A Sampling-Based Sentiment Analysis of Imbalanced Streamed Movie Reviews

Rafi Ramadhani, *Student, Sepuluh Nopember Institute of Technology,* Zulchair Asy'ari, *Fellow, Sepuluh Nopember Institute of Technology,* and Reza Wahyu Ramadhan, *Fellow, Sepuluh Nopember Institute of Technology,*

***Abstract*—Sentiment analysis has gained significant importance in analyzing individuals' attitudes and perceptions toward various products, services, and entertainment mediums, including movies. Evaluating the sentiment expressed in movie reviews can provide valuable insights into how users interpret and react to specific films. However, Movie review datasets often suffer from an imbalance in the distribution of positive and negative sentiment labels, which presents challenges for accurate sentiment classification. We propose an approach that tackles this problem by employing various sampling techniques, ... These techniques aim to manipulate the class distribution in the dataset, mitigating the bias towards the majority sentiment class and improving the performance of the sentiment classifier. We conducted extensive experiments using diverse movie review datasets to evaluate the approach's effectiveness. They compared the performance of the sampling-based approach with baseline methods, using evaluation metrics such as accuracy, precision, recall, and F1-score. The experimental results consistently demonstrated that . . ..**

***Index Terms*—IEEE, IEEEtran, journal, LaTeX, data stream, sentiment analysis, machine learning, data mining.**

I. Introduction

Sentiment analysis is a rapidly growing field in natural language processing that focuses on understanding and categorizing people's opinions and emotions. Sentiment analysis has become increasingly important in analyzing the attitudes and perceptions of individuals towards different products, services, and even entertainment mediums such as movies.Sentiment Analysis is the process of determining a value in opinion of consumer about certain topics or products. Sentiment Analysis can be done by evaluating the text contained within the opinion itself. It determines whether an opinion is positive or negative[1]. Internet Movie Database (IMDb) is a popular platform for users to share their opinions on movies. Users' expressions in the reviews and categorize them as positive, negative, or neutral. Examining the tone of movie reviews can reveal important information about how users interpret and respond to certain films. Thus, filmmakers, producers, and studios can use this information to determine which elements of their films are effective and which must be modified to appeal to audiences more effectively. The challenges in analyzing the sentiment of movies from the IMDb database are language complexity, subjectivity and context, and dataset quality.

M. Shell was with the Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, 30332 USA e-mail: (see http://www.michaelshell.org/contact.html).

J. Doe and J. Doe are with Anonymous University.

Manuscript received April 19, 2005; revised August 26, 2015.

Finding the best accuracy and performance between algorithms has been done by BalakrishnanGokulakrishnan[4] on a Twitter Data Stream. They have improved the accuracy of the classifiers they were using and they compared each algorithm based on their performances. For future work, they aim to try sampling and boosting techniques on different datasets.

This paper aims to compare performances between Support Vector Classifier(SVC), Random Forest, Naive Bayes, Gradient Boosting, Extra Tree, dan AdaBoost that has been oversampled on IMDb dataset simulated as a data stream. This study provides results of accuracy and performances between algorithms for classifying IMDb movie reviews. Classifier with the best accuracy and performance contributes to sentiment analysis the most.

II. Literature review

Sentiment analysis is a method of identifying opinions, emotions and value in natural language which is a type of subjectivity analysis that has been used in natural language processing studies[5]. It determines whether words or phrases are negative or positive based on the topic and context given.

Tun[9] proposed a method of sentiment analysis that performs fine-grained analysis to determine sentiment orientation and strength of reviewers by taking various aspects of a movie into consideration. sentiment classification for overall movie, director, story, cast, music and scene aspect has the accuracy of 75%, 86%, 80%, 83%, 81%, and 90%.

Movie reviews still pose a risk of having an imbalanced set of data that could lead to major problem to the development of classifiers in prediction and performance. in order to solve this problem, Dai[3] used an oversampling technique (WESMOTE) that synthesizes new training examples from text based on word embeddings. Krishna[5] proposed a method of using hybrid Deep Learning algorithms on IMDb movie reviews. The research shows that the hybrid CNN-BiLSTM technique reached 95% precision.

Other research done by Motz[6] who proposed a method of doing live sentiment analysis using machine learning and text processing algorithms. The two main components consist of trend analysis and sentiment analysis on Twitter API data stream is able to process only relevant data. Result shows that there is still room for improvement in this topic in terms of combining algorithm and the best accuracy achieved was 68.29%.

