Topic Modeling for Text Classification

Pinaki Prasad Guha Neogi, Amit Kumar Das, Saptarsi Goswami and Joy Mustafi

Abstract Topic models allude to statistical algorithms for finding out an extensive text body's latent semantic structures. Standing here in today's world, the measure of the textual data and information we come across in our day-to-day lives is basically beyond our handling limit. Topic models can provide a way out for us to understand and manage the vast accumulations of unstructured textual data and information. Initially emerged as a text-mining instrument, topic models have found applications in various other fields. This paper makes a thorough comparative study of LSA with that of commonly used TF-IDF approach for text classification and proves that LSA yields better accuracy in classifying texts. The novelty of the paper lies in the fact that we are using a much sparser representation than usual TF-IDF and also, LSA can get from the topic if there are any synonym words. This paper proposes a method, using the concept of entropy, which further increases the accuracy of text classification.

Keywords Text classification · Topic modeling · Latent Semantic Analysis (LSA) · TF-IDF · Entropy

P. P. G. Neogi (✉)
Department of CSE, Meghnad Saha Institute of Technology, Kolkata 700150, India
e-mail: ratul.ng@gmail.com

A. K. Das · S. Goswami
A.K. Choudhury School of Information Technology, Calcutta University, Kolkata 700106, India
e-mail: amitkrdas.kol@gmail.com

S. Goswami
e-mail: saptarsi007@gmail.com

J. Mustafi
MUST Research Club, Hyderabad 500107, India
e-mail: mustafi.joy@live.com

J. K. Mandal and D. Bhattacharya (eds.), *Emerging Technology in Modelling and Graphics*, Advances in Intelligent Systems and Computing 937,
https://doi.org/10.1007/978-981-13-7403-6_36

1 Introduction

We live and experience a daily reality where floods of information are generated relentlessly. Therefore, hunting down intuitions from the gathered data can turn out to be exceptionally monotonous and tedious. News can be considered as a contemporary observer of the society and can reveal to us a great deal about things that went right or off-base. However, the quantity of news articles being delivered on the planet even in a single day is remarkably prodigious and this unthinkably vast quantity of data is beyond manual or human processing capability. As indicated by Chartbeat, every 24 h, more than 92,000 articles are published on the web. But, of course, utilization of computers can be made to segregate helpful data from news writings, for example, names of individuals, associations, geological locations, political gatherings and basically any word that has got a meaningful denotation in Wikipedia. However, it is extremely precarious for a computer to nail down "What is this article about?" in a couple of words, simply like any human being would answer naturally. Topic modeling is a technique of employing computers for this labyrinthine task. Topic modeling was delineated as an instrument for organizing, comprehending and searching for tremendous amounts of textual data. Thus, the idea is to make application of computational categorization techniques for answering the inquiry "What is this news article about?" on a substantial scale. Fortunately for us, the BBC is always ready with hundreds of fresh news every minute, so data are definitely not a limiting factor.

In machine learning, topic model can thus be particularly elucidated as a natural language processing tool, used for detecting concealed semantic structures of textual information in an accumulation of textual records, termed as corpus. Usually, each record alludes to a continuous arrangement of words, similar to an article or a passage, where each passage or article comprises an arrangement of words. Topic modeling is basically an unsupervised learning algorithm to deal with clumping of textual reports, by finding out topics based on the analysis of their substance. Its fundamental working concept is much analogous to that of expectation-maximization work and K-means algorithm. Since we are clumping documents, in order to find out topics, we will have to take into consideration and process every single word in the document and allocate values to each depending on its distribution. This swells the volume of information we are working with and thus to handle the complex processing necessitated for grouping documents, we should have to make utilization of well-organized sparse data structures.

