Two Random FCNB-based models for Systematic Reviews

Bo Liu

December, 2013

1 INTRODUCTION AND BACKGROUND

In this project, I tried to use knowledge on Machine Learning to help experts reduce workload in Systematic Reviews for medical papers. A systematic review is a research study that collects and looks at multiple studies. Researchers use methods that are determined before they begin to frame one or more questions, then they find and analyse the studies that relate to that question. In this paper, I first review the Factorized Complement Naive Bayes (FCNB) by Matwin *et al*[4] and then propose two different ensemble randomized methods RFCNB-I and RFCNB-F for this specifc task. In the Experiments section, I applied the two methods to four different datasets and compare the performances with results from Cohen *et al* [1] and Matwin *et al* [4]. And the experiments show that these randomized ensemble algorithms can be useful for machine-learning-based automation of systematic reviews of drug class efficacy for disease treatement.

2 METHODS

I used a novel, modified, factorized complemental naive bayes (FCNB) as my basis classification algorithm. And then I proposed two novel randomized versions of FCNB. So in this section, I will introduce FCNB and the corresponding modified FCNB in 2.1 and propose the two RFCNBs in 2.2 and 2.3.

2.1 FCNB algorithm

The skewed data - more training examples for one class than another - can cause the decision boundary weights to be biased. [3] To deal with skewed training data, Rennie JD *et al* introduce a "complement class" version of Naive Bayes, called Complement Naive Bayes (CNB). And a number of experiments have been done to show that CNB can

be indeed employed in practical classification tasks with imbalanced datasets. But in our systematic review task, the demand of very high recall (no less than 95%) can not be fulfilled with CNB model. To tackle the poor recall of the original CNB, Matwin *et al*[4] added a heuristic weight factorization technique to the CNB algorithm. After adding the factor Fc∈[0, 1], the classification rule was changed as following:

$$l_{FCNB(d)} = argmax_c \left[logp(\theta_c) - F_c \sum f_i log \frac{N_{\bar{c}i} + \alpha i}{N_{\bar{c}} + \alpha} \right] \tag{1}$$

where $p(\Theta_c)$ is the class prior estimate, f_j is the frequency count of feature i (the word i) in document d, $N_{\bar{c}i}$ is the number of times feature i occurred in documents of classes other than c, and $N_{\bar{c}}$ is the total number of feature occurrences in classes other than c, α_i is a smoothing parameter (1 is commonly set in practice); α denotes the sum of the α_i.

From the above equation, we can find F_c is used to improve the recall on the minority class of relevant abstracts. Specifically, F_c can be set to 1 when we compute for the non-relevant class and $F_c<1$ will be used when c represents the relevant class. More details about this algorithm can be found in their papers. And the experiments show this method can work well in this specific SR task.

As a result, I decide to use this method as my basis algorithm. And before proposing the two randomized algorithm I made a minor modification to formula (1) :

$$l_{FCNB(d)^*} = argmax_c \left\{ F_c \left[logp(\theta_c) - \sum f_i log \frac{N_{\bar{c}i} + \alpha i}{N_{\bar{c}} + \alpha} \right] \right\} \tag{2}$$

As discussed above, $p(\Theta_c)$ is the class prior estimate, the value of $logp(\Theta_c)$ is fixed once the training data is given. Therefore, changing the position of the factor F_c will not influence the performance of the model but it will make it more practical to implement and easier to understand. To differentiate it from FCNB [4], FCNB* is used to represent the model.

With such a classifier for the SR task we can start to explore an ensemble technique to get a more robust and effective model. And the basic idea for the ensemble method is similar to Random Forests [9], which are a combination of decision trees. And in this paper, I applied

two randomization methods - random feature selection and random instance selection - while training the forest of bayes classifiers.

2.2 Random Feature Selection FCNB

To our knowledge, Naive Bayes is a probabilistic classifier based on the assumption of class conditional independence. Similarly, FCNB, another version of multinomial naive bayes, is also a learning algorithm with an assumption that any word's occurrence is independent of the occurrences of all the other words and itself. To decrease the influence of such an assumption, Prinzie A *et al* [5] propose a bagged classifier combining a 'forest' of NBs called Random Naive Bayes (RNB) and each NB is obtained on a bootstrap sample with m randomly selected features. And the experiments show that the predictive performance of RNB can outperform SVM.

