Indian Journal of Science and Technology, *Vol 9(39), DOI: 10.17485/ijst/2016/v9i39/100777, October 2016*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Semi-Supervised Distributional Vector Generation Techniques for Text Classification

Mohammed Abdul Wajeed*

Computer Science and Engineering Department, Jyothi Engineering College, Cheruthuruthy - 679531, Kerala, India; drwajeedphd@gmail.com,

Abstract

Text class has loved its privilege as a core studies area in text mining. Supervised, unsupervised are the 2 famous paradigms within the technique of type. Relatively novel method of classification is semi-supervised mastering which is midway among the supervised and unsupervised getting to know. With smaller schooling statistics units and taking the large without problems to be had unlabeled data, the procedure of studying in class is refined. There are versions in semi-supervised, transductive gaining knowledge of wherein the trained and untrained facts are given in advance the classifier is built, the goal is to expect the magnificence label of untrained data. The opposite version is inductive learning in which the labeled and unlabeled statistics is utilized in model constructing; goal of the version is to predict the unseen information magnificence label. The paper aims to using transductive getting to know to classifying the textual statistics with the aid of considering the phrases appearing in special parts of the record. The words performing inside the introductory and conclusion a part of the files may additionally play important function within the procedure of type, than the ones seemed in other parts. The approach employed could provide one of a kind weights to words primarily based on their presence in one-of-a-kind role of the document. Taking into consideration the above within the procedure of mapping the textual facts into numerical patterns editions of distributed vector generations are acquired. Taking into account large differences in the duration of the documents, distinct normalization techniques are employed which gave eights one-of-a-kind vectors. Non-parametric, most effective to put into effect ok-nearest neighbour algorithm is hired for free-go with the flow textual classification. The outcomes received conclude that semi-supervised textual class can be carried out without loss in category accuracy where restrained skilled records is to be had, as the accuracies of the gaining knowledge of model in supervised and emi-supervised coincide with each other.

Keywords: Distributional Vectors, KNN, Semi-Supervised, Text Classification, Transductive Learning

1. Introduction

Text type has loved as a core studies region in domain of text mining particularly after the generation of electronic textual information mills. Greater ever the fee of textual records technology has multiplied due to the usage of intra, internet utilization in enormous areas. The sole cause to shop the data is for evidence checking alone, so no effort is employed to save the information in categorized repository. Keeping in view of the future wishes of choice making, if the statistics is stored in categorized repositories, then navigation and use of it decision making becomes simpler. The most popular models for records classification commonly are supervised and unsupervised as given in[1]. Supervised type is a predictive version where the task is to educate a version based totally on training records, the education records is classified. The version constructed is used to assign a pre-defined elegance label to new facts as in[13]. Unsupervised mastering do not have education records, it corporations the given information into clusters primarily based at the similarity as in[6]. The statistics within the similar cluster are handled to belong to same magnificence. Semi-supervised is relative new process model which takes smaller education statistics units mixed with massive to be had unlabeled facts to categorise the records. Normally bag-of-words (bow)

*Author for correspondence

approach is employed in text category as in[4] as opposed to parts-of-speech (pos) technique as in[5]. The paper tries in exploring the statistics type the usage of the semi-supervised paradigm the use of the bow technique. The presence of a word within the report, its frequency alone is commonly taken into consideration in the manner of text class, but the impact of words incidence in distinct components of the report changed into no longer considered. Words going on in the advent and conclusion play essential function prompted to advise distributional vector technology techniques for textual facts type. The aim of the paper is to apply semi-supervised studying paradigm and obtain the accuracy of the classifier for the vectors generated that think about the presence of the word relative to its position in the document.

The paper is organized as follows. In the second section material and methods are outlined. In Section third the results and discussion are provided, finally the paper ends with the conclusion and future enhancements.

2. Materials and Methods

In[2] author affords the dataset (corpus) for the experiments performed in the paper. Though noise become determined in the corpus, however it was overlooked. the dataset had 5485 files d train as schooling records split throughout 8 training and 2189 files as take a look at information d test.

Inside the technique of semi-supervised mastering, using the limited training records, a classifier is built. Part of check records is subjected to the classifier; the results acquired later are introduced to the pre-present schooling data to generate new education statistics as shown in the discern 1. With the new schooling data a brand new classifier is built which is deemed to be superficial than the previously built classifier because of increase within the education information. The complete system is repeated, yielding a very last classifier.

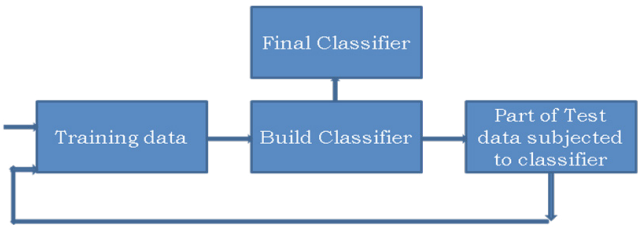

Figure 1. Semi-supervised approach in the process of learning.

