Journal of Statistics Applications & Probability
An International Journal

http://dx.doi.org/10.18576/jsap/110206

Evaluation of descriptive type answer using transformed weight and Cosine-SVM

K. Meena[1], *R. Lawrance* [2], *S. Suresh* [3] *and Moaiad Ahmad Khder* [4,*]

[1]Department of Computer Science, Ayya Nadar Janaki Ammal College, Sivakasi, India
[2]Department of Computer Applications, Ayya Nadar Janaki Ammal College, Sivakasi, India
[3]Department of Multimedia Science, Ahlia University, Kingdom of Bahrain
[4]Department of Computer Science, Applied Science University, Bahrain

Received: 4 Jan. 2022, Revised: 20 Feb. 2022, Accepted: 25 Feb. 2022.
Published online: 1 May 2022.

Abstract: Text Mining is the technique of obtaining high characteristic information from text. In recent years, applications of text mining are broadly used in the fields of multimedia, biomedical, patent analysis, anti-spam filtering of emails, linguistic profiling and opinion mining etc. To extract useful patterns from text, various tasks such as text preprocessing, feature extraction, pattern discovery and evaluation are performed on it. The proposed work has been developed as an efficient and effective classification algorithm for textual data base. This algorithm helps to evaluate the descriptive type answers collected from the learners and also eliminate the discrepancy in manual evaluation. The implemented framework preprocesses the documents in two steps. Initially, the documents have been pruned and stemmed to moderate the size of the documents. Also, some of the feature extraction methods have been analyzed and implemented for feature extraction. The existing feature extraction method Term-Frequency-Inverse Document Frequency (TF-IDF) assigns weight to the term, based on the occurrence. But the modified TF-IDF (M-TF-IDF) assigns weight to the term based on the occurrence and importance of the terms in the document. This weighting scheme is used to increase the accuracy of the classification algorithm. But this method does not consider semantic similarity of the term. Hence Latent Semantic Analysis (LSA) method is discussed to select the terms based on the semantic similarity. The combination of M-TF-IDF and LSA has assigned weight to the terms based on the importance and semantic similarity between the terms. The Support Vector Machine (SVM) algorithm classifies the text document which depends on the kernel functions and cost parameter. The proposed work has introduced cosine similarity function as decision making function. The implemented framework Cosine-SVM (CSVM) classifies the new test data in three steps. First, the cosine similarity value has been calculated between each group support vectors and the new test data. Then, the average is calculated between them and the similarity value has been checked. If the new test data has the highest similarity with any one group of support vectors, then the label of that group has been assigned to the test data.
The present work effectively and efficiently classifies the bench mark data set and hence it has also been used to evaluate the descriptive type answer written by the learners. This method has a number of benefits like increased reliability of results, reduced time and effort, reduced burden on the faculty and efficient use of resources.

Keywords: Text document, Feature extraction, Cosine-SVM classification, descriptive type answer

1 Introduction

Internet data is accessible for everyone in the world that forces internet an ascendant platform. Due to the flexibility and scope of the internet, various new technologies have been practiced continuously to enhance the perceptibility of its content. Social media is a collection of web-based applications which allow the user to create the information and to exchange it through the technological and ideological foundations. Internet consists of all information available in the world and thus arises the necessity for clustering and classifying of the information available on the internet so that the users can obtain the appropriate and valuable information. It is difficult for scheming and developing a standard algorithm

[*]Corresponding author e-mail: moaiad.khder@asu.edu.bh

or technique to retrieve data from all the web pages [1]. In the present work, an algorithm has been designed and developed for classifying the text in an efficient and effective manner.

A text available on the internet can be effectively classified by using the proposed algorithm. Data mining presents various techniques to detect useful information from the databases. These techniques have the ability to assist many organizations to focus on the delightful information in their data warehouses. Steps involved in data mining process are depicted in Figure 1.

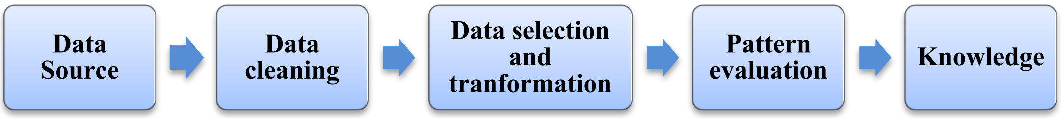

Figure 1. Data mining processing steps

Data mining function is separated into the descriptive and predictive model. A descriptive model enables us to find out the prototype and associations in a sample data and its functions are clustering, summarization, association rules and sequence discovery [2]. The predictive model enables one to forecast the values of result using known results from a diverse set of sample data and some of its tasks are classification, regression and time series analysis. Classifying documents into pretend grouping, based on the class specified, is recognized as content classification. It is an automatic arrangement of texts into groups, as specified by the user. Classification consists of predicting a definite effect based on the known input and also the methods to predict the value of an attribute based on the values of other attributes. There are various classification techniques available such as neural network, memory-based reasoning, Support Vector Machine, Naïve Bayes and Bayesian belief, decision tree based methods, rule based methods, etc. [2][3][4]