This paper is organized as follows: Sect. 2 gives a brief overview of the related works in the fields we are focusing upon; Sect. 3 narrates the TF-IDF approach for classifying text documents; Sect. 4 gives a concise account of Latent Semantic Analysis (LSA); Sect. 5 explains the concept of entropy in a text document; the proposed method is described under Sect. 6, including the algorithm (Sect. 6.3); Sect. 7 gives a detailed account of the dataset and coding environment; experimental result, along with comparison tables and confusion matrices, is presented under Sect. 8; and lastly, the conclusion is given in Sect. 9.

2 Related Works

Zelikovitz [1] applied the semantic analysis algorithms [2–4] for the purpose of classifying the short texts. A progression of latent semantic allocation (LSA) for the classification of short texts is the Transductive LSA. Transduction makes the utilization of the test examples for the purpose of selecting the hypothesis of the trainee to settle on decisions contrapose the test cases. Pu et al. [2] amalgamated independent component analysis (ICA) and latent semantic allocation (LSA) together. A large-scale classification framework of short text documents was established by Phan et al. [5], which is essentially in light of machine learning methods such as SVMs and maximum entropy and that of latent dirichlet allocation (LDA). In any case, their work primarily centered on how to apply it to Wikipedia and no intuition was given on if there exists a different approach to train the same model. Generally, what happens in case of web search is that, the search engine gets employed directly in this line of research. For instance, a kernel function was proposed by Sahami et al. [6] in light of search engine outcomes. The method was extended even more by the application of some machine learning algorithm by Yih et al. [7].

LDA was stretched out by Ramage et al. [8] to a supervised form and its applications were analyzed in micro-blogging environment. A strategy in light of labeled LDA was built up by Denial Ramage et al. for multi-labeled corpora [9].

In [10], using Naïve Bayes algorithm for text classification, a novel approach has been proposed for feature selection. In light of latent dirichlet allocation for topic extraction from source code, an approach was proposed by Maskeri et al. [11]. A technique in text mining, based on partly or incompletely labeled topics was proposed by Manning et al. [12]. In these models, implementation of unsupervised learning algorithms is made for topic models so as to find out the unrevealed topics within every single label, as well as unlabeled latent topics. A semi-supervised hierarchical topic model (SSHLDA) has been proposed in [13], where the newer topics are spontaneously expected to get explored.

3 TF-IDF for Text Classification

In the process of data retrieval, term frequency–inverse document frequency (TF-IDF) is a numerical approach to indicate the importance of any word in a document present in a corpus [14]. A rise in the TF-IDF value corresponding to a specific word is observed with the increase in the frequency of that word in the text document and is offset by the word frequency of the corpus. This adjusts the fact that the frequency of few words in a document is much higher compared to others.

On account of the term recurrence $tf(t, d)$, the most commonly utilized approach is to utilize the raw count of a term t in a document d, i.e., the number of occurrences of the term t in the given document d. If the raw count is denoted as $f_{t,d}$, then the most straightforward tf scheme is $tf(t, d) = f_{t,d}$. The inverse document frequency or IDF

provides a measure of the amount of information retrievable from a specific word in a document, regardless of whether the term is rare or common over all documents. It is the logarithmically scaled inverse fraction of the records containing the term, acquired by dividing the total quantity of reports by the quantity of reports containing the term, and afterward taking the logarithm of the quotient thus obtained.

$$idf(t, D) = \log[N \div |\{d \in D : t \in d\}|] \tag{1}$$

where N denotes the total quantity of documents in the corpus, $N = |D|$ and $|\{d \in D : t \in d\}|$ refers to the quantity of documents where the term t is present which means $tf(t, d) \neq 0$. If the term is missing from the corpus, the consequence will be a division by zero. Thus, the denominator is modified as $1 + |\{d \in D : t \in d\}|$.

Then, TF-IDF is computed as

$$\text{TF-IDF}(t, d, D) = \text{TF}(t, d) * \text{IDF}(t, D) \tag{2}$$

4 Latent Semantic Analysis (LSA)

To discover a topic in a gathering of textual files and documents, different techniques are utilized. Topic modeling algorithms, for the most part, are used to build up a model for browsing, searching and outlining extensive corpus of writings. Issue to consider being given a huge arrangement of messages, news articles, diary papers, and reports are to comprehension of the key data contained in set of records. The ultimate objective is to understand the key information contained in a huge corpus of news articles, emails, journal papers, etc. and keeping aside the unnecessary detailing. To extricate topics from huge corpus, various generative models are utilized for topic modeling, of which LSA is a frequently used one.

