2011 International Conference on Computational Intelligence and Communication Systems

Categorizing the Document using Multi Class Classification in Data Mining

Shweta Joshi
Department of Computer Engineering
Institute of Engineering & Technology
Devi Ahilya Vishwavidyalaya, Indore (M.P.), India
e-mail: joshii_shweta@yahoo.com

Bhawna Nigam
Department of Information technology
Institute of Engineering & Technology
Devi Ahilya Vishwavidyalaya, Indore (M.P.), India
e-mail: bhawnanigam@gmail.com

Abstract- **Classification is the process of dividing the data into number of groups which are either dependent or independent of each other and each group acts as a class .The task of Classification can be done by using several methods using different types of classifiers. But classification cannot be done easily when it is to be applied on text documents that is: document classification. The main purpose of this paper is to analyze the task multi-class document classification and to learn that how can we achieve high classification accuracy in the context of text documents. Naive Bayes approach is used to deal with the problem of document classification via a deceptively simplistic model: assume all features are independent of one another, and compute the class of a document based on maximal probability. The Naive Bayes approach is applied in Flat (linear) and hierarchical manner for improving the efficiency of classification model. It has been found that Hierarchical Classification technique is more effective then Flat classification .It also performs better in case of multi-label document classification. The dataset for the evaluation purpose is collected from UCI repository dataset in which some changes have been done from our side.**

Keywords- Data Mining, Document Classification, Multi-class Classification Multi-label Classification, Hierarchical Classification, Text categorization, Naïve Bayes classifier.

I. INTRODUCTION

Data Mining is the process of applying machine learning techniques for automatically or semi automatically analyzing and extracting knowledge from stored data. It is defined as non-trivial extraction of implicit, novel and actionable knowledge from large datasets. Data mining [1] can also be defined as technology which enables data analysis, exploration and visualization of very large databases using high level of abstraction. Data mining [2] models can be categorized as predictive models and descriptive models. Classification is one of the important aspects of data mining which is a predictive modeling technique. Classification techniques [3] are used in various real world problems with respect to application domain as well as for various research purposes. It is used to group the data instances into proper class i.e. Classify them. Classification is used to build structures from examples of past decisions that can be used to make decisions for unseen or future cases. Various algorithms[4]which are used for classification are decision tree learning, nearest neighbor, naïve's bayes classification[5], neural network, support vector machine, e.t.c. Classification techniques are used for web page classification, data classification e.t.c. Text categorization is the task of automated assigning of texts to predefined categories based on their content. Text categorization [6] [7] [8] is the primary requirement of Text Retrieval systems. As the amount of online text increases, the demand for text categorization for the analysis and management of text is increasing. Though the text is cheap, it is expensive to get the information that, to which class a text belongs to. This information can be obtained from automatic categorization of text at low cost, but building the classifier itself is expensive because it require a lot of human effort or it must be trained from texts which have themselves been manually classified. Document Classification [9] [10] is one of the aspect of text categorization is a fundamental learning problem of many information management and retrieval tasks. In the task of document classification when more than two classes exists then it can be termed as multi-class document classification. In case of multi-class document classification when a document belongs to more than one label then it can be termed as multi-label document classification [11]. It is usually performed in two stages: 1) the training phase and 2) the testing phase. During the training phase, sample documents are provided to the document classifier for each predefined category. The classifier uses machine learning algorithms to learn a class prediction model based on these labeled documents. In the testing phase, unlabelled documents are provided to the classifier, which applies its classification model to determine the categories or classes of the unseen documents. This training-testing approach makes the process of document classification a supervised learning task where unlabeled documents are categorized into known categories.

There exist so many algorithms based on the Naïve Bayes [12] Classifier to classify text. For applying Naïve Bayes Classifier, each word position in a document must be treated as an attribute and the value of that attribute to be the word found in that position. Naïve Bayes categorization is given by equation (1):

$$\Pr\ (\text{Category} \mid \text{Word}) = (\Pr\ (\text{Word}/\ \text{Category}).\ \Pr\ (\text{Category}))/\ \Pr\ (\text{Word}) \qquad (1)$$

Where, Pr=probability

978-0-7695-4587-5/11 $26.00 © 2011 IEEE
DOI 10.1109/CICN.2011.50

IEEE computer society

In our approach firstly we have used general Naïve Bayes approach for classifying the documents. The system is given a set of example documents which are used to train the classifier. For preprocessing the text documents stop words are removed. Then collection of frequently occurring words from each document is done. This is done by matching each word of the training document with the words contained in pre-defined vocabulary. The vocabulary is the collection of feature set from the training set documents which can be built by using any feature set extraction method. Then new documents are classified using Naïve Bayes approach but using derived feature sets. As the training dataset is arranged in a linear manner and the classification is performed in linear manner we will call this approach as flat or linear classification.

