International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 14 (2015) pp 34889-34893

Enhanced Text Categorization Using Normalized Term Weighting and Particle Swarm Optimization Techniques

N.Naveenkumar and K.Batri,

[1]Assistant Professor,
Department of Computer Science and Engineering,
SBM College of Engineering and Technolog,
Dindigul District, thillainaveen.n@gmail.com
[2]Assistant Professor,
Department of Computer Science and Engineering
PSNA College of Engineering and Technology,
Dindigul District, krishnan.batri@gmail.com

Abstract

Text Categorization (TC) is an important technology in the field of organizing a huge number of documents. Feature selection (FS) is most important part in text categorization to issue more efficient and accurate. It is commonly used to reduce the dimensionality of text datasets with large number of relevant features which would be problematic of the computation process. This work planning to implement Particle Swarm Optimization (PSO) technique based on normalized term weighting method for improving the performance of text categorization. A method has done feature selection implicit, since the regularities of higher term weight text. The experimental results are carried out from Reuters-21578 corpora. The performance of PSO based on term weighting and feature selection algorithm has been evaluated and compared with the TF-IG and TF- X^2 methods. The classification has been done using Support Vector Machine (SVM) and enhances the text categorization.

Keywords - Feature Selection, Particle Swarm Optimization, Support Vector Machine, Text categorization, Term Weighting

1. Introduction

Text categorization (TC) or classification is the task of assigning one or more predefined set of categories for natural language text. It is an active research field in information retrieval and machine learning. Feature selection (FS) is the process of selecting a subset of features available from the data as an application learning algorithm. FS plays a significant role of research field and effective care due to growing availability of text documents in electronic forms. Generally, dealing with dimensionality reduction for feature selection is essential pre-processing method to remove noisy features [1].

In TC represented as a vector of term t in document $d=\{t_1,t_2,\ldots t_n\}$ where n is the number of term in the document $t_n \in$ term size in d. A predefined set of categories $C_i = \{C_1,C_2, \ldots , C_k\}$with label y_i is the category x_i, $y_i \in \{C_1,C_2, \ldots . , C_k\}$are train the data of $(t_1,y_1),(t_2,y_2) \ldots . (t_n,y_i)$. An important term of word w_i is different distributional feature of the term t_i contributes to category of document. In feature selection approach uses distributional feature of words via the recently introduced information of bottleneck method, which generates more efficient representation of the documents. In variant of Particle Swarm Optimization (PSO) has been known as a novel population-based meta-heuristic algorithm [2].

For each specific term in a document collection consist of distinct feature. Hence, a term value of tf-idf uses a feature to predict the collection of documents. In major issues of text classification deal with high dimensionality of feature space to increase computational time and degrade accuracy [3]. Features are vector space representation of terms such as words and phrases are extracted from relative feature of documents. The common paradigm of FS methods using corpus statistics to remove non-informative terms and some rules of combining features to reduce feature space dimensions. Generally, a large set of features or may be thousand and more, many learning algorithms are cannot be support of computation time and memory limitations. Unlike Chi-Square (x^2)-statistics are sum of similarity does not use a scoring function for each category [4]. Information as retrieved weight of each word use similarity structure among the original documents, in probabilistic distribution of the each word of term weight assigned Information Gain Ratio (IGR) [5].

To reduce the feature documents using machine learning problem to apply subsequent learning algorithms for effective performance through avoiding over fitting problem. A statistical measures tf-idf variants, document frequency, information gain, Chi-Square, mutual information gain and information gain ratios are used in the feature selection [6]. Binary PSO used select the two variants of local and global neighbourhood best feature subset choose from the population of particle [11]. Here, TF-IDF and PSO techniques are implemented through heuristic approach for text categorization. A novel method can be utilized in the term weighting to issue the motivated subset of category for nearest feature of the term to find a document for determining and distinguishing the importance of these relative terms within the subset of appropriate features.

This paper is organized as follows: Section 1 introduced the concepts. Section 2, discuss the term weighting with feature selection methods. We describe in detail about feature selection for PSO implementation in Section 3. Sections 4 explain the study of method based in the experiments. Section 5 presents the Reuters 21578 dataset used to experimental study and results, which are related to precision, recall, f-measures, accuracy, and timing analysis, for entire dataset. Finally, concludes are given in Section 6.