In natural language processing, specifically distributional semantics, Latent Semantic Analysis (LSA) is a way of examining the connections between an arrangement of documents and the terms contained by them by creating an arrangement of ideas associated with the records and terms. In case of LSA, it is assumed that words those are near in their meaning appear in related pieces of texts (the distributional hypothesis). A matrix that keeps record of the word count in each paragraph (each of the paragraphs is represented by the columns and the rows represent the unique words) is built from a vast section of text and in order to minimize the row count, a mathematical approach called singular value decomposition or SVD (in which a term-by-document matrix, say X, is decomposed into three other matrices W, S and P, such that when multiplied together, they give back the matrix X with $\{X\} = \{W\}\ \{S\}\ \{P\}$. Refer to Fig. 1) is implemented, while conserving the comparability structure among columns.

Then, comparison between the words is made by taking into consideration the cosine of the angles between any two vectors formed by rows (or by taking the dot

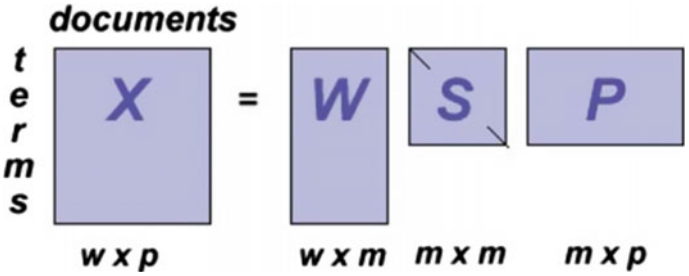

Fig. 1 Solitary Value Decomposition (SVD) of Latent Semantic Analysis (LSA)

product of the normalizations of the two vectors). If the value is near about 0, it means that the words are dissimilar, whereas if the value is near about 1, it means they are very much similar.

5 Entropy of Text Document

Specific conditions are required to be met for legitimate estimation of entropy of which ergodicity and stationarity are typical. Stationarity implies the fact that the statistical properties of a collection of words are totally independent of the position of the words in the text sequence, whereas ergodicity implies that the average properties of the troupe of all conceivable text sequences are matched with the statistical properties of an adequately large text series (e.g., [15–17]). Fitness is another such condition (i.e., a corpus of text documents has a limited arrangement of word types). While, in research articles, ergodicity and stationarity have a tendency to be exhibited and examined together, fitness is generally expressed independently, ordinarily as a component of the underlying setting of the issue (e.g., [15, 17]). A fourth general condition is the supposition that types are independent and identically distributed (i.i.d.) [15, 17, 18]. Consequently, for legitimate entropy estimation the normal prerequisites are either (a) ergodicity, stationarity and fitness, or (b) just fitness and i.i.d., as ergodicity and stationarity take after inconsequentially from the i.i.d. supposition.

For entropy estimation, a vital pre-requisite is the guess of the probabilities of word types. In a text document, each word of the types w_i has a token frequency given by f_i= freq (w_i). The probability of a word w_i, say $p(w_i)$ can be estimated by the so-called maximum likelihood method as

$$\hat{p}(wi) = \frac{f_i}{\sum_{j=1}^{V} f_i} \tag{3}$$

where the numerator denotes the frequency of the word type w_i taken into consideration (denoted by f_i) while the denominator denotes the summation of the frequencies

of each word type over an empirical vocabulary having size of $V = |\mathcal{V}|$. This gives the probability of a word type w_i.

