ISSN: 0974-5823 Vol. 7 No. 5 May, 2022

International Journal of Mechanical Engineering

An Efficient Adaboost Ensemble System for Predicting Frequent Itemset in Weblog Dataset

A. Dhana Praveena,
Research Scholar,
Department of Computer Science,
Mother Teresa Women's University,
Kodaikanal,

Dr. V. Selvi,

Assistant Professor,

Department of Computer Science,

Mother Teresa Women's university,

Kodaikanal.

Abstract

Frequent item set mining in Educational field is an emerging discipline, concerned with developingmethods for exploring the unique types of data that come from educational settings,and using those methods to better understand students, and the settings which they learn in. Its main objective is to analyze these types of data in order to resolve educational research issues. It becomes an imperative and important research topicthat discovering hidden and useful knowledge from such an extensive quantity of datato guide and develop education.In our proposed work the behavior and surfing characteristics of students are considered. This work elucidates the process of identifying the student behavioral pattern by tracing the web browsing behavior of the students hidden in the log files of universities web server. The raw data is preprocessed, to generate organized information.After completion of preprocessing, next is to frequent itemset mining from given large dataset for further analysis. Apriori algorithm is used to find closely related attributes using support and confidence measures. From closely related attributes a number of association rules are mined. Among these attributes, best attributes are selected using firefly method for efficient classification process.The classifiers such as KNN, SVM and AdaBoost with ensemble SVM - KNN classifierare used to classify frequent item set.Finally, the proposed framework achieves a classification accuracy of 97%.

Keywords: Frequent Itemset mining, Firefly, ADABoost

1. Introduction

Now a days, huge amount of data is available freely in the World Wide Web for user access. So we need to manage and organized various types of data that can be accesses by different user very proficiently. Most of the researcher around the world is mainly focus for an application of data mining techniques. A variety of methods is available in data mining to determine the concealed information in the Web. Web mining has been developed especially to focus on this research area and also need new approaches to fit the properties of the web data. In the web log database use frequent data mining is to discover the various types of patterns such as frequent item sets, sequences and tree patterns. The frequent item set mining (FIM) is also called as association rule mining and used to explore the repeated items, patterns or events in the data set. The different types of algorithms in the frequent item set mining is required to mine the hidden data items with less memory consumption and shorter run time. FIM plays a major role in performing many different tasks in the concept of data mining. The various other applications supported by FPM are bug detection in software, biological analysis of data etc., To find the information of students hidden in the log files of universities web server for generating the association rules to be used for analysis the data.

Many algorithms had been developed for the techniques of frequent itemset mining and but still need some improvement for the existing algorithm of frequent item set mining because enormous amount of data set have in the web mining. The various FPM algorithms support for frequent item set mining such as Apriori algorithm, FP-Growth algorithm, EClaT algorithm, TreeProjection algorithm, COFI algorithm, TM algorithm, P-Mine algorithm, LP-Growth algorithm, Can-Mining algorithm, EXTRACT algorithm etc., The algorithm used in this proposed work is Apriori algorithm. This algorithm is used to mine the frequent dataset in the web and also generating some association rules. Apriori algorithm is the method of iterative search techniques to find the k+1 from k item sets. This method is first scan the entire university web server by tracing the behavior of students' item-set by counting each of them and to satisfy the minimum threshold. The architecture proposed consists of models of pre-processing, , Feature Extraction, Feature Selection and Classification. It is described in the below figure.