But the linear classification takes more time and sometimes proves inaccurate when classifying a new document thus if the whole training set can be arranged in a hierarchic manner that is if we arrange the classes or labels containing the set of training documents into hierarchical order according to inter class relationship though we cannot able to decrease the training time but the time for classifying the new document can be decreased as, the comparison for matching the words should not be done for each class in the training set ,but it is to be done in hierarchic manner .This approach of classification we will call as hierarchical classification [13][14][15]. Thus the main aim of this paper is to focus on flat (non-hierarchical) multi-class classification methods as well as hierarchical multi-class [16] classification.

The rest of the paper is organized as follows: In section 2 a general model for document classification is present using flat classification and hierarchal classification is also described in this section. We move to experimental results in section 3 and finally we conclude the paper in section 4.

II. RELATED WORK

In this section all the work related to the task of document classification [17] is illustrated including classification approaches used in this paper with their description.

A. Description of document classification using Naïve Bayes Theorem

For the task of document classification, the Bayes theorem uses the fact that the probability of a particular document being annotated to a particular category, given that the document contains certain words in it, is equal to the probability of finding those certain words in that particular category, times the probability that any document is annotated to that category, divided by the probability of finding those words in any document, as illustrated in equation (1). Each document contains words which are given probability values based on the number of its occurrence within that particular document. Naïve Bayes classification is predicated on the idea that electronic documents can be classified based on the probability that certain keywords will correctly identify a piece of text document to its annotated category. At the basic level, a naïve Bayes classifier examines a set of text documents that have been well organized and categorized, and then compares the content of all categories in order to build a database of words and their occurrences. The database is used to identify or predict the membership of future documents to their right category, according to the probability of certain word occurring more frequently for certain categories. It thus overcomes the problems faced by static approaches, using static databases filled with pre-defined keywords. An advantage of the naive Bayes classifier is that it only requires a small amount of training data to estimate the parameters necessary for classification. Because independent variables are assumed, only the variances of the variables for each class need to be determined.

B. Applying Flat classification using Naïve Bayes Theorem

The Naïve Bayes classifier we have implemented performs its classification tasks starting with analyzing the text document by extracting words which are contain in the document. To perform this analysis, any extraction algorithm [18] [19] can be used to extract each individual word from the document to generate a list of words. In our work we will term this list as vocabulary.

|vocabulary| =the total number of distinct word set found within all the training data

This list is helpful when the probabilistic classifier calculates the probability of each word being annotated to each category. The list of words is then used to generate a table, containing the words which are extracted from the input document. The probabilistic classifier is needed to be trained with a set of well-categorized training dataset. Each individual word which will be matched with words contained in vocabulary from all training documents in the same category are extracted and listed in a list of words occurrence for the particular category. Based on the list of word occurrence, the trained probabilistic classifier calculates the posterior probability of the particular word of the new unlabeled document being annotated to particular category by using the formula which is shown as equation(1). The prior probability, Pr (Category) can be computed from equation (2)

Or equation (3):

$$\text{Pr (Category)} = \frac{\text{(Total number of words in a class)}}{\text{(Total number of words in training set)}} \qquad (2)$$

$$\text{Or, Pr (Category)} = \frac{\text{size of class}}{\text{Size of training dataset}} \qquad (3)$$

The category for an unlabeled document is represented by the category which has the highest posterior probability value, Pr (Category | Document).In case of multi-label [20] document classification the classes having higher posterior probability value can be assigned to a document.

C. The steps for preprocessing and classifying a new document can be summarized as follows:

- Remove periods, commas, punctuation, stop words. Collect words that have occurrence frequency more than once in the document. We called this collection of words as vocabulary.
- View the frequent words as word sets by matching the words which are in the vocabulary as well as training set documents.
- Search for matching word set(s) or its subset(containing items more than one) in the list of word sets collected from training data with that of subset(s) (containing items more than one) of frequent word set of new document.
- Collect the corresponding probability values of matched word set(s) for each target class.
- Calculate the probability values for each target class from Naïve Bayes categorization theorem.
- Categorize the document to a class having higher probability values.

Following the steps mentioned above, we can determine the target class of a new document. For multi-label[21][22] document classification a threshold[23] value can be assigned to classifier and a document can be categorized to those classes having probability values higher than threshold value.