It is assumed that a text is an arbitrary variable T, which is formed by method of drawing and concatenating tokens from a vocabulary of word types $\mathcal{V} = \{w_1, w_2, \ldots, w_W\}$, where the letter W represents the theoretical size of the vocabulary set and for every $w \in \mathcal{V}$, a probability mass function is obtained by $p(w) = Pr\{T = w\}$. From all these, the theoretical entropy T can be computed using the following relation [19]:

$$H(T) = -\sum_{i=1}^{W} p(w_i).\log_2 p(w_i) \tag{4}$$

where the term $p(w_i)$ presents the probability of the word type w_i, such that the summation of the probabilities of every word type is 1, i.e., $\sum_{i=1}^{W} p(w_i) = 1$. As seen from Eq. (4), each of the probability term is being multiplied with the logarithmic term $\log_2 p(w_i)$ according to Shannon's Entropy Equation. The resultant outcome that is obtained is negative and that is why the preceding $(-)$ sign is used so as to make the final outcome positive. For this situation, $H(T)$ can be viewed as the average data substance of word types. A pivotal walk toward assessing $H(T)$ is to dependably estimated the probabilities of word types $p(w_i)$.

6 Proposed Method

6.1 Text Classification Using TF-IDF Versus Text Classification Using Topic Modeling

TF-IDF can be utilized as attributes in a supervised learning setting (i.e., depicting the data/information of a word in a record relating to some suitable outcome) whereas topic modeling is generally an unsupervised learning problem (basically endeavoring to comprehend the topics of a corpus). One noteworthy contrast is that TF-IDF is at the word level, so a textual report that is about *car* may be afar from a report about *tire* when TF-IDF is utilized to represent them. On the contrary, since the words *car* and *tire* very often appear simultaneously in articles, they are likely to arrive from the same topic and as a result the topic modeling portrayal of these reports would be close. In case of TF-IDF, though the matrix has number of rows equal to the number of documents in the corpus set and number of columns equal to the size of the vocabulary set, most of the cells in the matrix are empty. In our case, the sparsity i.e., percentage of non-zero cells is around 2.37%. That means remaining 97.63% of the cells is zero, i.e., contains no information. Whereas in case of topic modeling (say LSA), the number of columns is drastically reduced (the number of columns is equal to the number of categories or topics in the corpus set). So, from all

these, we can infer that topic modeling is supposed to give better text classification accuracy, compared to TF-IDF. We made use of Latent Semantic Analysis (LSA) in our experiment.

6.2 Human Intervention in Deciding Categories of Document Having Very High Entropy Value

The root of the word entropy is in the Greek *entropia*, which signifies "a moving in the direction of" or "change." Up in the year 1868, the word was utilized to depict the estimation of disorder by the German physicist Rudolph Clausius. It basically refers to a numerical measure of the uncertainty of an outcome. Sometimes, it happens that a single textual document has almost equal inclination to more than one topic categories because the document contains words from different topics in almost equal proportions.

For example, suppose an illegal or criminal activity takes place in the educational sector. Now, a textual document related to this contains keywords related to both "Crime" and "Educational Sector" and sometimes it becomes difficult for machines to identify the most appropriate category and ends up giving unexpected outcomes. Such documents generally have a very high textual entropy value and hence, if the textual entropy of a specific document is beyond a certain threshold value (in our experiment, the threshold value is taken as 3.80), then human assistance is asked by the machine to help it decide the most appropriate category for the document. And moreover, text classification is very subjective, very difficult to match the classification done by human beings with that of the models. However, getting these tremendous volumes of textual data classified by humans is not possible. Here, we only route those observations which have a high uncertainty associated with the decision.

