3.53)}$	$92.45^{(+10.06,+10.15,+9.88)}$	$91.25^{(+0.54,+1.78,+7.21)}$
CTWS-Avg.	$80.39^{(-7.32,-6.73,-3.65)}$	$92.32^{(+9.93,+10.02,+9.75)}$	$90.59^{(-0.12,+1.12,+6.55)}$
CTWS-FFNN	$80.57^{(-7.14,-6.55,-3.46)}$	$92.35^{(+9.96,+10.05,+9.78)}$	$91.41^{(+0.70,+1.94,+7.37)}$
CTWS-GA	$80.55^{(-7.16,-6.57,-3.48)}$	$91.29^{(+8.90,+8.99,+8.71)}$	$91.31^{(+0.60,+1.84,+7.27)}$
CTWS-MR	$81.34^{(-6.37,-5.78,-2.69)}$	$92.27^{(+9.88,+9.97,+9.70)}$	$91.20^{(+0.49,+1.73,+7.16)}$

Note: Numbers in parentheses denote the performance in/decrease from TF.IDF.ICS$_\delta$F, TF.IDF.ICF, and TF.ICS$_\delta$F respectively

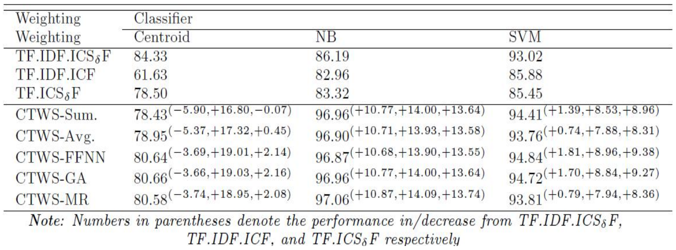
TABLE 6
Performance Comparison of CTWS with Efficient Weihting Approaches using F_1^M in 20Newsgroups



Weighting	Classifier		
	Centroid	NB	SVM
TF.IDF.ICS$_\delta$F	82.51	85.93	92.78
TF.IDF.ICF	60.86	79.38	85.34
TF.ICS$_\delta$F	76.89	83.17	84.15
CTWS-Sum.	$78.86^{(-3.65,+17.99,+1.97)}$	$96.99^{(+11.05,+14.19,+13.81)}$	$93.98^{(+1.20,+8.64,+9.83)}$
CTWS-Avg.	$79.28^{(-3.24,+18.41,+2.39)}$	$96.92^{(+10.99,+14.13,+13.75)}$	$93.27^{(+0.49,+7.94,+9.13)}$
CTWS-FFNN	$80.79^{(-1.72,+19.93,+3.90)}$	$96.89^{(+10.96,+14.09,+13.72)}$	$94.58^{(+1.80,+9.24,+10.43)}$
CTWS-GA	$80.78^{(-1.73,+19.92,+3.89)}$	$96.98^{(+11.05,+14.18,+13.81)}$	$94.34^{(+1.56,+9.00,+10.19)}$
CTWS-MR	$80.64^{(-1.87,+19.78,+3.75)}$	$97.07^{(+11.14,+14.28,+13.90)}$	$93.37^{(+0.59,+8.03,+9.22)}$

Note: Numbers in parentheses denote the performance in/decrease from TF.IDF.ICS$_\delta$F, TF.IDF.ICF, and TF.ICS$_\delta$F respectively

TABLE 7
Performance Comparison of CTWS with Efficient Weihting Approaches using F_1^μ in the 20Newsgroups