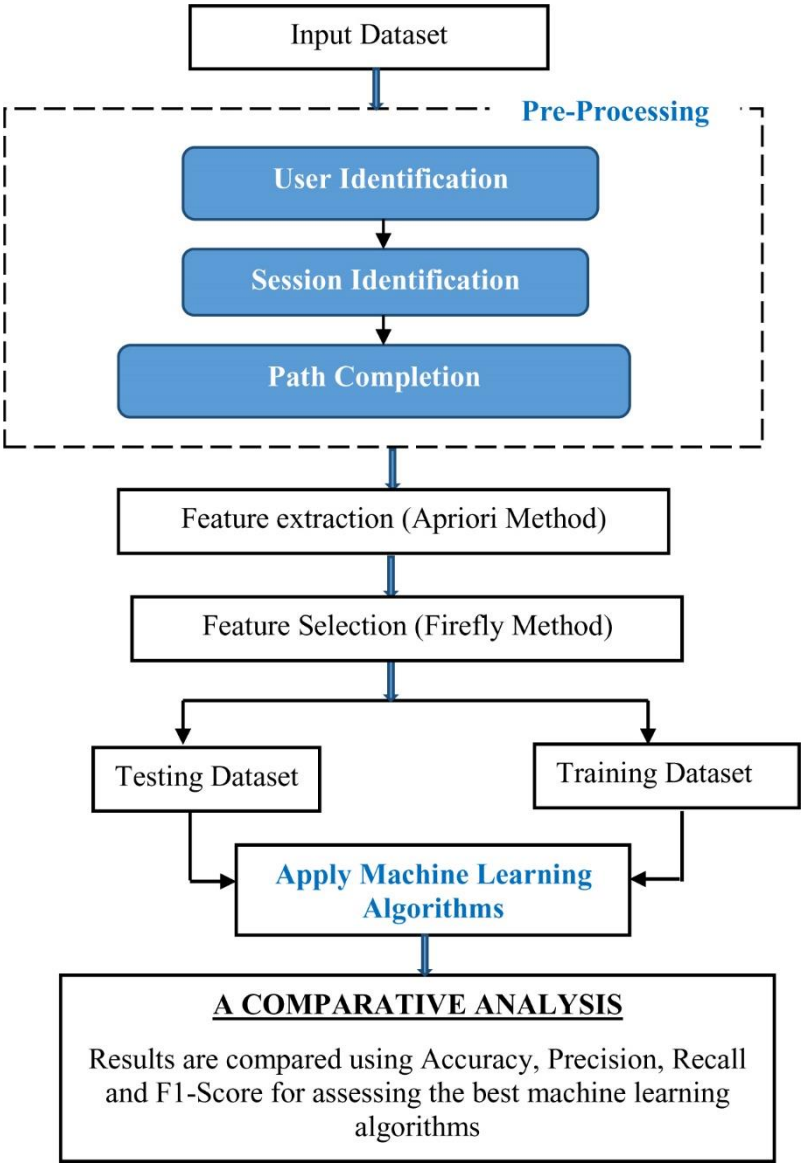

Fig1. System overview

2. Literature survey

Dhanashree L. Patil et al proposed a algorithm as Frequent Item-sets Mining Using Basic Time Cube had take the input data values has Database(S), min-S-up, density, basic time cube value (BTC) and the output has Set of frequent item-set. This algorithm is used to find out the how many times the particular data item-set is to be appearing in the data base web server. To find the frequent data item-sets, the algorithm use temporal data. The main feature of the algorithm is used the concept of basic time cube to regard as the time hierarchies in data mining process and this new proposed method is very effective [1].

C. Borgel et al [2] analysis various algorithm such as FPgrowth, apriori, TM and Eclat and find out the strength and weakness of these algorithm. It introduces the new algorithm as Split and Merge (SAM) with the help of analysis the various existing algorithm especially developed for frequent item-set mining. This algorithm is well suited for external data storage.

Chin-Hoong Chee et al [3] surveyed various frequent pattern mining algorithms and make the comparison and classify in to three types are pattern growth, tree based and join based . The problem in frequent pattern mining is more consumption of memory. So we need to develop the new algorithm to solve the problem of more memory consumption and new researcher aims to introduce find out the hidden patterns with short run time.Agrawal et al. [4] proposed the algorithm named as Apriori and many data item-sets are generated in recursive way. This algorithm is very effective for frequent item-set mining.

Varun Dixit et al. [13] conducted a survey on path completion and other techniques of web usage mining, they studied nearly sixteen research article related to the web usage mining and other techniques, out of their study they represents the web usage

mining process in detail, data cleaning, user identification, session identification and pattern discovery. In addition to that they also depicts the some of web usage mining application and also represents performance of web usage mining, what are the requirements to the web usage mining , represents the functionality of web usage mining with neat diagram. Finally they represent about the path completion techniques, in web log contains detailed log information, but some of the pages have back catch of page and link. This type of information not available in the web log file, it is available only the client machine alone, the path completion process to identify the missing catch pages and links, and added to the web log file.

B.Bhavani et al. [14] reviewed the ten papers five among them are represents the web usage mining techniques and remaining of them are web usage mining application related. Web usage mining consist the different techniques such as the data pre-processing, it includes the data-preprocessing, user, session identification and path completion. The another technique pattern discovery, it includes the data mining techniques such as Association, clustering, sequential pattern and classification. Pattern Analysis includes the OLAP, data and knowledge querying, usability analysis and visualizations technique. The author analysis the web usage mining applications such as the personalization of web content, perfecting and caching, support to the design and E-commerce applications.

3. Materials and Methods

3.1 Input Data: WEB LOG FILES

Web page access history is stored in file that is called web log files. It is automatically generated if the user clicks or requests a page. Each time of a page access log file should be updated automatically. A web log file is located in the following locations; 1. Web server 2. Web proxy server 3. Client browser.

Web server logs	**It provides more accurate and complete usage of data to web server.**
Web proxy serve logs	It takes HTTP request from user, gives them to web server, then result passed to web server and return to user. Client send request to web server via proxy server.
Client Browser logs	It can reside in client browser alone. In the form HTTP cookies.

Type of Web log files: There are four types web log files are there; Access Log file, Error Log file, Agent Log file, Referrer Log file.

- ***Access Log file-***All incoming request data's and information about client of server.
- ***Error Log file-***Internal Error generated by the server. The page is being requested by the client to the web server.
- ***Agent Log file-*** Information about user browser name and the other details.
- ***Referrer Log file-*** It contains the information about link and redirects visitors to site.

Web Log File Formats: It is the standardized text file format that is used by most ofthe web servers to generate the log files. The configuration of common log file format is given below in the box.

"%h %l %u %t \"%r\" %>s %b" THRC/access_log_common

eg: *127.0.0.1 RFC 1413 frank [20/Jan/2018:17:35:33 -0700] "GET*

/apache_pb.gif HTTP/1.0" 200 2326

3.2 Preprocessing

Pre-processing of data is the first stage in the web usage mining, it is accomplished through the different phases, and the first one is the Data cleaning. It is the primary role of the pre-processing. Web log file contains the lot of information's some of them are no needed for analysis they are removed in this stage. In second stage of the data cleaning process is User Identification; the cleaned web log file is on input for this, form the web log which user can access the web pages to be found with the different heuristics. Like the third one also Session Identification, this stage also take the cleaned web log and find sessions with different heuristics. The final stage of the pre-processing is the Path Completion.