For instance, let us examine one such text document from the dataset of this experiment "*M. Aswini of Madhuravoyal, a first year B.Com student at the Meenakshi Academy of Higher Education and Research was brutally murdered near her college in K.K. Nagar on Friday. According to the police report, the murderer had been stalking Aswini, asking her to marry him.*" Here, we can see that the document has words belonging to Topic 9 (Education Sector) like "*B.Com,*" "*student,*" "*Education,*" "*Research,*" "*college*", etc. and at the same time has words like "*brutally,*" "*murdered,*" "*police,*" "*murderer*", etc. that fit in Topic 0 (Crime). In spite of this being a clear report of crime (Topic 0), in order to describe the identity and whereabouts of the victim, references to education sector (Topic 9) is made. But for machine, it is rather hard to identify the most appropriate category and ended up giving unexpected outcome. When this very document was tested with only LSA model, 4 out of 10 times the outcome says that it belongs to Topic 9 and 6 out of 10 times the outcome says that it belongs to Topic 0. And when the entropy of the document is computed it is found to be 3.85 which is quite high indicating that the document has

high uncertainty. Hence, under such circumstances, the best way to classify the document under the most appropriate category is to ask for human decision. And using human intervention for documents having very high entropy improves the accuracy as every time such ambiguity is faced, human decision is taken into account and the category decided by human beings are taken as the ultimate category.

6.3 Proposed Algorithm

Input: Train Data and test Data
Output: Labels of Test Data

```
Step 1: Represent the documents in terms of topic
coverage distribution after applying LSA.
Step 2: The dataset is being split into test and train
set and the model is trained using the train set data
consisting of 80% of the data elements.
Step 3: For a test set textual document, predict label,
say 'P' (the dominant topic).
Step 4: For the test set textual document, compute the
entropy.
Step 5:      a) If the entropy value is beyond a
threshold value then human intervention is asked by the
machine to help it deciding the most appropriate category.
             b) Else, no human intervention is required and
the predicted label 'P' (as obtained in Step 3) is taken
into consideration.
```

The proposed algorithm is diagrammatically depicted in Fig. 2.

7 Dataset and Coding Environment

The dataset that is being used in this experiment is a self-made dataset of 870 news articles belonging to 10 different news categories (accidents, business, crime, education sector, entertainment, health and medicine, politics, science and technology, sports, travel and tourism). The corpus set consists of news reports of various different fields from across India, (news sources—The Times of India, The Indian Express and www.mid-day.com). The number of documents under each category is taken in varying proportions so as to study the effect of the number of documents in determining the accuracy in text classification. For example, under the category "Crime" we have taken 128 sample documents in our dataset, whereas the category "Business" has only 65 sample documents. All other categories contain number of documents in between 65 and 128. Statistical representation of the dataset, with different topics and their distribution, is depicted in Fig. 3.

The code has been generated using Python programming language. Various libraries of Python have found applications in this code, including *scikit-learn* and its

