Effective Garbage Data Filtering Algorithm for SNS Big Data Processing by Machine Learning

B. SRAVANI1, A. SHARANYA[2], M. AKHILA[3], G. SWATHI[4]

[1, 2, 3] *B. Tech Student, Dept. of CSE, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad*

[4] *Assistant Professor, Dept. of CSE, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad*

Abstract- Recently, as the use of social network services (SNS) increases in modern daily life, the amount of SNS data has become enormous. In addition, more and more efforts are being made to extract different pieces of information by collecting, processing, and analysing large amounts of SNS data. Although various pieces of information can be extracted from SNS data through big data processing, this is a resource-intensive task. Therefore, extracting information from SNS data requires considerable time and material resources. In this paper, we propose a data filtering algorithm that filters out junk data that has no data meaning in SNS data. The proposed algorithm improves the filtering accuracy by iterative learning based on the initial learning data. Experimental results show that the proposed algorithm has a filtering effect of more than 70% on experimental keywords.

Indexed Terms- Social network services, big data, machine learning, iterative learning.

I. INTRODUCTION

Due to the fast growth of social network services (SNS), the number of users has recently increased. As the number of mobile devices grows, so does the volume of data gather on social networking sites. SNS is frequently used for friendship and social interactions, but in recent years, its secondary usage for collecting, analysing, and acquiring various bits of information from large datasets on SNS has significantly increased. Therefore, by examining the data on SNS, it is possible to deduce information about a variety of flows and opinions on topics such as society, the economy, and politics. However, because the data on SNS is a mixture of relevant data, data from advertisements, and beneficial data for the research itself, it is highly difficult and time-consuming to analyse it successfully. Studies on stable data collection and storage as well as effective data processing with constrained computing resources have been done recently as interest in big-data processing has grown. The value of large data prior to processing, however, is the subject of less research and study [1].

Recently, the number of users of social network services (SNS) is increasing due to the explosive growth of mobile devices, and the amount of data generated on SNS is increasing correspondingly. SNS is widely used for social relations and friendship, but recently, it has been increasingly used for the secondary purpose of gathering and analyzing large datasets on SNS and obtaining various pieces of information. The data on SNS includes content related to opinions being expressed in various fields such as economy, society, and culture [2]. Therefore, by analyzing the data on SNS, information on various flows and opinions on topics such as society, economy, and politics can be extracted. However, it is very difficult and time consuming to accurately analyze the data on SNS as it consists of a mix between positive data that is helpful to the actual analysis, advertisement data, and irrelevant data. In recent years, as interest in big-data processing has increased, studies have been conducted on collecting and storing big data in a stable manner and more efficiently processing data using limited computing resources[3]. However, less research and fewer studies are available regarding the utility of big data before they are processed. Therefore, this study investigates how to effectively filter garbage data from big data, and thereby improve the accuracy and speed of the data analysis in real big-data processing as Figure 1. In particular, this study focuses on improving the filtering accuracy by including machine learning in the process of filtering garbage data. Therefore, in this study, we propose an algorithm that can improve the garbage data filtering accuracy of SNS big data by cyclic learning and prove the effectiveness of the algorithm through experiments. As a result, this work explores effective garbage data filtering from big data to enhance the correctness and speed of real-world big-data processing analysis. By introducing machine learning into the process of removing useless data, this study explicitly aims to increase filtering accuracy. Consequently, in this paper, we present a method that increases garbage data filtering accuracy using cyclic learning, and we use experiments to demonstrate the programme's efficacy.

II. LITERATURE SURVEY

Qiu et al.[4] There is no doubt that big data are now rapidly expanding in all science and engineering domains. While the potential of these massive data is undoubtedly significant, fully making sense of them requires new ways of thinking and novel learning techniques to address the various challenges. In this paper, we present a literature survey of the latest advances in researches on machine learning for big data processing. First, we review the machine learning techniques and highlight some promising learning methods in recent studies, such as representation learning, deep learning, distributed and parallel learning, transfer learning, active learning, and kernel-based learning. Next, we focus on the analysis and discussions about the challenges and possible solutions of machine learning for big data. Following that, we investigate the close connections of machine learning with signal processing techniques for big data processing. Finally, we outline several open issues and research trends.

