A Hybrid Approach of Association Rule & Hidden Makov Model to Improve Efficiency Medical Text Classification

Huda Ali Al-qozani
Department of Computer Science
University of Thamar
Thamar, Yemen

Khalil saeed Al-wagih
Department of Information Technology
University of Thamar
Thamar, Yemen

Abstract: Text classification problem is a set of documents be classified into a predefined set of categories, each document is classified based on a set of features (words). However, some of the words not relevant to a category which causes a gap between words relevance in a document. A lot of research articles in public databases, and The digitization of critical medical information such as lab reports, patients records, research papers, and anatomic images tremendous amounts of biomedical research data are generated every day. So that, the classification this data and retrieving information relevant to information users' needs have been a primary research issue in the field of Information Retrieval, and the adoption of classification has been applied to tackle this particular problem. In this paper, we propose a hybrid model for the classification of biomedical texts according to their content, using Association Rules and Hidden Markov Model as classifier. In order to demonstrate it, we present a set of experiments performed on OHSUMED biomedical text corpora. Our classifier compared with Naive Bayes and Support Vector Machine models. The evaluation result shows that the proposed classification is complete and accurate when compared with Naive Bayes and Support Vector Machine models.

Keywords: Hidden Markov Model, Association Rules, Biomedical Text, Text Classification, Machine learning, Text mining, Information Retrieved.

1. INTRODUCTION

The field of biomedical informatics has drawn increasing attention and has been growing rapidly. The amounts of biomedical research data are generated every day in public databases such as OHSUMED or elsewhere, has come to a growing realization that such data contains buried within it knowledge, knowledge that could lead to important discoveries in science, knowledge that could enable us accurately to predict the diseases. The knowledge that could enable us to identify the causes of and possible cures for lethal illnesses, a knowledge that could literally mean the difference between life and death. It has rightly been said that the world is becoming 'data rich but knowledge poor', These data need to be effectively organized and analyzed in order to be useful [18].

In the another side knowledge management practices often need to leverage existing clinical decision support, information retrieval (IR), and digital library techniques to capture and deliver tacit and explicit biomedical knowledge. Text mining techniques have been used to analyses research publications as well as electronic patient records [9]. The task of automatic classification is a relatively new IR sub field. Since Machine Learning (ML) serves as a theoretical foundation for the methodologies in this task, its scope is often referred to as the intersection of IR and ML[46].

Text classification (TC) may be formalized as the task of approximating the unknown target function f : D x C { - 1 , 1} that corresponds to how documents would be classified . The function f is the text classifier, C = { c1,c2,… ,cj,... ,c |C|} is a pre-defined set of categories and D is a set of documents. Each document is represented using the set of features, usually words, W = { w1, w2, . . . ,wk, . . . , wW } , with each one as a vector di = { wi1, wi2, . . . ,wik, . . . , wi | W |}, where wik describes each feature's representation for that specific document. When f (di,cj)= 1, di is a positive example or member of category cj , whilst when f (di,cj) = 0 it is a negative example of cj. The goal of this paper is to categorize electronic biomedical texts to one or more categories automatically[39]. The following part moves on to describe the methods used in different aspects of TC. The Naive Bayes (NB) model has been one of the more popular methods used in TC due to its simplicity and relative effectiveness [7, 27, 30]. However, the performance of the NB model has turned out to be inferior to other models such as Support Vector Machine (SVM) [19], k-Near Neighbor (KNN) [43], Neural Network (NN) [44]. The outcome of many studies confirms that there is no single TC model instead. Distinct models seem to be robust for different aspects of TC and within different contexts such as KNN-based models are easily scalable to large data sets [43], NN-based are best suitable for applications to obscure intrinsic structures [37], NB-based are appropriate for their simplicity and extensibility to web documents with links [26] and SVM-based may be used for their resistance to over-fitting and large dimensionality [14].

Hidden Markov Model (HMM) has been used to describe a sequential random process[41, 2]. Association Rule Mining (ARM) is to examine the contents of the database and find rules[7]. Another significant aspect of this study, the surveys of biomedical text mining [50, 49], journal [8], and book [3] indicate that general purpose text and data mining tools are not well-suited for the biomedical domain. The biomedical domain is highly specialized, but biomedical information is being created in text forms [40].

In this paper, a hybrid association rule and hidden markov model (AR-HMM) is investigated to prove the effectiveness of the proposed method, it is compared with SVM and NB. Rest of this paper is organized as follows: section 2, describes the methods and materials which used in this study, also present the performance measurements which are used to evaluate the categorization models. section 3, the results and discussion are presented, then reviews the most related work of Hidden markov model and association rules. in section 4, the conclusions. Finally, the last section presents the conclusions.

2. METHODOLOGY

TC is the process of assigning predefined category labels to new documents based on the classifier learnt from training examples. Text mining can be defined similar to data mining as the application of algorithms and methods from the field machine learning and statistics. To the dataset usually comprises the documents themselves, and the features are extracted from the documents automatically based on their content before the classification algorithm is applied. For this purpose it is necessary to pre-process the texts accordingly. See Figure 1 which can be divided text Classification into four components as displayed: 1- Dataset, 2- preprocessing, 3- learning, and 4- Evaluation.