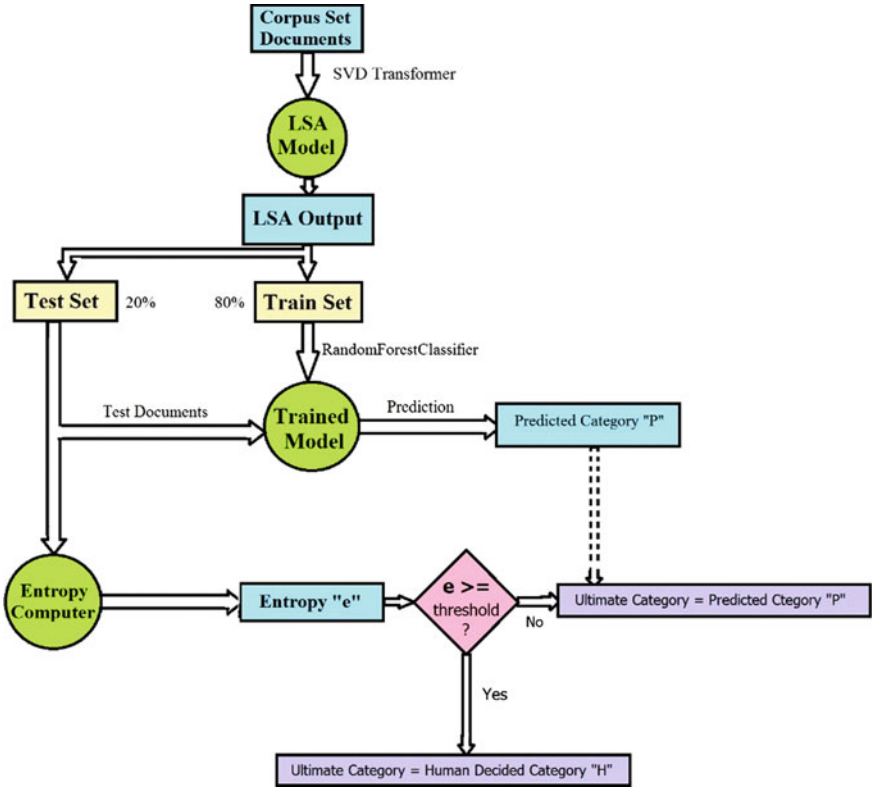

Fig. 2 Diagrammatic representation of the algorithm being proposed in this paper

various sub-libraries, *numpy*, *pandas*, *matplotlib.pyplot*, *re*, *spacy*, *keras*, *seaborn,* etc. Random Forest Classifier is the classification algorithm that has been used for this experiment for training the model based on the training set (80%) and the accuracy is then checked on the test set (20%). Random Forest Classifier helps to conquer numerous problems associated with many other classification algorithms (like that of decision trees), including Reduction in over-fitting (there is an impressively lower risk of over-fitting due to the reason that average of several trees are being taken into consideration) and less variance (By utilizing numerous trees, the likelihood of stumbling across a classifier that performs poorly is minimized due to the connection between the test and the train data.). Besides, it also has high accuracy and runs efficiently on large data sets.

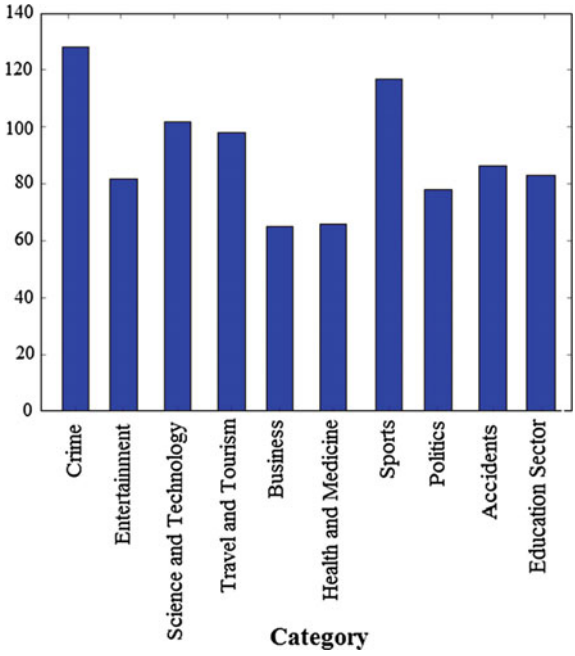

Fig. 3 Statistical representation of the dataset, with different topics and their distribution

8 Experimental Result

The proposed algorithm is given Sect. 6.3. At the beginning, the topic distribution (in percentage) of each document is presented in the form of a matrix using the Latent Semantic Analysis. After training the classification model (Random Forest Classifier) with 80% of the sample data, a prediction is made for the test set (20%) and at the same time, entropy of the text document is also computed. If the entropy value is beyond a threshold value, then human intervention is asked by the machine to help it deciding the most appropriate category. Else, the predicted category is taken as the outcome. Accuracy is calculated by dividing the number of documents correctly predicted by the model on the test set by the actual number of documents in that very category in the test set. The average accuracy obtained in case of TF-IDF text classification is around 65.4%. Whereas using Latent Semantic Analysis (LSA), the accuracy shows a drastic improvement (around 78.8%), which is further enhanced implementing the concept of entropy. When the entropy of the text is greater than or equal to 3.8, then based on human decision, the categories are chosen for better accuracy and surety. This gives an accuracy of around 87.6%. The comparison of the topic wise accuracies in all the three cases is depicted in Table 1. From Table 1, we can see that the topic wise, as well as the average accuracy in case of LSA is more as compared to TF-IDF, which is even increased implementing the concept of entropy. From Table 1, we can also see that Topic 0 having 128 sample documents has higher accuracy compared to that of Topic 4 which has only 65 sample documents. This