2. Related Work

Feature Selection is a common method for preceding term weighting for several reasons. In feature selection of a subset of terms is commonly learning on text data set used to text documents are characterized by reducing the feature vector. A weight of the term represent the Vector Space Model (VSM) of term frequency(tf) is defined as the number of times term occur in a document and inverse document frequency (idf) are number of documents in the corpus that contain the term.

Information theoretic of IG term measures obtained the prediction of presence or absence of term in document of category. Information gain of a term *t* is defined as probabilities are interpreted the even space of documents and counting the occurrence in the training set [6]. In eq. [1]

$$IG(t_i,c_k)= -\sum_{k=1}^{M} P(c_k) log P(c_k) + P(t_i) \sum_{k=1}^{M} P(\frac{c_k}{t_i}) log P(\frac{c_k}{t_i}) + P(\bar{t_i}) \sum_{k=1}^{M} P\left(\frac{c_k}{\bar{t_i}}\right) log P\left(\frac{c_k}{\bar{t_i}}\right) \quad (1)$$

The x^2 measures are common statistical test as divergence from distribution of term selection are independence of term t_i contain the category c_i. It is defined as x^2 as follows:

$$x^2(t_i,c_k)=\frac{N\times(AD-CB)^2}{(A+C)\times(B+D)\times(A+B)\times(C+D)} \quad (2)$$

In contingency table applied from the number of document occurrence in term t_i and category c_i for classified documents A, B, C and D. If x^2 value zero means t_i and c_k are independent. N is total number of documents [7].

3. Document Representation

The document of main function to represent are convert the terms which are strings to features to handle them and features are transformed from the full text version to a document vector which describes the contents of the document. The term frequency is simple choice to use the frequency of term in a document. In each document D are counts the terms tf for category C of words in each category of document fig.1.

	C_1	C_2	…….	C_n
D_1	tf(1,1)	tf(1,2)	……	tf(1,n)
D_2	tf(2,1)	tf(2,2)	……...	tf(2,n)
.	.	.		.
.	.	.		.
.	.	.		.
D_m	tf(m,1)	tf(m,2)	…….	tf(m,n)

Fig1. Document Representation

Here, a document is represented by multi-dimensional feature vector where each dimension corresponds to a weighted value of the regarding term within the document collection [8].

3.1 Term Frequency -Inverse Document Frequency

In TFIDF is the famous weight method which has defined a term frequency(tf) are number of times that term occurs in the document and inverse document frequency(idf) concerns the number of term occurrences across a collection of text. In problem of this weighting method that number of the documents becomes large, when the terms that have nearest frequency have almost equal weight, which makes the learning task more difficult [8].

A maximum normalization of term frequency (tf_{ik}) and inverse document frequency $idf(t_k,d_i)$ used to calculate the tf-idf weight (w_{ki}) as given bellows[2]:

$$w_{ki} = \frac{tf\times idf(t_k,d_i)}{maxtf} \quad (3)$$

Where w_{ki} is the weight of the term frequency k in document i.

3.2 Particle Swarm Optimization

PSO is a population based stochastic optimization technique, which was developed by Kennedy and Eberhart in 1995[10]. A basic variant of the PSO algorithm works by having a population (called swarm) of candidate solutions (called particles). The swarm consist of N number of particles moving around in an S-dimensional searching space. In this i^{th} particle is represented as $X_i=(x_{i1},x_{i2},\ldots x_{iD})$. The best previous position (p_{best}) of any particle is $P_i=(p_{i1},p_{i2},\ldots p_{iD})$. The global best particle, which are represent by g_{best}. The velocity of the particle i is $V_i=v_{i1},v_{i2},\ldots v_{iD})$. The particles are manipulated of the velocity according to following equations:

$$V_{id}=\omega.v_{id}+c_1.r_1().(p_{id}-x_{id})+c_2.r_2.(p_{gd}-x_{id}) \quad (4)$$

Where ω is the inertia weight, in c_1, c_2 denote the acceleration coefficients d=1,2,….S, and r_1 and r_2 are two random numbers uniformly distributed the range in [0,1]. The acceleration constants c_1 and c_2 in eq. (4) represent the weight have stochastic acceleration terms that appeal each particle toward p_{best} and g_{best} positions. The each particle then moves to a new potential position as follows:

$$x_{id} = x_{id} + v_{id} \quad (5)$$

The PSO algorithm procedure as represented as follows:

- Initialize a population of particles with random positions and random velocities on D dimensions in the future space.
  - Initialize P_i with copy of X_i and initialize the index of particle P_g (best position of neighbor particle) with the best fitness function value among the population.