Text classification is a promising field of study which is used in the fields of product management, academia, governance and marketing. Text classification is the activity of cataloging text in a natural way into a related group, using a pre-defined set. Classifying the text content helps the users to navigate easily and search inside an application or website. Text mining process consists of text document preprocessing, text transformation, feature selection, integration of data mining practices, elucidation and assessment [5]. This process is shown in Figure 2.

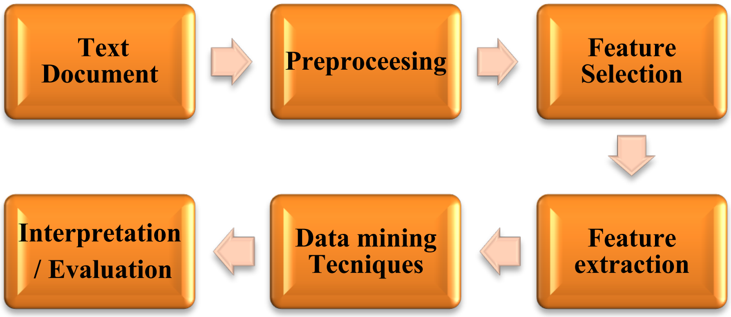

Figure 2. Text mining Process

The proposed work first analyses the problems that come across in the classification system and then suggest solutions to solve some of the problem faced in the society.

- Problems in classification system

  Some of the problems of the existing classification systems have been considered in this work.

These are:

- TF-IDF gives importance only to the uncommonness of the term.
- Existing methods do not consider the semantic information
- Support Vector Machine algorithm produces the classification result which depends on the kernel parameter and cost value.

- Problem in evaluating descriptive type answer

The present work focuses on the assessment of expressive type responds which will eradicate the inconsistency in manual evaluation.

The present work is necessitated by the desire to find a solution to the above addressed problems by the effective and efficient classification of text documents.

The software prototype being developed has used the modified TF-IDF and LSA for feature extraction. Support Vector Machine is used for classification but it based on the cost parameter and kernel. If the cost parameter and kernel are changed, the classification accuracy is also changed. To avoid this variation in classification accuracy, cosine similarity function is used as decision making function. The classification accuracy has been produced irrespective of the optimal separating hyper plane. Reuters-21578-R8 data set and descriptive type answer dataset are used to check the efficiency of the implemented algorithm. Assessment outcomes show that this algorithm outperforms other algorithms and also has less impact with the change of cost value and kernel functions. The novel automatic text classification algorithm evaluates the students' answers effectively and efficiently. Thus, the scope of the work to ensure an efficient and effective classification of text document has been fulfilled.

In this paper, the proposed work illustrates the need for the design and development of an automatic categorization algorithm to classify the text documents effectively and efficiently. The section 2 explains in detail the various works in the literature which are related to the present work. Section 3 discusses the importance of preprocessing, illustrates in detail the need for feature mining method, the offered feature withdrawal methods, the modified feature extraction method, the classification algorithm for text document and the Cosine-SVM based classification algorithm. The results obtained with the above algorithm are discussed with supportive figures and tables in section 4. Section 5 discusses the concluding remarks and scope for further extension of the present work.

2 Literature Review:

Researchers are developing a new algorithm for classifying the text document or modifying the existing algorithm with their ideas to produce better results than the algorithms available in the literature. This survey discusses the various works related to text mining, preprocessing, clustering and classification.

The Arabic text has been classified [6] with cosine method and Latent Semantic Indexing. It also proclaims that SVM and K-Nearest neighbors perform the classification better than the Naïve Bayes, neural network and Random Forest. [7] uses low rank matrix factorization to decrease the dimension of the text matrix for analyzing the text document. [8] provides a large analysis of different surveys based on Arabic text using text mining techniques.

[9] discuss different forms of feature assortment techniques such as embedded, filter, hybrid and wrapper with various classifiers such as decision tree, support vector machine, nearest neighbor, neural networks and Naïve Bayes for text document classification. [10] develop a new weighing scheme for term weighing instead of term frequency. This new weighing scheme is based on the combination of square

root and term frequency. [11] analyze the various feature selection methods and propose an algorithm, using dynamic programming, to automatically select features from the text document.

[12] discuss text classification to analyze the information related to suicidal feelings and behavior as expressed through the various social media.