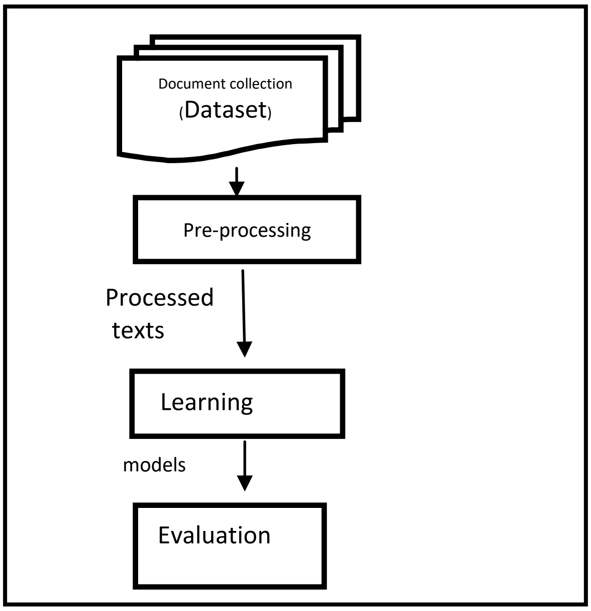


Figure1.Text classification process

In our context, the dataset comprises the relevant and non-relevant classes. Next, the preprocessing step such as (remove stop word and stemming) is applied. The features are extracted from the documents automatically based on their content. After that, apirior algorithm is applied on our dataset for each classes to determine the hidden states for hmm classifier, and classify the document by hmm algorithm. In the last, the model is evaluated which consists of a set of pre-classified documents in categories, the idea of building a classifier is based on the implicit relation between the characteristics of the document and its class by association words. See figure 2 which illustrates the overall process of the text Classification based on AR-HMM, and the following sub section is detailed the process.

2.1 Representation of Text Document

Typically, text documents are unstructured data. Before learning, one must transform them into a representation that is suitable for computing. Once the features in the documents, usually words or terms, are extracted, each document is represented in a vector space. also known as the bag-of-words(BOW), widely used in information retrieval [25]. This representation method is equivalent to an attribute value representation used in ML [4]. Each dimension of this space represents a single feature, whose importance in the document corresponds to the exact distance from the origin. Documents are thus points (vectors) in a|W|-dimensional vector space, where|W| denotes the dimension of the vocabulary or dictionary, W ={w1,w2,.. .,wk,...,w|W |}, every document is represented as a vector di =(wi1,wi2,...,wik,...,wi|W |), where wik describes each feature word in the dictionary, for the document i.

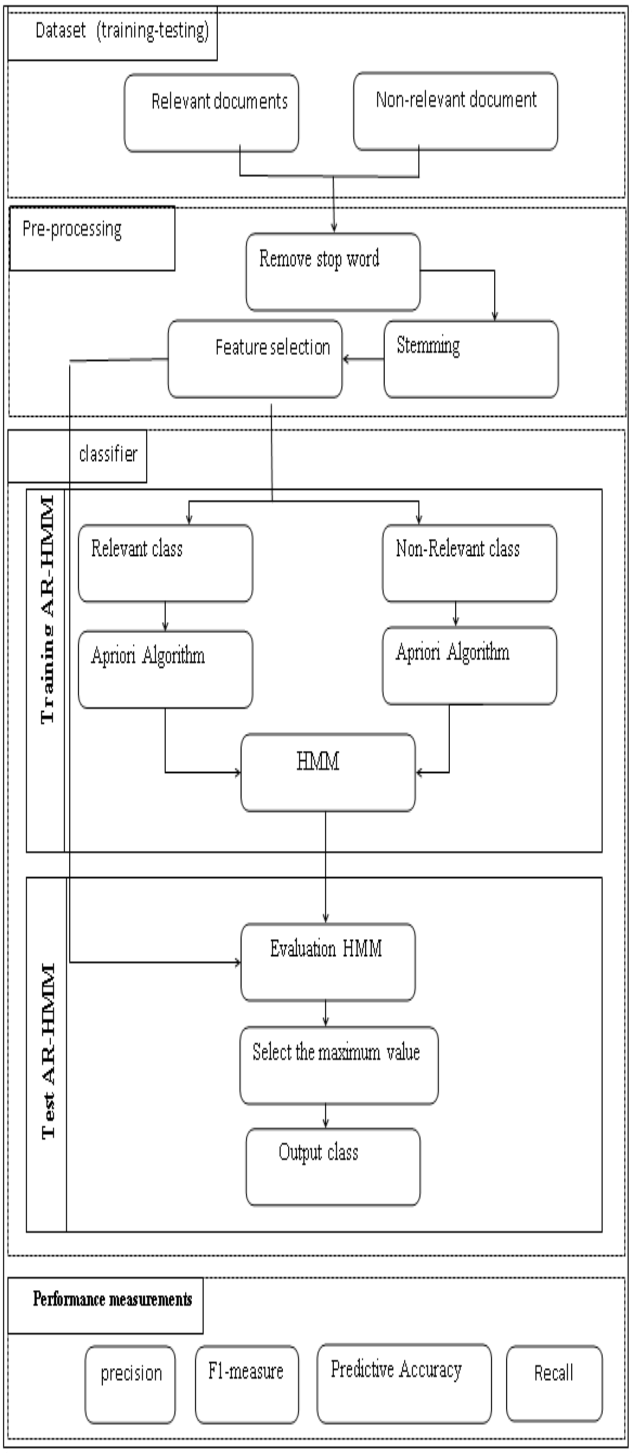
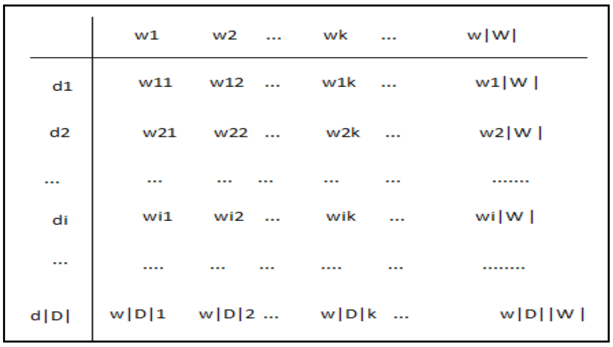