To combine such a method with FCNB*, I propose the model of Random Feature Selection FCNB (RFCNB-F), a combination of a 'forest' of FCNB classifiers, and each FCNB* classifier is trained from a random subset of given features. For instance, the corpus for a given set of training documents consist of 1000 different words and 10% is used as the random percentage. Then every time we train a FCNB* classifier, we randomly select 100 words from the 1000 corpus words as the training features and use all the documents as training instances. After training a number of classifiers (e.g. 100 classifiers), we can use a simple voting scheme to determine the label of one document.

2.3 Random Instance Selection FCNB

Another idea is to use all the features while selecting the training instances randomly. However, due to the highly imbalanced data, we can not achieve a satisfactory result if we only use a simple bagging method to select the instances randomly. So to make a good use of the minority class (relevant documents), I propose another bagging method for our Random Instance Selection FCNB (RFCNB-I). The bagging phase is easy to illustrate and implement:

First extract all the instances of the minority class from the training data;

Then extract a same amount instances of the majority class randomly from the training data;

Combine the two sets of instances to generate the final training set.

After obtaining this new training set, we can train a FCNB* model using classification rule of formula (2). Then repeat the whole process several times to get a combination of FCNB* classifiers. Lastly, with the same voting method as FCNB-F, i.e. Majority Voting, we can predict whether a document is relevant.

Additionally, we may find that the training set is 'balanced' after such a bagging process because the number of the relevant and non-relevant documents is equal for a specific RFCNB-I classifier. And questions like "why not use a MNB model instead of the CNB model" may be asked. The answer is that although for a specific classifier the training set is balanced, the training set for the whole RFCNB-I is still imbalanced because we randomly select the non-relevant documents many times which can cover all the data of the original training documents. And experiments has been done to show that MNB can not be employed in this scheme for the SR task.

3 EXPERIMENTS

In my experiments, I use the same 5×2 cross-validation method used in the paper[1] for the comparison work. In 5×2 cross-validation, the dataset is randomly partitioned to two same-size subsets. One dataset is for training while the other one is for evaluation and then exchange the roles of the two subsets. Repeat the above process 5 times which can produce $5 \times 2 = 10$ sets of scoring results. The 5×2 cross-validation approach is more realistic than the 10-fold cross-validation method because the latter one always overestimates performance.[6]

3.1 Data-Preprocessing

In this SR task, we are provided with 4 drug groups which are the same data used by Cohen *et al*[1] and Matwin *et al*[4], Estrogens, OralHypoglemics, Triptans and BetaBlockers. Each instance consist of five attributes: Class (Relevant and Non-relevant), Title (title of the document), Abstract (text of the abstract), Publication Type, MeSH (Medical Subject Headings which is a medical thesaurus[7], a hierarchical structure of descriptors (tags) representing the US National Library of Medicine's controlled vocabulary used for medical information indexing and retrieval). In Matwin's[4] FCNB/WE method, they employ a WE (weight engineering) technique, that is to give different weights to the two different types (frequency-based representation and binary representation) during data preprocessing phase. The results show this WE technique can be beneficial to the classification performance. But in this paper, I have not used such a WE technique and decide to leave it for a future study.

Drug Class Review	Abstracts	Relevant Abstracts	Terms (After Preprocessing)
Estrogens	368	80	1066
OralHypoglycemics	503	139	946
Triptans	671	218	748
BetaBlockers	2072	302	934

Table 1: Datasets' description after preprocessing

Instead, I decide to use the Bag-of-words (BOW) representation to code each text collection. Here I use R to transform the documents to a corpus and then transpose each document to a row of a document-term matrix after removing the numbers, punctuation and stopwords (SMART Word List[8]) in the corpus. Lastly, I removed 98% sparse terms from the term-document matrix (the resulting matrix contains only terms with a sparse factor of less than 98%). After finishing the preprocessing phase, the final datasets (showing in Table 1) can be obtained.

3.2 Evaluation-Metric

In this experiment, I use WSS@95%, the method introduced by Cohen *et al* [2] to evaluate the performance of the two RFCNB models. WSS@95% is a WSS interpolation for recall at 0.95. And WSS is defined as:

$$WSS = (TN + FN)/N - (1 - R) \tag{3}$$

where TN (true negatives) is the number of non-relevant abstracts correctly classified, FN (false negative) is the number of relevant abstracts incorrectly classified. N is the total number of abstracts in the test set and R represents the Recall.