D. Applying Hierarchical classification using Naïve Bayes Theorem

We have extended the classification step for input documents by performing the hierarchical classification using Naïve Bayes approach for the purpose of increasing the classification accuracy as well as speed. The limitation of flat classification is that in flat classification the predefined categories are treated individually and equally so that no structures exist to define relationships among them. In such a hierarchical structure document types become more specific as we go down in the hierarchy. In hierarchical structures relationships of dependence between the categories is identified, which provide a valuable information source for many problems. The basic step in hierarchical classification [24] [25] is to infer class relationships.

E. The steps for preprocessing and classifying a new document can be summarized as follows:

- Remove periods, commas, punctuation, stop words. Collect words that have occurrence frequency more than once in the document. We called this collection of words as vocabulary.
- Arrange the collection of training dataset in hierarchic manner by finding the inter-class relationship between the classes by comparing each class with one another according to words contained in the document.
- Whenever any new unlabeled document comes for testing it is matched in hierarchic manner with training classes.
- Thus in testing phase the test document follows only that path of hierarchy whose highest probability value matched with the test document.

In case of multi-label document classification [26] the document can be assigned to those classes which are having higher probability value than the specified threshold and they must be a part of hierarchy from root to leaf that is they must have some interrelationship between them.

F .Comparison of Flat classification with hierarchical classification

- Limitation of flat classification [27] is that as the number of possible categories increases the distinction between document classes get blurred. Whole in hierarchical structure as we go down in hierarchy document types become more specific.
- In flat classification relationship among documents cannot be identified thus it proves problematic in multi-label classification .While the hierarchical structure identify the relationship among the classes which allows for efficiency n both learning and representation.

G. Training Data [29]

The documents are collected from the classic Reuters - 21578 collection for the purpose of evaluation. It is a collection of 21578 newswire articles originally collected and labeled by Carnegie Group,Inc. and Reuters,Ltd. For the task of evaluation ten largest classes in the Reuters-21578 collection was taken and some classes are added in this collection for testing the accuracy of classifier. The whole collection of documents is divided into two parts ,one is considered as training document for developing model for classifying new documents of unknown class and another is used as test set for classifying new documents of unknown class . We have also tested our approaches on the 20 Newsgroups dataset. The 20 Newsgroups dataset is one of the most common datasets used by many text classification research groups to evaluate the performance of classification approaches. The 20 Newsgroups dataset is a collection of 20,000 Usenet articles from 20 different newsgroups with 1,000 articles per newsgroup. In our experiments using this dataset, every category was divided into two subsets. Some documents from each category were divided for training while the remaining documents were used for testing purposes. The developed model for classifying documents can be applicable to any dataset having collection of text documents. But an assumption is there that when we use the training set to learn a classifier for test data ,the training data and test data must be similar or from the same distribution.

III. PERFORMANCE EVALUATION

As discussed above in flat classification whenever a new document arrives which is to be categorized, its matching is done with each and every class one by one in linear manner while in hierarchical classification the document is matched with child of only that class of hierarchy whose frequency of matched words is highest with the test document skipping some classes whose frequency of matched words is less than this group of classes. Thus time for searching the class decreases as some classes are skipped. Even the accuracy of the classifier can be improved as if we take the example of Reuters-21578 dataset grouping the classes like wheat, corn, grain into single class or exchanges and organization into single class hierarchy reduces misclassification and also provide more effective multi-label classification. Even in 20 newsgroup dataset by form hierarchy of classes related to science, computers and others improves the classification accuracy.

In our work to test whether the document has been correctly classified or not, the predicted class is cross-checked with the document's actual class. For measuring the performance of the two approaches stated above, the following definitions of precision, recall, F1-measure and accuracy are used to find the effectiveness of document classifier. Accuracy of the classifier is determined by the percentage of the test data set that is correctly classified which is shown as equation (4).

$$\text{Accuracy}= \frac{\text{Number of correct prediction}}{\text{Total number of prediction}} \qquad (4)$$

Recall is determined by number of documents retrieved that are relevant with respect to total number of documents that are relevant. Thus,

$$\text{Recall}= TP/ (TP+FN) \qquad (5)$$

Precision is determined by number of documents retrieved that are relevant with respect to total number of documents that are retrieved. Thus,

$$\text{Precision} = TP/ (TP+FP) \qquad (6)$$

Thus,

$$\text{F1-measure}=(2*\text{recall}*\text{precision})/(\text{recall}+\text{precision}) \qquad (7)$$

Where,

TP (True Positive): The number of documents correctly classified to that class.