Figure 2.Text classification process

2.2 Stop Words and Stemming

Where rare words which do not provide any useful information such as(prepositions, determiners or conjunctions) are removed.

Another very important way to reduce the number of words in the representation is to use stemming. This is based on the observation that words in documents often have many morphological variants. For example we may use the words computing, computer, computation, computes, computational, computable and computability all in the same document. These words clearly have the same linguistic root. Putting them together as if they were occurrences of a single word would probably give a strong indication of the content of the document whereas each word individually might not [7]. Finally, all documents in the collection are mapped to a matrix called the term. The document matrix representing the feature space, each row of the matrix corresponds to a document, and the columns of the matrix correspond to the unique terms in the document collection. See figure 3 each intersection (wik) represents the TFIDF weight of term wk in document di to measure the word relevance.



	w1	w2	...	wk	...	w|W|
d1	w11	w12	...	w1k	...	w1|W |
d2	w21	w22	...	w2k	...	w2|W |
...	...	...	...	...	...	....
di	wi1	wi2	...	wik	...	wi|W |
...	....	...	...	....	...	....
d|D|	w|D|1	w|D|2	...	w|D|k	...	w|D||W |

Figure 3.The Term Matrix

2.3 The feature selection

Feature selection methods aim at choosing from the available set of terms a smaller set that more efficiently represents the documents. Feature selection is not needed for all classification algorithms as some classifiers are capable of feature selection themselves. However for other classifiers feature selection is mandatory, since a large number of irrelevant features can significantly impair classifier accuracy[39]. The feature selection method based in Information Gain that is implemented in WEKA[13] is used as the feature reduction algorithm for this dataset When building an AR-HMM model.

2.4 Association Rules

The aim of Association Rule Mining (ARM) is to examine the contents of the database and find rules, known as association rules[7]. The discovery of interesting association relationships among huge amounts of transaction records can help in many decision making processes. In most of the ARM is to evaluate rules from a two basic measures called support and confidence. Support(s) were defined as the parts of record that come together X and Y to the total number of records in the dataset. Confidence was calculated as percentage of transactions that contain X and Y to the total number of records that contain X, where if the percentage exceeds of confidence threshold [29].

2.5 Apriori algorithm

Apriori is a well-known association rule learning algorithm, for finding frequent items over transactional data sources .The initial idea of Apriori algorithm[1] is derived from the shopping cart transactions that strive about the set of items purchase frequently. Among different algorithms that can be used to derive frequent item sets, FP-growth (Frequent pattern growth) uses extend prefix-tree structure to store the database in a compressed form. From the various researchers found results in their work by using different applications, apriori algorithm is suitable and mostly utilized in their chosen domains. Also, apriori algorithm is widely applied in the domain of medical care. The common approach of the apriori algorithm is run into two passes. The first pass of the algorithm is to simply counts item occurrences to determine the large 1-item sets. Subsequently a second pass say k, consist of two stages. First one is the large item sets Lk-1 found in the (k-1)th pass that used to generate the candidate item sets Ck by the apriori function. Next, the database is scanned and the support of candidates in Ck is counted. Basic principles of the Apriori Algorithm are demonstrated as follows:

- To find the set of frequent 1-itemsets. Lk is completed through scan the data and accumulates the count of each item to see the minimum support in a new set called Lk.
- It uses Lk to find Ck+1 (the set of candidate 2itemsets) is a two-step process that first generates Ck+1 based on Lk and secondly prunes Ck+1 by getting rid of those Ck+1 itemsets using the apriori method.
- It is to find Lk+1: we do this by finding the support count for all the itemsets in Ck+1 and getting rid of those that are below the minsup.
- It continues step 2 and 3 until no new frequent (k+1)itemset are found.

2.6 The Classifier Algorithm

Many different types of supervised learners have been used in text classification, including probabilistic Naive Bayesian methods [38] [5], Bayesian Networks [45], Decision Trees [11], Decision Rules [33], Neural Networks [32], Support Vector Machines [24], Hidden Markov Model [20], and association rules [28]. In this paper, HMM been used as classifier algorithm.