3.3 Feature Extraction

In frequent pattern mining, the database consisting of a series of transactions and the aim is on finding frequently occurring itemsets in the whole database. Classical Apriori algorithm's process is divided into two stages. Initially, candidate item sets are generated and then association rules are also generated. Before starting these procedures, the value of threshold T is prescribed. After scanning the database Q, all the candidate itemsets are gained and then only those frequent itemsets are mined whose threshold value is greater than T. After completing the first scanning, we get only single itemsets and then successive iterations deals with n-itemsets until all the frequent itemsets are mined from Q. The use of support for pruning candidate itemsets is guided by the following Statistics.

i. If an itemset is frequent, then all of its subsets must also be frequent.

ii. If an itemset is infrequent, then all of its supersets must also be infrequent.

The pseudo code is illustrated below,

INPUT: D is a database, a support threshold T.
OUTPUT: A list of itemsets $F(Q, T)$.
PROCESS:
$C_1 \leftarrow \{\{i\} | i \in J\}$
$K \leftarrow 1$
while $C_K \neq \{\}$
# Compute the supports of all candidate itemsets
for all transactions $\{tid, I\} \in Q$ **do**
for all candidate itemsets $A \in C_K$ **do**
if $A \subseteq I$ **then**
$A.support{+}{+}$
Extract all frequent itemsets
$F_K = \{ A | A.support > T \}$
#Originate candidate itemsets
for all $A, B \in Fi$, $A[i] = B[i]$ for $1 \leq i \leq k-1$, and $A[k] < B[k]$ **do**
$I = A \cup \{B[k]\}$
if $\forall J \subset I, |J| = k : J \in F_k$ **then**
$C_{k+1} \leftarrow C_{k+1} \cup I$
$k{+}{+}$

3.4 Feature Selection

This section deals with the optimizers for an effective feature selection. To discuss the working mechanism of the proposed optimizer, the preliminary background firefly optimization algorithm is presented here.

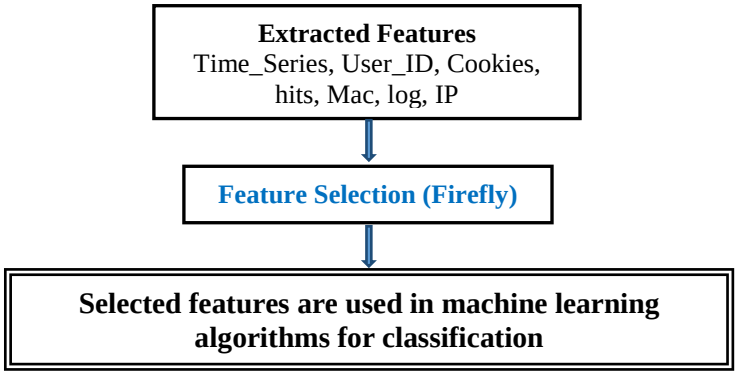

Fig.2. Feature Selection system

Feature selection is to be converted in to a more reliable and suitable form for the classifier to classify the given data. This paper introduced the ADABoost based ensemble algorithm with firefly optimized features to classify the frequent and infrequent in web log dataset. Meanwhile, the proposed algorithm has been compared with existing ML algorithms in terms of accuracy, precision and recall with the machine-learning algorithm.

a. Firefly (FF) algorithm:

Fireflies are winged beetles, which produce light and blinking at night. The light has no infrared or an ultraviolet frequency that is chemically produced from the lower abdomen is called bioluminescence. They use the flash light especially to attract mates or prey. The flash light also used as a protective warning mechanism to remind the fireflies about the potential predators.

Firefly algorithm was firstly developed by Yang, which is inspired by the flashing light of fireflies in the summer sky. The flashing light can attract mating partners or potential prey. Based on the attraction behavior, Yang built the original FA. The Firefly Algorithm was formulated with the following assumptions:

i. A firefly will be attracted to each other regardless of their sex because they are unisexual.
ii. Attractiveness is proportional to their brightness whereas the less bright firefly will be attracted to the brighter firefly. However, the attractiveness decreased when the distance of the two fireflies increased.
iii. If the brightness of both fireflies is the same, the fireflies will move randomly.

The generations of new solutions are by random walk and attraction of the fireflies. The brightness of the fireflies should be associated with the objective function of the related problem. Their attractiveness makes them capable to subdivide themselves into smaller groups and each subgroup swarm around the local models. Thus, FF is an excellent global optimizer based on swarm intelligence.

In FF, there is a set to initial solutions consisting of the initial population. Each firefly is regarded as a potential solution in the search space. Assume that N is the population size, and X_i is the ith solution in the population, where i = 1, 2… N.

The light intensity (I) usually decreases with the increase in distance. According to the literature, the light intensity can be defined as follows [15]:

$$I(r) = I_0 e^{-\gamma r^2}$$

Where I_0 is the initial light intensity and γ is called light absorption coefficient. The attractiveness β is defined as follows [14]:

$$\beta = \beta_0 e^{-\gamma r^2}$$

Where β_0 is a constant value and it is usually equal to 1.0. For any two fireflies Xi and Xj, their distance can be calculated by [14].

$$r_{ij} = \|X_i - X_j\| = \sqrt{\sum_{d=1}^{D} (x_{id} - x_{jd})^2}$$

Where X_{id} and X_{jd} are the d^{th} component of X_i and X_j, respectively. When X_j is brighter (better) than X_i, X_i is attracted to X_j. It means that X_i will move to X_j because of the attraction.

