Automated Genre Classification of Books Using Machine Learning and Natural Language Processing

Conference Paper · January 2019

DOI: 10.1109/CONFLUENCE.2019.8776935

Shikha Gupta
Netaji Subhas Institute of Technology
8 PUBLICATIONS **20** CITATIONS
SEE PROFILE

Mohit Agarwal
University
16 PUBLICATIONS **204** CITATIONS
SEE PROFILE

Automated Genre Classification of Books Using Machine Learning and Natural Language Processing

Shikha Gupta
Computer Engineering Department
NSIT
(Delhi University)
Delhi, India
shikha.gpt1@gmail.com

Satbir Jain
Computer Engineering Department
NSIT
(Delhi University)
Delhi, India
jain_satbir@yahoo.com

Mohit Agarwal
Department of Physics and Computer Science
Dyalbagh Education Institute
Agra,India
mail2mohitag@gmail.com

Abstract— **In today's world due to ever-increasing demand to make computers perform tasks of humans, machine learning is used. It is a tedious task to manually read the entire book and classify it based on its genre. Novice writers find it troublesome to figure out the genre of their book, which can affect its reach to the right audience. The proposed method gains knowledge from a large number of words from the books and transforms them into a feature matrix. During transformation, the size of the initial matrix is reduced using Wordnet and Principle Component Analysis. Then, the AdaBoost classifier is applied to predict the genres of the books.**

Keywords— Natural Language Processing, Machine Learning, Genre Classification, WordNet, TF-IDF, Decision Tree, AdaBoost, Principle Component Analysis (PCA).

I. INTRODUCTION

Machine Learning [16] is the core of categorization, which can find many applications ranging from personalized articles [2] of the web to classifying books into different categories. A problem with categorizing books which contains a huge amount of words. If one wants to get information from this data, then one must reduce the dimensions [3] of feature matrix by clustering the related features; otherwise, the model will suffer from the curse of dimensionality. Additionally, in a real environment, the amount of labeled data available is limited.

In the proposed approach, similar words are clustered into the same class using Natural Language Processing [19] and Principle Component Analysis [4] that reduces the number of features [5] on which machine learning would work. Unlabeled and labeled data is used to understand the structure of data and used before the training of machine learning model. Labeled data generated from those books whose genre is known and unlabeled data generated from books whose genre is not known. Therefore, labeled data used in training, and the addition of unlabeled data to it improves the accuracy of the model. In this paper, genre classification is scaled up for documents from few to thousands of lines; thus, elevating it to a new, practical level.

The rest of the paper comprises of following sections: Section 2, discusses the prior work along with Section 3, that describes the proposed approach, followed with Section 4, in which we discuss the results and performance of machine learning model. Finally, the conclusion is given in Section 5.

II. PRIOR WORK

In [5] Lazaro S.P. Busagala created a lexicon with all the different words in the text and a vector with a count of frequencies of lexicon words in the text. Term selection and Principle Component Analysis does dimensionality of the feature vector. Various Discriminant Functions used for experimentation includes support vector machines. The main drawback of this approach is that the feature vector generated is very large which will, in turn, affect the performance of the machine learning model.

In [9] Prafulla Bafna presented an approach for text classification. This approach removes stop words from the text and TF-IDF is applied to normalize the data. Agglomerative clustering and fuzzy k means are applied to get the clusters. This approach overcomes the disadvantage of [5] by using TF-IDF to normalize the data. There have been a plethora of approaches made for text classification. All the approaches have more or less a common structure. In general sense, the words in the text are first represented in different ways [18]. Then, the weights for the terms are computed. Finally, a classifier is used to classify the text. Dimension reduction can be performed at any stage of the method. Apart from this structure, text classification is done using only labeled data or using labeled and unlabeled data.

In the proposed approach, it is assumed that each text has one to one association with the genre. In other words, one text cannot have more than one genre. There is no exogenous knowledge available. That is, metadata such as publishing date, publication source, author, for the texts is not assumed to be available. The input is only the text of the book and its genre. Bag of senses is used to represent the features which are explained in the transforming tokens using Wordnet. The added benefit of the bag of senses is that it performs dimension reduction based on similar term extracted from Wordnet. The weight for each term is computed using TF-IDF. PCA is used to reduce the dimension of the feature matrix further. For classification, boosting for ensemble learning is used.