2.6.1 Hidden markov model

The Hidden Markov Model(HMM) is a powerful statistical tool for modeling generative sequences that can be characterized by an underlying process generating an observable sequence. A generic Hidden Markov model is illustrated in Figure 4, where the Xi represent the hidden state sequence The Markov process which is hidden behind the dashed line's determined by the current state and the A matrix. We are only able to observe the Oi, which are related to the (hidden) states of the Markov process by the matrix B.

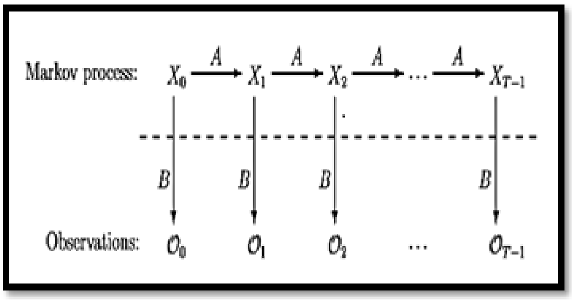


Figure 4. Hidden Markov Model

the most probable observations for the first state are the most relevant words in the corpus. a Hidden Markov model is proposed to represent a predefined category c as follows:

- The union of words from the training corpus is taken as the set of observation symbols V . For each word, there is a symbol vk. The set of possible observations is the same for every HMM, taking into account all words in the corpus, regardless of their category.
- states represent ranking positions. Therefore, states are ordered from the first rank to the last one. The state transitions are ordered sequentially in the same way, forming a left-right HMM [6] without self-state loops.
- The observation output probability distribution of each state is defined according to the training corpus and category c. A word/observation vk will have a higher output probability at a given state si if the word appears frequently with the same ranking position that si represents.
- The initial probability distribution p is defined by giving probability 1 to the first state s0. Once the two Hidden Markov models are created and trained (one for each category), a new document d can be classified by, first of all, formatting it into an ordered wordlist Ld in the same way as in the training process.

Then, as words are considered observations in our HMM, we calculate the probability of the word sequence Ld being produced by the two HMMs. That is, P(Ld—R) and P(Ld—N) need to be computed, where R is the model for relevant documents and N the model for non-relevant documents. The final output class for document d will be the class represented by the HMM with the highest calculated probability.

2.7 Performance measurements

measuring the performance of a classifier is by its predictive accuracy, i.e. the proportion of unseen instances it correctly classifies. However this is not necessarily the case. There are many other types of classification algorithm as well as :

- True Positive: positive instances that are correctly classified as positive.
- False Positive: negative instances that are erroneously classified as positive.
- False Negative: positive instances that are erroneously classified as negative = 1 - True Positive Rate.
- True Negative: negative instances that are correctly classified as negative.
- Precision: Proportion of instances classified as Positive that are really positive.
  precision =TP /TP + FP. (1)
- F1 Score A measure that combine Precision and Recall.
  F1 =2×precision×Recall/ Precision + Recall. (2)

3. RESULTS AND DISCUSSION

To demonstrate the efficiency of the algorithm, a set of experiments is presented which have been performed on the OHSUMED biomedical corpus [17], each document in the set has one or more associated category from 23 disease categories.

3.1 Preparation of Datasets

One of these categories is elected as relevant and consider the others as non-relevant. Five categories are chosen as relevant: Neoplasms (C04), Digestive (C06), Car-dio (C14), Immunology (C20) and Pathology (C23).The other 18 categories are considered as the common bag of non-relevant documents. For each one of the five relevant categories, corpora need to be pre-processed. Every document is formatted into a vector of feature words which have been described the word occurrence frequencies. All the different words that appear in the training corpus are candidates for feature words. To reduce the initial feature size, standard text pre-processing techniques are used. A predefined list of stopwords common English words is removed from the text and a stemmer based on the Lovins stemmer is applied. The feature selection method based in Information Gain is used as the feature reduction algorithm, ending up with five distinct matrices. That split into two matric relevant and non-relevant, for each matric was been inputted as input to Aporior algorithm. The feature wordset was sampled by the selection of the higher 100 from the results of above steps, a different corpus is created in the way mentioned above.

Most common (traditional) way of representing documents for text ARM purposes is the Vector Space Model where documents are represented as a single, high dimensional, numeric vector d (where d is a subset of some vocabulary V). A major concern of the apriori algorithm is the high computational time needed to find frequent rule items from all possible candidates at each level. The output form the apriori was presented the number of state based on the support threshold. We use text classification models such as NB and SVM with these classifiers using the same corpus in order to compare them with camper with our hybrid model were applying a gaussian the kernel of SVM.