The movement of fireflies is defined as follows [13]:

$$x_{id} = x_{id} + \beta \cdot (x_{jd} - x_{id}) + \alpha \cdot (rand - 0.5)$$

Where α is called step factor and rand is a random value uniformly generated in the range [0, 1].

3.5 Supervised Machine Learning Algorithms

The majority of machine learning techniques use the concept of supervised machine learning (SML). This model is use the labeled dataset which have the parameters of both input and output. This supervised machine learning method is very effective and produce more accuracy when compare to the Unsupervised machine learning technique.

The main techniques of supervised machine learning are regression and classification. Regression is used to predict the single output value with the help of the trained data. The output of regression has continuous value in the particular range. The main aim of this technique is used to predict the value much closer to the value of actual output and then find out the error value of the regression model [16].

- If the value of error is small, then the accuracy level is to be greater.
- If the value of error is large, then the accuracy level is to be smaller.

The various supervised machine learning algorithms are SVM (Support Vector Machine), Linear Regression, random forest and the Decision Trees. The main challenges of SML are preprocessing and data preparation. The accuracy level is to be decreased due to the incomplete input value of trained data.

Advantage

- With the help of existing experience, the supervised learning in Machine Learning is used to collecting the data or produces the output of data.
- Using the previous experience, help you to optimize the performance criteria.
- It also solves various real-time computations problems.

Disadvantage

- The real challenge of the supervised machine learning is classify the huge data.
- Lot of computational time needed to train the supervised learning.

While training the system, the data is divided in to the ratio of 3:1 such as 80:20. The ratio of 80% is used for the data training and the ratio of 20% data is used for the testing data. In the training data, in the ratio of 80% itself both the input and output are feed. Different machine learning algorithms are used to build the model and each model has their own logic. After build the model, and then it is tested. At the time of testing the model, the remaining ratio of 20% data is fed in the input and this model will predict some value of data and then it will compare with the original actual output to find out the accuracy.

4. ML Algorithms used in this research

4.1 K-Nearest Neighbor (KNN)

KNN algorithm plays an important role in machine learning system. It belongs to the supervised learning area and have numerous applications in intrusion detection, pattern recognition, and so on. These KNNs are applied in realistic consequences where non-parametric methods are needed [16].

These techniques do not create any presumptions about data distribution. In the certain dataset, the KNN method classifies the correlatives into clusters which are recognized by a specific characteristic. The most idea for this method is that it's similar output for similar training samples. For the input population nearest value is identified that's ready to assign classes to all or any the samples.

Consider $\mathrm{X}_i = \{x_1, x_2, \dots, x_{iN}\}$ and $\mathrm{X}_j = \{x_1, x_2, \dots, x_{jN}\}$ the sample population, thus to measure the similarity between them and the distance is calculated as given.

$$\mathrm{Dist}(\mathrm{X}_i, \mathrm{X}_i) = \sqrt{\sum_{m=1}^{N} \left(\mathrm{x}_{im} - \mathrm{x}_{jm}\right)^2}$$

In the above equation, Euclidean distance is described that evaluates similarity among two pixel points. Hence, the pixels obtain the category to which a number of them commonly resemble.

Advantage

- Versatile that is used in both classification and regression
- Produce better accuracy for huge amount of data

Disadvantage

- Does not support well in high dimensional space.
- For large amount of memory needed so more expensive

4.2 Support Vector Machine (SVM)

This model is used to analyze the data in both the classification and regression analysis. SVM is tone of the robust prediction methods that can be used in statistical methods [9]. The main task of support vector machine is used to classify the data. For example, some data may belong to one of two classes, and to find out which classes a new data point will be in. suppose the data point is viewed as X-dimensional vector and then we want to know whether we can separate such data points as (X-1) dimensional hyper-plane. There are many hyper-plane in support vector machine to classify the data points. This method can be called as linear classifier. Support Vector Machine can solve various problems in real world applications [10] [11].

Advantage

- More reliable and produce more accuracy
- Work well when there is a clear margin of separation between classes.
- Produces better results in high dimensional spaces.
- Work well in both unstructured and semi- structured data
- More useful in non-linear data.

Disadvantage

- It need full labeling of data input.
- Its avoids estimating probabilities on finite data
- Problems occur in multi-class SVM because the labels are drawn from a finite set of several elements.
- Not suitable for huge amount of data.

4.3 Proposed: AdaBoost with Ensemble SVM-KNN algorithm

In classification algorithms, each one has its own advantage and disadvantage. So, AdaBoost with Ensemble SVM-KNN algorithm is compared with above mentioned algorithms to achieve the highest accuracy then others. The working mechanism of proposed algorithm is explained in the below section. In proposed technique, K nearest neighbor technique finds the distance between test sample and training sample. An important task of KNN is to find out the neighbors first, and then it will classify the query sample on the majority class of its nearest neighbors. The proposed KNN-SVM ensemble classification approach can be used effectively for frequent itemset detection with low computational complexity in the training and detection stage. The lower computational complexity property is gained from KNN classification approach that does not require construction of a feature space. KNN algorithm has been used in the proposed hybrid approach KNN-SVM as the first step in the pancreatic tumor detection, and then the SVM method is employed in the second stage as a classification engine of this hybrid model.

