Centroid Training to Achieve Effective Text Classification

Libiao Zhang*, Yuefeng Li†, Yue Xu‡, Dian Tjondronegoro§, Chao Sun¶
*†‡¶School of Electrical Engineering and Computer Science, §School of Information System,
*†‡§¶Faculty of Science and Engineering, Queensland University of Technology, Brisbane, QLD 4001, Australia
Email: *L39.zhang@student.qut.edu.au, †y2.li@qut.edu.au, ‡Yue.xu@qut.edu.au, §Dian@qut.edu.au, ¶elliot.s@bluhex.com

***Abstract*—Traditional text classification technology based on machine learning and data mining techniques has made a big progress. However, it is still a big problem on how to draw an exact decision boundary between relevant and irrelevant objects in binary classification due to much uncertainty produced in the process of the traditional algorithms. The proposed model CTTC (Centroid Training for Text Classification) aims to build an uncertainty boundary to absorb as many indeterminate objects as possible so as to elevate the certainty of the relevant and irrelevant groups through the centroid clustering and training process. The clustering starts from the two training subsets labelled as relevant or irrelevant respectively to create two principal centroid vectors by which all the training samples are further separated into three groups: POS, NEG and BND, with all the indeterminate objects absorbed into the uncertain decision boundary BND. Two pairs of centroid vectors are proposed to be trained and optimized through the subsequent iterative multi-learning process, all of which are proposed to collaboratively help predict the polarities of the incoming objects thereafter. For the assessment of the proposed model, F_1 and $Accuracy$ have been chosen as the key evaluation measures. We stress the F_1 measure because it can display the overall performance improvement of the final classifier better than $Accuracy$. A large number of experiments have been completed using the proposed model on the Reuters Corpus Volume 1 (RCV1) which is important standard dataset in the field. The experiment results show that the proposed model has significantly improved the binary text classification performance in both F_1 and $Accuracy$ compared with three other influential baseline models.**

***Index Terms*—Text classification, Centroid vector, Centroid optimization, Multi-learning, clustering**

I. Introduction

Text classification is the process of classifying an incoming stream of textual documents into predefined categories through the classifiers learned from the training samples, labelled or unlabelled. Many traditional models for text classification have been put forward by field researchers in different ways and levels, such as k-Nearest Neighbors (kNN) [1], Support Vector Machines (SVM) [2], Naive Bayes [3], Rocchio Similarity [4] and rule-based methods.

Although there has been a continuous improvement of text classification technology, we found that too much noises are produced by traditional ways to cause the uncertainty of text classification. Feature is the essential element to represent textual documents [5], but unsuitable feature number, inferior feature quality or imperfect feature weighting algorithms will probably bring about much noise which may arouse reduction of text classification performance. The knowledge acquired by current machine learning and data mining techniques inevitably contains much noise. Under such situation, we found that it is difficult for a clear decision boundary to be drawn by a traditional binary text classifier. Therefore, with the features selected and weighted by specific algorithms, most document sets can only be grouped into three rather than two groups by a traditional binary text classifier because the knowledge learned from the training samples cannot help classify the documents at one stroke including the training set itself [6].

The proposed model CTTC (Centroid Training for Text Classification) addresses the above problems. It tries to set up an uncertain decision boundary through partitioning the training samples into three regions and iteratively improve the certainty of the relevant and irrelevant groups, and absorb and resolve the uncertain objects in the third group so as to make the knowledge of document relevancy more unambiguous. It starts from calculation of two main centroid vectors C_P and C_N by clustering the relevant and irrelevant training subsets, and further regroups the training samples into three regions using the two centroid vectors gained, with all the indeterminate objects collected into a boundary region BND, the objects with most relevant possibility to the topic stored into the POS region, and those with most irrelevant possibility to the topic collected into the NEG region. Through above iteratively training process, it filters as many uncertain objects gradually and save them into BND region to make the other two regions POS and NEG of greater certainty. During the training process the two main centroid vectors C_P and C_N and two other auxiliary centroid vectors B_P and B_N formed from the BND region are expected to be trained and optimized successively in the multi-learning process to reach the optimal condition. Development of vector space theory make it possible to represent and operate the documents in the type of vectors. Although Rocchio classification also involves the operation of centroids, the centroids have not been optimized through multi-learning process. The evaluation of the text classification is another key issue that the paper addresses. We have chosen F_1 and $Accuracy$ as the key evaluation measures. The F_1 measure is emphasized and used for the performance assessment of the proposed model. The proposed model aims to pursue substantial improvement on F_1 with the $Accuracy$ guaranteed not to be reduced. The calculation of F_1 depends on two factors, the $Precision$ and $Recall$ which can together reflect both the real situation of relevant and irrelevant ratio

and their improvement degrees in the testing process.