3.2 Experiments and results

The proposed algorithm AR-HMM, SVM and NB are implemented by c# programming language, where the SVM was applied using a Gaussian kernel. The tests were made with these classifiers using the same corpus in order to compare SVM and NB with the AR-HMM. Table 1 shows the results obtained from the preliminary analysis of the experiments were carried out for the proposed AR-HMM, NB and SVM. The results have shown the performance as follows: The AR-HMM outperforms NB and SVM in accuracy, recall and F1 measure for N class with each corpus. While The AR-HMM outperforms NB and SVM in accuracy for R class with each C4, C14, C20 corpus. The NB outperforms AR_HMM and SVM in F1 measure for R class except for C20 corpus, also SVM outperforms AR-HMM and NB in the precision measure for R and N class with C4, C6, and C14.

3.3 Discussion and Related Work

3.3.1 Discussion

This section discusses the performances of classification by showing accuracy and F1 measures, the average for all corpus has taken with R and N classes. The accuracy is measured by the effectiveness and efficiency of the classifier and F1 is combined precision and recall measures . In the partially, Table 2 shows the results of accuracy measure on R and N classes. According to N class AR-HMM outperforms NB and SVM for each corpus, the AR-HMM gets 77%, 61%, 80%, 94%, 56% with C4, C6, C14, C20, and C23 respectively. In the case of R class AR-HMM gets 77%, 80%, 94%, with C4, C14, C20 respectively. While NB outperforms AR-HMM and SVM in C6 and C23. As table.2, the average for all corpus presents the order of accuracy algorithms, whereas the AR-HMM algorithm gets74%, then NB algorithm gets 66%, finally SVM algorithm gets 50% for R class. In N class, the AR-HMM gets 74%, the SVM gets 53%, and NB algorithm gets 34% as shown in figure 5 .

Table 1. The Result of Accuracy, Precision, Recall, F1, Kappa Measures With 65 The Minimum Sup on R and N Classes

	R				N			
Corpus	Measure	AR_HMM	NB	SVM	Measure	AR_HMM	NB	SVM
c4	Accuracy	**0.773**	**0.773**	0.646	Accuracy	**0.773**	0.227	0.665
	Precision	0.799	0.834	**0.915**	Precision	0.754	0.166	**0.916**
	Recall	**0.716**	0.696	0.588	Recall	**0.828**	0.145	0.603
	F1	0.755	**0.759**	0.716	F1	**0.789**	0.155	0.727
c6	Accuracy	0.610	**0.663**	0.489	Accuracy	**0.610**	0.337	0.530
	Precision	0.395	0.899	**1.000**	Precision	0.742	0.101	**1.000**
	Recall	0.483	**0.576**	0.378	Recall	**0.667**	0.144	0.398
	F1	0.434	**0.702**	0.548	F1	**0.702**	0.119	0.569
c14	Accuracy	**0.802**	**0.802**	0.686	Accuracy	**0.802**	0.198	0.642
	Precision	0.833	0.856	**0.921**	Precision	0.779	0.144	**0.921**
	Recall	0.736	**0.744**	0.615	Recall	**0.863**	0.136	0.581
	F1	0.781	**0.796**	0.738	F1	**0.819**	0.140	0.707
c20	Accuracy	**0.947**	0.425	0.437	Accuracy	**0.947**	0.575	0.563
	Precision	**1.000**	0.990	0.003	Precision	0.927	0.010	**0.997**
	Recall	**0.833**	0.156	0.004	Recall	**1.000**	0.003	0.422
	F1	**0.909**	0.270	0.004	F1	**0.962**	0.005	0.593
c23	Accuracy	0.561	**0.622**	0.261	Accuracy	**0.561**	0.378	0.272
	Precision	**0.670**	0.530	0.436	Precision	**0.473**	0.470	0.456
	Recall	0.509	**0.680**	0.389	Recall	**0.637**	0.417	0.399
	F1	0.579	**0.596**	0.411	F1	**0.543**	0.442	0.425

Table 2.The accuracy measure on R and N classes

Corpus	AR_HMM		NB		SVM	
	R	N	R	N	R	N
C4	**0.773**	**0.773**	**0.773**	0.227	0.646	0.665
C6	0.61	**0.61**	**0.663**	0.337	0.489	0.53
C14	**0.802**	**0.802**	**0.802**	0.198	0.686	0.642
C20	**0.947**	**0.947**	0.425	0.575	0.437	0.563
C23	0.561	**0.561**	**0.622**	0.378	0.261	0.272
Average	**74%**	**74%**	%66	34%	%50	%53

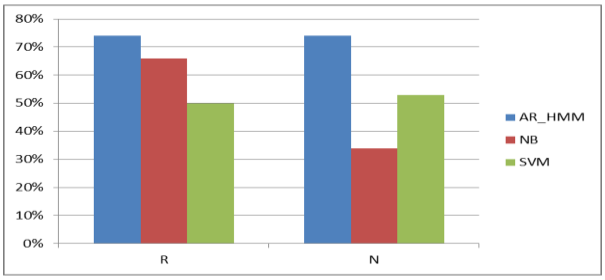


Figure 5. Average Accuracy For N and R Classes

In the table 3, shows the results of F1 measure on R and N classes. In the case N class, The AR-HMM gets 78%, 70%, 81%, 96%, 54% with C4, C6, C14, C20, and C23 respectively. In the case R, AR-HMM gets 90%, with C20. while NB have been getting 75%, 70%, 79%, 59% with C4, C6, C14, C23 respectively.