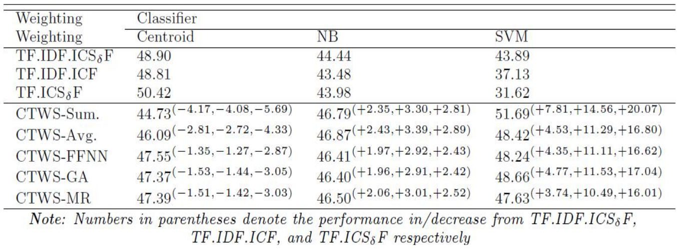
Weighting	Classifier		
Weighting	Centroid	NB	SVM
TF.IDF.ICS$_\delta$F	84.33	86.19	93.02
TF.IDF.ICF	61.63	82.96	85.88
TF.ICS$_\delta$F	78.50	83.32	85.45
CTWS-Sum.	$78.43^{(-5.90,+16.80,-0.07)}$	$96.96^{(+10.77,+14.00,+13.64)}$	$94.41^{(+1.39,+8.53,+8.96)}$
CTWS-Avg.	$78.95^{(-5.37,+17.32,+0.45)}$	$96.90^{(+10.71,+13.93,+13.58)}$	$93.76^{(+0.74,+7.88,+8.31)}$
CTWS-FFNN	$80.64^{(-3.69,+19.01,+2.14)}$	$96.87^{(+10.68,+13.90,+13.55)}$	$94.84^{(+1.81,+8.96,+9.38)}$
CTWS-GA	$80.66^{(-3.66,+19.03,+2.16)}$	$96.96^{(+10.77,+14.00,+13.64)}$	$94.72^{(+1.70,+8.84,+9.27)}$
CTWS-MR	$80.58^{(-3.74,+18.95,+2.08)}$	$97.06^{(+10.87,+14.09,+13.74)}$	$93.81^{(+0.79,+7.94,+8.36)}$

Note: Numbers in parentheses denote the performance in/decrease from TF.IDF.ICS$_\delta$F, TF.IDF.ICF, and TF.ICS$_\delta$F respectively

6 Related Work

Ren and Sohrab [21] conducted their experiment based on eight weighing algorithms, where term frequency (TF) is incorporated with global weights including inverse document frequency incorporated with inverse class space density frequency (TF.IDF.ICS$_\delta$F), relevance frequency (TF.RF), probability based (TF.PB), odds ratio (TF.OR), mutual information (TF.MI), coefficient correlation (TF.CC), inverse document frequency (TF.IDF), and inverse document frequency incorporated with inverse class frequency (TF.IDF.ICF). TF.IDF.ICS$_\delta$F is novel with SVM as shown in the results. The TF.IDF.ICS$_\delta$F method showed its superiority for a majority of the Reuters-21578 and all the categories of the 20Newsgroups datasets using SVM. However, the performances with other classifiers like naïve bayes (NB) and centroid, where TF.IDF.ICS$_\delta$F was unable to show its superiority comparing with other

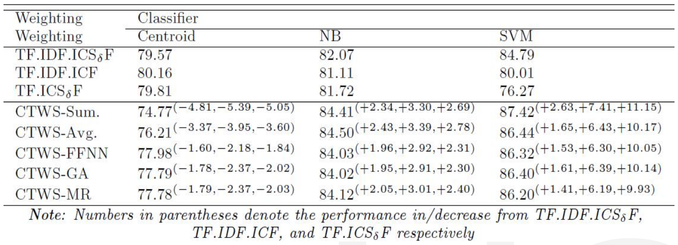
TABLE 8
PERFORMANCE COMPARISON OF CTWS WITH EFFICIENT WEIHTING APPROACHES USING F_1^M IN RCV1-V2



Weighting	Classifier		
Weighting	Centroid	NB	SVM
TF.IDF.ICS$_\delta$F	48.90	44.44	43.89
TF.IDF.ICF	48.81	43.48	37.13
TF.ICS$_\delta$F	50.42	43.98	31.62
CTWS-Sum.	$44.73^{(-4.17,-4.08,-5.69)}$	$46.79^{(+2.35,+3.30,+2.81)}$	$51.69^{(+7.81,+14.56,+20.07)}$
CTWS-Avg.	$46.09^{(-2.81,-2.72,-4.33)}$	$46.87^{(+2.43,+3.39,+2.89)}$	$48.42^{(+4.53,+11.29,+16.80)}$
CTWS-FFNN	$47.55^{(-1.35,-1.27,-2.87)}$	$46.41^{(+1.97,+2.92,+2.43)}$	$48.24^{(+4.35,+11.11,+16.62)}$
CTWS-GA	$47.37^{(-1.53,-1.44,-3.05)}$	$46.40^{(+1.96,+2.91,+2.42)}$	$48.66^{(+4.77,+11.53,+17.04)}$
CTWS-MR	$47.39^{(-1.51,-1.42,-3.03)}$	$46.50^{(+2.06,+3.01,+2.52)}$	$47.63^{(+3.74,+10.49,+16.01)}$