- For each particle, to evaluate the fitness value in the population.
- Get the p_{best} value.
  - If the fitness value of current particle i is better than p_{best}, and then set the fitness value of as a new p_{best} of particle i.
- Get the g_{best} value.
  - If the fitness values are population's overall previous best.
- Update the velocity and position of the particles according to (4) and (5)
- Until a termination criterion is met, usually best fitness value or a maximum number of iterations.

4 Experimental Setup

4.1 Input Data Preparation

The data analysis in the experiment is retrieved from Reuters corpus of newswire articles "Reuters -21578, Distribution 1.0" resides in Reuters Ltd. and Carnegie Group, Inc. This corpus used in research and development of natural language-processing, information-retrieval and machine learning systems. Reuters newswire has formatting of the documents and organization of data files was done in 1997 by David D.Lewis. Datasets are used to utilize the standard format of modApte train and test split [9].

Input documents retrieved from top 10 largest categories of Reuters-21578 according to the ModApte Split, 9980 documents partitioned into a training set of 7193 documents and a test set of 2787 documents. The stop words are removed from SMART stop list for the further process of term retrieval. Porter stemming algorithm has been used to remove the punctuations, numbers and other unwanted special character in the document.

Document datasets are preprocessed into category-of-words for vector space representation of normalized weight in the eq.(3). PSO based feature selection method is used in the best feature from the population whole feature space.

Support Vector Machine (SVM) performs the classification by constructing to independent dimensionality of feature space. Its measure the complexity of hypotheses based on the margin with assign a separate the data, not the number of features. This means that we can generalize even in the presence of some many features, if our data is separable with a wide margin using functions from the hypothesis space.

4.2 Performance Measures

The performance measures such as precision, recall, accuracy and computational time are used to evaluate the performance.

$$recall = \frac{Number\ of\ documents\ retrived\ that\ are\ relevant}{Total\ number\ of\ documents\ those\ are\ relevant} \Rightarrow \frac{tp}{(tp+fn)} \quad (7)$$

$$precision = \frac{Number\ of\ documents\ retrived\ that\ are\ relevant}{Total\ number\ fo\ documents\ that\ are\ retrived} \Rightarrow \frac{tp}{(tp+fp)} \quad (8)$$

Where TP-the number of true positives
FP-the number of false positives
FN-the number of false negatives

To evaluate performance measures analysis to the calculation accuracy for the categorization of the text.

5 Results and Discussions

5.1 Feature Selection Performance Analysis

We have used the Reuters- 21548 for training and testing the text classifier. Forever we select the different distinctive features, split into {2000, 4000, 6000 …20000} from ten document training sets to improve the classification accuracy and computational processing time. The performance measures of TFIDF-PSO are framed using precision, recall, f-measure, accuracy, and computational time to implemented and compared in TF-IG, TF- X^2 feature selection algorithm.

5.2 Classification Comparative Analysis

In Table-5.1 shows that performance analysis of TF-IG SVM is compared with number of features in 2000's, TF-IG SVM accuracy in percentage, precision, recall and computation time in micro seconds. The features are classified into 2000's of 10 sets. The precession values sustained the range of 10000 to 20000 features, because relevant features occurred similarly in dataset.

Table 5.1- Performance analysis of TF-IG SVM

Number of Features	Accuracy (%)	Precision	Recall	Time(µs)
2000	76.77	0.801	0.7415	826
4000	78.35	0.8474	0.7768	1640
6000	78.79	0.855	0.7802	1696
8000	80.72	0.8746	0.7996	2055
10000	83.2	0.8851	0.8212	2486
12000	83.2	0.8851	0.8222	2461
14000	83.19	0.8854	0.8218	2454
16000	83.17	0.8856	0.8214	2500
18000	83.19	0.8853	0.8215	2469
20000	83.23	0.8867	0.822	2505
Avg	**81.381**	**0.86912**	**0.80282**	**2109.2**

In Table-5.2 shows that the performance of TF-X2 SVM. The mean value has been calculated for the comparison of irrespective of document features value. The features range from 2000 to 6000 TF-X2 SVM accuracy has improved from 74.03% to 79.84%. It has a drastic changes form 79% to 81.9% and 81.9% to 84.5% subsequently in the

range of 8000 and 10000 after that sustained in the ranges of 84%.