Comparison between algorithms to find which one is better to Classify sentiment analysis has been done by Sai[7] who

applied few algorithms to compare their accuracy between Bag Of Words(BOW) and TF-IDF technique. Among all of the algorithms tested, they have found that for BOW technique the KNN algorithm is the least accurate with 54percent accuracy. For TF-IDF technique, the classifier with the least accuracy is Decision Tree with 71percent accurate.

Other work done by Satrya[8] showed that by combining approaches to accomplish sentiment analysis is very beneficial. Machine Learning combined with Ensemble learning using rule-based approach on indonesian police-related twitter dataset showed that they were able to raise positive sentiments 0.83percent higher than negative.

III. Methodology

A. Data Preprocessing

In the movie review dataset, there is sometimes an imbalance within the dataset between the minority and the majority class. in order to prevent bias in training the dataset that could affect the result of some algorithms to ignore the minority class, we balanced the dataset using the oversampling approach.

Oversampling is a technique to balance imbalanced data by increasing the number of samples in the minority class while maintaining the data information from the majority class and enhancing that from the minority class[10].

B. Building Trained Model

In this paper we compare the accuracy and the perfomances of sentiment analysis in classifying IMDb Movie Reviews among these algorithms below.

1) Support Vector Machine (SVM): Support vector machine (SVM) is a machine learning algorithm used for both classification and regression. The Support Vector Machine method is a statistical classification approach which is based on the maximization of the margin between the instances and the separation hyper-plane. It is a non-probabilistic binary linear classifier, that has the ability to linearly separate the classes by a large margin, it become one of the most powerful classifier capable of handling infinite dimensional feature vectors[1].

2) Random Forest: Random forest is a machine learning algorithm that belongs to the family ensemble methods. The algorithm combines multiple decision trees to form a forest, where each decision tree makes predictions independently. following on study done by Chen[2], they showed that random forest outperformed another deep learning algorithm on a legal text classification, proposing the usefulness of domain-specific concepts as attributes for text classification in a specific domain.

3) Naive Bayes: Naive Bayes Classifier is a quite popular machine learning algorithm that used for text classification. It is based on the principles of Bayes' theorem and assumes that features (words or terms) are conditionally independent given the class label.

TABLE I
Result of 1000 random data as data training

2*Name	Mean				Stan
	Accuracy	F1 Score	Recall	Precission	Acc
Naive Bayes	0.86295	0.862841062	0.862924112	0.864039066	0.00
Extra Trees	0.8613000001	0.861293304	0.86129439	0.861354712	0.00
Random Forest	0.84305	0.8430401	0.84304447	0.843120694	0.00
SVC	0.899425	0.899420692	0.899431682	0.89950932	0.00

4) Gradient Boosting: Gradient Boosting Classifier is a machine learning algorithm that belongs to the family of boosting methods. It is commonly used for text classification tasks, where the goal is to assign predefined categories or labels to text documents based on their content. Gradient Boosting combines multiple weak classifiers, typically decision trees, in a sequential manner to create a strong classifier.

5) Extra Tree: Extra Trees Classifier, also known as Extremely Randomized Trees, is a machine learning algorithm that belongs to the ensemble learning family. It is a variant of the Random Forest algorithm and can be used for text classification tasks. Similar to Random Forest, the Extra Trees Classifier builds an ensemble of decision trees to make predictions.

6) AdaBoost: AdaBoost (Adaptive Boosting) is a machine learning algorithm that belongs to the family of boosting methods. It can be used for text classification tasks, where the goal is to assign predefined categories or labels to text documents based on their content. The AdaBoost classifier works by iteratively training a sequence of weak classifiers, typically decision trees, and giving more weight to misclassified samples in each iteration.

IV. Experiment and analysis

This research will be conducted with 2 types of experiments to evaluate the performance of several classification algorithms. The first experiment is the classification with constant data training and data testing, and the second one is the classification with stream simulation.

A. Dataset

In this research, we use IMDB Movie Ratings Sentiment Analysis from Kaggle as the dataset. In this dataset, we found that the dataset has 20019 data labeled as negative responses and 19981 data labeled as positive responses, with a total of 40000 data. Because of our data was imbalanced, we try to balance our data first with SMOTE which used oversampling approach.

B. Experiment 1: Classical model classification

A. Random Samples

We gather random data from the simulated data stream for 1000, 10000, and 20000 rows. After that we applied 10-fold cross-validation classification using all of the proposed algorithms and we analyze their own performances on accuracy, precision, recall, and F1-Score.

B. Balanced classes

We gather random data for 1000, 10000, and 20000 rows from the simulated data stream. After that we applied 10-fold cross validation classification using all of the proposed algorithms and we analyze their own performances on accuracy, precision, recall, and F1-Score.