It is very important from the public health point of view. This paper constructs an algorithm for automatically detect the suicide content available on the media. [13] compare different automatic classification methods using 41 social media datasets of varying size and also with different languages. This comparison shows that the marketing departments mainly depend on Linguistic Inquiry and Word Count (LIWC) and SVM. Support Vector Machine classification performance is medium on the social media datasets. [14] discuss and provide a general idea about text classification, various steps in text classification, feature extraction methods, classification algorithms and evaluation measures of various classification algorithms related to text.

[15] discuss about different feature extraction methods, dimensionality reduction methods, many machine learning algorithms and different evaluation methods. In many applications, text classification needs some advanced text classification machine learning methods. This paper discusses about the feature extraction methods of Word2Vec, Term Frequency, and Global Vectors for word representation and Term Frequency-Inverse Document Frequency. [16] use text classification algorithm on doctor review data set. [17] uses embedded word representation to represent the document and has been used to increase the classification accuracy.

[18] examine and classify hierarchical text using various machine learning algorithms. The working of Hierarchical text classification is useful like the work of a librarian and is used in medical applications also. [19] propose an algorithm to take out composite characteristics from the text manuscript. SVM is used to take into account of the various types of actions from videos [20]. Since it uses more than one SVMs to complete a task, it proves that the SVM produces the best classification performance by selecting appropriate features. Fuzzy metric has been used to derive the informative sentence from the text document and after that the document is automatically assessed [21] and this motivates the researchers to evaluate the descriptive type answer using SVM. Euclidean-SVM can be used for text document classification [22].

3 Proposed Work with Algorithm

The objective of preprocessing progression is to choose the important traits from the text document and to remove irrelevant and redundant features from it. Several machine learning algorithms have different advantages such as developed automated learning algorithms that are robust to erroneous and unfamiliar input and also produce more accurate result to the supplied input.

Tokenization and normalization are the important components of preprocessing. The tokenizations steps are separate the text into words using white spaces, line breaks or punctuation marks. Tokenization is used for efficient processing of text document.

Tokenized text should be normalized before further processing. This process consists of a series of steps such as change all the text to the same case (lower or upper), convert numbers to their corresponding equivalent words or remove numbers, remove punctuation and eliminate stop words. These tokenized words are stemmed using Porter stemming algorithm. Stemming is the method of eliminate affixes (infixes, prefixes and suffixes) from the given word and produces the root form of the word. This process is used to save the time and memory space. It is an important step in the text mining process.

This work analyzed different feature extraction methods and proposed transformed weight and semantic similarity method for extracting vital characteristics. [23] applied TF-IDF for text classification.

[24] proposed feature extraction procedure using M-TF-IDF and Latent Semantic Analysis. This weight is calculated by using the sum of number of documents and the count of number of words in the document. It assigns the weight depend on the significance and occurrence of the word. The relationships between the words are determined using LSA. Finally, words with the modified weights are selected and it is given as input to the SVM.

SVM is a machine learning method which is used to curtail the mistake with the help of the Structural Risk Minimization principle [25]. Text classification can be done with SVM classification [26, 27] and Bayesian classification techniques [28]. The proposed work tries to eliminate this deviation [29].

Support Vector Machine used to construct the model. The model has been predicted with the help of cosine method. Cosine function is used to determine the resemblance stuck between the two vectors and it returns the angle between them, based on the orientation of vectors. If the two vectors have the same orientation; the cosine of angle between these two vectors is 1 and zero if they are related to each other by 900. It is used to evaluate the viewpoint among two n-dimensional vectors using equation (1) below.

$$\text{Cosine-similarity(X,Y)} = \frac{\sum_{i=1}^{n} X*Y}{\sqrt{\sum_{i=1}^{n} X^2}\sqrt{\sum_{i=1}^{n} Y^2}} \qquad (1)$$

If the cosine similarity function returns 1, then it means that the two vector values are similar and if it returns -1then the vector values are dissimilar.

In this proposed work, SVM is used for selecting the support vectors and the Cosine similarity function is used as decision making function in this work and hence the name Cosine-SVM. The SVM and cosine similarity function are used in the training and testing phases respectively. This method does not depend on the cost parameter value C and kernel function. The implemented framework classifies the new test data in three steps. First, the cosine similarity value has been calculated between the new test data and for each group support vectors. Then the average between them is calculated and the similarity value has been checked. Assigned the label depend on the similarity value.