Time analysis has been done in micro seconds in the ranges of 2000's of documents up to 20000 in 10sets. In particularly 2000's documents taken 918 µs of less time and 4000's documents are certainly increased 1528µs.

Table 5.2 - Performance analysis of TF-X2 SVM

Number of Features	Accuracy (%)	Precision	Recall	Time (µs)
2000	74.03	0.755	0.7304	918
4000	79.13	0.7992	0.7808	1528
6000	79.84	0.8001	0.7856	1598
8000	81.9	0.8233	0.8054	1723
10000	84.56	0.8421	0.8312	2175
12000	84.57	0.8416	0.8312	2159
14000	84.72	0.8428	0.831	2158
16000	84.63	0.8418	0.8311	2185
18000	84.69	0.8421	0.831	2161
20000	84.62	0.8425	0.8304	2197
Avg	**82.269**	**0.82305**	**0.80881**	**1880.2**

In Table-5.3 shows that performance analysis of TF-IDF-PSO SVM are compared with number of documents in 2000's, TF-IDF- PSO SVM accuracy in percentage, precision, recall and computation time in micro seconds. The documents are classified into 2000's of 10 sets.

Table 5.3 - Performance analysis of TF-IDF-PSO SVM

Number of Features	Accuracy (%)	Precision	Recall	Time(µs)
2000	81.51	0.8457	0.7852	958
4000	82.53	0.8848	0.7991	1759
6000	82.79	0.8937	0.8191	1772
8000	82.85	0.9057	0.8069	2150
10000	83.14	0.9214	0.8141	2575
12000	83.14	0.9222	0.8146	2580
14000	83.19	0.9229	0.814	2589
16000	83.24	0.923	0.8141	2570
18000	83.23	0.9226	0.8144	2571
20000	83.15	0.9231	0.8148	2579
Avg	**82.887**	**0.90651**	**0.80963**	**2210.3**

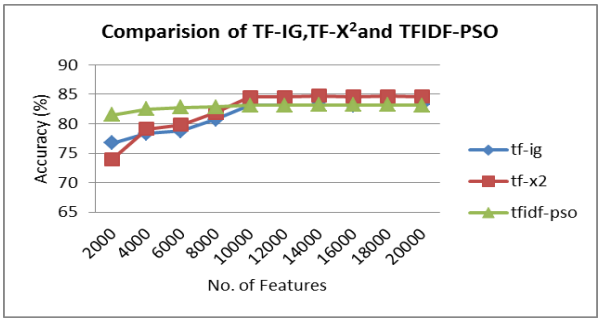

Figure:- 1 Comparison of TF-IG, TF-X^2 and TFIDF-PSO in terms of accuracy and number of features

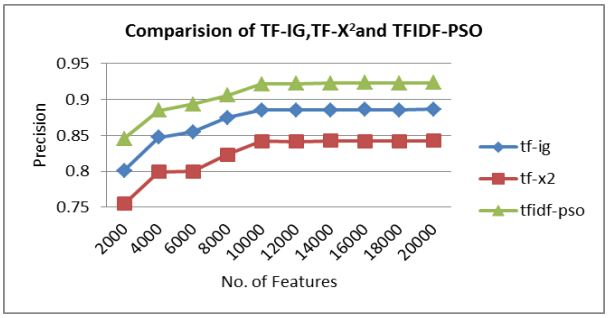

Figure:- 2 Comparison of TF-IG, TF-X^2 and TFIDF-PSO in terms of precision and number of features

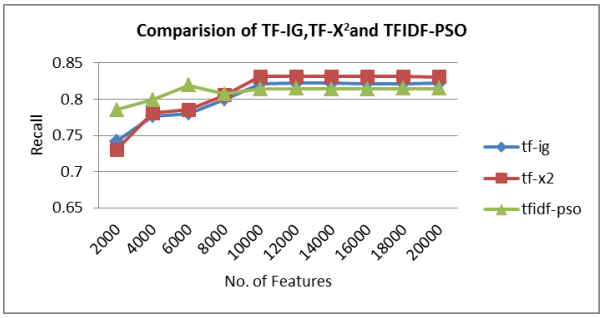

Figure 3 Comparison of TF-IG, TF-X^2 and TFIDF-PSO in terms of recall and number of features

