Computers and Electrical Engineering 81 (2020) 106518

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

Effective information retrieval and feature minimization technique for semantic web data ☆

C.S. Saravana Kumar*, R. Santhosh

Department of Computer Science and Engineering, Karpagam Academy of Higher Education, Coimbatore, Tamil Nadu, India

ARTICLE INFO

Article history:
Received 8 July 2019
Revised 15 November 2019
Accepted 15 November 2019
Available online 21 November 2019

Keywords:
Feature extraction
Dimensionality reduction
Semantic web mining
Text mining
Data mining
Feature selection
Information retrieval
Feature vector

ABSTRACT

The Internet contains both structured and unstructured data. The enormous flow of Internet data creates challenges in relation to effective information retrieval. Semantic Web Mining explores Web addresses using ontological and semantic structures. For effective information retrieval in Web Mining and Text Mining, text feature extraction plays an important role. The effectiveness of the text processing is determined by the complexity and dimensionality reduction of the feature vector. In this paper, a new approach is proposed based on the semantic structure of the Web data. It combines both feature extraction and feature selection techniques for data mapping and retrieval, involving standard features for effective text mapping. This process reduces the dimension complexity in the feature vector for effective information retrieval.

1. Introduction

The basic problem in information retrieval is extracting the features of text sentences for text classification [1]. Without processing, text cannot be used directly for similarity measurement, as the required results cannot be obtained with the desired accuracy. For effective text processing, first we need to explore the semantic structure of the Web efficiently. It is necessary to identify and strongly define the relationships between data. Tagging data using appropriate techniques plays a key role. In this article, we present our own data description approach. Exploring the semantic structures efficiently helps to extract and select features to properly represent the vector space with reduced dimensions. Ensuring the vector space remains manageable is highly important. Generally, dimensionality reduction algorithms [2] are classified into two types, namely feature extraction and feature selection algorithms. Feature extraction algorithms [3,4] reduce the vector space through algebraic transformations and a new feature set is created from the base set. Feature selection algorithms focus on reducing the vector space by considering the subset features from the base set. In this paper, we combine both feature extraction and feature selection to reduce the dimensions of the feature vector space.

☆ Reviews processed and recommended for publication to the Editor-in-Chief by Guest Editor Dr. P. Pandian.

* Corresponding author.
E-mail address: csskbaba@yahoo.co.in (C.S. Saravana Kumar).

https://doi.org/10.1016/j.compeleceng.2019.106518
0045-7906/

1.1. Feature extraction

The word frequency model is widely used for text classification, Web Mining and Text Mining. It is a frequently used method for feature extraction, in which the frequency of each word is counted and maintained in the text feature vector for similarity identification. Similarity identification is conducted by calculating the distance using dot product and cosine similarity. The model considers different text attributes for the feature vector to be framed. Using the feature vector, the classifier is trained for text classification. Training the classifier when the text dataset involved is huge demands the inclusion of a greater number of text features for better classification accuracy. The complexity of the text feature model will increase when there is a demand for more text features, thus reducing the text classification accuracy. Another common approach involved in feature extraction is principal component analysis (PCA), in which dimensionality reduction is achieved by eliminating unnecessary features using a variance calculation. Variance calculations become complex and noise removal becomes challenging if there are a greater number of features involved. In our proposed approach, we consider the use of limited features to form a feature vector that maintains the same hierarchy, even for a huge volume of data. Another popular technique used for feature extraction is the random forests technique, in which the attribute is selected when it serves as the best split to attain the desired class. The attribute with the best split will be retained and will be considered in the vector space, but a problem arises when the extracted attribute is not the best split; then it impacts the vector space model, affecting the accuracy of information retrieval. Although the backward feature elimination and forward construction methods are effective, they are time-consuming and more suitable for small datasets.

1.2. Feature selection

The objective of feature selection is to identify which feature is more useful for effective classification, thus reducing the dimensionality. Numerous existing techniques can be used for this purpose and the popular ones are pruning, random, accuracy and balanced accuracy, Chi-squared, information gain, probability ratio, bi-normal separation, weighted sampling and clustering. Pruning removes rare and common words, while clustering is a method in which related features are considered together as a single feature. All of the popular methods in feature selection are highly suitable for small datasets; when the dataset is huge, then feature selection remains complex.

In this article, we propose a new approach for exploring semantic structures in the Web. The dimensionality reduction involves a feature extraction and feature selection technique with new standard features used for data mapping. Section 2 explains the word frequency approach and PCA in detail, Section 3 explains the proposed approach for feature dimensionality reduction, Section 4 compares the results of the existing methods with our own approach and Section 5 provides the conclusion followed by references.

The proposed approach will offer a substantial advantage when a greater number of training sentences are involved in training the model, as the proposed algorithm maintains a higher level of accuracy as well as improved performance for larger data sets.

2. Related work

2.1. Feature extraction techniques

2.1.1. Bag-of-Words model

Bag-of-Words [5] or Word frequency model is a common way of representing text when modelling with machine learning algorithms. Bag-Of-Words model is a simple technique to extract features from text. Bag-Of-Words focuses on a couple of points, familiar vocabulary and its frequency present in the text. Bag-Of-Words is explained using the following texts [6],

1 Peter likes to watch football. Scott likes football too.
2 Peter also likes to watch rugby games.

The above sentences can be represented in the word count as [7],

1 {"Peter":1, "likes":2,"to":1,"watch":1, "football":2,"Scott":1,"too":1} and converting it into the vector it is represented as [2,1,1,2,1,1]
2 {"Peter":1, "also":1,"likes":1, "watch":1,"rugby":1,"games":1} and converting it into the vector it is represented as [1,1,1,1,1,1]

Here each word is counted for its frequency [8] and the feature vector is defined with the Bag of Words model. Based on the feature vector the text is classified. Representing the feature vector involving bag of words will become complex when a greater number of features or words are involved in text classification, and hence it is disadvantageous.

2.1.2. Principal component analysis

Principal Component Analysis (PCA) [9] is a widely used dimensionality reduction method for a larger data set. In this approach variables or features having larger boundaries will dominate the variables with smaller boundaries. For example a variable with the range 0 to 100 will dominate the variable with the range 0 to 10 which will lead to biased results.

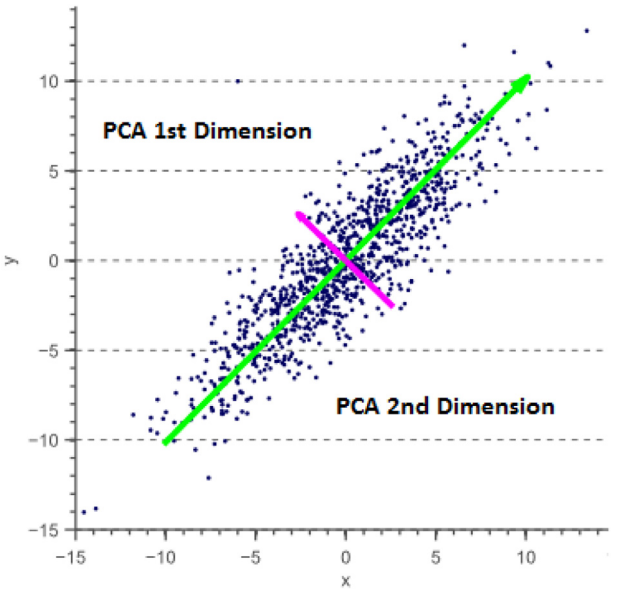

Fig. 1. Two variables are correlated by adding noise.