III. PROPOSED APPROACH

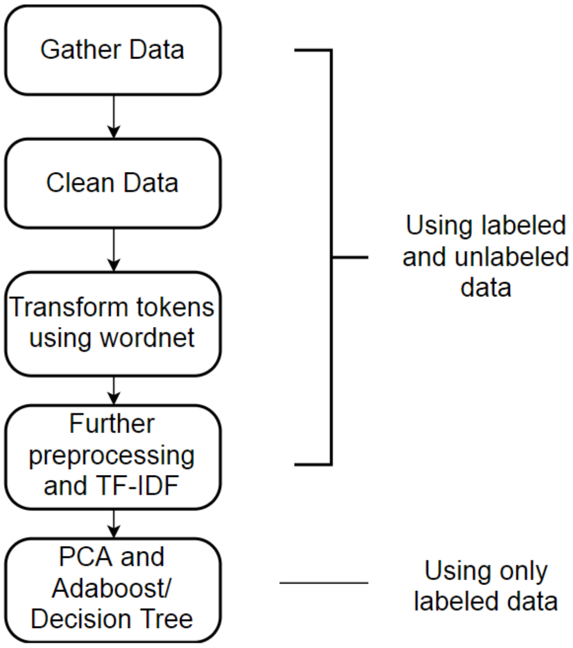
Gather Data
Clean Data
Transform tokens using wordnet
Further preprocessing and TF-IDF
PCA and Adaboost/ Decision Tree
Using labeled and unlabeled data
Using only labeled data

Fig 1. Steps used for implementing the proposed approach

3.1. Get Data

First of all, data was collected to test the proposed scheme on it. Websites are used to collect the text format books [18]. Python's library Beautiful Soup was used to scrape data from the website. It makes it easy to get data by traversing the Document Object Model structure of a web page. Nearly, 3600 books were collected through this website adding up to a total size of approximately 3 GigaBytes.

3.2. Clean Data

The data scraped is in unstructured form. Files have several lines of unwanted data like license and information about Gutenberg initiative. The text between “Start of This Project Gutenberg Ebook” and “End of This Project Gutenberg Ebook” is extracted and then extract the tokens [20] from the text, and perform various operations on these tokens. Tokens which are not listed in stop words [20] are selected. In other words, we can say that remove the words which are present in most of the documents (such as a and the) and thus are irrelevant for categorization. The tokens are then grouped to form a frequency table. A word can have many possible senses, for instance, wood can be used to refer a piece of wood as well as to refer a geographical area with many trees. In the proposed approach, first, we select the frequent sense for word sense disambiguation, and after that frequency table is obtained for most frequent sense of the token at the end.

3.3. Transforming tokens using wordnet

The data obtained from the previous step is huge, and the feature matrix it makes is sparse. In the presented method, the most frequent sense is transformed using Wordnet [12]. All the word sense are converted to the 1393-word sense with the help of Wordnet. From a word sense, the whole path to that sense can be obtained in Wordnet. This feature of Wordnet is exploited to reduce the number of senses and better cluster them to related senses.

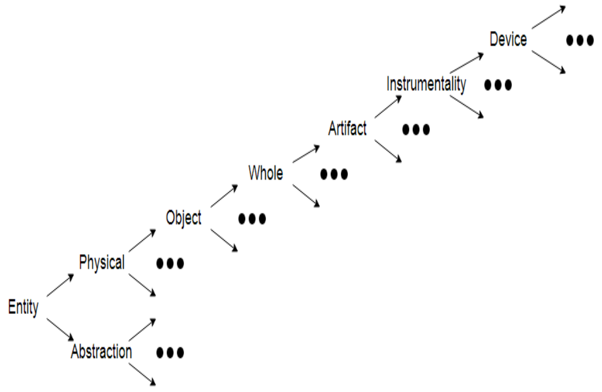

Fig.2 Hierarchy of words in Wordnet

3.4. Further Preprocessing

The total number of words of a book is added as a feature. Books with more than two genres are removed and deletes the features which contain zero in each row. Moreover, excludes the books whose absolute number of words are zero. TF-IDF [8][9] is evaluated [13] from the Eq.1 as defined below to generate the feature matrix. The unlabeled data facilitates to compute the term weight for the respective genre in the TF-IDF [10][11] step. Till now both labeled and unlabeled data is used in TF-IDF. Further, in the paper only labeled data would be processed.