C. Experiment 2: Datastream classification

In this experiment, we initiate 80 % of the dataset for initialization of the model that we build, and all the datasets would be randomized in the range of 1 to 100 data every minute later, after 15 minutes we triggered the system to remodel the training data based on every single data that we try to classify.

Figure 1a 2a 3a 4a 5a 6a shows the mean, f1, recall, and precision sometimes hit 0 and sometimes hit 1 which has to be never happened in the real world. The event that hit 0 or 1 is caused by our random number generated between 1 and 100, so if the random number is so small it may be caused by the model accuracy being 1 or 0. For the performance could be seen that Naive Bayes does not really consume memory significantly while training the new model compared to another algorithm, but Naive Bayes is stable to consume more CPU resources in 2 hours of simulating the data stream.

Figure 1a 2a 3a 4a 5a 6a shows the accuracy of the algorithm that we use to classify the sentiment into negative or positive. Based on the picture it shows that Adaboost, Gradient Boosting, and Naive Bayes accuracy is distributed around 0.8 or 80% of the accuracy. But Extra Tree, Random Forest, and Support Vector Machine distributed around 0.9 or 90% or more. while Random Forest is slightly around 1 or 100% while predicting the sentiment. Figure 1b 2b 3b 4b 5b 6b shows the precision of the algorithm that we use to classify the sentiment. Precision shows the proportion of positive predictions that are actually correct. In this case, the precision of Adaboost and Gradient Boosting distributed around 0.8 or 80%, and on the other side Extra tree classifier, Random Forest, and SVC mostly have better precision. Figure 1c 2c 3c 4c 5c 6c shows recall of the algorithm. Recall or sensitivity shows the proportion of actual positive instances correctly identified by the model. This experiment showed that Extra Tree and Random Forest classifiers have a better performance in classifying the positive sentiment. On the other side, the Adaboost classifier shows the worst performance on classifying the positive sentiment. Figure 1d 2d 3d 4d 5d 6d shows the F1 score of the algorithm. F1 score shows The harmonic mean of precision and recall, providing a balanced measure between the two. This experiment also shows Extra Tree and Random Forest have the better F1 score with Adaboost and Gradient Boosting have the lower F1 score. A high F1 score close to 1 indicates a model that has both high precision and high recall, suggesting good overall performance. A low F1 score closer to 0 indicates that the model has either low precision, low recall, or both, suggesting poor performance.

Figure 7a 7b 7c 7d 7e 7f shows consumable of computer resource by doing this experiment. Adaboost, Extra Tree, Gradient Boosting, and Random Forest classifiers show the need of using RAM in particular time series or show the algorithms using a high memory while re-train the model, On the other side Naive Bayes and SVC show the need of using CPU while re-train the model, Especially Naive Bayes shows stable to consume a high CPU either re-train the data or predicted the text that showed Naive Bayes has more complexity in the algorithm rather than the other algorithm that we use for this experiment.

TABLE II
MEAN OF THE ACCURACY

name	accuracy	precision	recall	f1-score
svc	0,9481777	0,9518184	0,9481777	0,9483182
random forest	0,9681520	0,9702490	0,9681520	0,9681852
nb	0,8959449	0,9034277	0,8959449	0,8958279
gradient boosting	0,8009341	0,8166880	0,8009341	0,7996701
extra tree	**0,9717636**	**0,9736189**	**0,9717636**	**0,9717993**
adaboost	0,8091311	0,8212302	0,8091311	0,8088404

TABLE III
STANDARD DEVIATION OF THE ACCCURACY

name	accuracy	precision	recall	f1-score
svc	0,0497305	0,0455899	0,0497305	0,0491354
random forest	0,0365329	0,0341241	0,0365329	0,0368645
nb	0,0752644	0,0733693	0,0752644	0,0762665
gradient boosting	0,0878767	0,0883729	0,0878767	0,0902029
extra tree	**0,0313473**	**0,0280731**	**0,0313473**	**0,0312633**
adaboost	0,0879722	0,0893127	0,0879722	0,0894462

TABLE IV
VARIANCE OF THE ACCURACY

name	accuracy	precision	recall	f1-score
svc	0,0024731	0,0020784	0,0024731	0,0024142
random forest	0,0013346	0,0011644	0,0013346	0,0013589
nb	0,0056647	0,0053830	0,0056647	0,0058165
gradient boosting	0,0077223	0,0078097	0,0077223	0,0081365
extra tree	0,0009826	0,0007881	0,0009826	0,0009773
adaboost	0,0077391	0,0079767	0,0077391	0,0080006

To simplify the research result we calculate the mean, variance, and standard deviation of the accuracy, f1, recall, and precision as shown in the table. Based on our research we find that the best accuracy, f1, recall, and precision is on Extra Tree Classifier, but mean doesn't show the data distribution of the accuracy, so we calculate standard deviation and variance to show how the data distribution was. Based on the standard deviation and variance we know the best data distribution of accuracy, f1 score, recall, and precision is also with Extra Tree Classifier.