Adaboost is an iterative boosting approach to improve the classification of the weak classifiers. At the initial stage, the Adaboost algorithm will allocate variant weights to each observations. After a few iteration, the weight impose on the misclassified observations will increase, and vice versa, the correctly classified will have lesser weights. The weights on the observations are the indicators as to which class the observation belongs to, thus lower the misclassification of the observations while extremely improve the performance of the classifiers at the same time. That mainly aims at reducing variance, boosting is a technique that consists in fitting sequentially multiple weak learners in a very adaptive way: each model in the sequence is fitted giving more importance to observations in the dataset that were badly handled by the previous models in the sequence [17].

Algorithm: AdaBoost with ensemble SVM - KNN classifier

Input: web log dataset with class label (frequent or infrequent) i.e. (X_1, C_1), (x_2, C_2)... (X_n, C_n); Feature pool F= {f_m, m=1... n}; Number of iterations = R

Initialization: Weight of each features

$$\frac{1}{N}; \forall \text{ i } (\text{i}= 1,..., \text{N}); \sigma = 1000$$

For r = 1 to R do:

(a) Generate a training set by sampling with {$w_i(r)$}

(b) Train base classifier h_r ((Proposed Hybrid SVM - KNN Classifier)) using this training set

1. Apply SVM classifier on optimized data set with K-fold cross-validation and K=5.
2. Update the weights.
3. According to Wolfe dual form, weight minimization is

$$Minimize : w(\alpha) = -\sum_{i=1}^{N} \alpha_i + \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} y_i y_j \alpha_i \alpha_j k(X_i, X_j)$$

$$Subject - to : \sum_{i=1}^{N} y_i \alpha_i = 0, \forall_i; 0 \leq \alpha_i \leq C$$

4. Predict the test web log class using the cross validated model with minimum weight.
5. Apply weighted K-Nearest Neighbor Classifier with number of nearest neighbors K=5 on optimized data set.
6. Apply K-fold cross validation with K=5.
7. Weight contribution of each k neighbor
8. Set initial weights of KNN = updated minimum weights of SVM.
9. X_t is test optimized dataset

$$\hat{f}(X_t) \leftarrow \frac{\sum_{i=1}^{k} w_i f(X_i)}{\sum_{i=1}^{k} w_i}$$

10. Predict the test web log data class using the cross validated model with minimum weight.
11. Take weighted average of predictions from both the models.

(c) Compute the training error of h_r:

$$\epsilon_r = \sum_{i=1}^{N} w_i(r).I[C_i \neq h_r(X_i)]$$

Where I ∈ (-1, 1), I_A is indicator of A; we assume ($\in_r$<0.5)

Set:

$$\alpha_r = log(\frac{1 - \epsilon_r}{\epsilon_r})$$

(We have $\alpha_r > 0$)

Update the weights by:

$$w_i^{'}(r+1) = w_i(r)exp(\alpha_r I[C_i \neq h_r(X_i)])$$

$$w_i(r+1) = \frac{w_i^{'}(r+1)}{\sum_i w_i^{'}(r+1)}$$

Output:

$$h(x) = Sign(\sum_{r=1}^{R} \alpha_r h_r(X))$$

AdaBoost with ensemble SVM-KNN as component classifier for pancreatic classification. Proposed scheme gives classification accuracy of 97% for web log data classification. Results reveal that proposed AdaBoost with ensemble SVM-KNN outperforms other methods.

5. Results and Discussions

In order to experiment and evaluate the proposed methodology, the proposed algorithm is implemented in the python 3.6 with Anaconda 3.vb distribution with Sci-kit machine learning packages runs on Intel i7 CPU with the 2TB hard disc, 8GB RAM with Windows 10 Operating Systems. The raw dataset used in this experiments can be seen in below table.

Time_Series	User ID	cookies	hits	Mac	log	IP
1.59E+09	31818	800	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:b6:fd:6e:08:00	sxc	2.89E+09
1.59E+09	3449	697	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:1f:ef:f8:08:00	sxc	2.89E+09
1.59E+09	29756	697	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:12:8a:ce:08:00	sxc	2.89E+09
1.59E+09	61521	696	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:1f:f6:82:08:00	sxc	2.15E+09
1.59E+09	46088	696	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:1f:f6:82:08:00	sxc	2.15E+09
1.59E+09	26073	800	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:b6:fd:6e:08:00	sxc	2.89E+09
1.59E+09	28942	697	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:1f:ef:f8:08:00	sxc	2.89E+09
1.59E+09	39284	44	117.240.141.242	00:16:76:38:f6:ab:20:d8:0b:d5:05:f1:08:00	sxc	1.55E+09
1.59E+09	18957	697	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:12:8a:ce:08:00	sxc	2.89E+09
1.59E+09	36190	696	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:1f:f6:82:08:00	sxc	2.15E+09
1.59E+09	7331	696	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:1f:f6:82:08:00	sxc	2.15E+09
1.59E+09	35477	800	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:b6:fd:6e:08:00	sxc	2.89E+09
1.59E+09	36018	697	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:1f:ef:f8:08:00	sxc	2.89E+09
1.59E+09	3150	697	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:1f:ef:f8:08:00	sxc	2.89E+09
1.59E+09	16880	696	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:1f:f6:82:08:00	sxc	2.15E+09
1.59E+09	0 DF	175	255.255.255.255	ff:ff:ff:ff:ff:ff:44:d9:e7:cc:c1:3d:08:00	sxc	2.89E+09
1.59E+09	65072	800	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:b6:fd:6e:08:00	sxc	2.89E+09
1.59E+09	20532	697	255.255.255.255	ff:ff:ff:ff:ff:ff:ac:84:c6:1f:ef:f8:08:00	sxc	2.89E+09
1.59E+09	54321	45	117.240.141.242	00:16:76:38:f6:ab:20:d8:0b:d5:05:f1:08:00	sxc	2.99E+09