$$W(i,j) = t\int i,j \times \log(N/df_i) \qquad (1)$$

Where
$t\int i,j$ = Number of occurrences of i in j.
df_i = Number of documents containing i.
N = Total number of words.

3.5. Applying machine learning

The feature matrix is sparse which will reduce the performance of the training model. To overcome this problem Principal Component Analysis [4] is applied to reduce the dimensions [14] of the feature matrix. Then, the feature matrix is divided into two parts: the training set and test set. Decision Tree Classifier [6] model is applied to training set to learn from it. Moreover, AdaBoost Classifier [7][17] is used to improve the accuracy of Decision Tree [15] by reducing bias and variance. After the model is trained on the training set, is used to predict genre on the test set.

IV. Results

The following hardware and software configuration is used to implement the proposed scheme and collected the accuracy on test and training result.

TABLE I. System Hardware Configuration

Hardware	Specifications
Operating System Type	64-bit operating system.
Processor	Intel i5-5200U CPU @ 2.20GHz × 4
Available memory	7.7 GiB

Table II. System Software Configuration

Software	Specifications
Operating System	Ubuntu 16.04 LTS
Language	Python 2.7.12

Using the proposed technique as described above, 80% of the data set is used to train the model and to evaluate the training score. Rest 20% is used to test the accuracy of the model (test score). Figure 3 and 4 represents the results of the test set.

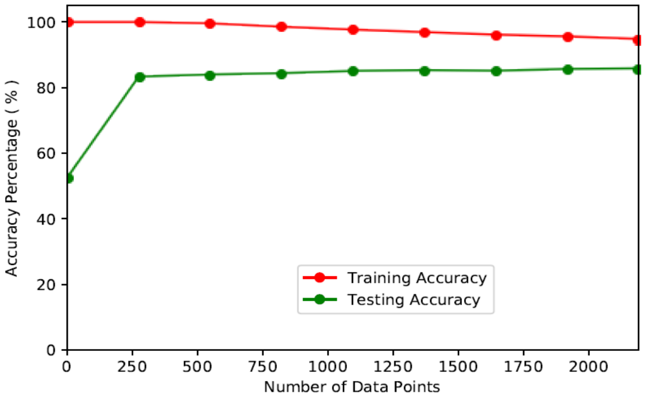

Fig. 1. Performance on labeled data

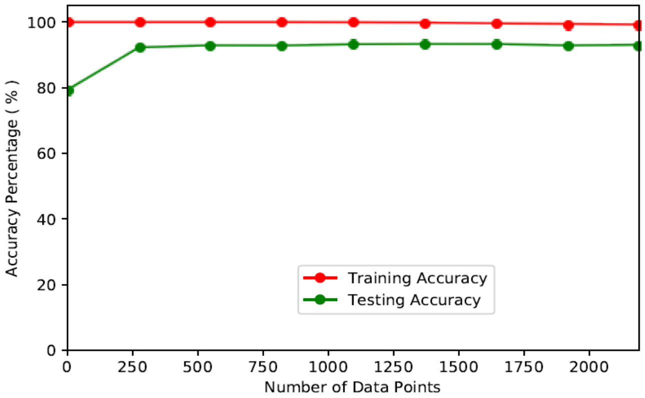

Fig. 2. Performance on labeled and unlabeled data

Figure 3 shows the performance on the test set without considering unlabeled data. It has observed that this scheme gives an accuracy of 81.18%. Figure 4 shows the performance on the test set after adding the unlabeled data to the preprocessing step. It has observed that this technique gives an accuracy of 92.88%. The increase in accuracy is due to better-computed term weight by TF-IDF. It is because more data is available due to the presence of unlabeled data which will help to compute better term weight of the features in TF-IDF.