A large number of experiments have been completed based on the proposed model using the standard textual dataset RCV1 [7], and the comparison analysis has been completed between the proposed model and the baseline models. The experiment results show that our proposed model has significantly improved the text classification performance in F_1 and $Accuracy$.

In this paper, section II introduces the related technologies and known algorithms in text classification area. The construction process of the proposed CTTC model is described detailedly in section III. The knowledge optimization approach through centroid vector training is presented in section IV. The evaluation metrics and the related issues are introduced in section V. Section VI finalizes the whole paper with the conclusion.

II. Related work

The research in the similar field and the often used technologies related to text classification will be reviewed in this section as they have certain relations with the topic and contribute to the discussion of the key issues in the paper.

Document representation is one of the most important steps for text classification, in which related documents are represented by single or multiple informative features to ease the automatic operation of the documents in the subsequent steps. Feature selection plays a significant role in document representation for the purpose of text classification because a document vector is composed of a set of weighted features, and the feature number and feature quality affect the performance of text classification. Feature selection aims to help build up the documents' vectors by selecting a subset of key features for describing all the related documents, and remove irrespective or noise features according to corpus statistics to increase the scalability, efficiency and accuracy of a text classifier. A number of popular term weighting functions have been developed and used such as $tf * idf$ (term frequency and inverse document frequency), Latent Semantic Analysis (LSA), Probabilistic LSA (pLSA), Latent Dirichlet Allocation (LDA) [8], semantic structure, belief revision method , relevance frequency (RF), pattern deploying method [9], BM25 [10].

BM25 is a well-known probabilistic scoring function for feature selection. From the experiments completed on the proposed model in the paper, it is found that the BM25 performs better than TF*IDF. We use the following scoring function to estimate the weight of term t extracted from relevant documents as follows:

$$W(t) = \frac{tf \cdot (k_1 + 1)}{k_1 \cdot ((1-b) + b\frac{DL}{AVDL}) + tf} \cdot \log \frac{\frac{(r+0.5)}{(n-r+0.5)}}{\frac{(R-r+0.5)}{(N-n-R+r+0.5)}} \quad (1)$$

where N is the total number of training documents; R is the number of relevant documents; n is the number of documents which contain term t; r is the number of relevant documents which contain term t; tf is the term frequency; DL and *AVDL* are the document length and average document length, respectively; and k_1 and b are the experimental parameters. We also use the BM25 with the parameters tuned in [11] (i.e.,$k_1 = 1.2$ and $b = 0.75$).

The text classification algorithm can be categorized in three ways including unsupervised, supervised and semi-supervised methods. In recent years, many classification algorithms have been invented for classifying electronic documents. It mainly addresses the supervised classification methods such as Naive Bayes, Support Vector Machines (SVM) and Rocchio. SVM can be applied to classify both linear and non-linear data, but its algorithm has relatively low efficiency [12]. Bayesian classifiers can be regarded as probabilistic models and it uses Bayes law to calculate the reverse probability of the model parameters for given functions. It assume that all the features in a certain class are irrelevant to each other and one feature does not affect other features [13]. Rocchio algorithm of classification is a vector space model for text classification presented by Rocchio in 1971. This method is easy to implement as well as efficient in computation, but it has a potential disadvantage that the performance will be reduced when the documents belonging to a category naturally form separate clusters. [14].