On the grounds that we're exploring semi-supervised mastering paradigm, a hundred and forty files from the education set are chosen as education files. A simple heuristic is employed in choice of the a hundred and forty documents. 10 files from each class (leaving a category which has few files) that have huge wide variety of words in it are chosen, and the rest of the facts is merged with the take a look at records. It's far assumed that the schooling and take a look at records are within the identical distribution.

In[7] presents in element the entire process of producing lexicon, which contains all of the words available within the dtrain. lexicon set is a fixed which has all the phrases that exist in the training document set. Phrases are the functions in textual content classification; characteristic reduction is commonly hired before the category process starts. stop words are the words that do not play any function within the category manner, so they're excluded within the method of generating the lexicon set. To be able to lessen the lexicon set size in addition, phrases root bureaucracy alone are considered as individuals of the lexicon set. Stemming algorithm from[8] is hired to get the root phrases. In this experiment the lexicon size acquired turned into 14,822. in the procedure of reducing the features, phrases that frequency under a threshold in the set of educate record are discarded. however to get perception of semi-supervised studying paradigm all phrases are considered whose frequency is extra than one are taken into consideration.

Dealing with huge quantity of features that clearly exists in text categorization is a night time mare. Characteristic selection is an important step in text processing. The other change to function reduction is feature selection and characteristic extraction. Feature selection refers the technique of locating a subset of the unique functions, which may be received the usage of both filtering method or by way of using wrapper method. Then again function extraction refers back to the technique of remodeling the facts from the excessive-dimension area to a lower size space. Such conversion in dimension can be a transformation from in linear as inside the case of fundamental aspect analysis (PCA), or can be in some other nonlinear transformation. In well known we have information benefit, gain ratio, odds ratio, gini-index, chi-square and so forth strategies as function choice techniques for supervised studying. Techniques like record frequency, term frequency, and inverse-report frequency

are taken into consideration as characteristic choice in unsupervised mastering.

3. Results and Discussion

To claim the effect of the distribution of the phrases within the file distributional features are explored in place of contemplating by myself phrases presence inside the files as in[14]. Words that appear inside the introduction and end of the file play critical role in the system of decision making of the class type than the ones seem within the other components of the report. Taking into account the placement of the words inside the record the want to discover distributional features turns into inevitable. The documents are divided into 4 elements on adhoc basis, introduction and end as the first and final component. We find the frequency of the words appearance in each of the element, consequently binary and different vectors are generated.

3.1 Binary Vector Era

Simplest the primary and remaining part of the record is considered. The first part of the record has the introduction; the ultimate part of the report has the precis or the conclusion of the record. Words appeared in the first and the remaining file plays critical role in figuring out the category of the report. On the way to capture the effect of words appearance in first and closing a part of the file, binary vector is generated as given inside the equation below

$$binary(t_i,d) = (C_0 + C_{n-1}) > 0 ? 1 : 0 \quad (1)$$

Wherein binary (ti,d) is the binary vector detail access for the time period ti inside the file wide variety d, c0 is the range of instances the term seemed within the first (introductory) part of the report, cn-1 is the variety of times the word seemed in the final (conclusion, summary) part of the file. When you consider that it is binary vector bin(ti,d) includes either 1 or 0. Words frequency has no function to play in binary vector era.

3.2 Frequency Vector Generation

Rather than taking the words appearance by myself in the first and remaining a part of the record, quantity (frequency) of instances the phrase look is taken into consideration inside the first and the remaining components of the file

$$frequency(t_i,d) = (C_0 + C_{n-1}) \quad (2)$$

where frequency(t_i,d) is the frequency vector element entry for the term t_i in the document d, C_0 is the number of times the term t_i appeared in the first (introductory) part of the document and C_{n-1} is the number of times the term appeared in the last (conclusion) part of the document.

3.3 Weighted Frequency Vector Generation

In preference to ignoring the presence of the words in different parts of the report, unique weight-age for every a part of the file is explored and the general weight-age of the words look within the file is calculated

$$wfreq(t_i,d) = W * (C_0 + C_{n-1}) + C_1 + C_2 \quad (3)$$

where wfreq(t_i,d) is the weighted frequency vector element entry for the term t_i in the document d, W is the weight assigned which take any arbitrary value, 2 is used in the experiment, C_0 is the number of times the term t_i appeared in the first part of the document and C_{n-1} is the number of times the term appeared in the last part of the document, C_1,C_2 referees to the number of times the terms appeared in the second and third part respectively.