Algorithm steps

1. Collect the text documents.
2. The size of text document has been reduced with the help of tokenization and stemming.
3. Construct the Document-Term matrix.
4. The text document transformed using the M-TF-IDF and Latent Semantic Analysis.
5. Extract the high similarity and most occurrence words from the document using the proposed weight method.
6. The selected words weight vectors given as input to the SVM.
7. The support vectors produced by the SVM used for further processing.
8. To predict the label of new document,
    i. The cosine similarity function used to calculate the similarity among the new data and the support vectors of each group.
    ii. Similarity value has been checked. If the new data has the high resemblance with any of support vectors group, assigned that group label to the new document.

9. Produce the results of classification.

4 Results and Discussion

The planned text classification algorithm has been executed in R-software and the algorithm performance is assessed with the two text corpus, namely Reutuers-21578 R8 dataset and descriptive type answer dataset.

To check the classification result of the implemented algorithm, tests have been approved out on both SVM and the proposed Cosine-SVM (CSVM) classifier. The implementation is carried out for both classifiers by using the three kernel functions and cost parameters.

4.1 Dataset

Descriptive type answers dataset and Reuters-21578-R8 dataset and are used for text mining analysis of this proposed work. The Reuters-21578-R8 has 118 categories but only 8 out of 118 categories are used here. Descriptive type answers data set consists of students' answer as 700 documents and 4 labels such as two, three, four and five marks.

4.2 Performance of the proposed algorithm

Table 1 compares the performance of the classification algorithms with SVM and CSVM classifiers for linear, nonlinear and RBF kernels by varying the cost parameter from 10 to 100000. The SVM with linear kernel has an accuracy ranging from 88.3% to 78.54% as given by the first row of Table 1. With the same variation in cost parameter, the variation in percentage of accuracy is from 78.5 to 68.6 and 80.43 to 68.22 for SVM with nonlinear and RBF kernels respectively as in the second and third rows of Table 1.

Table 1. Reuters-21578-R8 dataset classification accuracy using SVM and CSVM classifiers

Kernels	**Accuracy (%)**			
	Cost Parameter(C)			
	10	**1000**	**10000**	**100000**
M-TF-IDF+LSA+SVM-linear	**88.3**	84.44	82.18	78.54
M-TF-IDF+LSA+SVM-polynomial	78.56	73.52	72.65	68.6
M-TF-IDF+LSA+ SVM-RBF	80.43	78	68.22	68.22
M-TF-IDF+LSA+CSVM-linear	**92.21**	92.21	92.21	90.11
M-TF-IDF+LSA+CSVM-polynomial	90	90	90	90
M-TF-IDF+LSA+ CSVM-RBF	91.45	91.45	91.45	90.78

The CSVM classifier with the linear kernel produces the accuracy ranging from 92.21% to90.11% when cost parameters are 10, 1000, 10000 and 100000 as shown in the fourth row of the Table 1. For the similar variation of cost parameter, CSVM classifier with the polynomial kernel recorded the same accuracy value of 90% as given by the fifth row of Table 1. The CSVM with RBF kernel produces accuracies between 91.45% and 90.78% for a similar cost parameter variation as presented in the last row of Table 1. The comparison is shown in Figure 3.