Table 3. Measure F1 for R and N classes

Corpus	AR_HMM		NB		SVM	
	R	N	R	N	R	N
C4	0.755	0.**789**	**0.759**	0.155	0.716	0.727
C6	0.434	0.**702**	0.**702**	0.119	0.548	0.569
C14	0.781	0.**819**	0.**796**	0.140	0.738	0.707
C20	0.**909**	0.**962**	0.270	0.005	0.004	0.593
C23	0.579	0.**543**	0.**596**	0.442	0.411	0.425
Average	**69%**	**76%**	63%	17%	48%	60%

The average for all test corpus in above table shows the results as :In R class, the AR-HMM algorithm gets 69%, NB algorithm gets 63%, and SVM algorithm gets 48%. According to N class the AR-HMM gets 76%, the SVM gets 60%, then NB algorithm gets 17 % for as shown in figure 6. In summary, for the informants in this analysis AR-HMM outperforms NB and SVM.

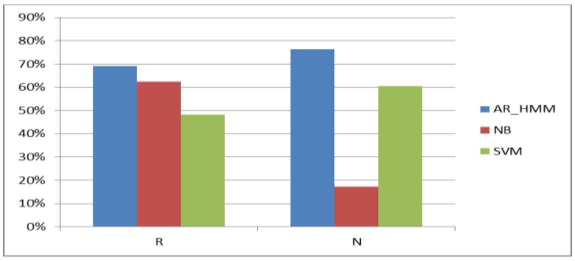


Figure 6. Average F1 For N and R Classes

3.3.2 *Related Work*

The theory of HMMs was developed in the late 1960s. HMM was used as a statistical model for sequential process application in temporal pattern recognition , i.e . speech[35], handwriting [36] and bioinformatics [41], [47]. the model has been extended to the text-related task such as information retrieval [31] information extraction [13], [26] text summarization [15] text categorization [6], [12], [42], [2] also the model has been turned to the hybrid and novel model [41], [22], [21] .In [31], the research use HMM in an information retrieval model. Given a set of documents and a query Q, the system searches a document D relevant to the query Q. It computes the probability that D is the relevant document in the users mind, given query Q, i.e P(D is R—Q), and ranks the documents based on this measure. The HMM is viewed as a generator of the query, and is used to estimate the probability that each document will be produced in the corpus. In the anther research[47], the research use the previous idea in a similar approach. They describe the text classification as the process of finding a relevant category c for a given documented. They implement a Hidden Markov Model to represent each category. Thus, given a document d, the probability that a document d belongs to category c is computed on the specific HMM model c.

In[17], The main idea of the article lies in setting up an HMM classifier, combining x2 and an improved TF-IDF method and reflecting the semantic relationship in the different categories. The process shows the semantic character in different documents to make the text categorization process more stable and accurate.by [6] proposed novel two tier prediction framework and present probabilistic model such as Markov model and association rule mining. The models gives better prediction accuracy without compromising prediction time but suffers to scale on larger datasets.

In [41] use hmm in an original model for the classification of biomedical texts stored in large document corpora. The model classifies scientific documents according to their content using information retrieval techniques and Hidden Markov Models, they present a set of experiments which have been performed on OHSUMED biomedical corpus, a subset of the MEDLINE database, and the Allele and GO TREC corpora. their classifier is also compared with Naive Bayes, k-NN and SVM techniques.

In the anther hand, Classification rules are concerned with predicting the value of a categorical attribute that has been identified as being of particular importance. Agrawal et al [1] introduced the AIS (Agrawal, Imielinski, Swami) algorithm for mining association rules, the algorithm focuses on improving the quality of databases along with the required functionality to process queries and consequent association rules are generated. In [2], the study presented an improved algorithm named apriori for As association rule mining in 1994 and found more efficient. Apriori is an influential algorithm for mining frequent itemsets for Boolean association rules. In [16], worked and designed a tree structure pattern mining algorithm called Frequent Pattern (FP)-Tree algorithm. The FP-Tree algorithm generates frequent itemsets by scanning the database only twice without any iteration process for candidate generation. The first one is FP-Tree construction process and the next one is the generation of frequent patterns from the FP-Tree through a procedure called FP-growth.

The FP-Growth Algorithm [15] is an alternative way to find frequent itemsets without using candidate key generations, thus improving performance. For so many, a divide-and-conquer strategy has been using. Here the database had been storing in the primary storage and to calculate the support of all generated set of patterns. In [28] presents a system for discovering association rule from the collections of unstructured documents called Extract Association Rules from Text (EART). The EART system has treated texts only not images or figures. The study[34] presented Continuous Association Rule Mining Algorithm (CARMA), an algorithm to compute large itemsets online.The algorithm needs, at most, two scans of the transaction sequence produce all large itemsets. During the first scan -Phase-I, the algorithm continuously con-structs a lattice of all potentially large itemsets. Phase-II initially removes all itemsets which are trivially small, i.e. itemsets with max Support below the last user-specified threshold. On the anther side[48] propose a new classification approach called classification based on multiple classification rules (CMR). It combines the advantages of both associative classification and rule-based classification.