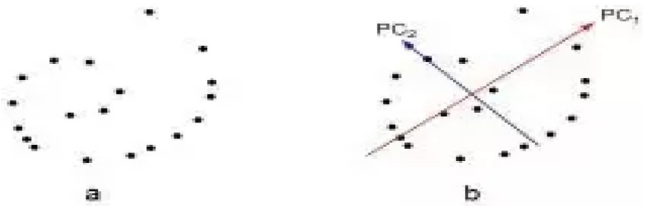

Fig. 2. Two variables with no correlation.

To prevent this problem standardization of the data set is to be performed by transforming data to comparable scales. In PCA principal component is identified by calculating eigen value and eigen vector of the variables and arranging the same in descending order and collecting the required variables [10]. The collected principal components form the feature vector. PCA approaches do have disadvantages as it relay on linear assumption. Let us consider two variables to represent size in cm and inch (refer to Fig. 1). Now these two variables are linearly correlated.

Centimeter is represented in inch (2.54 cm = 1 inch) [11], where as if there are variables which cannot be correlated (refer to Fig. 2) then PCA is not suitable and accuracy cannot be maintained [12].

Another disadvantage in using PCA is that it relies on orthogonal transformation. If the principal components are orthogonal to others then it will be difficult to find the variable with larger boundaries making PCA less reliable.

2.1.3. Random forest

For dimensionality reduction Random Forest technique [13] will generate possible trees against the target attribute. Statistical usage of different attributes is calculated and using the same the most informative subset of features is found (refer Fig. 3). If an attribute is often selected as best split then it is retained. In a tree we calculate how many times an attribute is selected as best split and based on it the attribute is ranked [14]. Attributes with higher rank are considered in the dimensional space. The disadvantage of using random forest is selecting the attribute for the best split. If proper attribute is not selected for the best split then accuracy remains a challenge [15].

2.1.4. Backward feature elimination

In Backward Feature Elimination [16] approach from the data set n input features is considered and removing n-1 feature the model is trained. Each iteration k produces a model trained with n-k features. Selecting the maximum tolerable error rate we define the smallest number of features required to attain the maximum performance with the machine learning algorithm [17]. The disadvantage of this approach is time complexity and is more suitable only for lower data set [18].

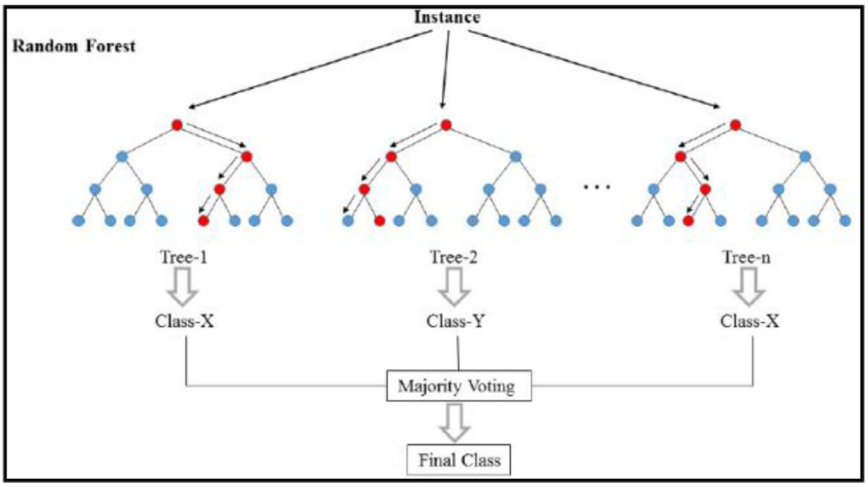

Fig. 3. Random forest for text classification.

2.1.5. Forward feature construction

This approach is converse to Backward Feature Selection [18], where we start with only one feature first and then keep adding up the features until the highest performance is achieved. Again similar to Backward Feature Elimination, Forward Feature Construction has high time complexity and it is not suitable for larger data set [19].

2.1.6. Sparsity

As most of the data within the dataset involve less number of words, the resulting matrix will have many feature values that are zeros (typically more than 99% of them) [20]. For instance a collection of 10,000 short input text data will use a vocabulary with a size in the order of 100,00 unique words in total while each input data will use 100 to 1000 unique words individually. To efficiently handle these kind of data we will typically use sparse representation [21].

2.2. Feature selection

2.2.1. Pruning

The number of features involved in large datasets will be huge and pruning technique reduces the number of features by removing frequently used words and rarely used words [22]. Pruning is commonly performed in word frequency technique which follows Zipf distribution, which states frequency of a word is proportional to 1/rank where rank is the place the corresponding word holds in the list of words and 1 is the fitting factor. By involving stopwords frequently used words may be removed [23].

2.2.2. Random

In the Random [24] feature selection technique features are ranked randomly and based on different trials the rank of the feature is changed. The best ones are kept and the others are eliminated. This is suitable and efficient for small volume of data and it is time consuming and less efficient for larger datasets.

2.2.3. Accuracy and balanced accuracy

Accuracy involves calculation of true positives (tp) and false positives (fp) to identify how many times the correct feature has been selected. Balanced accuracy takes the difference between true positive rate (t_{pr}) and false positive rate (f_{pr}) to identify the correct feature selection [10].

$$\text{Accuracy} = \text{tp} - \text{fp} \tag{1}$$

$$\text{Balanced Accuracy} = |t_{pr} - f_{pr}| \tag{2}$$

The problem with these techniques is it will involve more negative features.

2.2.4. Chi-Squared

Chi-Squared technique [25] for feature selection is mainly useful in pattern recognition. It assumes feature is independent of the target class. It does not work with smaller datasets as Chi-Squared distribution involves high volume of data for probability and statistics. Chi-Squared function is given as follows,

$$X^2(x,y) = \frac{B\,(Q_{VpDp} * Q_{VnDn} - Q_{VnDp} * Q_{VpDn})^2}{Q_{Vp} * Q_{Vn} * Q_{Dp} * Q_{Dn}} \quad (3)$$

where B is the total number of input data. A is the positive samples. x is the number of sample that contain the subsequence v, y is the number of negative samples that contain the subsequence v. Dp is the positive class and Dn is the negative class.