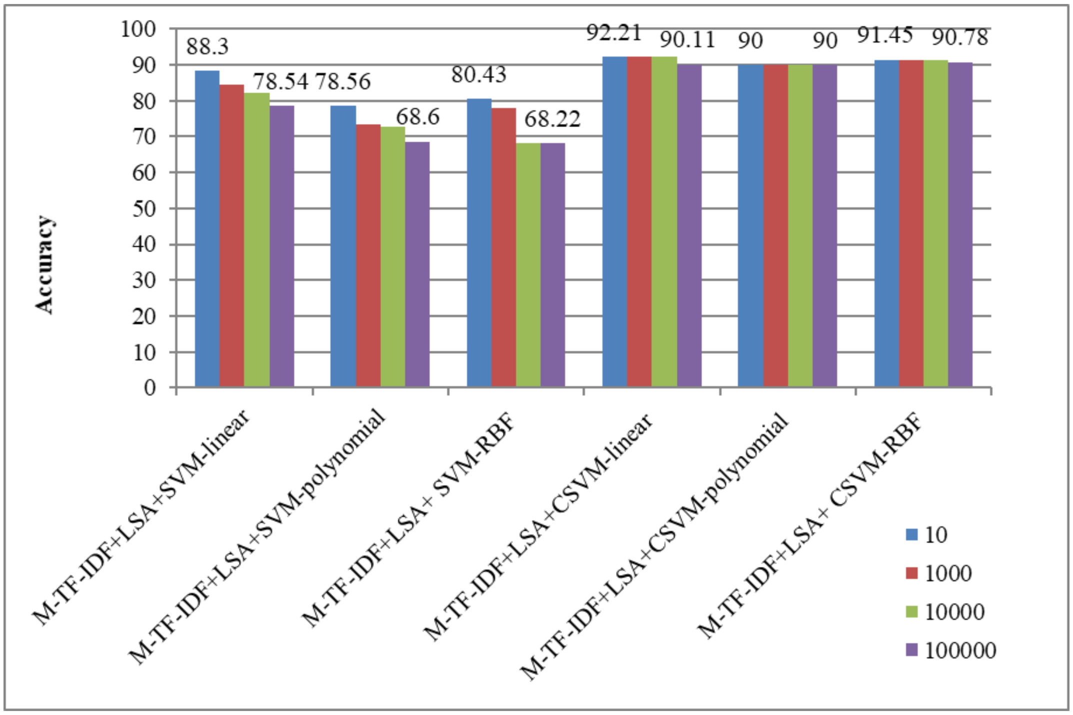

Figure 3. Classification performance of SVM and CSVM of the Reuters-21578-R8 dataset

The variation of accuracy, with the variation in cost parameter C, is appreciable in SVM for all kernels whereas the variation of accuracy with C is negligible in CSVM for all type of kernels. In fact, CSVM gives the maximum accuracy even for the lowest value of C=10 with the linear kernel. When compared with the SVM, CSVM accuracies are better and also it proves that the accuracy does not depend on the kernel and cost parameter value.

Euclidean-SVM [18] produces an accuracy of 84.73% only for the same dataset and SVM-NN [30] classifier produces an accuracy of 81.48% only for Reuter-21578 R8 dataset whereas the CSVM classifier has produced 92.21% accuracy for the same dataset.

Figure 4 shows the classification performance of SVM and CSVM of the descriptive type answer dataset.

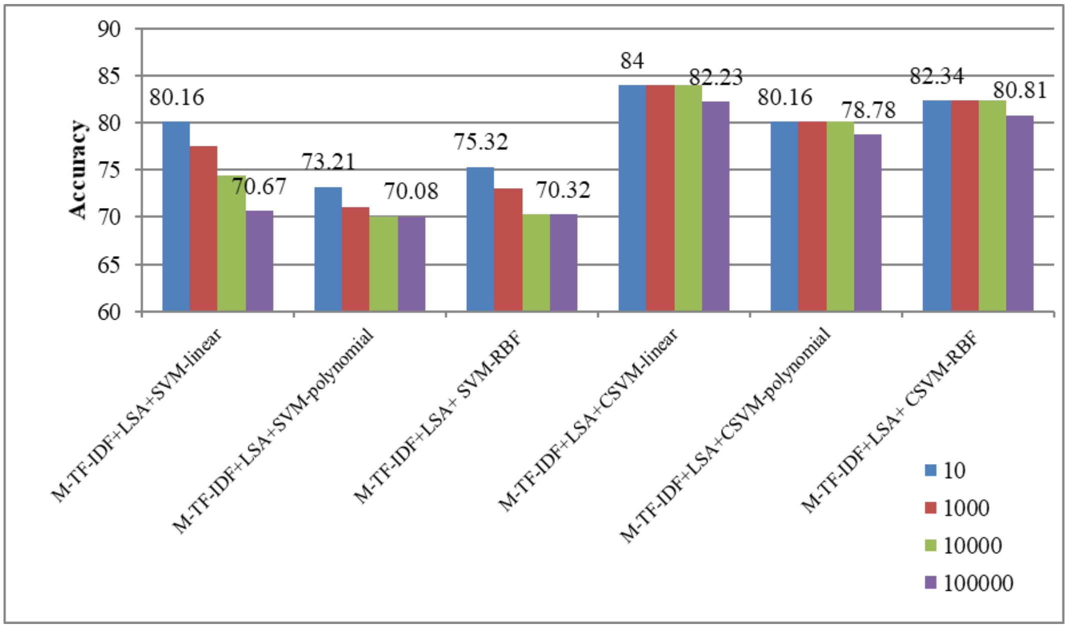

Figure 4. Classification performance of SVM and CSVM of the descriptive type answer dataset

The comparison of SVM with CSVM classifiers is shown in table 2.