5.1 Performance Measures parameters

The evaluation is carried out for the different algorithms with the following parameters such as TP, FP, TN, FN, and Precision, Recall, F1_Score and classification accuracy [18].

TP - Number of frequent instances is correctly classified as frequent features

TN - Number of infrequent instances is correctly classified as infrequent features

FP - Number of frequent instances is wrongly classified as infrequent features

FN - Number of infrequent is wrongly classified as frequent features

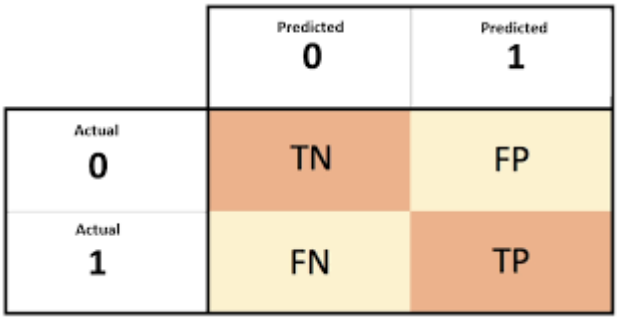

Fig.6. Confusion Matrix Format

- Accuracy value is the proportion of the accurate number of predictions. It can be determined using the below equation:

$$\text{Accuracy} = \frac{(TP + TN)}{(\text{TP} + \text{TN} + \text{FP} + \text{FN})}$$

- Precision is the ratio of predicted positive examples which really are positive

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{TN}}$$

- Recall also called hit rate or sensitivity; it measures how much a classifier can recognize positive examples

$$\text{Recall} = \frac{\text{TN}}{\text{TP} + \text{TN}}$$

- 'F1_Score' is the 'Harmonic Mean' of recall with precision.

$$\text{F1\_Score} = \frac{(2 \times \text{Precision} \times \text{Recall})}{(\text{Precision} \times \text{Recall})}$$

- The error rate is obtained by using the below Equation as

$$\text{Error rate} = (\text{FP} + \text{FN}) / (\text{TP} + \text{FP} + \text{TN} + \text{FN})$$

In the evaluation scenario, web log data is considered for the classification and different comparative analysis are shown in below table.

Table.2 Classification accuracy of the various classifiers disease / non-disease cotton plant classification

Classifiers/ Measures	SVM-KNN	Adaboost – Ensemble	SVM	KNN
Correctly Classified	205	260	290	201
Incorrectly Classified	45	10	80	69
Classification Accuracy (%)	87	97	82	81

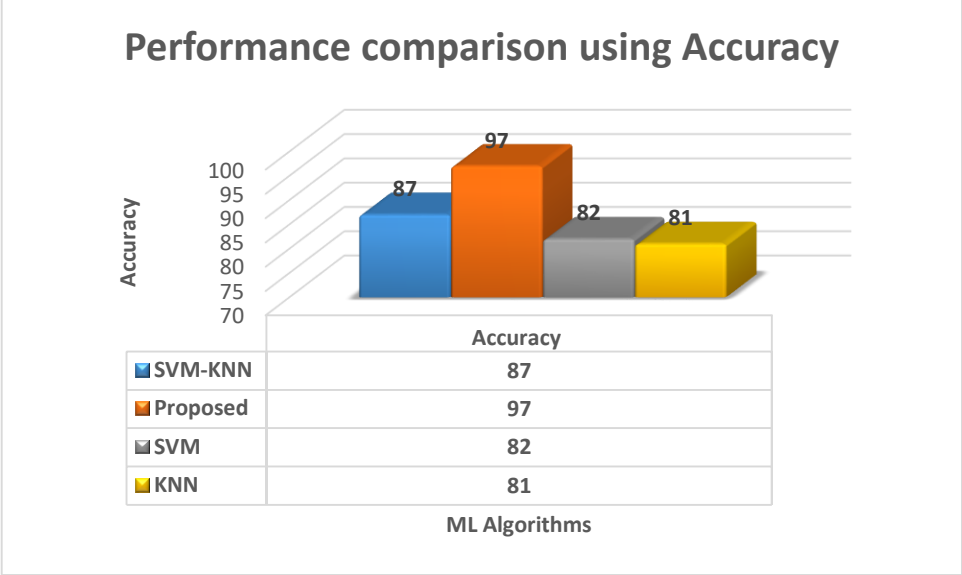

Fig. 7. Performance Comparison with Accuracy

The above tables display the classification accuracy for normal / abnormal case with respect to classified dataset and number of without disease / with disease case respectively using various ML classifiers.