2.2.5. Information gain

Information Gain measures the decrease in entropy when the feature is given rather than not given. In terms of entropy equation information gain is represented as decrease in entropy of a feature having knowledge of other and it is given as follows,

$$I(A; B) = H(A) - H(A|B) = H(B) - H(B|A) \quad (4)$$

Information gain can be given by substituting the entropy equation as follows,

$$I(A; B) = \sum a \sum b(a, b) \log p(a, b)/p(a)p(b) \quad (5)$$

2.2.6. Probability ratio

Probability ratio is calculated by considering number of true positives and number of false positives. True positive is calculated by considering number of true positives divided by number of true cases. False positive is calculated by considering number of false positives divided by number of negative cases. The probability ratio represents the percentage of positives or negatives for a given sample. Based on it same ranking is done for a feature and they are identified for relevancy.

2.2.7. Sampling-based technique

Sampling based technique is an unsupervised feature selection technique where small number of features which preserves the geometric of data is chosen. A (n-(x-n)) diagonal matrix is chosen where the entries are random and each entry is ranked and maintained separately as Rank Matrix. Based on the Rank matrix the feature of diagonal matrix is altered.

2.2.8. Clustering

Clustering is a widely used technique to reduce dimensionality of the feature space while processing text data by clustering related word feature to single feature. Word clustering is best effective for text classification when smaller datasets are involved. Clustering generically has a high accuracy in sparse data.

2.3. Resource description framework (RDF) and ontology

Resource Description Framework was developed by Word Wide Web Consortium (W3C) along with Xtensible Markup Language (XML) and Uniform Resource Identifier(URI) as distribution standards. These serve as metadata for the data available on the web. Ontology where it identifies the relationship that exists between the data in the web is achieved using RDF. Web Ontology Language is developed based on these standards where using the same machine can easily interpret the data available on the web efficiently. With these metadata available data retrieval system still finds the difficulty in retrieving the user required information due to the complex nature of RDF as well as inability of the systems to exactly map the relationship between the data with these metadata available.

3. System architecture

3.1. T (Text)-Order algorithm for dimensionality reduction

Dimensionality reduction is achieved in our proposed algorithm by involving defined features (refer to Fig. 4) which handles text classification effectively for different types of data sets. From the input text data a list of noun and verb is collected. The type of noun and its number is identified. All these data are collected from the input data.

3.1.1. Mood feature

Mood of the input text data is identified from the verb and it is classified into three major types namely subjunctive, imperative and indicative (refer to Fig. 5).

If the input text data convey a wish or a suggestion then we classify the input text data under Subjunctive mood. The same is identified by involving conditions like,

(i) Input text sentence starts with the word "If".
(ii) Validating the presence of the words like "ask", "demand", "determine", "insist", "doubt", "order", "pray", "recommend", "regret", "require", "suggest", "wish", "request", "were", "be", etc.,

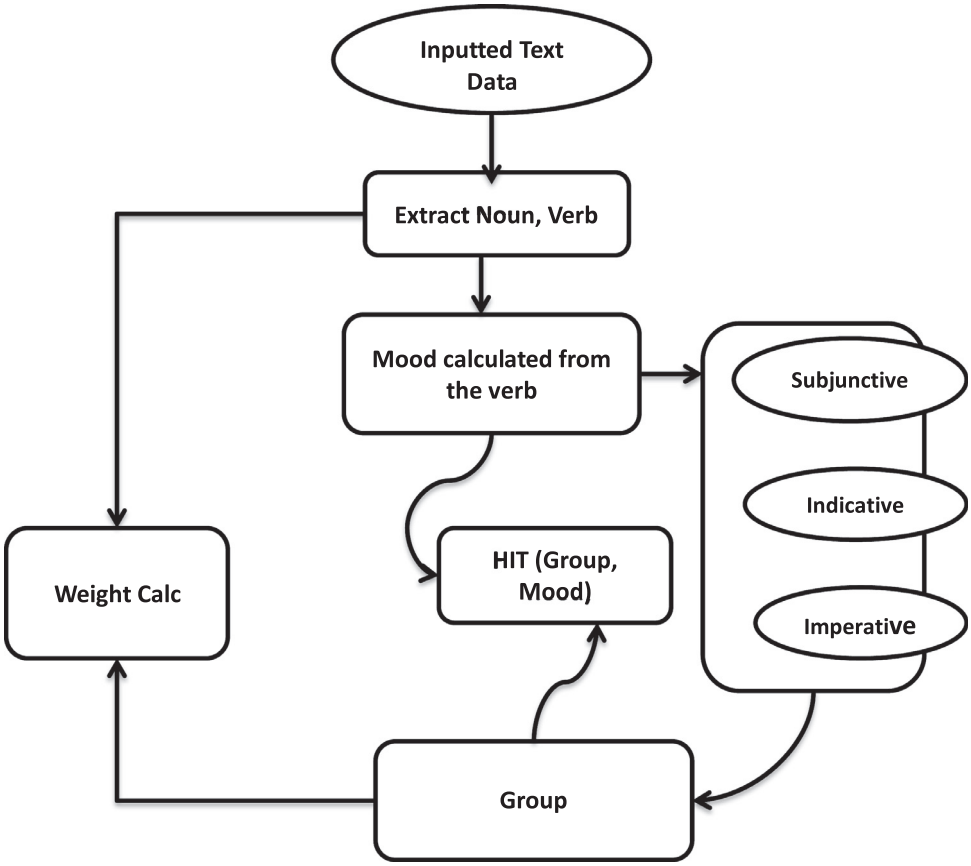

Fig. 4. Fixed features involved in the feature vector for text classification.

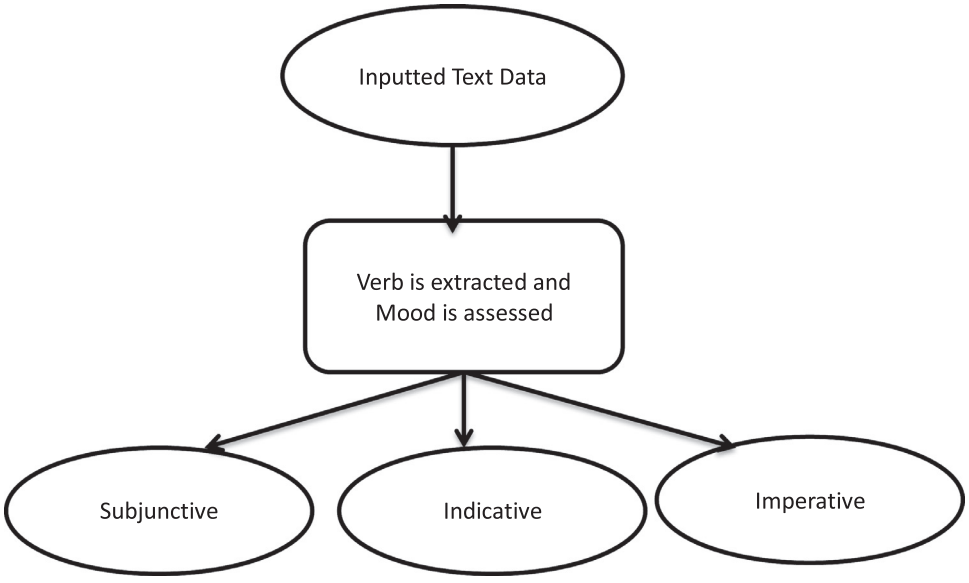

Fig. 5. Major type of moods.