III. Construction of the proposed model

Suppose that we have a traditional binary text classifier CF, we try to describe one of the traditional text classification ideas firstly as follows by mathematical methods, where a training document set D in which all the document objects are labelled as either relevant or irrelevant, and assumed to be stored in subset D^+ and D^- separately. F is used to keep a set of term features, the key words extracted from D.

$$D = D^+ \cup D^-, \quad F = \{f_1, f_2, \ldots, f_n\}$$

Then, for each document $d \in D$, it can be represented as a vector $\vec{d}$ by the term weights after all the terms in F are assigned some sort of weights by different feature weighting algorithms such as TF*IDF or BM25.

$$\vec{d} = (TW(f_1), \quad TW(f_2), \ldots, TW(f_n))$$

Through different ways to calculate the relevancy of the training documents to the specified topic or query, we can get the ranked list of the documents so as to categorise them into different groups according to their ranked positions in the list that express the relevancy degree.

Based on above consideration, the classifier CF will partition the document set D into relevant (R) and non-relevant (NR) parts if given certain threshold as a watershed between the two parts, regardless of whether such boundary can directly be obtained or not:

$$CF : D \longrightarrow \{ \text{ R, NR } \}$$

However, much research completed in the similar fields shows that it is hard to label all the documents in a document set with relevant or irrelevant polarity at one stroke by any

traditional text classifier, even for classifying the training documents when applied the knowledge learned from themselves [6]. In such case, it is inappropriate to assume that the binary text classification can be reached directly with a high precision by means of traditional text classification way.

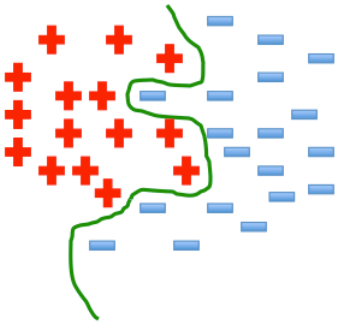

Fig. 1. Irregular nonlinear boundary

As shown in Figure 1, it is usually hard to find a clear boundary, which can be accurately described by means of mathematics, between relevant and irrelevant groups of documents. The "+" denotes the relevant documents and the "-" denotes the irrelevant ones in the figure, but it is almost impossible to describe the curve with any exact math equation as there are always many strange cases containing some unexpected or irregular data points. Even exists, it is not guaranteed to be applied to the prediction of the incoming testing documents because of the different situation in the testing document set. Therefore, we can only find an uncertain boundary instead by any traditional text classifier as shown in Figure 2, the proposed approach focuses on setting up the uncertain decision boundary to absorb the uncertainties through centroid training to achieve more reasonable classifier to indirectly achieve the final purpose.

Inspired by center-based clustering, in the training process, the training samples are proposed to be partitioned into three rather than two different regions including POS, NEG and BND, and the two main centroid vectors C_P and C_N are firstly generated from the training document sets D^+ and D^- which are replaced with POS and NEG in subsequent iterations, and B_P and B_N are formed from BND simultaneously and also updated iteratively with the advance of the training process.

Next, the vector $\vec{u}$ of each incoming document $u \in U$ is compared to the pair of centroid vectors C_P and C_N with its euclidean distance computed by the selected algorithms for vectors so as to be accurately predicted through the specified decision making rules described in Section IV-C.

IV. Centroid training and optimization

A. Vector space of textual document set

The document representation is one of the most important steps to realize a text classifier, which is to transform the

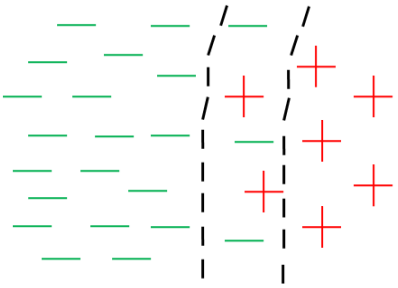

Fig. 2. Illustration of uncertain boundary

documents into the specified types of data that are applicable for the classification algorithms. In the paper, a document is proposed to be represented from full text version into a document vector and it involves feature selection, feature weighting and vector computation.

Feature selection is a key step of text classification as the feature number and feature quality affect the performance of text classification. Experiments show that the feature selection is not the more the better because too many features may bring more confusion, but once too few, it may also cause performance degradation due to information loss of the represented documents. The features are selected based on their weights and the feature weights are proportional to their relevancy with the topic which we are concerned about. In the project, the number of the keywords, i.e., the extracted features, is set to 150 based on the previous research experience. In the process, TF*IDF is firstly used as the feature weighting method for feature selection and the training and testing of text classification, and BM25 is also mainly considered. After repeated testing and comparative study, it proves that BM25 has extremely improved the performance of the proposed model.