V. CONCLUSION

The conclusion goes here.

APPENDIX A
PROOF OF THE FIRST ZONKLAR EQUATION

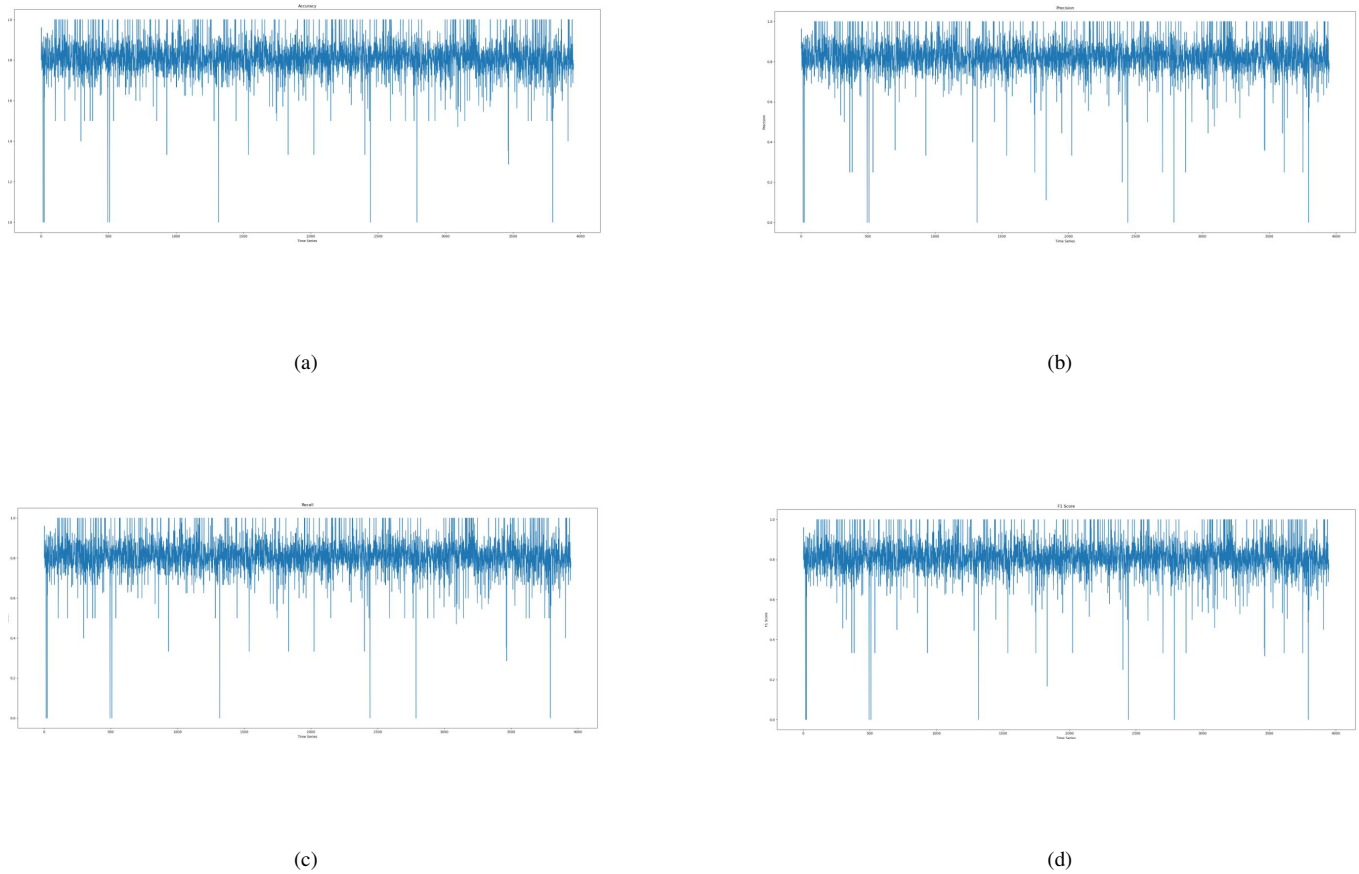
Appendix one text goes here.

APPENDIX B

Appendix two text goes here.

ACKNOWLEDGMENT

The authors would like to thank...