Table 1 Accuracy of text classification using TF-IDF, LSA and LSA with human decision based on entropy value

Topics	Accuracy		
	TF-IDF	LSA	LSA (with human decision based on entropy value)
Topic 0	0.75	0.93	1.00
Topic 1	0.68	0.75	0.81
Topic 2	0.84	0.89	0.95
Topic 3	0.80	0.86	0.93
Topic 4	0.28	0.5	0.64
Topic 5	0.47	0.68	0.79
Topic 6	0.70	0.9	1.00
Topic 7	0.70	0.82	0.88
Topic 8	0.66	0.83	0.87
Topic 9	0.66	0.72	0.89
Average	0.654	0.788	0.876

Fig. 4 Confusion matrix of text classification using TF-IDF

also gives a hint toward the fact that increasing the number of documents leads to accuracy gains for LSA models. The confusion matrices of text classification using term frequency–inverse document frequency (TF-IDF), Latent Semantic Analysis (LSA) and that of LSA with human decision based on entropy value are shown in Figs. 4 and 5a, b, respectively.

9 Conclusion

The proposed method suggested in this paper presented the utilization of entropy in enhancing the accuracy of text classification. In this paper, Latent Semantic Anal-

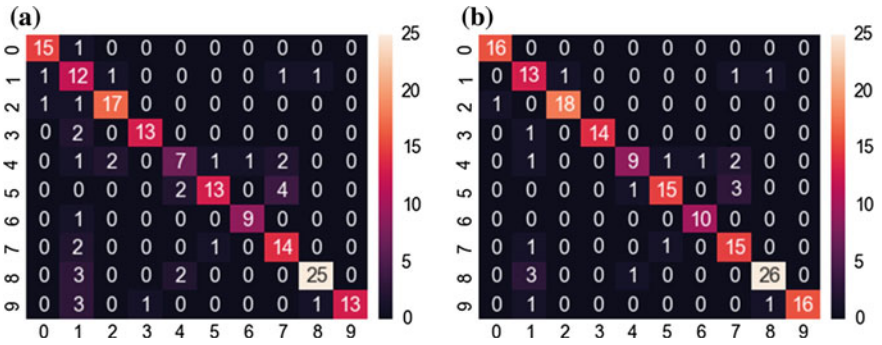

Fig. 5 **a** Confusion matrix of text classification using LSA, **b** Confusion matrix of text classification using LSA with human decision based on entropy value (the proposed method)

ysis has been utilized for text classification, which uses a much lower-dimensional representation of documents and words than usual TF-IDF. Though the number of rows in case of both TF-IDF and LSA remains the same, being the number of textual documents in the corpus, the number of columns is huge in case of TF-IDF (equal to the size of the vocabulary set), whereas in case of LSA it is equal to only the number of topics/categories. Additionally, in spite of having so many columns, most of the cells in case of TF-IDF are empty; in case of our dataset, the sparsity is only 2.37%, remaining 97.63% cells are empty (for TF-IDF). And moreover, text classification is very subjective, very difficult to attain the same accuracy for models as that of the classification done by human beings. However, getting these tremendous volumes of textual data classified by humans is also impractical. Here, only those documents are sent for human assessment which has a high uncertainty associated with the machine's decision, thus boosting up the accuracy in text classification.

Acknowledgements We gratefully acknowledge the contribution of MUST Research Club and CU Data Science Group. MUST Research Club is a non-profit organization registered under Society Act of India. MUST Research Club is dedicated to promote excellence and competence in the field of data science, cognitive computing, artificial intelligence, machine learning and advanced analytics for the benefit of the society. Calcutta University Data Science Group is a group to create solutions to solve societal problems, as well as form generic solutions to common data science and engineering issue. This is a forum which brings together researchers, industry practitioners, scholars, interns and form groups.