Once the feature weighting method has been determined, each feature will be assigned a score so that the keywords can be selected by the specified algorithms based on their weights to help compute vector for each related document and build up the vector space of the corresponding document set for subsequent training and testing.

B. Centroid clustering and training

According to CTTC, it firstly generates the two basic centroid vectors by clustering the two given labelled training subsets and use them to further divide the whole set of training documents into three regions so as to trigger the subsequent iterative process of centroid training and optimization in the training stage.

Next, the detailed process of centroid training will be further described as follows. The two centroid vectors C_P and C_N are proposed to be generated by clustering the relevant and irrelevant training subsets D^+ and D^- of D, and the training samples are further regrouped into three regions using the two centroid vectors gained and then the iterative training process for the two pairs of optimal centroid vectors starts. Specifically,

two matrixs are set up corresponding with the two subsets D^+ and D^-, which are formed by using documents as rows and the keywords as columns. Every line of the matrix is filled with the BM25 feature values of all the keywords that occur in corresponding document of the related subset, thus one line of the matrix refers to the vector of the document. Two sorts of matrix corresponding with training relevant and irrelevant subsets are all built up by the same methods, and used for calculation of centroid vectors. The centroid clustering is completed by calculating the average of feature weights of all the documents in the subset vertically to construct one centroid vector for the subset of documents.

Assumed that there is a general classifier CF for binary classification built as described in Section III, in order to model the uncertainty that happens in traditional text classification, after the new knowledge has been gained in the type of centroid vectors, we try to extend the classifier $CF \Longrightarrow CF'$, where CF' is able to classify the document D into positive (POS), negative (NEG) and boundary (BND) regions by comparing the distance from each document vector in D to the two main centroid vectors C_P and C_N:

$$CF'\text{: } D \longrightarrow \{\text{ POS, BND, NEG }\}$$

The three regions are defined as follows:

Definition 1. $CF(d) =$ "R" *and* $d \in D^+ \Longrightarrow CF'(d) =$ "POS"

Definition 2. $CF(d) =$ "NR" *and* $d \in D^- \Longrightarrow CF'(d) =$ "NEG"

Definition 3. $\{CF(d) =$ "NR" *and* $d \in D^+\}$ *or* $\{CF(d) =$ "R" *and* $d \in D^-\} \Longrightarrow CF'(d) =$ "BND"

Based on the above definitions, some properties about the three regions including POS for positive region, NEG for negative region, BND for Boundary region divided by CF' are deducted as follows:

Property 1. *If* $d \in POS$ *then* $d \in D^+$

Property 2. *If* $d \in NEG$ *then* $d \in D^-$

Property 3. *If* $d \in D^+$ *and* $d \in BND$ *then* $CF(d) =$ "NR"

Property 4. *If* $d \in D^-$ *and* $d \in BND$ *then* $CF(d) =$ "R"

The four centroid vectors can be generated respectively from POS region, NEG region and the two parts of BND region through the three region evolution progressively. C_P is the centroid vector for the positive region and C_N is the centroid vector for the negative region, B_P is the centroid vector from the part of BND region which are in D^+, B_N is the centroid vector from the part of BND which are in D^-. Between the C_P and C_N centroid vectors, there is a central line in the boundary region. Figure 3 gives the centroid production and optimization process. Theorem 1 indicates the relations between the four centroid vectors. Figure 4 gives schematic representation of the four centroid vectors.