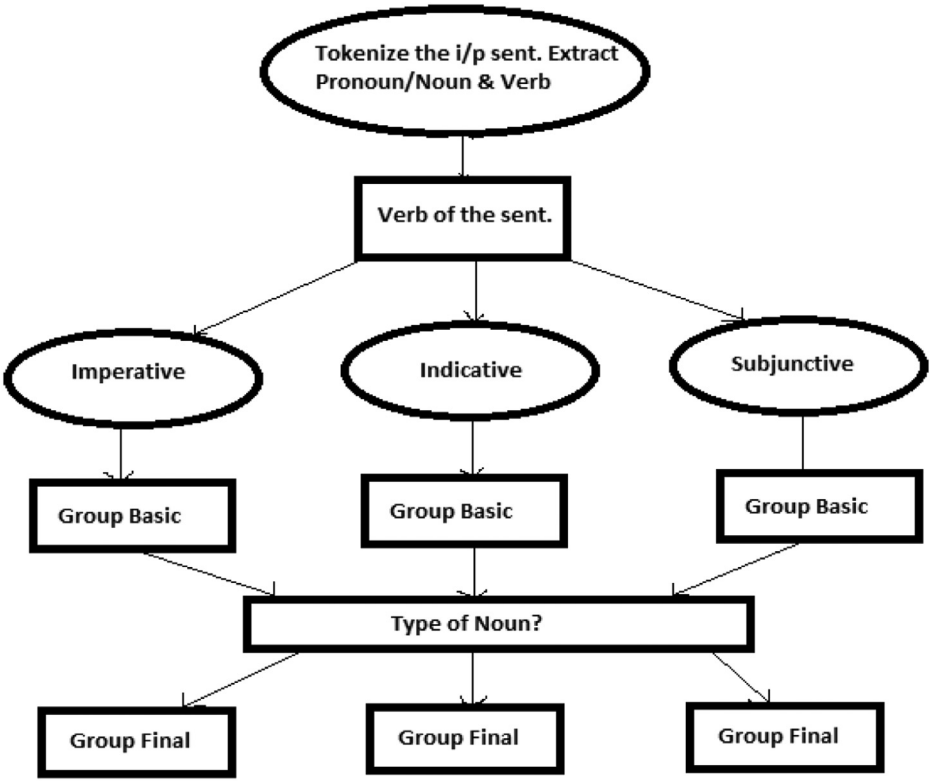

Fig. 6. Logic describing how we detect the group of the input data.

(iii) Third person activity in the input text data should contain "s" or "es".

We determine by involving tagger and tokenizing the input sentence for parts of speech. We count the nouns present and add "s" and "es" to the nouns present which exclude proper noun and personal pronoun. Now with the nouns modified we tag the sentence again and look for the nouns count. If the following condition (refer to Eq. (6)) is satisfied, mood of the input data is classified as subjunctive.

$$\text{Noun}_{(\text{Pre})} > \text{Noun}_{(\text{Post})} \tag{6}$$

If the input sentence makes a statement then the mood of the input data is classified as indicative. We mark mood of the input sentence as indicative based on two conditions,

(i) If input text data contain words that makes a statement.
(ii) Verifying whether third person verb ends with the word "s".

If the input sentence forms a command or request then the mood of the input sentence is classified as Imperative.

3.1.2. Group feature

Group feature extracts the group of the input data based on the logic described in Fig. 6.

From the input data noun and verb are extracted. Using the nature of verb the "mood" of the input sentence is identified as explained above. Based on the mood basic group is identified and the type of noun group is finalized.

3.1.3. Weight & TWeight feature

Weight calculation [9] involves two stages where at first A cosine similarity is calculated between group and noun(s) of the input sentence which is given in Eq. (7) below,

$$A = \text{Cos}(\text{Group}, \text{Noun}) \tag{7}$$

B cosine similarity is calculated between words of the input sentence and the noun which is given in the Eq. (8) below,

$$B = \text{Cos}(\text{Words}, \text{Noun}) \tag{8}$$

Final weight is calculated by dividing B by A as shown in Eq. (9).

$$W = B/A \tag{9}$$

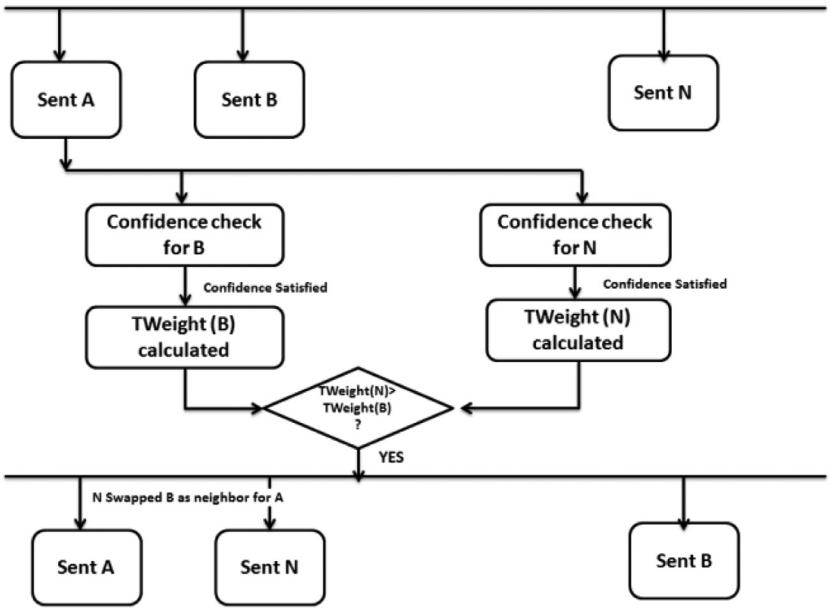

Fig. 7. Flowchart of T-rank algorithm.

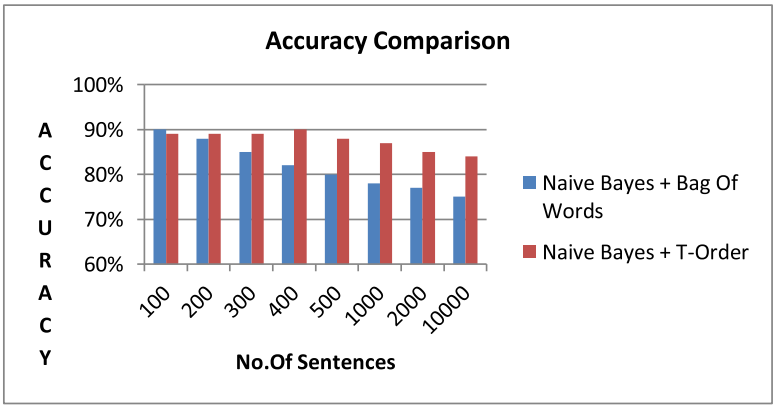

Fig. 8. Comparison of accuracy between Naive Bayes with Bag of Words and Naive Bayes with T-order.