Note: Numbers in parentheses denote the performance in/decrease from TF.IDF.ICS$_\delta$F, TF.IDF.ICF, and TF.ICS$_\delta$F respectively

TABLE 9
PERFORMANCE COMPARISON OF CTWS WITH EFFICIENT WEIHTING APPROACHES USING F_1^μ IN THE RCV1-V2



Weighting	Classifier		
Weighting	Centroid	NB	SVM
TF.IDF.ICS$_\delta$F	79.57	82.07	84.79
TF.IDF.ICF	80.16	81.11	80.01
TF.ICS$_\delta$F	79.81	81.72	76.27
CTWS-Sum.	$74.77^{(-4.81,-5.39,-5.05)}$	$84.41^{(+2.34,+3.30,+2.69)}$	$87.42^{(+2.63,+7.41,+11.15)}$
CTWS-Avg.	$76.21^{(-3.37,-3.95,-3.60)}$	$84.50^{(+2.43,+3.39,+2.78)}$	$86.44^{(+1.65,+6.43,+10.17)}$
CTWS-FFNN	$77.98^{(-1.60,-2.18,-1.84)}$	$84.03^{(+1.96,+2.92,+2.31)}$	$86.32^{(+1.53,+6.30,+10.05)}$
CTWS-GA	$77.79^{(-1.78,-2.37,-2.02)}$	$84.02^{(+1.95,+2.91,+2.30)}$	$86.40^{(+1.61,+6.39,+10.14)}$
CTWS-MR	$77.78^{(-1.79,-2.37,-2.03)}$	$84.12^{(+2.05,+3.01,+2.40)}$	$86.20^{(+1.41,+6.19,+9.93)}$

Note: Numbers in parentheses denote the performance in/decrease from TF.IDF.ICS$_\delta$F, TF.IDF.ICF, and TF.ICS$_\delta$F respectively

weighting methods. The results showed that for two different datasets in terms of NB and centroid classifier the distinct categorical performances are saturated for any term weighting methods. Not a single weighting method was consistently outperformed over other approaches. Some cases information-element-based or TF.IDF approaches were outperformed than class-indexing-based method.

Sohrab, Fattah, and Ren [27] discussed different text features, including sentence relative length, sentence inclusion of numerical data, sentence inclusion of named entity, sentence resemblance to the title, sentence centrality, and sentence position to enhance automatic text summarization. In this approach, first judge the effect of individual feature parameter score with different compression ratio on summarization performance. Therefore, the sum of all normalized feature parameter is constructed to address summarization task. The experiment's results showed that the sum of all normalized feature parameter approach outperforms the individual feature parameter.

Fattah and Ren [3] investigated on different models, including GA, MR, FFNN, GMM, and PNN to combine with the sum of all normalized feature parameter. The experimental results showed that the results of different models with the sum of all normalized feature parameters are promising to enhance automatic text summarization.

7 CONCLUSION

In this work, we investigated the effectiveness of proposed combined-term-weighting-scheme (CTWS) including CTWS-Sum, CTWS-Avg., CTWS-FFNN, CTWS-GA, and CTWS-MR approaches with other different document-indexing-, class-indexing- and information-element-based weighing approaches using a centroid, Naïve Bayes, and SVM classifiers applied on the Reuters-21578, 20Newsgroups, and RCV1-v2 / LYRL2004 datasets as benchmarks collection.