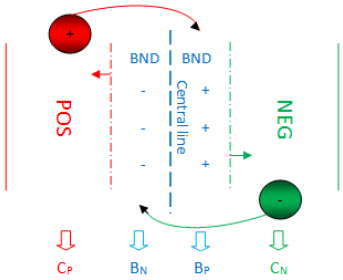

Fig. 3. Three-region evolution and centroid optimization process

Theorem 1. *Let* B^+ *and* B^- *be two subset of documents, where* $B^+ = \{d \in BND \cap d \in D^+\}$, $B^- = \{d \in BND \cap d \in D^-\}$. *If* B^+ *and* B^- *have been gained by classifier* CF', *then all the documents in* B^+ *must be below the central line, whereas all the documents in* B^- *must be above the central line.*

Proof: **If** there is a document $d \in B^+$, **then** according to the definition of B^+, it should be $d \in D^+$, **suppose** it is above the central line, it must be $d \in POS$, which is against the property of B^+: $d \in BND$, **therefore** d is below the central line. **In the same way,** any document $d \in B^-$ must be above the central line, as shown in Figure 4. ■

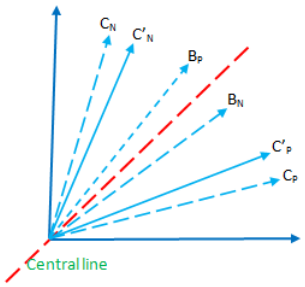

Fig. 4. Four kinds of centroid vectors and the central line

The optimization of the centroid vectors is proposed to be completed through an iteration process. With the training process progressing, the boundary region will gradually absorb as many uncertain training documents as possible so that the two main centroid vectors are moving away from each other accordingly until the distance between them no longer changes, as shown in Figure 3. Figure 4 is an example result after applied the classifier CF' on training document set from which we can clearly see that the two main centroid vectors C_P and C_N have been changed to C'_P and C'_N. As is known from the experiments that the larger the gap between the centroid vectors C_P from POS and and C_N from NEG is, the easier it would be made to separate the training or incoming documents apart into binary categories.

C. Relevancy prediction

Whether a testing document is relevant or not depends on the Euclidean distance between its vector and the two centroid

vectors. However, in the training stage, the documents around the two centroid vectors are not always categorized into right groups that the two centroid vectors represent, especially those nearer to the central line. Therefore, it will be more reasonable if we add some other conditions to help make more righteous decision for document relevancy prediction.

Inspired by the theory of Standard Deviation which is commonly used to measure the degree of confidence in statistical conclusions, a new method has been invented as the supplementary strategy to improve the performance of incoming document relevancy prediction. In practice, if dis($\vec{u}$,$Centroid$) $< meanSquareDis$, then there will be a high possibility that document u is relevant, where $\vec{u}$ is the vector of document $u \in U$, $Centroid$ is one of the main centroid vectors from POS or NEG, $meanSquareDis$ is the average squared distance from $\vec{u}$ to $Centroid$, as shown in Equation 2 and 3, in which N is the number of the training document set POS or NEG, and F is the selected feature set and k is an experiment parameter that influences the performance and needs to be adjusted in the experiments. The specific decision making methods that assist the relevancy prediction of the testing documents are described in Algorithm 1.

Let $\vec{u_i} = (w_1, w_2, ..., w_{|F|})$, $Centroid = (w'_1, w'_2, ..., w'_{|F|})$, then we get:

$$dis^2(\vec{u_i}, Centroid) = \sum_{j=1}^{|F|}(w_j - w'_j)^2 \tag{2}$$

$$meanSquareDis = k * \frac{\sum_{i=1}^{N} dis^2(\vec{u_i}, Centroid)}{N} \tag{3}$$

To predict the polarity of each incoming document for testing, we try to follow six scenarios that cover all typical spatial location of the incoming document vectors for relevancy analysis and decision-making of relevancy prediction, as illustrated in Figure 5. Specifically, the red mark "+" and blue mark "-" represent the positive centroid vector C_P and negative centroid vector C_N respectively; The dotted line refers to the central line that locates in the middle of the positive and negative centroid vectors and symmetrically separates them so as to separate the whole document space; The u_1, u_2, u_3, u_4, u_5 and u_6 sequentially denote the six types of incoming document vectors in different six situations corresponding with different orientation and distance, three of which locate at the left side of the central line, and the other three locate at the right side.

For document u_1:

As seen from Figure 5, if $dis(u_1, C_P) < dis(u_1, C_N)$, u_1 is close to positive centroid and far away from the negative centroid. Further, we check which side of the central line it locates at by testing if $dis(u_1, C_N)^2 > dis(u_1, C_P)^2 + dis(C_P, C_N)^2)$. In this case, document u_1 is predicted as relevant.