(a) (b)

(c) (d)

Fig. 1. (a) Adaboost Accuracy. (b) Adaboost Precission.(c) Adaboost Recall. (d) Adaboost F1. All the graph show in datastream mining

REFERENCES

[1] Yassine Al Amrani, Mohamed Lazaar, and Kamal Eddine El Kadiri. Random forest and support vector machine based hybrid approach to sentiment analysis. *Procedia Computer Science*, 127:511–520, 2018.

[2] Haihua Chen, Lei Wu, Jiangping Chen, Wei Lu, and Junhua Ding. A comparative study of automated legal text classification using random forests and deep learning. *Information Processing & Management*, 59(2):102798, 2022.

[3] Hong-Jie Dai and Chen-Kai Wang. Classifying adverse drug reactions from imbalanced twitter data. *International Journal of Medical Informatics*, 129:122–132, 09 2019.

[4] Balakrishnan Gokulakrishnan, Pavalanathan Priyanthan, Thiruchittampalam Ragavan, Nadarajah Prasath, and AShehan Perera. Opinion mining and sentiment analysis on a twitter data stream, 12 2012.

[5] Muddada Murali Krishna, Balaganesh Duraisamy, and Jayavani Vankara. Hybrid deep learning techniques for sentiment analysis on imdb datasets. In *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, pages 689–693. IEEE, 2022.

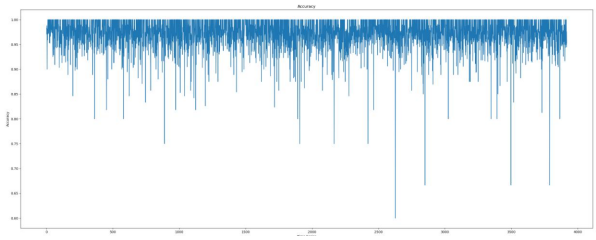
[6] Andrew Motz, Elizabeth Ranta, Adan Sierra Calderon, Quin Adam, Fadi Alzhouri, and Dariush Ebrahimi. Live sentiment analysis using multiple machine learning and text processing algorithms. *Procedia Computer Science*, 203:165–172, 2022.

[7] Lakshmi Sai, Kondepudi Yasaswi, Vellanki Rakesh, Mandadapu. Varun, Mentem Yeswanth, and Jonnalagadda Surya Kiran. Comparative study of algorithms for sentiment analysis on imdb movie reviews. 03 2023.

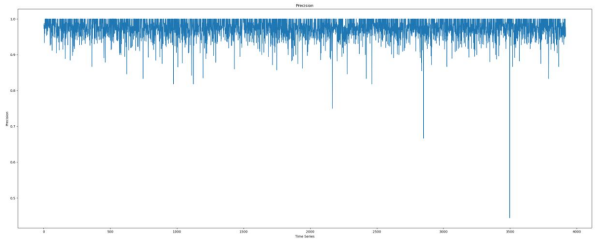
[8] Wahyu Fadli Satrya, Ria Aprilliyani, and Emny Harna Yossy. Sentiment analysis of indonesian police chief using multi-level ensemble model. *Procedia Computer Science*, 216:620–629, 2023.

[9] Tun Thura Thet, Jin-Cheon Na, and Christopher S.G. Khoo. Aspect-based sentiment analysis of movie reviews on discussion boards. *Journal of Information Science*, 36:823–848, 11 2010.

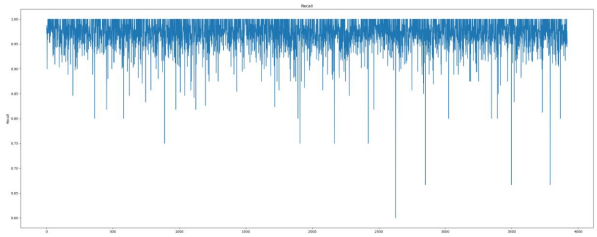
[10] Zhen Wei, Li Zhang, and Lei Zhao. Minority-prediction-probability-based oversampling technique for imbalanced learning. *Information Sciences*, 622:1273–1295, 2023.



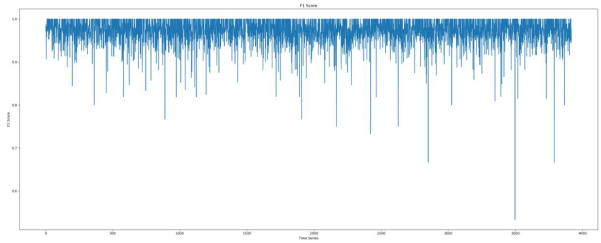
(a)



(b)