References

1. S. Zelikovitz, Transductive LSI for short text classification problems, in *Proceedings of the 17th International Flairs Conference* (2004)
2. Q. Pu, G. Yang, Short-text classification based on ICA and LSA, in *Advances in Neural Networks ISNN 2006* (2006), pp. 265–270

3. S. Zelikovitz, H. Hirsh, Using LSI for text classification in the presence of background text, in *Proceedings of 10th International Conference on Information and Knowledge Management* (2001), pp. 113–118
4. B. Wang, Y. Huang, W. Yang, X. Li, Short text classification based on strong feature thesaurus. J. Zhejiang Univ. Sci. C **13**(9), 649–659 (2012)
5. X. Phan, L. Nguyen, S. Horiguchi, Learning to classify short and sparse text & web with hidden topics from large-scale data collections, in *IW3C2* (2008)
6. M. Sahami, T.D. Heilman, A web-based kernel function for measuring the similarity of short text snippets, in *WWW '06: Proceedings of the 15th International Conference on World Wide Web* (2006), pp. 377–386
7. W.-T. Yih, C. Meek, Improving similarity measures for short segments of text, in *AAAI'07: Proceedings of the 22nd National Conference on Artificial Intelligence* (2007), pp. 1489–1494
8. D. Ramage, D. Hall, R. Nallapati, C.D. Manning, Labeled LDA: a supervised topic model for credit attribution in multi-labeled corpora, in *EMNLP '09: Proceedings of the Conference on Empirical Methods in Natural Language Processing* (Association for Computational Linguistics, 2009), pp. 248–256
9. D. Ramage, D. Hall, R. Nallapati, C.D. Manning, Labeled LDA: a supervised topic model for credit attribution in multilabeled corpora, in *Proceeding of the 2009 Conference on Empirical Methods in Natural Language Processing* (2009)
10. S. Dey Sarkar, S. Goswami, A. Agarwal, J. Aktar, A novel feature selection technique for text classification using Naïve Bayes. Int. Sch. Res. Not. **2014**, Article ID 717092. https://doi.org/10.1155/2014/717092
11. G. Maskeri, S. Sarkar, K. Heafield, *Mining Business Topics in Source Code using Latent Dirichlet Allocation* (ACM, 2008)
12. D. Ramage, C.D. Manning, S. Dumais, Partially labeled topic models for interpretable text mining, San Diego, California, USA (2011)
13. D. Ramage, E. Rosen, Stanford Topic modelling Toolbox, Dec 2011. [Online]. Available: http://nlp.stanford.edu/software/tmt/tmt-0.4
14. A. Rajaraman, J.D. Ullman, Data Mining. Mining of Massive Datasets (PDF) pp. 1–17 (2011). https://doi.org/10.1017/cbo9781139058452.002. ISBN 978-1-139-05845-2
15. Ł. Dębowski, Consistency of the plug-in estimator of the entropy rate for ergodic processes, in *Proceedings of the 2016 IEEE International Symposium on Information Theory (ISIT)*, Barcelona, Spain 10–15 July 2016, pp. 1651–1655
16. J. Jiao, K. Venkat, Y. Han, T. Weissman, Minimax estimation of functionals of discrete distributions. IEEE Trans. Inf. Theory **61**, 2835–2885 (2015)
17. A. Lesne, J.L. Blanc, L. Pezard, Entropy estimation of very short symbolic sequences. Phys. Rev. E **79**, 046208 (2009)
18. G.P. Basharin, On a statistical estimate for the entropy of a sequence of independent random variables. Theory Probab. Appl. **4**, 333–336 (1959)
19. C.E. Shannon, W. Weaver, *The Mathematical Theory of Communication* (The University of Illinois Press, Urbana, IL, 1949)