Normalizing the vectors become inevitable, to reduce the impact of features having more value, dominating the features having lesser value.

3.4 Binary Vector Normalized with Unique Words

As the length of the document increases the probability of words appearance in the document also increases. In order to neutralize the length impact, normalizing the vector is explored.

$$bin\_u(t_i,d) = (C_0 + C_{n-1} > 0 ? 1 : 0) / \sum_{i=0}^{n-1} UC_i \quad (4)$$

Where, bin_u(t_i,d) is the binary normalized, with unique words vector. The numerator value is binary vector generation. UC_i is the number of unique words in each of the part of the document. N-1 is the total number of parts in the document which is arbitrarily choose as 4 in the present experiment.

3.5 Binary Vector Normalized with all Words

Instead of normalizing the vector elements with the number of unique words we normalize the vector elements

with the number of all words in the document which results in binary vector normalized with all words.

$$bin\_all(t_i,d) = (C_0 + C_{n-1} > 0 ? 1 : 0) / \sum_{i=0}^{n-1} C_i \quad (5)$$

where bin_all(t_i,d) is the binary normalized, with all words vector. The numerator value is binary vector generation. C_i is the number of all words in each of the part of the document. N-1 is the total number of parts in the document which is arbitrarily chooses as 4 in the present experiment.

3.6 Frequency Vector Normalized with all Words

It is similar to the binary normalized in addition to normalizing the binary vector we normalize the frequency vector with all words

$$freq\_all(t_i,d) = (C_0 + C_{n-1}) / \sum_{i=0}^{n-1} C_i \quad (6)$$

where freq_all(t_i,d) is the frequency normalized, with all words vector. The numerator value is frequency vector generation. C_i is the number of all words in each of the part of the document. N-1 is the total number of parts in the document which is arbitrarily chooses as 4 in the present experiment.

3.7 Frequency Vector Normalized with Unique Words

In addition to normalizing the vector elements with all words, frequency vector is normalized with unique words appeared in the document.

$$freq\_u(t_i,d) = (C_0 + C_{n-1}) / \sum_{i=0}^{n-1} UC_i \quad (7)$$

where freq_u(t_i,d) is the frequency normalized, with unique words vector. The numerator value is frequency vector generation. UC_i is the number of unique words in each of the part of the document. N-1 is the total number of parts in the document which is arbitrarily chooses as 4 in the present experiment.

3.8 Weighted Frequency Vector Normalized with Unique Words

In the process of exploring the comprehension of distributed features we obtain the weighted frequency vector normalized with unique words based on the below equation.

$$w\_freq\_u(t_i,d) = W * (C_0 + C_{n-1}) + C_1 + C_2 / \sum_{i=0}^{n-1} UC_i \quad (8)$$

where w_freq_u(t_i,d) is the weighted frequency vector normalized, with unique words vector. The numerator value is frequency vector generation multiplied by a weight W which is taken as 2. UC_i is the number of unique words in each of the part of the document. N-1 is the total number of parts in the document which is arbitrarily chooses as 4 in the present experiment.

3.9 Weighted Frequency Vector Normalized with all Words

In addition to obtaining the weighted frequency vector normalized with unique words, weighted frequency vector normalized with all words is determined using the below equation.

$$w\_freq\_all(t_i,d) = W * (C_0 + C_{n-1}) + C_1 + C_2 / \sum_{i=0}^{n-1} C_i \quad (9)$$

where w_freq_all(t_i,d) is the weighted frequency vector normalized, with all words vector. The numerator value is frequency vector generation multiplied by weight W which is taken as 2 in the experiment. C_i is the number of all words in each of the part of the document. N-1 is the total number of parts in the document which is arbitrarily chooses as 4 in the present experiment.

3.10 Implementation

Ok-nearest neighbour set of rules is also known as instance primarily based learning set of rules[9–11]. The training tuples are defined by 'n' capabilities, in our case the features are words inside the lexicon set. Every tuple represents a factor in an n-measurement pattern space. Whilst given an unseen tuple, the k-nearest neighbor classifier searches the pattern area for special values of okay (which can take any value 1 via some arbitrary quantity), the schooling tuples which can be closest to the unseen tuple. Relying at the value of ok, okay training tuples are used which might be near to the unseen tuple.

For one of a kind okay values, the experiment changed into repeated. The elegance label of the check statistics is considered relying on most of the people of the tuples from the train statistics which might be very close to the

record in consideration. In case of a tie, in which identical numbers of train documents belong to 2 or more elegance labels then arbitrary the tie is resolved. Whilst okay price is 1 in k-nn, the gap between the training and a specific test documents is measured, the elegance with the nearest training information is taken as the magnificence of the test records as in[3,12]. In case of k price 2 we take 2 smallest distances, and if both belong to equal elegance than the take a look at tuple also belong to the equal magnificence, in case of tie an arbitrary consensus is used to resolve the conflict.