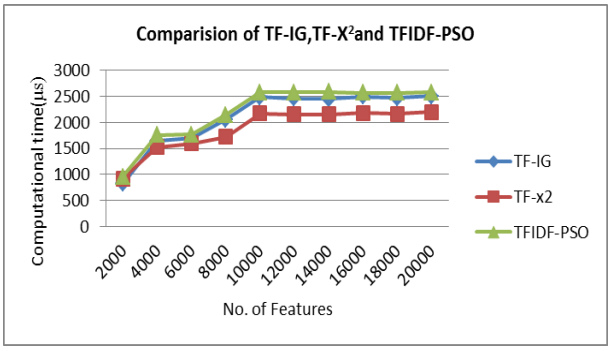

Figure 4 Comparison of TF-IG, TF-X^2 and TFIDF-PSO in terms of computational time and number of features

In Fig.1, 2, 3 and 4 shows that the ranges of result features. The experimental results starts from 2000 to 20000 separated into 10 sets compared with computational accuracy in percentage. Most of the results sets of TF-IG, TF- X^2 and TFIDF-PSO are occur in the computational accuracy on Reuters 21578 data corpus using SVM. TFIDF-PSO shows consistent performance irrespective of that number of documents. It can be conformed fig.1 can identify that variations in performance for TFIDF-PSO is minimal compared to TF-IG and TF-X^2. So we can use PSO method any applications irrespective of the corpus size.

6 Conclusion

TFIDF- PSO-based feature selection algorithm has been combined and executed in the Reuters 21578 dataset as a input data set. The experimental results proved that the developed TFIDF- PSO algorithm compared with TF-IG and TF-X^2 in terms of better classification accuracy. Statistical approaches like TF-IG and TF-X^2 have result the best classification accuracy. From the obtained result analysis, TFIDF-PSO has precision and accuracy has been improved and the performance shown in the feg.1. Interaction in the TFIDF-PSO enhances progress toward the solution. Generally heuristic based feature selection approach which needs more computational time compared to other feature selection methods. We are trying to minimize the computation time without compromising the performance the in our future work.

References

[1] XU Junling1, XU Baowen1,2†, ZHANG Weifeng3, CUI Zifeng1, ZHANG Wei, A New Feature Selection Method for Text Clustering, Wuhan University Journal of Natural Sciences Vol.12 No.5,pp: 912 – 916, 2007.

[2] Bilal M. Zahran and Ghassan Kanaan, Text Feature Selection using Particle Swarm Optimization Algorithm, World Applied Sciences Journal 7 (Special Issue of Computer & IT): 69-74, 2009.

[3] Akiko Aizawa, An information-theoretic perspective of tf–idf measures, Information Processing and Management 39, pp.45–65, 2003.

[4] Yao-Tsung Chen, Meng Chang Chen, Using chi-square statistics to measure similarities for text categorization, Expert Systems with Applications 38, pp.3085–3090, 2011.

[5] T Mori, M Kikuchi, K Yoshida, Term weighting method based on information gain ratio for summarizing documents retrieved by IR systems, Journal of Natural Language Processing, 9, 2001.

[6] F Debole, F Sebastiani, Supervised term weighting for automated text categorization, 18th ACM Symposium on Applied Computing, pp. 784-788, 2003.

[7] Moyotl-Hernández, E., & Jiménez-Salazar, H, An analysis on frequency of terms for text categorization. Procesamiento del lenguaje natural, 33, pp.141-146, 2004.

[8] C. Deisy, M. Gowri, S. Baskar, S.M.A. Kalaiarasi, N. Ramraj, A Novel Term Weighting Scheme MIDF for Text Categorization, Journal of Engineering Science and Technology, Vol. 5, No. 1 pp. 94 – 107, 2010.

[9] The reuters-21578 text categorization test collection. http://kdd.ics.uci.edu/databases/ reuters21578/reuters21578.html.

[10] James Kennedy[1] and Russell Eberhart[2], Particle Swarm Optimization, Proc. of IEEE. International Conference on Neural. Networks (ICNN), Vol.IV, pp.1942-1948, 1995.

[11] R.Parimala, R.Nallaswamy, Feature Selection using a Novel Particle Swarm Optimization and It's Variants, I.J. Information Technology and Computer Science, 5, 16-24 2012.