For document u_2:

Refer to Figure 5, if document u_2 locates between the centroid vectors C_P and C_N but around the centroid vector B_N, specifically between B_N and the central line. Under such circumstance, we can also know that $dis(u_2, C_P) < dis(u_2, C_N)$ and it locates at the right side of C_P, but the $dis(u_2, C_P)$ is greater than $meanSquareDis$. So the document u_2 is predicted as irrelevant.

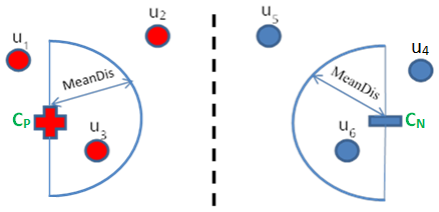

Fig. 5. Six scenarios for polarity prediction of incoming documents

For document u_3:

Document u_3 is similar to u_2, but it actually locates between the positive centroid vector C_P and B_N, and the distance $dis(u_3, C_P)$ is not greater than $meanSquareDis$. In this case, it has a greater chance that u_3 is relevant. Therefore, u_3 is predicted as relevant.

For document u_4:

The scenario of document u_4 is quite similar with u_1, however, it is on the right side of the negative centroid, showing that u_4 is close to negative centroid C_N and far away from the positive centroid C_P ($dis(u_4, C_N) < dis(u_4, C_P)$). At the meantime, it shows that u_4, C_N and C_P form an obtuse triangle based on 2-D perspective. Therefore, it is predicted as an irrelevant document.

For document u_5:

The scenario of document u_5 is quite similar with u_2, so the similar decision making can also be applied for it. We can similarly calculate the average distance from all the documents in the NEG region to the negative centroid vector C_N based on Equation 3, and the distance from document u_5 to the C_N is bigger than meanSquareDis so that it is predicted as relevant.

For document u_6:

The scenario of document u_6 is quite similar with u_3, but the document u_6 locates between the negative centroid vector C_N instead of C_P, and B_P, and the distance $dis(u_6, C_P)$ is not greater than meanSquareDis. Therefore, u_6 is predicted as irrelevant.

Algorithm 1 describes the decision rules and decision making for the testing stage of text classification by the proposed approach. If the document is relevant then $y = 1$, otherwise $y = -1$. According to the experiment results, the best result of the proposed model has been gained when the parameter $k = 2.1$.

Algorithm 1 Decision making for polarity prediction of testing documents

Input:

C_P, C_N, POS, NEG

$U = \{u(x, y) \mid 1 \leq x \leq n\}$ (A set of incoming document without label for testing)

Part 1: Processing the first three scenarios of incoming documents

Output:

$U = \{u(x, y) \mid 1 \leq x \leq n, y \in \{-1, 1\}\}$(The set of testing documents labelled)

Initiate:

$meanSquareDis = 0$

Procedure:

Calculate the $meanSquareDis$ and based on the input POS and C_P

for i=1 to n **do**

  if $(dis(u(i, y), C_P)) \leqslant dis(u(i, y), C_N)$ **then**

    if $(dis(u(i, y), C_N)^2 > dis(u(i, y), C_P)^2 + dis(C_P, C_N)^2)$ **then**

      $y = 1$

    else

      if $(dis(u(i, y), C_P) \leqslant meanSquareDis)$ **then**

        $y = 1$

      else

        $y = -1$

      end if

    end if

  else

    $y = -1$

  end if

end for

Part 2: Repeat the similar sequential operations with Part 1 to process the last three scenarios of incoming documents, with **meanSquareDis calculation based on NEG and** C_N.

V. Experiments and evaluations

A. Data collection

The Reuters Corpus Volume 1 (RCV1) consists of 100 topics of semi-structured document set; The number of documents contained in each topic of RCV1 is different, and each topic is a separate unit which is composed of two parts, training set and testing set with relevance judgements in which all the documents has been labelled attribute of relevancy with the topic. RCV1 is totally comprised of 806,791 documents that cover a very large spectrum of topics, all of which are news stories in English wrote by Reuters journalists between August 20, 1996 and August 19, 1997 [7]. All of the documents in RCV1 are formatted as XML pages. The first 50 topics were developed by National Institute of Standards and Technology (NIST) and the relevance attribute of each document in it has been labelled by the personnel of NIST. The last 50 topics have been completed manually through fusion of different categories in Reuters and the relevance attributes of those documents have been labelled by the machine learning process rather than by humans. Therefore, the first 50 topics are more reliable and the quality of the latter 50 topics is relatively low [15].