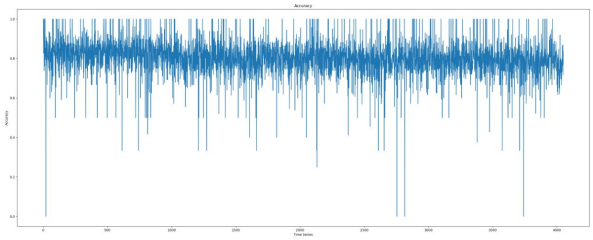


(c)

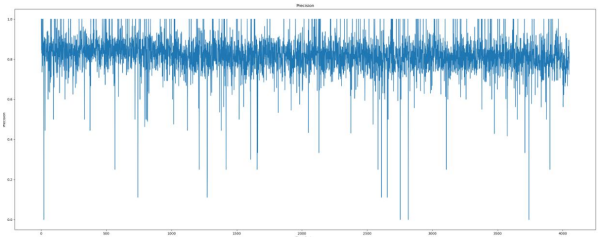


(d)

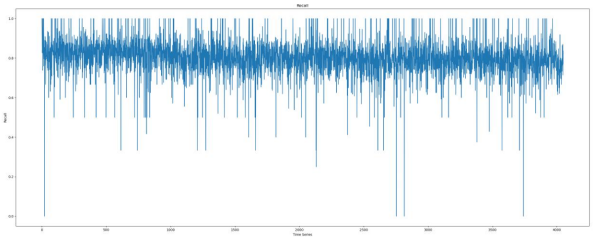
Fig. 2. (a) Extra Tree Classifier Accuracy. (b) Extra Tree Classifier Precission.(c) Extra Tree Classifier Recall. (d) Extra Tree Classifier F1. All the graph show in datastream mining



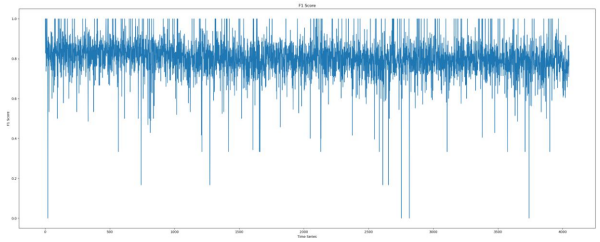
(a)



(b)

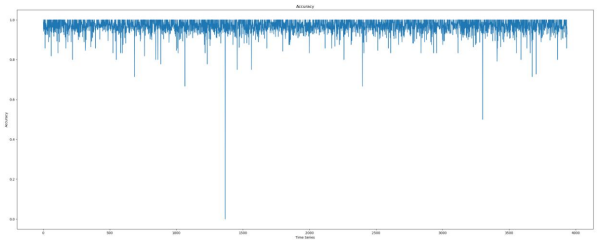


(c)

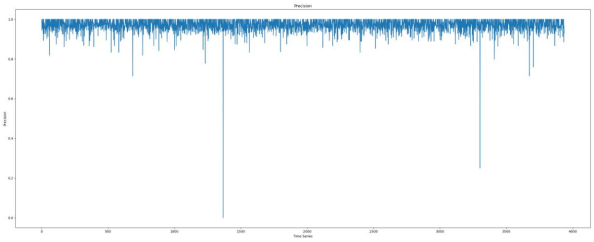


(d)

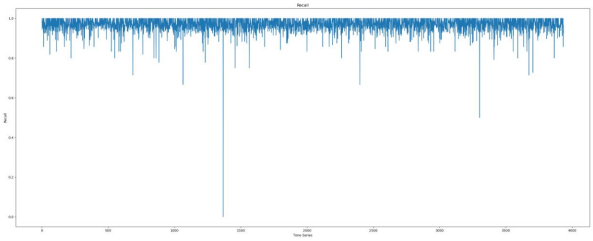
Fig. 3. (a) Gradient Boosting Accuracy. (b) Gradient Boosting Precission.(c) Gradient Boosting Recall. (d) Gradient Boosting F1. All the graph show in datastream mining



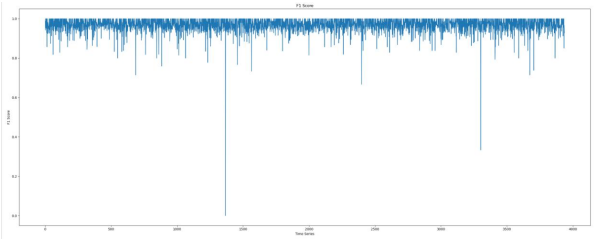
(a)



(b)

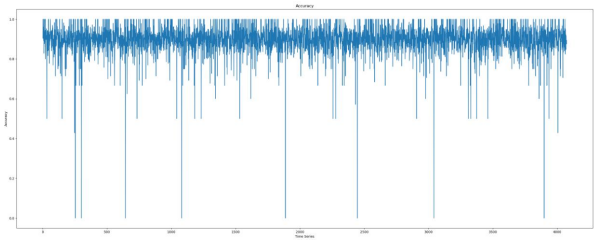


(c)