After analysing the result, four conclusions seem warranted. First, several traditional weighting approaches like TF.CC, TF.MI, TF.OR, TF.PB, and TF.RF are performing low even in categorical or in overall performances. At this point, the CTWS is very effective to assist to enrich in categorical and overall performances.

Second, the models GA, FFNN, MR are very effective to generate global weight for the combinational process. CTWS-FFNN, CTWS-GA, and CTWS-MR are very effective in SVM and NB classifiers. In the 20Newsgroup dataset, the CTWS-MR approach is outperformed in NB classifier.

Third, this study results indicate that the CTWS including, CTWS-Sum, CTWS-Avg., CTWS-FFNN, CTWS-GA, and CTWS-MR all are performing more than 80%, 91%, and 90% for the Reuters-21578, 78%, 96%, and 93% for the 20Newsgroups, and 74%, 84%, and 86% over the centroid, NB, and SVM classifiers respectively. These weighting schemes have enhanced ATC.

Future work possible ideas may be conducting experiment for very large scale multi-labeled hierarchical text classification. It may be interesting to study CTWS behavior for large scale dataset where certain corpus has thousands of categories and for a certain document; one or more categories are assigned in order to address multi-labeled hierarchical classification.

ACKNOWLEDGMENT

The paper is based on results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

REFERENCES

[1] Chen J., Huang H., Tian S., and Qua Y. (2009). Feature selection for text classification with Naïve Bayes. Expert Systems with Applications, 36(3 / 1), 5432-5435.

[2] Debole F., and Sebastiani F. (2003). Supervised term weighting for automated text categorization. In SAS-2003: Proceedings of the 18th annual ACM Symposium on Applied Computing (pp. 784-788).

[3] Fattah M. A., and Ren F. (2011). GA, MR, FFNN, PNN, GMM based models for automatic text summarization. Computer Speech and Language, 23(1), 126-144.

[4] Flora S., and Agus T. (2011). Experiments in term weighting for novelty mining. Expert Systems with Applications, 38(11), 14094-14101.

[5] Fuhr N., and Buckley C. (1991). Probabilistic learning approach for document indexing. ACM Transactions on Information Systems, 9(3), 223-248.

[6] Guo Y., Shao Z., and Hua N. (2010). Automatic text categorization based on content analysis with cognitive situation models. Information Sciences, 180, 613-630.

[7] Han E. H., and Karypis G. (2000). Centroid-based document classification: analysis and experimental results, In PKDD-2000: Proceedings of the 4th European conference on Principles of Data Mining and Knowledge Discovery (pp. 424-431).

[8] Joachims T. (2001). A statistical learning model of text classification

for support vector machines. In SIGIR-2001: Proceedings of the 24th annual international ACM SIGIR conference on Research and Development in Information Retrieval (pp. 128-136).

[9] Kang B., and Lee S. (2005). Document indexing: A concept-based approach to term weight estimation. Information Processing and Management, 41(5), 1065-1080.

[10] Kansheng S., Jie H. Hai-tao L., Nai-tong Z. and Wen-tao S. (2011). Efficient text classification method based on improved term reduction and term weighting. The Journal of China Universities of Posts and Telecommunications, 18(1), 131-135.

[11] Ko Y., and Seo J. (2009). Text classification from unlabeled documents with bootstrapping and feature projection techniques. Information Processing and Management, 45(1), 70-83.

[12] Lan M., Tan C. L., Su J., and Lu Y. (2009). Supervised and traditional term weighting methods for automatic text categorization. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31(4), 721-735.

[13] Lee C., and Lee G. G. (2006). Information gain and divergence-based feature selection for machine learning-based text categorization. Information Processing & Management, 42(1), 155-165.

[14] Lewis D. D., and Ringuette M. (1994). A Comparison of two learning algorithms for text categorization. In Proceedings of the 3rd annual symposium on Document Analysis and Information Retrieval (pp. 81-93).