4. CONCLUSION

Text classification is becoming a crucial task to analysts in different areas. In the last few decades, the production of textual documents in digital form has increased exponentially. Their applications range from web pages to scientific documents, including emails, news and books. This paper investigated hybrid hidden Markov model with association rules in automatic classification of biomedical text. were we use the apiroir algorithm to determinate the size of number of state hidden for hmm and we present a set of experiments which have been performed on OHSUMED biomedical corpus, a subset of the MEDLINE database. Our classifier outperforms commonly used text classification techniques like Naive Bayes and SVM techniques. In the whole process, there are still some areas that could be improved. firstly, our model were trained using a limited number of documents and terms second, using a method of data mining like a neural networks that can be make the rules dynamic or implementing use the algorithm in many application such as computational biology “DNA”, text retrieval, web searching, and handwriting.

5. REFERENCES

[1] Agrawal, R., Imielin´ski, T. and Swami, A.1993, “Mining association rules between sets of items in large databases,” in Acm sigmod record, vol. 22, pp. 207–216, ACM.

[2] Agrawal, R., Srikant, R. et al. , 1994, “Fast algorithms for mining association rules,” in Proc. 20th int. conf. very large data bases, VLDB, vol. 1215, pp. 487–499.

[3] Ananiadou, S. and McNaught, J. 2006.Text mining for biology and biomedicine. Artech House London.

[4] Androutsopoulos, I., Koutsias, J., Chandrinos, K. V. and Spyropoulos, C. D. 2000, “An experimental comparison of naive bayesian and keyword-based anti-spam filtering with personal e-mail messages,” in Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 160–167, ACM.

[5] Apt´e, C., Damerau, F. and Weiss, S. M. 1994, “Automated learning of decision rules for text categorization,” ACM Transactions on Information Systems (TOIS), vol. 12, no. 3, pp. 233–251.

[6] Awad, M.A, and Khalil, I. 2012, Prediction of user’s web-browsing behavior: Application of markov model. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 42(4):1131–1142.

[7] Bramer, M. 2007. Principles of data mining, vol. 180. Springer.

[8] Chapman, W. W. and Cohen, K. B. 2009, “Current issues in biomedical text mining and natural language processing,” Journal of biomedical informatics, vol. 42, no. 5, pp. 757–759.

[9] Chen, H., Fuller, S. S., Friedman, C. and Hersh, W. 2006. Medical informatics: knowledge management and data mining in biomedicine, vol. 8. Springer Science & Business Media.

[10] Conroy, J. M. and O’leary, D. P. 2001, “Text summarization via hidden markov models,” in Proceedings of the 24th annual international ACMSIGIR conference on Research and development in information retrieval, pp. 406–407, ACM.

[11] Dumais, S., Platt, J., Heckerman, D. and Sahami, M. 1998, “Inductive learning algorithms and representations for text categorization,” in Proceedings of the seventh international conference on Information and knowledge management, pp. 148–155, ACM.

[12] Frasconi, P., Soda, G. and Vullo, A. 2002, “Hidden markov models for text categorization in multi-page documents,” Journal of Intelligent Information Systems, vol. 18, no. 2-3, pp. 195–217.

[13] Freitag, D. and McCallum, A. 2000, “Information extraction with hmm structures learned by stochastic optimization,” AAAI/IAAI, vol. 2000, pp. 584–589.

[14] Glover, S. Rosenbaum, D. A. Graham, J. and Dixon, P. 2004, “Grasping the meaning of words,” Experimental Brain Research, vol. 154, no. 1, pp. 103–108.

[15] Grahne, G. and Zhu, J. 2005, “Fast algorithms for frequent itemset mining using fp-trees,” IEEE transactions on knowledge and data engineering, vol. 17, no. 10, pp. 1347– 1362.

[16] Han, J., Pei, J. and Yin, Y. 2000, “Mining frequent patterns without candidate generation,” in ACM sigmod record, vol. 29, pp. 1–12, ACM.

[17] Hersh, W., Buckley, C., Leone, T. and Hickam, D. 1994, “Ohsumed: An interactive retrieval evaluation and new large test collection for research,” in SIGIR94, pp. 192–201, Springer.

[18] Izenman, A J. 2008, Modern Multivariate Statistical Techniques: Regression, Classification, and Manifold Learning.

[19] Joachims, T. 1998. “Text categorization with support vector machines: Learning with many relevant features,” Machine learning: ECML-98, pp. 137– 142.

[20] John, G. H., Kohavi, R., Pfleger, K. et al. , 1994, “Irrelevant features and the subset selection problem,” in Machine learning: proceedings of the eleventh international conference, pp. 121–129.

[21] Krishnalal, G and Rengarajan, S and Srinivasagan, KG, 2010, ’ A new text mining approach based on HMM-SVM for web news classification ’,International Journal of Computer Applications,vol.1,no.19,pp.98–104,Citeseer .

[22] Khosronejad, M. and Sharififar, E., Torshizi, H. A.and Jalali, M., 2013,’ Developing a hybrid method of Hidden Markov Models and C5. 0 as a Intrusion Detection System ’,International Journal of Database Theory and Application,vol.6,no.5,pp.165–174

[23] Kairong Li, G., Chen and Cheng , J..2011 ,“Research on hidden markov model based text categorization process,” in International Journal of Digital Content Technology and its Application, pp. 244–251.