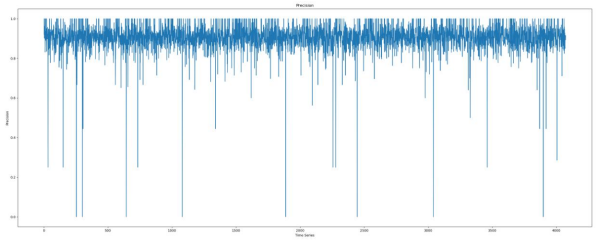


(d)

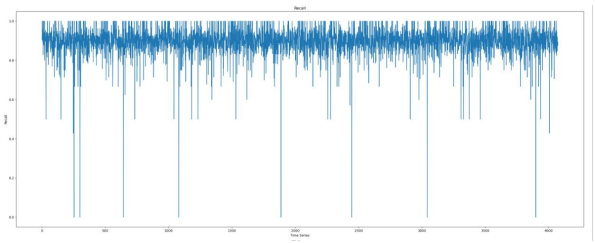
Fig. 4. (a) Random Forest Classifier Accuracy. (b) Random Forest Classifier Precission.(c) Random Forest Classifier Recall. (d) Random Forest Classifier F1. All the graph show in datastream mining



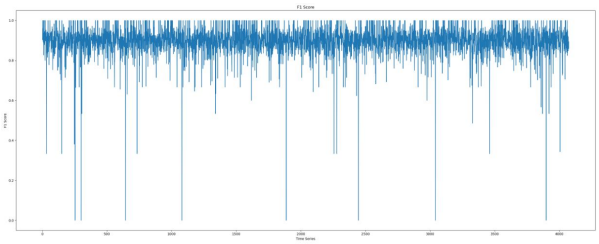
(a)



(b)

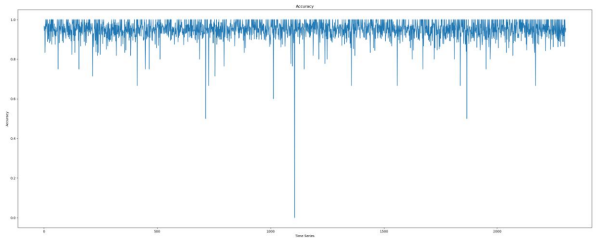


(c)

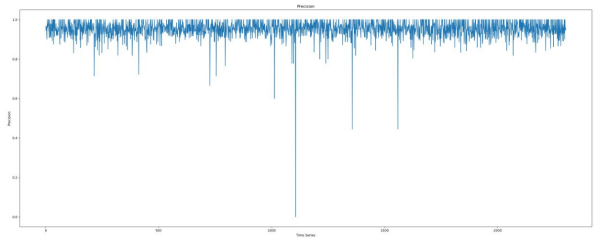


(d)

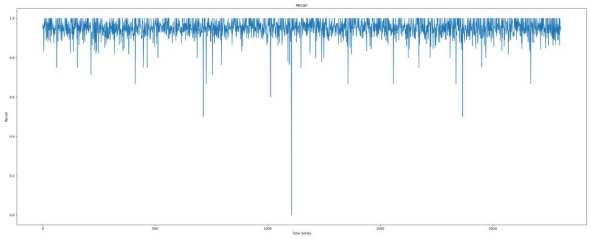
Fig. 5. (a) Naive Bayes Classifier Accuracy. (b) Naive Bayes Classifier Precission.(c) Naive Bayes Classifier Recall. (d) Naive Bayes Classifier F1. All the graph show in datastream mining



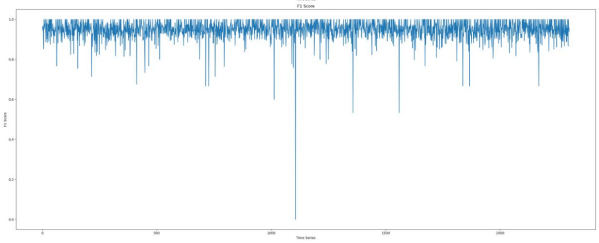
(a)



(b)