Table 2. Comparison of SVM with CSVM classifiers

Dataset	M-TF-IDF+LSA+ SVM Accuracy (%)	M-TF-IDF+LSA+Cosine-SVM Accuracy (%)
Reuters-21578 R8 dataset	88.3	92.21
Descriptive type answer dataset	80.16	84

Figure 5 shows explicitly that the Cosine-SVM classifier approach achieves high accuracies, without transforming the original input into high dimensional vector by using kernel functions and also avoids the selection of C value. If the cost parameters value is small, a greater number of support vectors have been identified which provides more information to make the classification decision. Hence, the Cosine-SVM approach is more suitable for accurate classification and even though the same support vectors are used in SVM and Cosine-SVM, the Cosine-SVM outperforms the SVM classifier. In SVM classifier, to find the correct category of the test data point, the test is calculated with the alpha standards and they play a major role in the classification. If the support vectors are wrongly weighted, it may lead to misclassification whereas in the Cosine-SVM classifier, such type of calculation is not necessary, because in the Cosine-SVM classifier, the similarity value is calculated between the experiment data value and every category support vectors. Here the support vectors play a major role and do not depend on the alpha weight values. This leads to the state that the accurateness of Cosine-SVM classifier is better than the SVM classifier even though both use the same support vectors for classification.

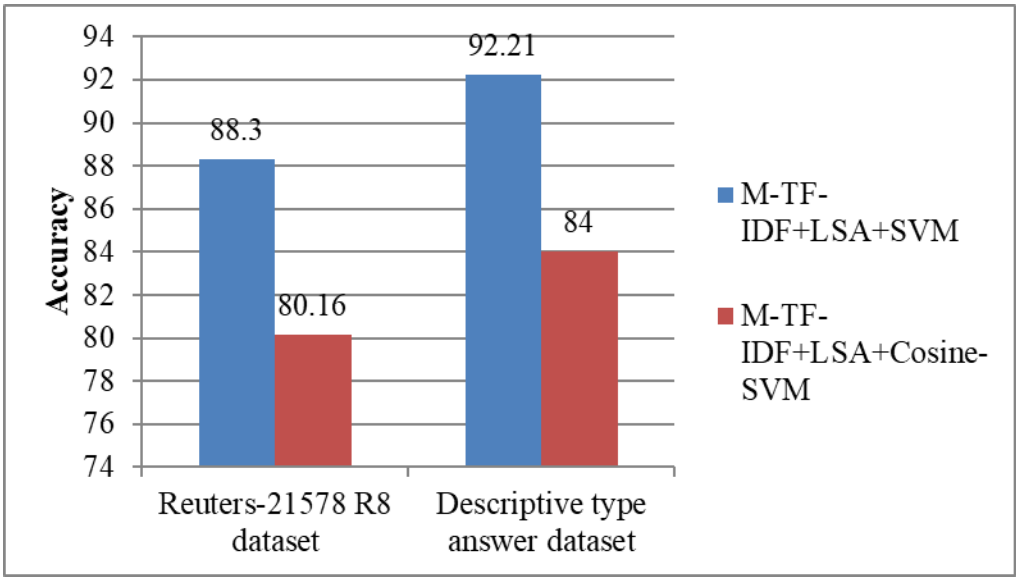

Figure 5. Performance comparison of SVM and CSVM for the two data sets

Table 3 shows the classifier performance using the recall, precision and F1-score. The recall percentage indicates the number of relevant results produced by the classifier and the precision percentage indicates how much of the results are related. The recall and precision being returned by the classifier CSVM are higher than those by SVM classifier which shows that the CSVM classifier is better than SVM classifier. [31] evaluate the descriptive type answer using relation based feature extraction method and it shows only 85% precision and recall. But this proposed method shows 88% recall and precision.

Table 3.Classifier performance using recall, precision and F1-score

Dataset	**M-TF-IDF+LSA+SVM**				**M-TF-IDF +LSA+Cosine-SVM**			
	Recall (%)	**Precision (%)**	**F1 Score (%)**	**Training time (Seconds)**	**Recall (%)**	**Precision (%)**	**F1 Score (%)**	**Training time(Seconds)**
Reuters-21578-R8 dataset	0.86	0.8264	0.8429	0.1962	0.9166	0.9081	0.9123	0.0783
Descriptive type answer dataset	0. 838	0.809	0.823	0.1711	0.8861	0.875	0.8773	0.0625

The training time taken by the CSVM is much less when compared with that of SVM and the time analysis is shown in Figure 6. The testing time depends on the count of support vector and data dimension. When the value of C is small, more support vectors are generated and if the support vectors are more, then the time taken for calculating the average similarity is also high. In the CSVM classification, the change in accuracy is differing slightly when the cost parameter C is changed which tells that the CSVM has a lesser amount of reliance on the cost parameter value.