Total weight, TW calculation involves weight calculation between group and noun of the current sentence with the group and noun of the neighbour sentence (refer Eq. (10)),

$$TW = \mathrm{Cos}\left(G_{current}N_{current}, G_{neighbour}N_{neigbour}\right) \tag{10}$$

3.1.4. Hit feature

This feature keeps track of the frequency between group and mood and plays an important part in placing the data in the feature vector for effective data retrieval.

3.1.5. T-RDF

The above explained features for vector space also acts as an efficient metadata which clearly identifies and maintains the relationships of the inputted data which can be interpreted by the classifiers efficiently for training as well as retrieving the requested data.

3.2. T(Text)-Rank algorithm for effective retrieval of data

We have extracted and placed the required features in order to process the text and here we propose a new algorithm namely T-Rank for effective retrieval of data. Inputted data are in the vector space and at frequent intervals our proposed algorithm executes and re-arranges the neighbour data. Steps of T-Rank (refer to Fig. 7) are as follows,

Table 1
Performance measure comparison between the existing feature extraction techniques and the proposed approach.

Category	Performance Measure	Bag of Words - Feature Reduction	PCA - Feature reduction	Random Forest - Feature reduction	T-Order - Feature Reduction
Naive Bayes	Accuracy	75.2	75.8	77.1	88
Decision Tree	Accuracy	81.08	80.34	82.76	90
SVM	Accuracy	85.1	84.25	82.57	86

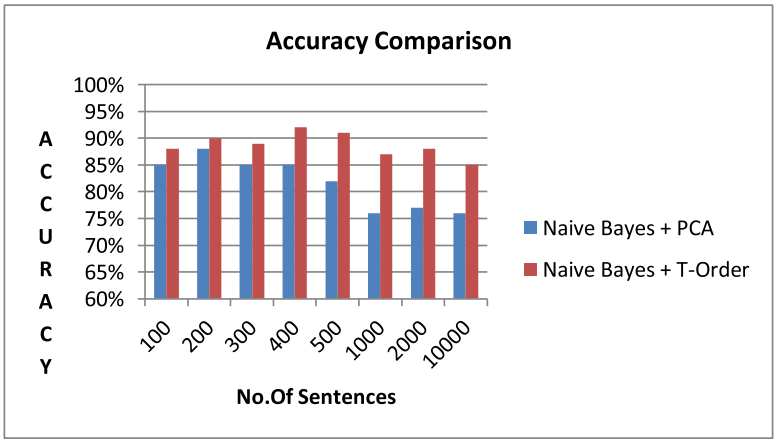

Fig. 9. Comparison of accuracy between Naive Bayes with PCA and Naive Bayes with T-order.

STEP 1: Calculate the confidence factor.

Confidence factor is calculated involving the group between the current and the neighbour data. Initial positioning of data is based on the mood and the weight (refer to Eq. (4)) with the neighbour data. Confidence is calculated at intermediate times where group count of an input data is divided by group count of the neighbouring data and if the confidence is greater than 70 (refers to Eq. (11)) then it is considered for weight calculation. The order remains the same: otherwise the position of neighbour data will be swapped with the data that have the highest confidence.

$$TC_f = \left(G_{(B)}/G_{(A)} * 100\right) > 70 \tag{11}$$

where TC_f represents Confidence factor
$G_{(B)}$ represents Group of neighbor
$G_{(A)}$ represents Group of current data

STEP 2: Calculate TWeight.

TWeight is calculated using the below Eq. (12),

$$TW_{(B)} = \mathrm{Cos}\left(G_{(A)} + N_{(A)}, G_{(B)} + N_{(B)}\right) \tag{12}$$

where Cosine similarity is calculated between the group and noun of sentence A with the group and noun of sentence B.

STEP 3: If the TWeight of n^{th} neighbor is calculated and if the weight is greater than the current neighbor B of A then N is swapped with B to be the neighbor of A. It is validated by using Eq. (13),

$$TW_{(B)} \text{ is } > TW_{(B)} \tag{13}$$

then the structure of A connects to N.

4. Results and discussion

We have used raw BBC news dataset for our experimental purpose.
Table 1 above compares the existing feature extraction techniques for feature reduction like bag of words, PCA and Random Forest with our own proposed approach for dimensionality reduction. The techniques are introduced along with classification algorithms and the accuracy is calculated. Results convey that accuracy is higher when classification algorithms along with traditional feature extraction techniques are compared with classification algorithms along with our approach (refer figure from 8–13).

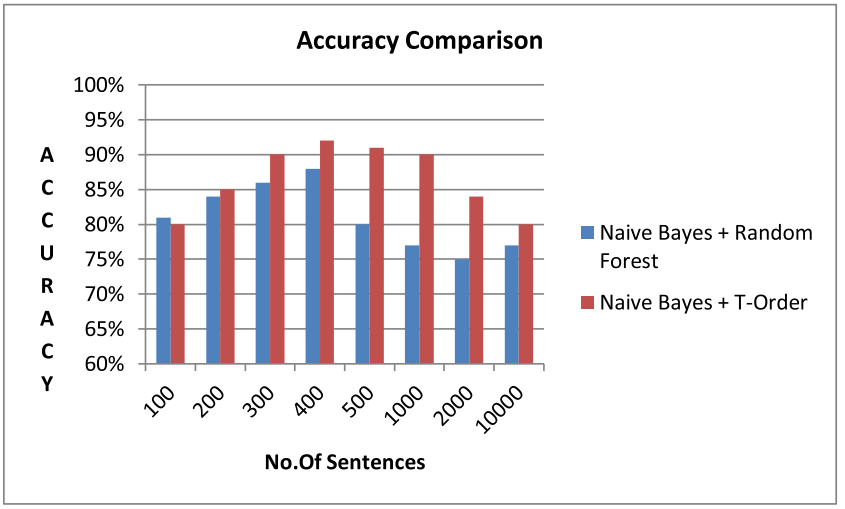

Fig. 10. Comparison of accuracy between Naive Bayes with Random Forest and Naive Bayes with T-order.

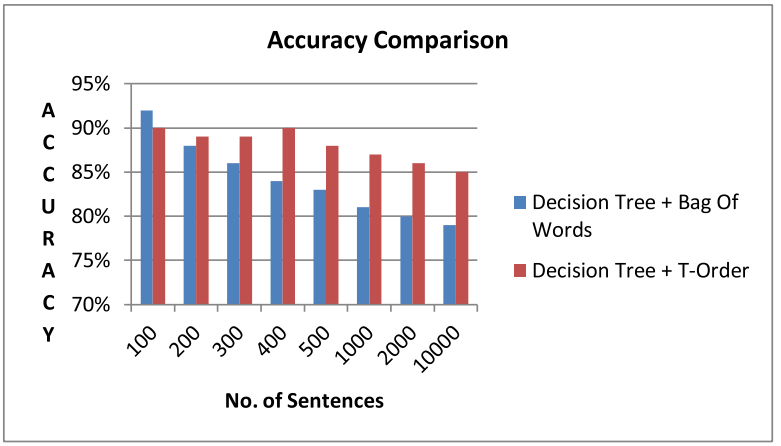

Fig. 11. Comparison of accuracy between Decision Tree with Bag of Words and Naive Bayes with T-order.