3.3 Results

To be able to compare the performance of the two RFCNB models (RFCNB-F and RFCNB-I), I performed the experiments to evaluate the two models at the same time. In other words, each time the dataset is split to half, I will use the same training subset to fit the

Drug Review Topics	Best VP	FCNB	FCNB/WE	RFCNB-F	RFCNB-I
Estrogens	12.8	22.0	**37.5**	28.2	27.3
BetaBlockers	22.0	27.0	**36.7**	15.1	23.3
OralHypoglycemics	3.4	6.1	8.5	10.4	**13.2**
Triptans	0.9	14.1	27.4	33.2	**35.2**
Sum	39.1	69.2	110.1	86.9	99
Average	9.775	17.3	27.525	21.725	24.75

Table 2: Work saved over sampling results, in percentages, for 5×2 cross validation experiments with RFCNB-F and RFCNB-I

two models and use the same other evaluation subset to evaluate them.

As for factor selection, I used two different schemes. For RFCNB-F model, everytime before training a model, I split the training set (without feature selection) to half and used one subset to train a FCNB* model with a initial factor (e.g. 0.8) then calculate the recall using this model to predict the other subset. If the recall is greater than 0.95, the factor will increase by 0.01. The experiments show that the factor achieved by this process can works well if we substract the factor by 0.03. As for RFCNB-I model, I use a manual method instead but I believe the process can also be implemented by a simple automatic technique.

Table 2 contains the main detailed results of my experiments in comparison with the results of Cohen *et al*[1] and Matwin *et al*[4]. Please note that the two models RFCNB-F (column 5) and RFCNB-I (column 6) are based on FCNB* (Formular (2)) while FCNB (column 3) is the method used by paper [4] and based on Formula (1). To train a RFCNB-F, I use 20% as feature selection percentage and train 100 classifiers to combine one model. At the same time, to train a RFCNB-I, I train 30 FCNB* classifiers to form a RFCNB-I, 30 seems to be small but experiments show it is enough for the task.

3.4 Discussion

The results in Table 2 shows that RFCNB-F and RFCNB-I can be recommended as alternative algorithms for Systematic reviews. Overall, the two random algorithms have beaten the scores obtained by Cohen *et al*[1] except the score for BetaBlockers by RFCNB-F. And we can find for the other 3 datasets, the scores have been increased significantly by both random algorithms comparing to the VP method. To my surprise, there are two results of RFCNB-I which are even greater

than the scores of FCNB/WE - Triptans (35.2 vs 27.4) and OralHypoglycemics (13.2 vs 8.5). When we compare the performances of the two ensemble methods, we can see except for the dataset Estrogens, RFCNB-I can achieve better WSS scores than the feature selection method to some extent. As a result, I prefer to choose RFCNB-I for other datasets in future work.

In future work, to improve the peformance of the instance-selection random FCNB (RFCNB-I), I prepare to integrate FCNB/WE with the instance-selection method. Because the experiments[4] show this modified FCNB with weight engineering can perform much better than original FCNB. I believe this will be a good try to replace FCNB* with FCNB/WE while training a RFCNB-I model. Addtionally, implementing such a model is quite pracical because the instance-selection method does not conflict with this weight engineering modification.

4 REFERENCES

1. Cohen AM, Hersh WR, Peterson K, *et al.* Reducing workload in systematic review preparation using automated citation\classification. J Am Med Inform Assoc 2006;13:206-19.

2. Cohen A. Optimizing feature representation for automated systematic review work prioritization. AMIA Annu Symp Proc 2008:121-5.

3. Rennie JD, Shih L, Teevan J, *et al.* Tackling the poor assumptions of Naïve Bayes text classifiers. In: Proc Int Conf on Machine Learning. 2003:616-23.

4. Matwin S, Kouznetsov A, Inkpen D, Frunza O. A new algorithm for reducing the workload of experts in performing systematic reviews. Journal of the American Medical Informatics Association 2010;17:446–53.

5. Prinzie A, Van den Poel D. Random multiclass classification: Generalizing random forests to random mnl and random nb[C]//Database and Expert Systems Applications. Springer Berlin Heidelberg, 2007: 349-358.

6. Dietterich TG. Approximate statistical tests for comparing supervised classification learning algorithms. Neural Comput 1998;10:1895-924.

7. United States National Library of Medicine. Medical subject headings. http://www.nlm.nih.gov/mesh/overview.html.

8. English stopwords from the SMART information retrieval system. http://jmlr.org/papers/volume5/lewis04a/a11-smart-stop-list/english.stop

9. Breiman L. Random forests[J]. Machine learning, 2001, 45(1): 5-32.