Table.3 Performance comparison with various evolution parameters

Algorithm used	TP	FP	TN	FN	Precision	Recall	F1-Score
SVM	241	33	158	38	83	82	83
KNN	241	31	160	38	81	80	81
SVM-KNN	267	24	138	21	87	87	87
Adaboost – Ensemble	260	10	180	0	97	97	97

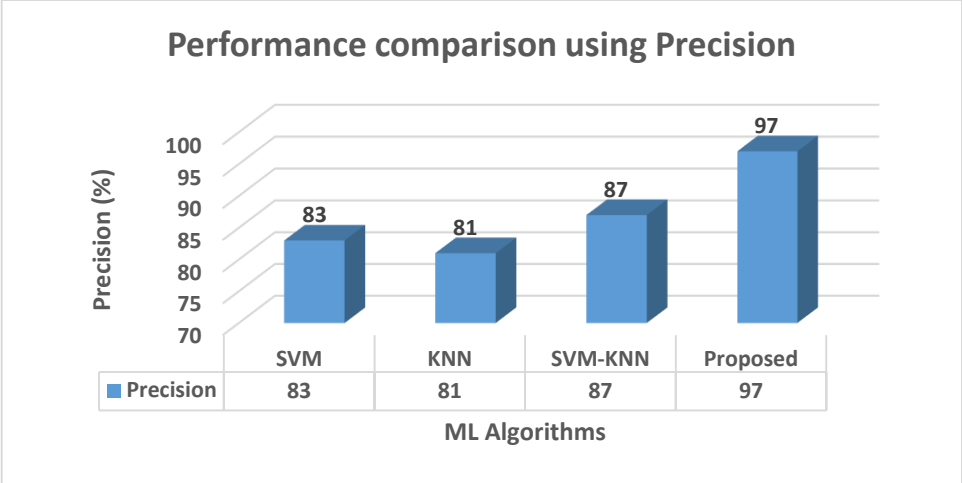

Fig.6. Performance Comparison Precision

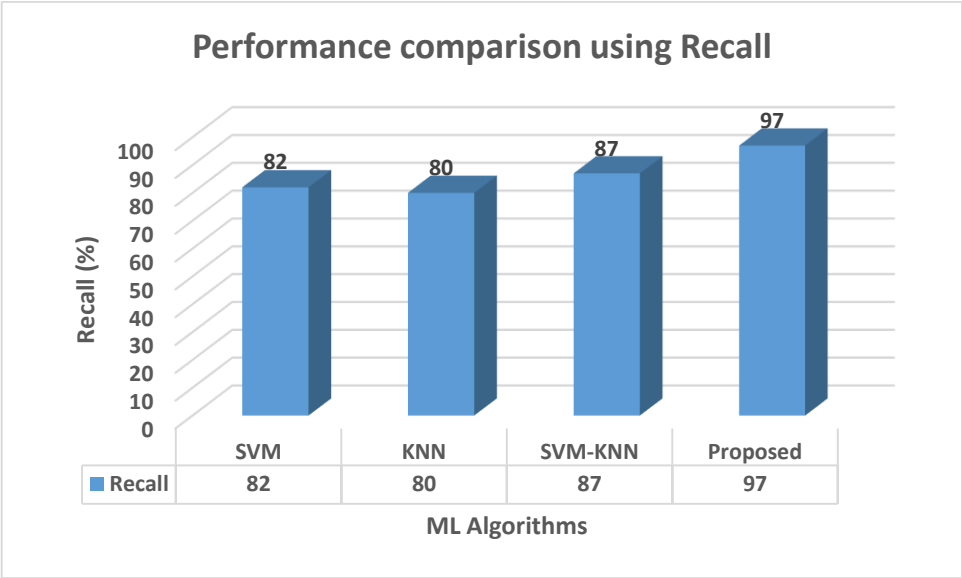

Fig.6. Performance comparison using Recall

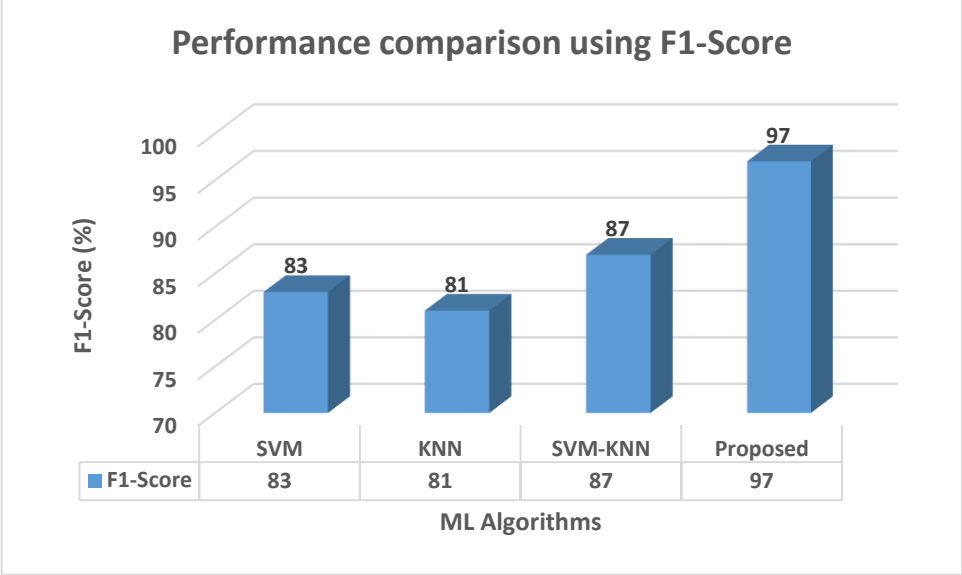

Fig.6. Performance comparison using F1-Score

Classification performance of the proposed ensemble classifier is better than the other single classifiers and it is illustrated from the results in above tables and figure. The proposed classifier yields better classification accuracy, because it has a regularization parameter, which avoids over-fitting. The above tables exhibits the performance metrics comparison of the existing ML classifiers with proposed ensemble classifier, for frequent / infrequent case respectively.

Conclusion

Among the traditional methods considered, KNN, SVM were the most commonly used. These models are highly applicable in classification process. The main contributions of this paper are: (i) A comparison of different approaches, evaluated with various experiments, on how ML can be applied for solving existing methods challenges and sequencing problems in a hybrid flow environment, (ii) a comparison of the best identified SVM-KNN strategy with Adaboost approaches concerning solution quality and computational efficiency. The proposed method shows promising results and the work confirms that the proposed model outperforms other models for frequent itemset prediction from web log data.

References:

1. Dhanashree L. Pati," Efficient Method for Mining Frequent Itemsets using Temporal Data", 2018 International Conference on Information, Communication, Engineering and Technology (ICICET),978-1-5386-5510-8/18/$31.00 ©2018 IEEE
2. C. Borgel et al," Simple Algorithms for Frequent Item Set Minin", Advances in Machine Learning II, SCI 263, pp. 351–369, pringer-Verlag Berlin Heidelberg 2010.
3. Chin-Hoong Chee1, Algorithms for frequent itemset mining: a literature review,Springer, 24 March 2018
4. R. Agrawal, T. Imielinski, A. N. Swami, R. Agrawal, T. Imielinski, and A. N. Swami, "Mining association rules between sets of items in large databases, sigmod conference," Proc Sigmod, vol. 22, no. 2, pp. 207– 216, 1993
5. Breiman, L. (1996). "Bagging Predictors". Machine Learning. 24 (2): 123–140. doi:10.1007/BF00058655.