TN (True Negative): The number of documents correctly rejected from that class.

FP (False Positive): The number of documents incorrectly rejected from that class.

FN (False Negative): The number of documents incorrectly classified to that class.

For evaluating the performance of both the approaches used in this paper if we take speed as criteria the hierarchical classification takes almost same or sometime more time in training phase but when it is to be applied on new document for finding its class that in testing phase its searching time is less then or sometime almost half with respect to flat classification. On the other hand if take accuracy as criteria by grouping the inter-related classes into hierarchy gives more better results in some condition(when the classes have some inter-relationship among them) with respect to flat classification otherwise its results will be same as that of flat classification. Also in case of multi-label classification the labels identified for a class are more accurate as they have some inter-relationship among them. The classes are first are grouped into hierarchy according to their inter-relationship and only those classes of hierarchy which matches best can be annotated as labels to a new unlabeled document.

IV. EXPERIMENTAL RESULTS

The work of classifying a new document depends on the word sets generated from training documents. So the number of training documents is important in formation of word sets used to determine the class of a new document. The greater number of word sets from training documents reduces the possibility of failure to classify a new document. We implemented both the approaches discussed above. We experimented with the datasets mentioned above in conjunction with the Naive Bayes [28] classifier learning algorithm. For performance evaluation, we used the Accuracy, Precision and Recall metrics that were presented in the previous section. In the following two figures both the techniques are compared with each other. The comparison is done on the basis accuracy and F1-Measure calculated for both the techniques. Accuracy and F1-Measure are compared with respect to size of training set. The training set consists of some classes from Reuters-21578 dataset along with some additional classes added for testing purpose. The x-axis in both the figures represents the number of documents used for training the classification model. As for accuracy the hierarchical technique is still ahead of flat classification technique except in some cases in which output of both the techniques are almost same Also hierarchical classification takes less time for finding out the class of a new testing document.

The results obtained on performing the experiments are:

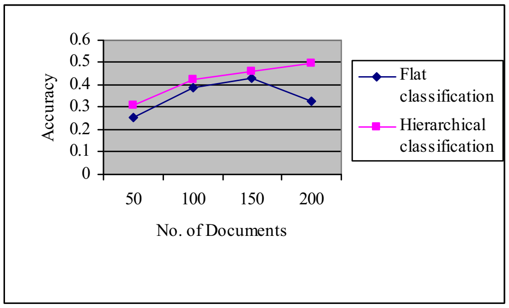

Fig1. Experimental Results

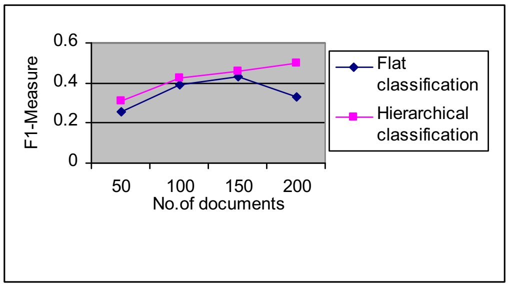

Fig2. Experimental results

V. CONCLUSION AND FUTURE WORK

In last two decades various researchers has experimented with the task of Text Document Classification using machine learning algorithms. The main aim of all this work is to improve the efficiency and accuracy of classifier. We have found that the Naïve Bayes approach we have used performs well with even large datasets. Generating hierarchy of the available training classes and then applying classifier model can improve classification performance in most cases even in multi label classification. But the further research is needed to build statistically significant and meaningful hierarchy .Even for efficient text classification it is required to get strong hierarchy information which needs further investigation. Combining different classification approaches instead of single one along with hierarchic structure of classes also provide an avenue for future research.

REFERENCES

[1] Data Mining: An AI Perspective, Xindong Wu1, Senior Member, IEEE.

[2] Jiawei Han and Micheline Kamber, 2001. "Data Mining: Concepts and Techniques", Morgan Kaufmann Publisher: CA.

[3] Survey of Classification Techniques in Data Mining, Thair Nu Phyu, Proceedings of the International Multi Conference of Engineers and Computer Scientists 2009 Vol IIMECS 2009, March 18 - 20, 2009, Hong Kong.

[4] Tom M.Mitchell, "Machine Learning," Carnegie Mellon University, McGraw-Hill Book Co, 1997.

[5] S. B. Kim, H. C. Rim, D. S. Yook and H. S. Lim, Effective Methods for Improving Naïve Bayes Text Classifiers, In Proceeding of the 7th Pacific Rim International Conference on Artificial Intelligence, Volume, 2417, 2002.