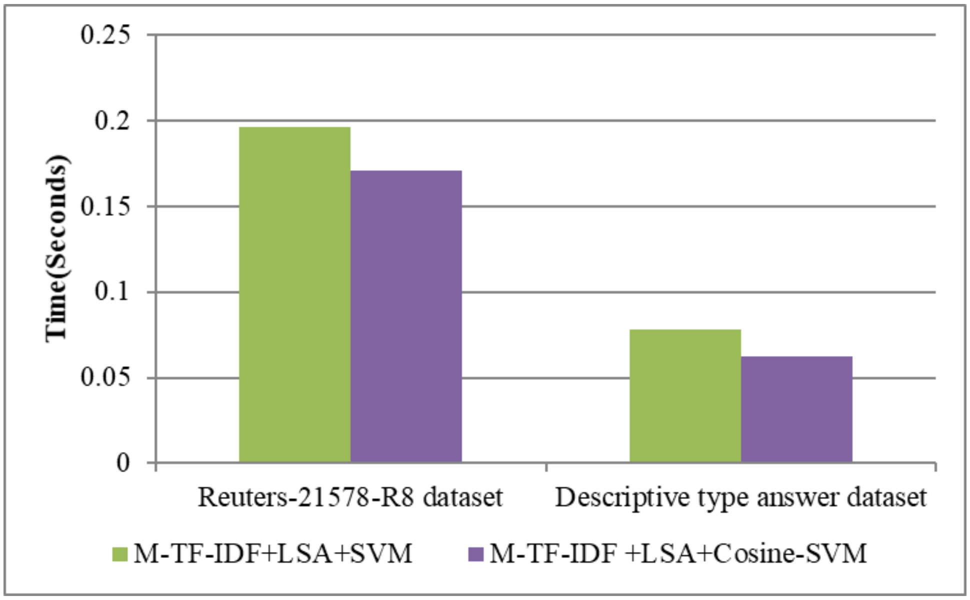

Figure 6.Time analysis of SVM and CSVM classifiers on the two data sets

5 Conclusion

The proposed work developed an algorithm for automatic classification of a text document efficiently and effectively. The text document may be of different size which can be reduced by using preprocessing techniques such as pruning and stemming and the important features are extracted from the document by using the approach being discussed.

In the first implemented approach, the weights are assigned to the features, depend on the significance, occurrence and semantic similarity of the term. These documents are further classified using SVM. Even though SVM classifies the text documents effectively, its classification efficiency depends on the cost parameter C and kernel functions. To avoid these variations in the classification accuracy, cosine similarity measurement is used as an alternative to the optimal separating hyperplane in the SVM.

This developed algorithm uses the SVM algorithm to find the support vectors of each category but to find and allocate the tag to the fresh test document, cosine function is used which finds the likeness among the support vectors of each category and the new test document. If the tested document has maximum resemblance value with some one of the categories, then allot the tag of that category to the new test document.

The same approach effectively and efficiently classifies the document in the bench mark data sets and hence it used to assess the descriptive type answer written by the learners.

The expressive category answers are collected from the learners and are evaluated with the help of the implemented algorithm. The inconsistency in the physical assessment of descriptive answers is eliminated using the proposed algorithm. It would be of more use to the academic institutions to publish their results efficiently and much earlier than before. Different approaches have been proposed and used for evaluating the learners' short answers. This proposed work assesses the student descriptive type answer.

The proposed algorithm dealing with the methods related to machine learning which is sufficient for the analysis of data being considered in this work.

The implemented approach can further be extended in the following directions:

- Some modifications can be made to other feature extraction methods and classification algorithms to classify the text document more efficiently.
- The evaluation of descriptive type answers can also be extended to the evaluation of essay type answers.
- Due to the phenomenal growth in the text document size, deep learning algorithms may be considered for classification.
- In analyzing the text data, big data analysis could provide improvement in the evaluation of descriptive type answer.

Competing interests: The authors declare that they have no competing interests.

References

[1] Khder, M. A. (2021). Web Scraping or Web Crawling: State of Art, Techniques, Approaches and Application. International Journal of Advances in Soft Computing & Its Applications, 13(3).

[2] Wael Fujo, S., Subramanian, S., & Ahmad Khder, M. (2022). Customer Churn Prediction in Telecommunication Industry Using Deep Learning. Information Sciences Letters, 11(1), 24.

[3] Khder, M. A., Fujo, S. W., Sayfi, M. A. (2021). A roadmap to data science: background, future, and trends. International Journal of Intelligent Information and Database Systems., 14(3), 277-293, 2021.

[4] Abazeed, A., & Khder, M. (2017). A Classification and Prediction Model for Student's Performance in University Level. J. Comput. Sci., 13(7), 228-233.

[5] Lee, D. L., Chuang, H., &Seamons, K. (1997). Document ranking and the vector-space model. IEEE software, 14(2), 67-75.

[6] Al-Anzi, F. S., &AbuZeina, D. (2017). Toward an enhanced Arabic text classification using cosine similarity and Latent Semantic Indexing. Journal of King Saud University-Computer and Information Sciences, 29(2), 189-195.