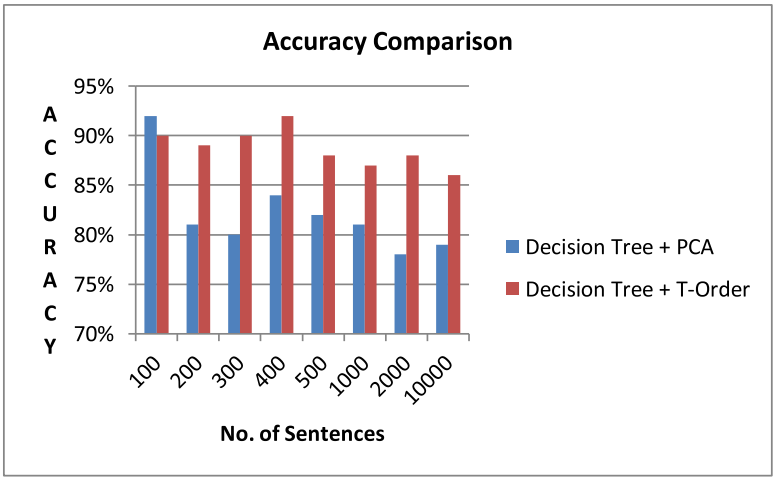

Fig. 12. Comparison of accuracy between Decision Tree with PCA and Naive Bayes with T-order.

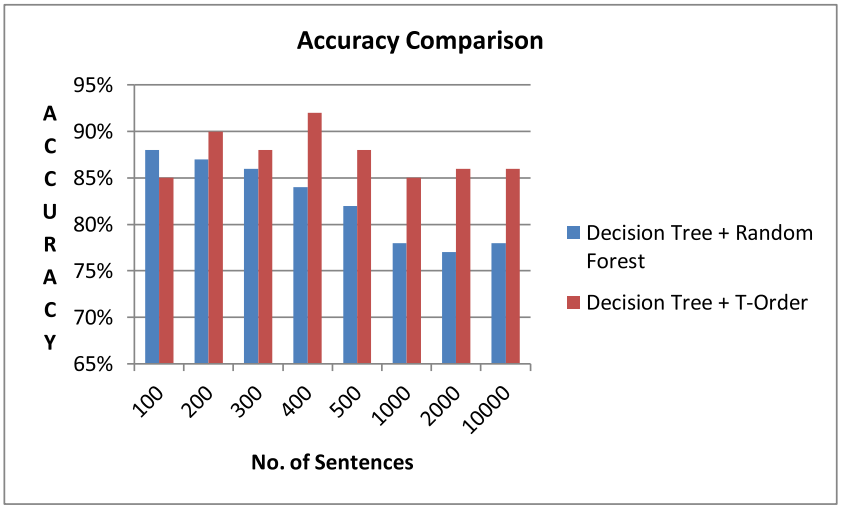

Fig. 13. Comparison of accuracy between Decision Tree with Random Forest and Naive Bayes with T-order.

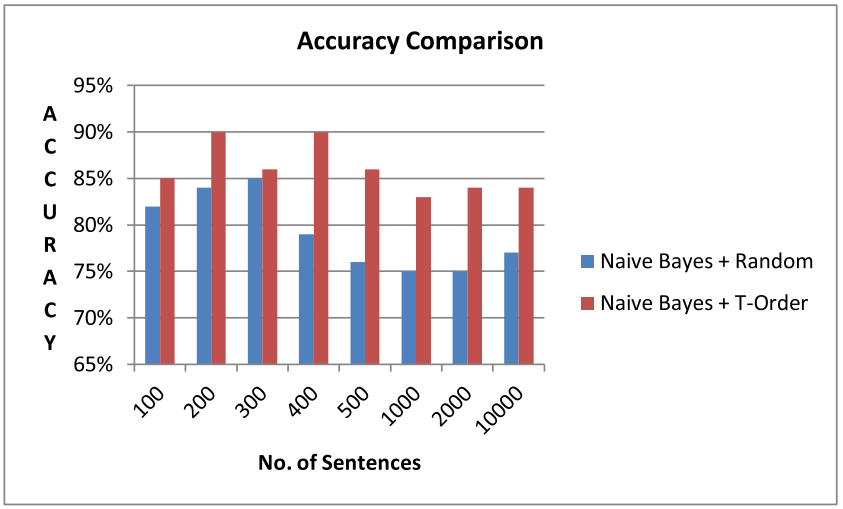

Fig. 14. Comparison of accuracy between Naive Bayes with Random and Naive Bayes with T-order.

Table 2 above compares the existing feature selection techniques for feature reduction like Radom, Sampling Measure and Clustering with our own proposed approach for dimensionality reduction. The techniques are introduced along with classification algorithms and the accuracy is calculated. Results convey that accuracy is higher when classification algorithm along with traditional feature selection techniques is compared with classification algorithms along with our approach as shown below in Figs. 14–19

The advantage of proposed approach will be huge when a greater number of training sentences are involved to train the model where the proposed algorithm maintains accuracy of higher level as well as shows improved performance for higher data set.

Table 2
Performance measure comparison between the existing feature selection techniques and the proposed approach.

Category	Performance Measure	Random	Sampling Based Technique	Clustering	T-Order - Feature Reduction
Naive Bayes	Accuracy	77	78	80	87
Decision Tree	Accuracy	75	79	82	90

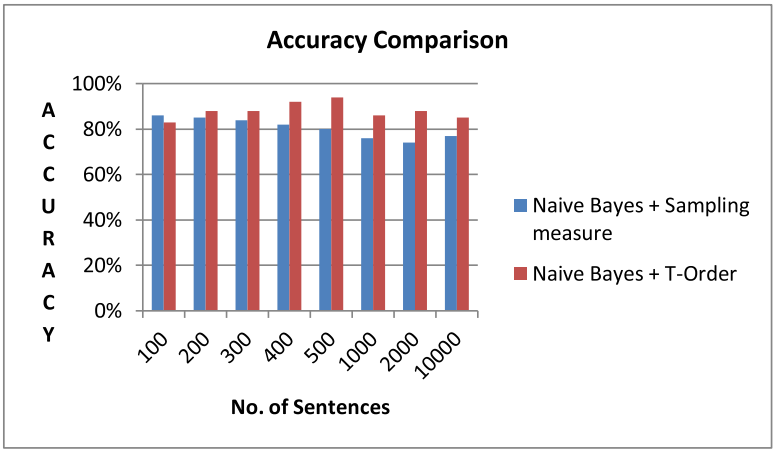

Fig. 15. Comparison of accuracy between Naive Bayes with sampling measure and Naive Bayes with T-order.