The researchers in IF/IR or the similar fields often conduct their experiments on RCV1 data sets to test the effectiveness or efficiency of the algorithms or the research plans designed by them. Therefore, each topic of RCV1 is devised into two different sets shouldering with different tasks: training sets and testing sets. The training sets consists of a total of 5,127 news stories, mainly provide necessary training seeds for machine learning purpose, while the testing sets contains the 37,556 news stories, are used as tested objects. Both of these two sets consist of relevant and irrelevant documents labelled clearly for the purpose of convenient utilisation in the test.

B. Baseline models and evaluation metrics

In order to make a comprehensive evaluation, we have chosen three types of classifiers with different algorithms as the baseline models. Support vector machine (SVM) is a statistical method that can be used to find a hyperplane that best separates two classes [16]. SVM represents the decision boundary using a subset of training data, known as support vectors and is one of state of the art of text classifier. Naive Bayes is a simple probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions. Naive Bayes is very popular in binary classification problem [19]. The Rocchio algorithm [4] has been widely adopted in the area of text categorization [20]. It can be used to build the profile for representing the concept of a topic which consists of a set of relevant and irrelevant documents.

Precision (p), Recall(r) are two basic parameters for evaluation of the proposed model. In the paper, the effectiveness of text classification is measured by two key measures: F_1 and $Accuracy$ (Acc). F_1 is stressed as it is one of the most important metrics of comprehensive assessment [2].

$$F_1 = \frac{2PR}{P+R}, \quad F_1^M = \frac{\sum_{i=1}^{|\mathcal{C}|} F_{1,i}}{|\mathcal{C}|}$$

where F_1^M is the macro average of F_1 for all the tested topics, and $F_{1,i}$ is the F_1 of topic i. For the calculation of $Accuracy$,

$$Acc = \frac{TP+TN}{TP+FP+TN+FN}, \quad Acc^M = \frac{\sum_{i=1}^{|\mathcal{C}|} Acc_i}{|\mathcal{C}|}$$

where Acc^M is the macro average of $Accuracy$ for all the the tested topics, and Acc_i is the $Accuracy$ of topic i.

C. Experiment results

The comparison between the proposed CTTC model and the baseline models has been completed mainly by the two measures of F_1 and $Accuracy$. The best result by the proposed model CTTC is compared with three influential baseline models as shown in Table I based on RCV1 Dataset. In Table I, we found that the proposed model has got an average increase of 9.28% for $Accuracy$ and 65.73% for F_1 compared with the other three baseline models. The $Accuracy$ value got by the

proposed model exceeds SVM model which has the highest $Accuracy$ value in all the baseline models, and the F_1 value has also been extremely improved by the proposed model at 116.71% compared with SVM model. The improvement situation of the proposed model can be seen clearly from Figure 6.

TABLE I
THE RESULTS OF EXPERIMENTS ON RCV1

No	Models	F_1	Accuracy
1	SVM	19.39%	85.45%
2	NaiveBayes	26.87%	81.62%
3	Rocchio	33.86%	70.13%
4	CTTC-SD-BM25-TF	**42.02%**	**85.79%**
5	Average %chg	**65.73%**	**9.28%**

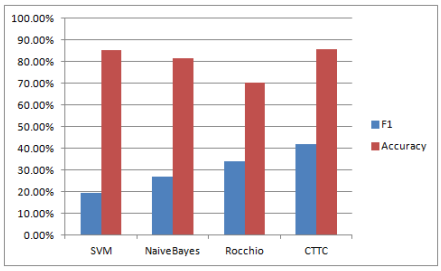

Fig. 6. Comparison between CTTC and baseline models

From Table I, it indicates that the proposed model has the highest score in both F_1 and $Accuracy$ on standard RCV1 dataset, especially in F_1 that best reflects the real situation of text classification performance. Therefore, the proposed approach has gained the best performance on RCV1 compared with the three influential baseline models.