6. Angshuman Paul, Dipti Prasad Mukherjee, Prasun Das, AbhinandanGangopadhyay, AppaRaoChintha "Improved Random Forest for Classification" IEEE Transactions on Image Processing (Volume: 27 , Issue: 8 , Aug. 2018).
7. Gareth, James; Witten, Daniela; Hastie, Trevor; Tibshirani, Robert (2015). An Introduction to Statistical Learning. New York: Springer. pp. 315. ISBN 978-1-4614-7137-0.
8. Mehtaa, Dinesh; Raghavan, Vijay (2002). "Decision tree approximations of Boolean functions". Theoretical Computer Science. 270 (1–2): 609–623. doi:10.1016/S0304-3975(01)00011-1.
9. Cortes, Corinna; Vapnik, Vladimir N. (1995). "Support-vector networks" (PDF). Machine Learning. 20 (3): 273–297. CiteSeerX 10.1.1.15.9362
10. Joachims, Thorsten (1998). "Text categorization with Support Vector Machines: Learning with many relevant features". Machine Learning: ECML-98. Lecture Notes in Computer Science. Springer. 1398: 137–142.
11. Barghout, Lauren. "Spatial-Taxon Information Granules as Used in Iterative Fuzzy-Decision-Making for Image Segmentation". Granular Computing and Decision-Making. Springer International Publishing, 2015. 285–318.
12. Mazaher Ghorbani and Masoud Abessi, "A New Methodology for Mining Frequent Itemsets on Temporal Data" IEEE Transactions on Engineering Management, Volume: PP, Issue: 99, Year-June 2017.
13. Varun Dixit and Abisshek Dwivedi (2017), "A Survey on path completion and various techniques in web usage mining", International Journal of LNCT, Vol 1(1) pp:16-21.
14. Bhavani B, Dr. Sucharita V and Dr. Satyanarana K.V.V. (2017), "Review on Techniques and Applications Involved in Web Usage Mining", International Journal of Applied Engineering Research, Volume 12, Number 24, pp.15994-15998.
15. S. Mehta, M. Singh and N. Kaur, "Firefly Algorithm for Optimization of Association Rules," 2020 6th International Conference on Signal Processing and Communication (ICSC), 2020, IEEE, pp. 143-148.
16. S. Ray, "A Quick Review of Machine Learning Algorithms," 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), 2019, IEEE, pp. 35-39.
17. Udhav Bhosle, Jyoti Deshmukh "Mammogram classification using AdaBoost with RBFSVM and Hybrid KNN–RBFSVM as base estimator by adaptively adjusting c and C value" Springer, 2018.
18. Mehdi Bahrami, Ali Shabani, Mohammad Reza Mahmoudi, Shohreh Didari, "Determination of Effective Weather Parameters on Rainfed Wheat Yield Using Backward Multiple Linear Regressions Based on Relative Importance Metrics", Complexity, vol. 2020, Article ID 6168252, 10 pages, 2020.