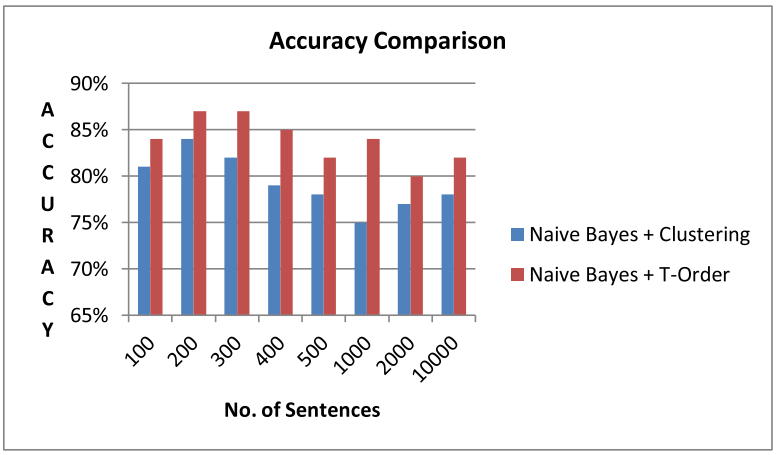

Fig. 16. Comparison of accuracy between Naive Bayes with clustering and Naive Bayes with T-order.

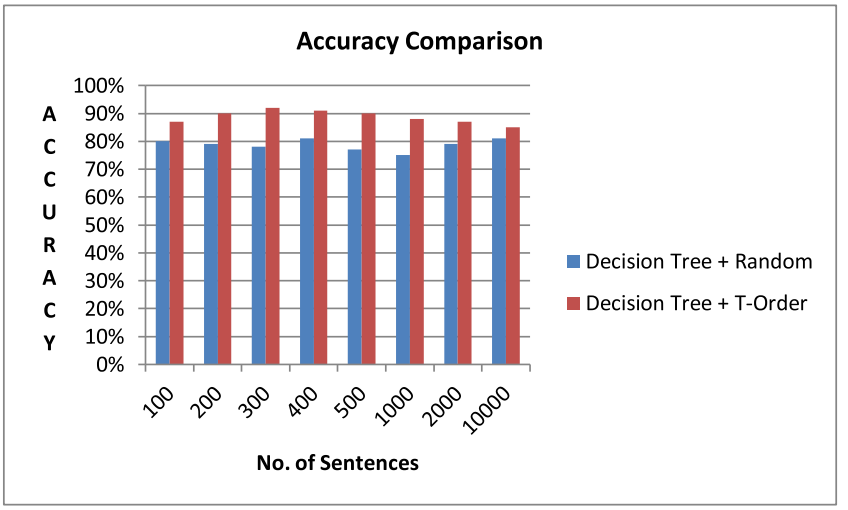

Fig. 17. Comparison of accuracy between Decision Tree with random and Naive Bayes with T-order.

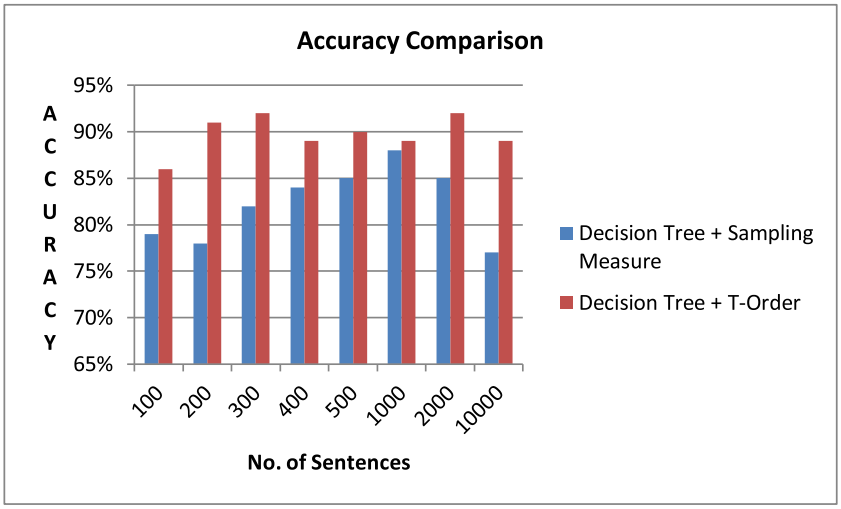

Fig. 18. Comparison of accuracy between Decision Tree with sampling measure and Naive Bayes with T-order.

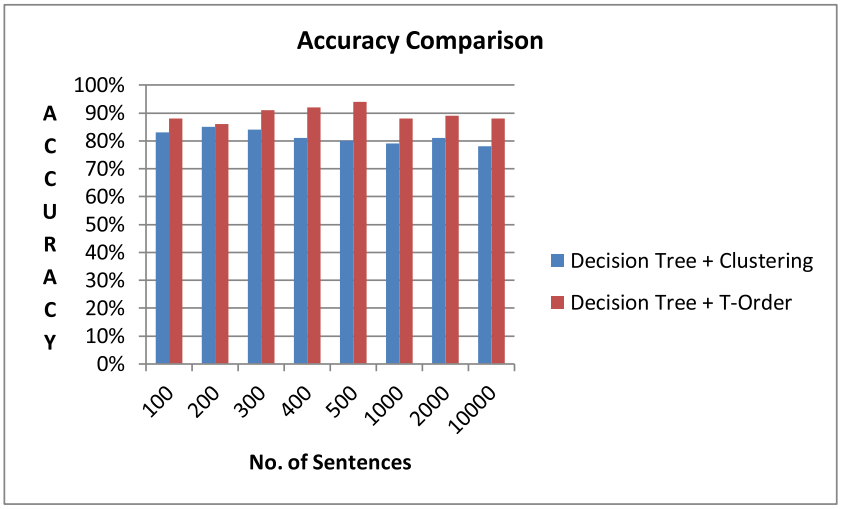

Fig. 19. Comparison of accuracy between Decision Tree with clustering and Naive Bayes with T-order.