[24] Lam, W., Ruiz, M., and Srinivasan, P. 1999, “Automatic text categorization and its application to text retrieval,” IEEE Transactions on Knowledge and Data engineering, vol. 11, no. 6, pp. 865–879.

[25] Larkey, L. S. and Croft, W. B. 1996, “Combining classifiers in text categorization,” in Proceedings of the 19th annual international ACM SIGIR conference on

Research and development in information retrieval, pp. 289– 297, ACM.

[26] Leek, T. R. 1997. Information extraction using hidden Markov models. PhD thesis, University of California, San Diego.

[27] Lewis, D. D. 1998. "Naive(Bayes) at forty: The independence assumption in information retrieval," in European conference on machine learning, pp. 4–15, Springer.

[28] Mahgoub, H. and Ro¨sner, D. 2006, "Mining association rules from unstructured documents," in Proc. 3rd Int. Conf. on Knowledge Mining, ICKM, Prague, Czech Republic, pp. 167–172.

[29] Manimaran, J. and Velmurugan, T. 2013," A Survey of Association Rule Mining in Text applications", IEEE, pp.698-702.

[30] McCallum, A., Nigam, K. et al. , 1998, "A comparison of event models for naive bayes text classification," in AAAI-98 workshop on learning for text categorization, vol. 752, pp. 41–48, Citeseer.

[31] Miller, D. R., Leek, T. and Schwartz, R. M. 1999, "A hidden markov model information retrieval system," in Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 214–221, ACM.

[32] Mladeni´c, D. 1998, "Feature subset selection in text-learning," Machine learning: ECML-98, pp. 95–100.

[33] Moulinier, I., Raskinis, G., and Ganascia, J. 1996, "Text categorization: a symbolic approach," in proceedings of the fifth annual symposium on document analysis and information retrieval, pp. 87–99.

[34] Olmezogullari, E. and Ari, I. 2013, "Online association rule mining over fast data," in Big Data (BigData Congress), 2013 IEEE International Congress on, pp. 110–117, IEEE.

[35] Rabiner, L. R. 1989, "A tutorial on hidden markov models and selected applications in speech recognition," vol. 77, FEBRUARY.

[36] Rothacker, L. and Fink, G. A. 2016 ,"Robust output modeling in bag-of features hmms for handwriting recognition," in Frontiers in Handwriting Recognition (ICFHR), 15th International Conference on, pp. 199– 204, IEEE.

[37] Schweighofer, E. and Merkl, D. 1999. "A learning technique for legal document analysis," in Proceedings of the 7th international conference on Artificial intelligence and law, pp. 156–163, ACM.

[38] Sebastiani, F. et al. , 1999, "A tutorial on automated text categorisation," in Proceedings of ASAI-99, 1st Argentinian Symposium on Artificial Intelligence, pp. 7–35, Buenos Aires, AR.

[39] Silva, C. and Ribeiro, B. 2009. Inductive inference for large scale text classification: kernel approaches and techniques, vol. 255. Springer.

[40] Simpson, M. S. and Demner-Fushman, D. 2012, "Biomedical text mining: A survey of recent progress," in Mining text data, pp. 465–517, Springer.

[41] Vieira, A. S. Iglesias, E. L. and Borrajo, L. 2014. "T-hmm: a novel biomedical text classifier based on hidden markov models," in 8th International Conference on Practical Applications of Computational Biology & Bioinformatics (PACBB 2014), pp. 225–234, Springer.

[42] Xu, R., Supekar, K. S. , Huang, Y., Das, A. K., Garber, A.M .2006,' Combining Text Classification and Hidden Markov Modeling Techniques for Structuring Randomized Clinical Trial Abstracts ,McGill University ,AMIA.

[43] Yang, Z. 1997. "Paml: a program package for phylogenetic analysis by maximum likelihood," Computer applications in the biosciences: CABIOS, vol. 13, no. 5, pp. 555– 556.

[44] Yang, Y. and Liu, X. 1999. "A re-examination of text categorization methods," in Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 42–49, ACM.

[45] Yang, Y. and Pedersen, J. O. 1997, "A comparative study on feature selection in text categorization," in Icml, vol. 97, pp. 412–420.

[46] Yi, K. 2005. Text classification using a hidden Markov model. McGill University.

[47] Yi, K. and Beheshti, J. 2009, "A hidden markov model-based text classification of medical documents," Journal of Information Science, vol. 35, no. 1, pp. 67–81.

[48] Zhou, Z. 2014, "A new classification approach based on multiple classification rules," Mathematical Problems in Engineering, vol. 2014.

[49] Zweigenbaum, P. Demner-Fushman, D. Yu, H. and Cohen, K. B. 2007, "Frontiers of biomedical text mining: current progress," Briefings in bioinformatics, vol. 8, no. 5, pp. 358–375.

[50] Zweigenbaum, P. and Demner-Fushman, D. 2009, "Advanced literature-mining tools," in Bioinformatics, pp. 347–380, Springer.