[15] Lewis D. D., Schapire R.E., Callan J.P., and Papka R. (1996). Training algorithms for linear text classifiers. In SIGIR-96: Proceedings of the 19th annual international ACM SIGIR conference on research and development in information retrieval (pp. 298-30).

[16] Lewis D. D., Yang Y., Rose T., and Li F. (2004). RCV1: A new Benchmark Collection for text categorization Research. Journal of Machine Learning Research, 5, 361-397.

[17] Liu Y., Loh H., and Sun A. (2009). Imbalanced text classification: A term weighting approach. Expert Systems with Applications, 36(1), 690-701.

[18] Luo Q., Chen E., and Xiong H. (2011). A semantic term weighting scheme for text classification. Expert Systems with Applications, 38(10), 12708-12716.

[19] Maldonado S., Weber R. E, Basak J. (2011). Simultaneous feature selection and classification using kernel-penalized support vector machines. Information Sciences, 181, 115-128.

[20] Ogura H., Amano H., and Kondo M. (2011). Comparison of metrics for feature selection in imbalanced text classification. Expert Systems with Applications, 38(5), 4978-4989.

[21] Ren F., and Sohrab M. G. (2013). Class-indexing-based term weighting for automatic text classification. Information Sciences, 236, 109-125.

[22] Salton G. (1975). A theory of indexing. Bristol, UK.

[23] Salton G., and Buckley C. (1988). Term-weighting approaches in automatic text retrieval. Information Processing and Management, 24(5), 513-523.

[24] Salton G., Wong A., and Yang C. S. (1975). A Vector Space Model for Automatic Indexing. Association of Computing Machines, 18(11), 613-620.

[25] Salton G., Yang C. S., and Yu C. T. (1973). Contribution to the theory of indexing. Proceeding IFIP Congress 74, Stockholm, American Elsevier, New York.

[26] Sebastiani F. (2002). Machine learning in automated text categorization. ACM Computing Surveys, 34(1), 1-47.

[27] Sohrab M. G., Fattah M. A., and Ren F. (2008). The Best Feature Parameter and HMM for Text Summarization, Research in Computing Science, 38, 152-161.

[28] Sohrab M. G., and Ren F. (2012). Class-Indexing: The effectiveness of class-space-density in high and low-dimensional vector space for text classification. In CCIS-2012: Proceedings of 2nd international conference of Cloud Computing and Intelligence Systems (pp. 2034-2042).

[29] Sparck K. J. (1972). A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation, 28(1), 11-21.

[30] Sparck K. J. (2008). Index term weighting, Information Storage and Retrieval, 9, 619-633.

[31] Tan S. (2008). An improved centroid classifier for text categorization, Expert Systems with Applications, 35(1/2), 279-285.

[32] Theeramnukkong T., and Lertnattee V. (2004). Effect of term distributions on centroid-based text categorization. Information Sciences 158, 89-115.

[33] Tzeras K., and Hartmann S. (1993). Automatic indexing based on Bayesian inference networks. In SIGIR-93: Proceedings of the 16th annual international ACM SIGIR conference on research and development in information retrieval (pp. 22-34).

[34] Wan C. H., Lee L. H., Rajkumar R., Isa D. (2012). A hybrid text classification approach with low dependency on parameter by integrating K-nearest neighbour and support vector machine. Expert Systems with Applications, 39, 11880-11888.

[35] Wu X., and Kumar V. et. al, (2008). Top 10 algorithms in data mining. Knowledge Information Systems, 14, 1-37.

[36] Xia R., Zong C., and Li S. (2011). Ensemble of feature sets and classification algorithms for sentiment classification. Information Sciences, 181, 1138-1152.

[37] Yang Y. (1999). An evaluation of statistical approaches to text categorization. Journal of Information Retrieval, 1(1/2), 67-88.

[38] Zhang W., and Gao F. (2011). An Improvement to Naive Bayes for text classification. Procedia Engineering, 15, 2160-2164.