VI. CONCLUSION

The paper proposed an innovative model CTTC (Centroid Training for Text Classification) in which the approximation approach by training two pairs of centroid vectors to text classification has been put forward, and it has accomplished the following three major tasks.

It presented a strategy of progressive realization for text classification by dividing the training documents into three regions which are positive, negative and boundary regions in order to reduce the impact of the uncertainties of text classification. It invented an innovative method for knowledge refinement by optimizing the centroid vectors to set up the binary text classifier and significantly improved the relevancy prediction accuracy. It developed a more reasonable assessment system based on F_1 and $Accuracy$ to evaluate the effectiveness due to the unique framework of the proposed model, in which the F_1 measure is emphasized as it can better reflect the overall performance improvement of the final classifier than $Accuracy$. The proposed model has been evaluated based on the experiment results using the Reuters Corpus Volume 1 (RCV1) which is important standard dataset in the field.

The experiment results show that the proposed model has significantly improved the binary text classification performance in both F_1 and $Accuracy$ compared with three other influential baseline models. It has been concluded that the proposed CTTC model is quite effective and promising.

ACKNOWLEDGMENTS

This paper was partially supported by Grant DP140103157 from the Australian Research Council (ARC Discovery Project).

REFERENCES

[1] T. Cover and P. Hart, "Nearest neighbor pattern classification," *Information Theory, IEEE Transactions on*, vol. 13, no. 1, pp. 21–27, 1967.
[2] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1–47, 2002.
[3] G. H. John and P. Langley, "Estimating continuous distributions in bayesian classifiers," in *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1995, pp. 338–345.
[4] J. Rocchio, "Relevance feedback in information retrieval," *SMART Retrieval System Experimens in Automatic Document Processing*, pp. 313–323, 1971.
[5] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *ICML*, 1997, pp. 412–420.
[6] L. Zhang, Y. Li, C. Sun, and W. Nadee, "Rough set based approach to text classification," in *2013 WI/IAT and IEEE/WIC/ACM International Joint Conferences*, vol. 3. IEEE, 2013, pp. 245–252.
[7] Y. Li, A. Algarni, and N. Zhong, "Mining positive and negative patterns for relevance feature discovery," in *Proceedings of SIGKDD*. ACM, 2010, pp. 753–762.
[8] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.
[9] S. Wu, Y. Li, and Y. Xu, "Deploying approaches for pattern refinement in text mining," in *Proceedings of ICDM'06. Sixth International Conference on Data Mining.*, 2006, pp. 1157–1161.
[10] J. S. Whissell and C. L. Clarke, "Improving document clustering using okapi bm25 feature weighting," *Information retrieval*, vol. 14, no. 5, pp. 466–487, 2011.
[11] N. Zhong, Y. Li, and S. Wu, "Effective pattern discovery for text mining," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 24, no. 1, pp. 30–44, 2012.
[12] L. Khan, M. Awad, and B. Thuraisingham, "A new intrusion detection system using support vector machines and hierarchical clustering," *The VLDB Journal—The International Journal on Very Large Data Bases*, vol. 16, no. 4, pp. 507–521, 2007.
[13] J. L. Carroll, "A bayesian decision theoretical approach to supervised learning, selective sampling, and empirical function optimization," Ph.D. dissertation, Brigham Young University, 2010.
[14] A. Zeng and Y. Huang, "A text classification algorithm based on rocchio and hierarchical clustering," in *Advanced Intelligent Computing*. Springer, 2012, pp. 432–439.
[15] D. D. Lewis, Y. Yang, T. G. Rose, and F. Li, "Rcv1: A new benchmark collection for text categorization research," *The Journal of Machine Learning Research*, vol. 5, pp. 361–397, 2004.
[16] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.
[17] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," *Machine Learning: ECML-98*, pp. 137–142, 1998.
[18] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 161–168.
[19] V. Metsis, I. Androutsopoulos, and G. Paliouras, "Spam filtering with naive bayes-which naive bayes?" in *CEAS*, 2006, pp. 27–28.
[20] C. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval*. Cambridge University Press Cambridge, 2008, vol. 1.