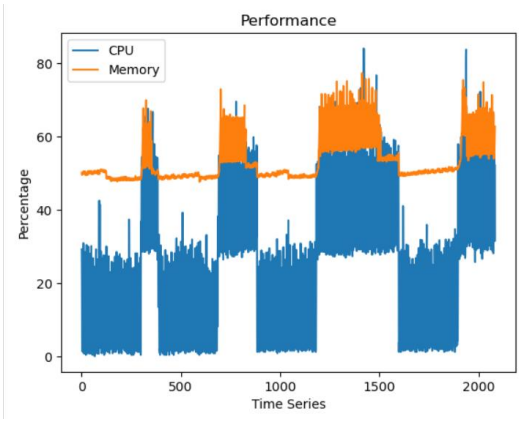


(c)

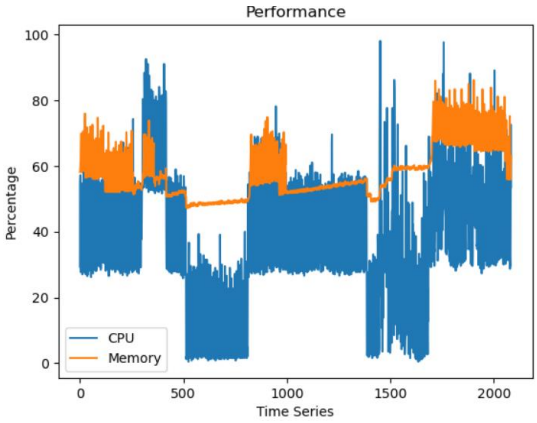


(d)

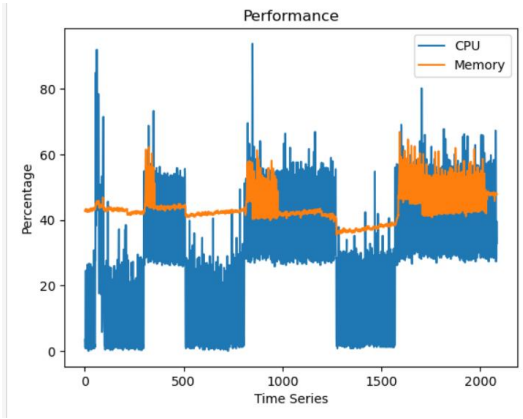
Fig. 6. (a) SVC Accuracy. (b) SVC Precission.(c) SVC Recall. (d) SVC F1. All the graph show in datastream mining



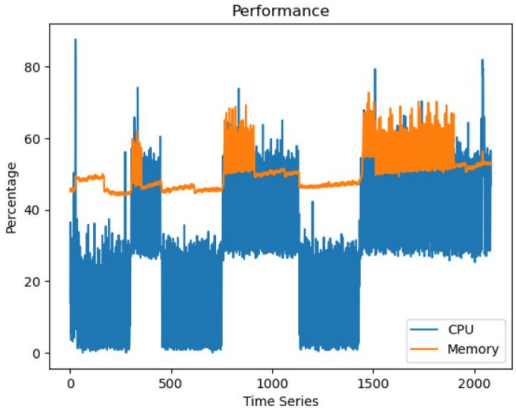
(a)



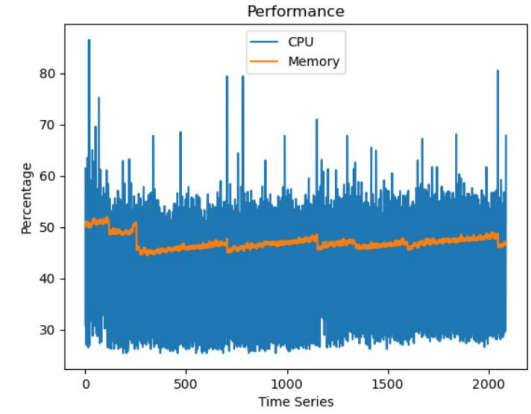
(b)



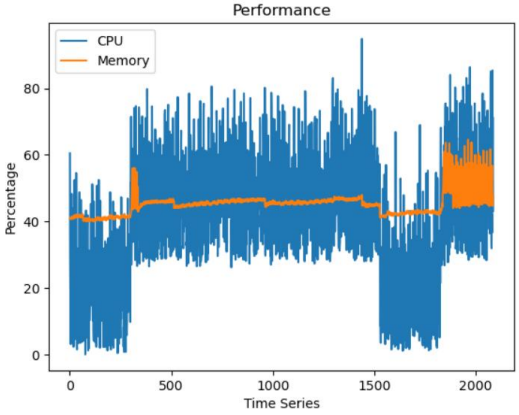
(c)



(d)



(e)



(f)

Fig. 7. (a) Adaboost Performance in datastream. (b) Extra Tree Classifier Performance in datastream.(c) Gradient Boosting Performance in datastream. (d) Random Forest Performance in datastream. (e) Naive Bayes Classifier Performance in datastream. (f) SVC Performance in datastream