5. Conclusion

Feature vector dimensionality reduction is gaining more importance as the demand for the extraction of user-requested data increases. Positioning and extracting the features are of particular importance in achieving this objective. The existing techniques and bag-of-words approach works well in feature reduction, but when the complexity of the data set increases in terms of its volume, a very high number of features are extracted and involved in the vector space model. This leads to increased architecture complexity, resulting in inappropriate data extraction. Principal component analysis relies on linear assumptions and orthogonal transformations, making it unsuitable for many kinds of datasets. Other approaches, such as random forests, backward feature elimination and forward feature construction involve increased time complexity. The proposed approach considers only limited features, which can handle all types of data set (reduced space complexity). It is also arranged and ranked in a new method for the effective retrieval of data, which increases the accuracy of data extraction and reduces the time complexity. The proffered method achieves the dimensionality reduction by using fixed features that effectively handle the text classification for a variety of datasets and utilize the novel algorithm T (text)-Rank for effective information retrieval. Further, the performance measurement and the comparison of the proposed approach for the category of naïve Bayes, the decision tree and the support vector machine with the existing feature extraction techniques using the raw BBC news data shows the competence of the proposed feature extraction techniques in terms of accuracy. Moreover, the simulation results obtained for the proposed approach of dimensionality reduction proves that the T (text)-Order dimensionality reduction for the category of naïve Bayes and the decision tree shows high accuracy when compared with the classification algorithms using traditional feature selection techniques.

Declaration of Competing Interest

All author states that there is no conflict of interest.

References

[1] Jingjing Cai, Jianping Li, Wei Li, Ji Wang, Deep learning model used in text classification, pp. 123–126, 2018.
[2] Elhadad Mohamed K, Badran KhaledM, Salama Gouda I. A novel approach for ontology-based dimensionality reduction for web text document classification. In: IEEE/ACIS 16th International Conference on Computer and Information Science; 2017. p. 373–8.
[3] Shah Foram P, Patel Vibha. A review on feature selection and feature extraction for text classification. In: International conference on Wireless Communications, Signal Processing and Networking; 2016. p. 2264–8.
[4] Chen J, Huang H, Tian S, Qu Y. Feature selection for text classification with Naive Bayes. Expert Syst Appl April 2009;36(3) Part 1.
[5] Liu Tinglin, Liu Jing, Liu Qinshan, Lu Hanqing. Expanded bag of words representation for object classification. In: 16th IEEE International Conference on Image Processing; 2009. p. 297–300.
[6] Pei Yan. Linear principal component discriminant analysis. In: IEEE International Conference on Systems, Man and Cybernetics; 2015. p. 2108–13.
[7] Valecha Harsh, Varma Aparna, Khare Ishita, Sachdeva Aakash, Goyal Mukta. Prediction of consumer behaviour using random forest algorithm. In: IEEE International Conference on Electrical, Electronics and Computer Engineerin; 2018. p. 1–6.
[8] Karnan M, Kalyani P. Attribute reduction using backward elimination algorithm. In: International Conference on Computational Intelligence and Computing Research; 2010. p. 1–4.
[9] Kumar, Saravana C.S.; Amudhavalli, P.; Santhosh, R.; Kalaiarasan, C., T structured semantic weight relationship algorithm combined with decision trees for data extraction, Volume 16, Number 2, February 2019, pp. 735–739, 2019.
[10] Forman G. An extensive empirical study of feature selection metrics for text classification. J Mach Learn Res 2003;3:1289–305.
[11] Dasgupta A, Drineas P, Harb B, Josifovski V, Michael WM. Feature selection methods for text classification. In: KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining; 2007. p. 230–9.
[12] Dhillon IS, Mallela S, Kumar R. A divisive information theoretic feature clustering algorithm for text classification. J Mach Learn Res 2003;3:1265–87.
[13] Mintz Mike, Bills Steven, Snow Rion, Jurafsky Dan. Distant supervision for relation extraction without labeled data. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2. Association for Computational Linguistics; 2009. p. 1003–11.
[14] Binding C, May K, Tudhope D. Semantic interoperability in archaeological datasets: data mapping and extraction via the CIDOC CRM. In: International Conference on Theory and Practice of Digital Libraries. Springer; 2008. p. 280–90.
[15] Bansal Srividya K. Towards a semantic extract-transform-load (ETL) framework for big data integration. In: 2014 IEEE International Congress on Big Data. IEEE; 2014. p. 522–9.
[16] Resch Bernd, Summa Anja, Sagl Günther, Zeile Peter, Exner Jan-Philipp. Urban emotions—Geo-semantic emotion extraction from technical sensors, human sensors and crowdsourced data. In: Progress in location-based services 2014. Springer; 2015. p. 199–212.
[17] Baumgartner R, Henze N, Herzog M. The personal publication reader: illustrating web data extraction, personalization and reasoning for the semantic web. In: European Semantic Web Conference. Springer; 2005. p. 515–30.
[18] Silva Bruno, Cardoso Jorge. Semantic data extraction for B2B integration. In: 26th IEEE International Conference on Distributed Computing Systems Workshops (ICDCSW'06). IEEE; 2006. p. 16. -16.
[19] Pueyo, Luis Garcia, Vanja Josifovski, Amitabh Saikia, Jie Yang, Mike Bendersky, Srinidhi Viswanatha, and Marc-Allen Cartright. Generating and applying data extraction templates. U.S. Patent Application 10/216,838, filed February 26, 2019.
[20] Pueyo, Luis Garcia, Vanja Josifovski, Amitabh Saikia, Jie Yang, Mike Bendersky, Srinidhi Viswanatha, and Marc-allen Cartright. "Generating and applying data extraction templates." U.S. Patent Application 15/394,610, filed February 26, 2019.
[21] Chen, Ye, Jin Zhang, and Chi Zhang. Automatic natural language processing based data extraction. U.S. Patent Application 10/204,146, filed February 12, 2019.
[22] Li W, He C, Fang J, Zheng J, Fu H, Yu L. Semantic segmentation-based building footprint extraction using very high-resolution satellite images and multi-source gis data. Remote Sens (Basel) 2019;11(4):403.
[23] Zhang Ying, Li Chaopeng, Chen Na, Liu Shaowen, Du Liming, Wang Zhuxiao, Ma Miaomiao. Semantic-Based geospatial data integration with unique. Geospatial Intelligence: Concepts, Methodologies, Tools, and Applications: Concepts, Methodologies, Tools, and Applications 2019:254.
[24] Kumar S, Amudhavalli P, Santhosh R, Kalaiarasan C. T structured semantic weight relationship algorithm combined with decision trees for data extraction. J Comput Theor Nanosci 2019;16(2):735–9.
[25] Hulsebos, Madelon, Kevin Hu, Michiel Bakker, Emanuel Zgraggen, Arvind Satyanarayan, Tim Kraska, Çağatay Demiralp, and César Hidalgo. Sherlock: a deep learning approach to semantic data type detection. (2019), arXiv:1905.10688.

C. S. Saravana Kumar received his BE in CSE from Coimbatore Institute of Engineering and Information Technology in 2008 and ME in CSE from PSG College of Technology in 2010. He is currently working as a Senior Software Engineer in an IT Firm. His current research interests include Data Mining, Text Mining, Paraphrase Identification, Semantic Web Mining, Semantic Association and Supervised Learning.

R. Santhosh received his BTech in IT from KSRCT in 2006, ME in Software Engineering from SREC in 2009 and PhD in Computer Science and Engineering from Karpagam Academy of Higher Education in 2016. He is currently working as an Associate Professor in the Department of CSE, KAHE. His current research interests include Cloud Computing and Distributed and Parallel Computing.