V. CONCLUSION

This paper proposes a technique which predicts the genre of books. NLP is used to convert the text in the books into a feature matrix then it is reduced and used in the machine learning model. The performance of the proposed technique is optimized with various parameters like depth of the tree in Decision Tree Classifier, some estimators in AdaBoost and number of components in Principle Component Analysis. The developed scheme is scalable and can also be applied to predict genres of news articles and blogs.

References

1. Sebastiani, Fabrizio. "Machine learning in automated text categorization." *ACM computing surveys (CSUR)* 34.1, 2002,1-47.
2. Wen, Hao, Liping Fang, and Ling Guan. "A hybrid approach for personalized recommendation of news on the Web." *Expert Systems with Applications* 39.5 (2012): 5806-5814.
3. Kumar, V. Arul, and N. Elavarasan. "A Survey on Dimensionality Reduction."
4. Wold, Svante, Kim Esbensen, and Paul Geladi. "Principal component analysis." *Chemometrics and intelligent laboratory systems* 2.1-3 (1987): 37-52.
5. Lazaro S.P. Busagala, Wataru Ohyama, Tetsushi Wakabayashi and Fumitaka Kimura: Machine Learning with Transformed Features in Automatic Text Classification, http://www2.hi.info.mieu.ac.jp/publication/archive/busagala_Camera-ready.pdf
6. Safavian, S. Rasoul, and David Landgrebe. "A survey of decision tree classifier methodology." *IEEE transactions on systems, man, and cybernetics* 21.3 (1991): 660-674.
7. Schapire, Robert E. "The boosting approach to machine learning: An overview." *Nonlinear estimation and classification*. Springer, New York, NY, 2003. 149-171.
8. Oh, KyoJoong, et al. "Research trend analysis using word similarities and clusters." *International Journal of Multimedia and Ubiquitous Engineering* 8.1 (2013): 185-196.
9. Bafna, Prafulla, Dhanya Pramod, and Anagha Vaidya. "Document clustering: TF-IDF approach." *Electrical, Electronics, and Optimization Techniques (ICEEOT), International Conference on*. IEEE, 2016.
10. Al-Talib, Ghayda A., and Hind S. Hassan. "A study on analysis of SMS classification using TF-IDF Weighting." *International Journal of Computer Networks and Communications Security*1.5 (2013): 189-194.
11. Albitar, Shereen, Sébastien Fournier, and Bernard Espinasse. "An effective TF/IDF-based text-to-text semantic similarity measure for text classification." *International Conference on Web Information Systems Engineering*. Springer, Cham, 2014.
12. Miller, George A. "WordNet: a lexical database for English." *Communications of the ACM* 38.11 (1995): 39-41.
13. Sebastiani, Fabrizio. "Machine learning in automated text categorization." *ACM computing surveys (CSUR)* 34.1 (2002): 1-47.

14. Zahedi, M., and A. Ghanbari Sorkhi. "Improving text classification performance using PCA and recall-precision criteria." *Arabian Journal for Science and Engineering* 38.8 (2013): 2095-2102.
15. Patel, Bhaskar N., Satish G. Prajapati, and Kamaljit I. Lakhtaria. "Efficient classification of data using decision tree." *Bonfring International Journal of Data Mining* 2.1 (2012): 06-12.
16. Ikonomakis, M., Sotiris Kotsiantis, and V. Tampakas. "Text classification using machine learning techniques." *WSEAS transactions on computers* 4.8 (2005): 966-974.
17. Schapire, Robert E., and Yoram Singer. "BoosTexter: A boosting-based system for text categorization." *Machine learning* 39.2-3 (2000): 135-168.
18. De Dillmont, Therese. *Encyclopedia of Needlework*. Editions Th. de Dillmont, 1987.
19. Park, Sanghoon, et al. "Text Mining Metal–Organic Framework Papers." *Journal of chemical information and modeling* 58.2 (2018): 244-251.
20. Loper, Edward, and Steven Bird. "NLTK: The natural language toolkit." *Proceedings of the ACL-02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics-Volume 1*. Association for Computational Linguistics, 2002.