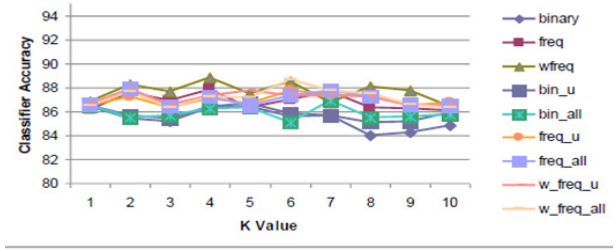

Figure 2. Text classifier accuracy for K value varying from 1 to 10.

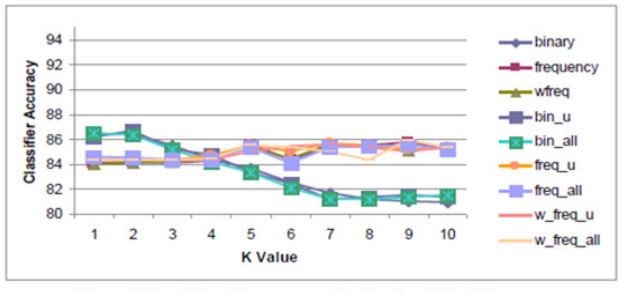

Figure 3. Text classifier accuracy for K value 10 to 100.

Primarily based at the similarity between the training and check tuples we achieve confusion matrix which is a good device for studying, how first-class the classifier has categorized the tuples for different lessons may be obtained. But we give here the accuracy of the classifier for k fee various from 1 to 10 for euclidean similarity metrics in Figure 2. The overall performance of weighted frequency vector is better than the alternative vector strategies. Parent 3 gives the accuracy of the classifier for k values varying from 10 to one hundred for euclidean similarity metrics. From the figures we find the accuracy of the semi-supervised learning to be very close to above 85%. We draw the realization that semi-supervised can be carried out in times wherein we've got constrained schooling facts to reap high accuracy of the classifier.

4. Conclusion

Textual content category is a totally essential studies region in textual content mining. Using bag-of-phrases approach the paper attempts to categorise the facts by way of exploring the distributional capabilities for semi-supervised class paradigm. For okay values various from 1 to ten weighted frequency vector gave excellent consequences and the binary vector least in terms of the classifier accuracy. For okay cost various from 10 to one hundred a few vectors gave higher effects till k price 50 while different vectors gave higher accuracy consequences for ok value above 50. Feature discount techniques may be carried out, and the overall performance of the semi-supervised classifier can be ascertained.

5. References

1. Sebastiani FS, et al. An improved boosting algorithm and its application to automated text categorization. Proceedings of CIKM-00, 9th ACM International Conference on Information and Knowledge Management' McLean, VA. 2000. p. 78–85.
2. Available from: http://www.daviddlewis.com/resources/testcollections/reuters21578/
3. Raschka. Naive bayes and text classification I-introduction and theory. 2014 Oct.
4. Wang C, et al. Text classification with heterogeneous information network kernels. 13th AAAI Conference on Artificial Intelligence; 2016.
5. Collobert R, et al. Natural language processing (almost) from scratch. The Journal of Machine Learning Research. 2011; 12(2011):2493–537.
6. Nayef N. Text zone classification using unsupervised feature learning. 13th International Conference on Document Analysis and Recognition; 2015. p. 776–80.
7. Wajeed MA, et al. Text classification using machine learning. Journal of Theoretical and Applied Information Technology. 2009; 7(2):119–23.
8. Available from: tartarus.org/~martin/PorterStemmer
9. Cunnigham P, et al. K-nearest neighbour classifiers. Technical Report; UCD-CSI 2007-4.
10. Aha DW, et al. Instance-based learning algorithms. Machine Learning. 1991; 6:37–66.
11. Yeung C-MA, et al. A k-nearest-neighbour method for classifying web search results with data in folksonomies. 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology; 2008. p. 70–6.

12. Harish BS, et al. Representation and classification of text documents: A brief review. IJCA. 2010; Special Issue on RTIPPR(2):110–9.
13. Wajeed MA, et al. Text classification using KNN classifier. International Journal of Computer Science and System Analysis. 2009 Jul-Dec; 3(2):83–7.
14. Wajeed MA, et al. Supervised and semi-supervised learning in text classification using enhanced KNN algorithm (A comparative study of supervised and semi-supervised classification in text categorization) International Journal of Intelligent Systems Technologies and Applications Journal. 2012; 11(3/4):179–95.