[7] Acharya, A., Goel, R., Metallinou, A., &Dhillon, I. (2019, July). Online embedding compression for text classification using low rank matrix factorization. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 33, pp. 6196-6203).

[8] Salloum, S. A., AlHamad, A. Q., Al-Emran, M., &Shaalan, K. (2018). A survey of Arabic text mining. In Intelligent Natural Language Processing: Trends and Applications (pp. 417-431). Springer, Cham.

[9] Deng, X., Li, Y., Weng, J., & Zhang, J. (2019). Feature selection for text classification: A review. Multimedia Tools and Applications, 78(3), 3797-3816.

[10] Dogan, T., &Uysal, A. K. (2019). On Term Frequency Factor in Supervised Term Weighting Schemes for Text Classification. Arabian Journal for Science and Engineering, 1-16.

[11] Huang, C., Zhu, J., Liang, Y., Yang, M., Fung, G. P. C., & Luo, J. (2019). An efficient automatic multiple objectives optimization feature selection strategy for internet text classification. International Journal of Machine Learning and Cybernetics, 10(5), 1151-1163.

[12] Desmet, B., &Hoste, V. (2018). Online suicide prevention through optimised text classification. Information Sciences, 439, 61-78.

[13] Hartmann, J., Huppertz, J., Schamp, C., &Heitmann, M. (2019). Comparing automated text classification methods. International Journal of Research in Marketing, 36(1), 20-38.

[14] Kobayashi, V. B., Mol, S. T., Berkers, H. A., Kismihók, G., & Den Hartog, D. N. (2018). Text classification for organizational researchers: A tutorial. Organizational research methods, 21(3), 766-799.

[15] Kowsari, K., JafariMeimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., &Brown, D. (2019). Text classification algorithms: A survey. Information, 10(4), 150.

[16] Rivas, R., Montazeri, N., Le, N. X., &Hristidis, V. (2018). Automatic classification of online doctor reviews: evaluation of text classifier algorithms. Journal of medical Internet research, 20(11), e11141.

[17] Sinoara, R. A., Camacho-Collados, J., Rossi, R. G., Navigli, R., &Rezende, S. O. (2019). Knowledge-enhanced document embeddings for text classification. Knowledge-Based Systems, 163, 955-971.

[18] Stein, R. A., Jaques, P. A., &Valiati, J. F. (2019). An analysis of hierarchical text classification using word embeddings. Information Sciences, 471, 216-232.

[19] Wan, C., Wang, Y., Liu, Y., Ji, J., & Feng, G. (2019). Composite Feature Extraction and Selection for Text Classification. IEEE Access, 7, 35208-35219.

[20] Qian, H., Mao, Y., Xiang, W., & Wang, Z. (2010). Recognition of human activities using SVM multi-class classifier. Pattern Recognition Letters, 31(2), 100-111.

[21] Goularte, F. B., Nassar, S. M., Fileto, R., &Saggion, H. (2019). A text summarizationmethod based on fuzzy rules and applicable to automated assessment. Expert Systems with Applications, 115, 264-275.

[22] Lee, L. H., Wan, C. H., Rajkumar, R., & Isa, D. (2012). An enhanced Support Vector Machine classification framework by using Euclidean distance function for text document categorization. Applied Intelligence, 37(1), 80-99.

[23] Jing, L. P., Huang, H. K., & Shi, H. B. (2002, November). Improved feature selection approach TFIDF in text mining. In Proceedings. International Conference on Machine Learning and Cybernetics (Vol. 2, pp. 944-946). IEEE.

[24] Meena, K., &Lawrance, R. (2019). An automatic text document classification using modified weight and semantic method. International Journal of Innovative Technology and Exploring Engineering, 8(12), 2608-2622.

[25] Vapnik, V. (2013). The nature of statistical learning theory. Springer science & business media.

[26] Joachims, T. (2002). Learning to classify text using support vector machines (Vol. 668). Springer Science & Business Media.

[27] Shawe-Taylor, J., &Cristianini, N. (2004). Kernel methods for pattern analysis. Cambridge university press.

[28] Yu, B., & Yang, Z. (2009). A dynamic holding strategy in public transit systems with real-time information. Applied Intelligence, 31(1), 69-80.

[29] Meena, K., &Lawrance, R. (2017). Text classification algorithm (TCLS) using Support Vector Machine. Journal of Advanced Research in Dynamical and Control Systems, 9, 1068-1090.

[30] 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 neighbor and support vector machine. Expert Systems with Applications, 39(15), 11880-11888.

[31] Nandini, V., & Uma Maheswari, P. (2020). Automatic assessment of descriptive answers in online examination system using semantic relational features. The Journal of Supercomputing, 76(6), 4430-4448.