[6] Yang, Y., & Liu, X. (1999). A re-examination of text categorization methods. Proceedings of the 22nd Annual International Conference on Research and Development in Information Retrieval (SIGIR'99) (pp. 42ñ49). ACM Press.

[7] D. D. Lewis, R. Ghani, D. Mladenic, I. Moulinier, and M. Wasson. Workshop proceedings. In 3rd Workshop on Operational Text Classification (OTC), in conjunction with SIGKDD, 2003.

[8] Cohen, W. (1995) Learning to classify English text with ILP methods, In Advances in ILP. IOS Press

[9] Shantanu Godbole, Abhay Harpale, Sunita Sarawagi, and Soumen Chakrabarti.. Document classification through interactive supervision of document and term labels. In Proc. of ECML/PKDD, 2004.

[10] Alexandrin Popescul, Lyle H. Ungar, Steve Lawrence, and David M. Pennock. Statistical relational learning for document mining. In Proceedings of IEEE International Conference on Data Mining (ICDM- 2003), pages 275–282, 2003.

[11]Multi-Label Classification: An Overview,Grigorios Tsoumakas,Ioannis Katakis,Dept. of Informatics, Aristotle University of Thessaloniki, 54124, Greece

[12] A. McCallum and K. Nigam, "A Comparison of Event Models Text forNaive Bayes Text Classification," AAAI-98 Workshop on "Learning for Categorization"

[13] Sun, A., & Lim, E.-P. (2001). Hierarchical text classification and evaluation. Proceedings of the 2001 IEEE International Conference on Data Mining (ICDM'01) (pp. 521ñ528). IEEE Computer Society.

[14] S. Dumais and H. Chen. Hierarchical classification of web content. In SIGIR '00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval,pages 256–263, New York, NY, USA, 2000.

[15] Daphne Koller and Mehran Sahami. Hierarchically classifying documents using very few words. In Proc. of ICML, 1997.

[16] Ryan Rifkin and Jason D. M. Rennie. Improving multi-class text classification with the support vector machine. In AI Memo, AIM-2001- 026,MIT, 2001.

[17] Shantanu Godbole, Abhay Harpale, Sunita Sarawagi, and Soumen Chakrabarti. Document classification through interactive supervision of document and term labels. In Proc. of ECML/PKDD, 2004.

[18] David D. Lewis, 1992. "Feature Selection and Feature Extraction for Text Categorization, appeared in Speech and Natural Language", Proceedings of a workshop held at Harriman, New York, February 23-26, 1992. Morgan Kaufmann, San Mateo, CA, pp. 212-217.

[19] Soucy, P. & Mineau, P. (2001), A simple feature selection method for text classification. In Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence (pp. 897–902).

[20] Nadia Ghamrawi and Andrew McCallum. Collective multi-label classification. In Proc. of CIKM, 2005.

[21] Tsoumakas, G., Katakis, I., Vlahavas, I.: Mining Multi-label Data. In: Maimon, O., Rokach, L. (eds.) Data Mining and Knowledge Discovery Handbook, 2nd edition, Springer, Heidelberg. (2010)

[22] Nadia Ghamrawi and Andrew McCallum. Collective multi-label classification. In Proc. of CIKM, 2005.

[23] Yiming Yang. A study of threshold strategies for text categorization.In Proc. of SIGIR, 2001.

[24] Shantanu Godbole. Exploiting confusion matrices for automatic generation of topic hierarchies and scaling up multi-way classifiers. In Technical report, IIT Bombay, 2002.

[25] L. Cai and T. Hofmann. Hierarchical document categorization with support vector machines. In CIKM'04: Proceedings of the thirteenth ACM international conference on Information and knowledge management, pages 78–87, New York, NY, USA, 2004.

[26] Schietgat, L., Blockeel, H., Struyf, J., Džeroski, S., Clare, A.: Decision Trees for Hierarchical Multilabel Classification: A Case Study in Functional Genomics.Lecture Notes in Computer Science, LNAI, Vol. 4213, 18-29. (2006)

[27] [Mit98] Tom Mitchell. Conditions for the equivalence of hierarchical and flat Bayesian classifiers. Technical note, Online at http://www.cs.cmu.edu/~tom/hierproof.ps, 1998.

[28] Text classification and Naïve Bayes. DRAFT! © April 1, 2009 Cambridge University Press.

[29] http://archive.ics.uci.edu/ml/datasets.html