Jarrah et al. [5] studied effective machine learning methods for processing big data. They explored data modelling methods and analysed the efficiencies of the model and algorithm. Landset et al. [6] classified tools for machine learning as processing engines, machine learning frameworks, and learning algorithms, and analysed their association. Machine learning frameworks such as Mahout, MLlib, H2O, and Samoa were also examined side by side. Xing et al. [7] analyzed and compared implementation engines such as MapReduce, Spark, Flink, Storm, and H2O in the Hadoop ecosystem, a typical machine learning architecture. In addition, machine learning libraries and frameworks such as Mahout, MLlib, and Samoa were examined. Chen et al. [8] proposed an algorithm for disease prediction based on machine learning for healthcare big data and demonstrated the effectiveness of the proposed algorithm through experiments. In addition, we can find some studies suggesting a big data processing method using various machine learning.

This paper focuses on the specific problem of Big Data classification of network intrusion traffic. It discusses the system challenges presented by the Big Data problems associated with network intrusion prediction. The prediction of a possible intrusion attack in a network requires continuous collection of traffic data and learning of their characteristics on the fly. The continuous collection of traffic data by the network leads to Big Data problems that are caused by the volume, variety and velocity properties of Big Data. The learning of the network characteristics requires machine learning techniques that capture global knowledge of the traffic patterns. The Big Data properties will lead to significant system challenges to implement machine learning frameworks. This paper discusses the problems and challenges in handling Big Data classification using geometric representation-learning techniques and the modern Big Data networking technologies

With the emerging technologies and all associated devices, it is predicted that massive amount of data will be created in the next few years – in fact, as much as 90% of current data were created in the last couple of years – a trend that will continue for the foreseeable future. Sustainable computing studies the process by which computer engineer/scientist designs computers and associated subsystems efficiently and effectively with minimal impact on the environment. However, current intelligent machine-learning systems are performance driven – the focus is on the predictive/classification accuracy, based on known properties learned from the training samples. For instance, most machine-learning-based nonparametric models are known to require high computational cost in order to find the global optima. With the learning task in a large dataset, the number of hidden nodes within the network will therefore increase significantly, which eventually leads to an exponential rise in computational complexity.

III. DATA FILTERING SYSTEM

In this section, we introduce the data filtering system. The core building blocks of the system are explained, and the functional roles of the components are discussed.

The proposed system consists of various components including a data classifier generator, a data classifier, and a data analyser. The data classifier generator is a unit tasked with performing data classification. The initial learning data are input, and the learned data are generated through morphological analysis, weighting, and application of the classification algorithm, and a data classification module is created based on the generated data. The data classifier receives target SNS data, namely sentences, to be filtered and performs classification into three groups of words: first, garbage data, then second, advertisement data, and third, definite (positive) data. The first and second groups are input once again in the data classifier generator and used to generate the classification module, and sentences included in the second and third group are entered into the pattern analyzer generator, which generates a pattern analysis module. This is accomplished through the sequence of morpheme analysis, application of classification algorithm, and use of vocabulary database by pattern type. The data analyzer receives sentence data included in the second and third group words as determined by the data classifier and generates various pieces of information. Data analyzers are not implemented in this study because they are off the subject of this study. Figure 2 shows the structure of proposed SNS garbage data filtering system.

- SYSTEM ARCHITECTURE

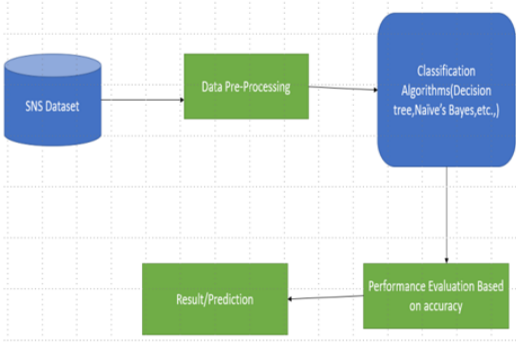

SNS Dataset
Data Pre-Processing
Classification Algorithms(Decision tree,Naïve's Bayes,etc.,)
Performance Evaluation Based on accuracy
Result/Prediction

Fig.1 System architecture

- B. Design of Data Classifier Generator

In the proposed system, the data classifier generator plays a key role. This module classifies an initial sentence data into the first, second or third group by referring to the data learned. In particular, in the proposed system, the machine learning system is implemented by executing the Mahout module in Hadoop system.

- NAÏVE'S BAYES:

It is a simple multiclass classification algorithm with the assumption of independence between every pair of features. Naive Bayes can be trained very efficiently. Within a single pass to the training data, it computes the conditional probability distribution of each feature given label, and then it applies Bayes' theorem to compute the conditional probability distribution of label given an observation and use it for prediction.

- RANDOM FOREST

It is a meta estimator that fits a number of decision tree classifiers on various sub samples of the dataset and use averaging to improve the predictive accuracy and control over fitting. It is a meta estimator that fits a number of decision tree classifiers on various sub samples of the dataset and use averaging to improve the predictive accuracy and control over fitting.

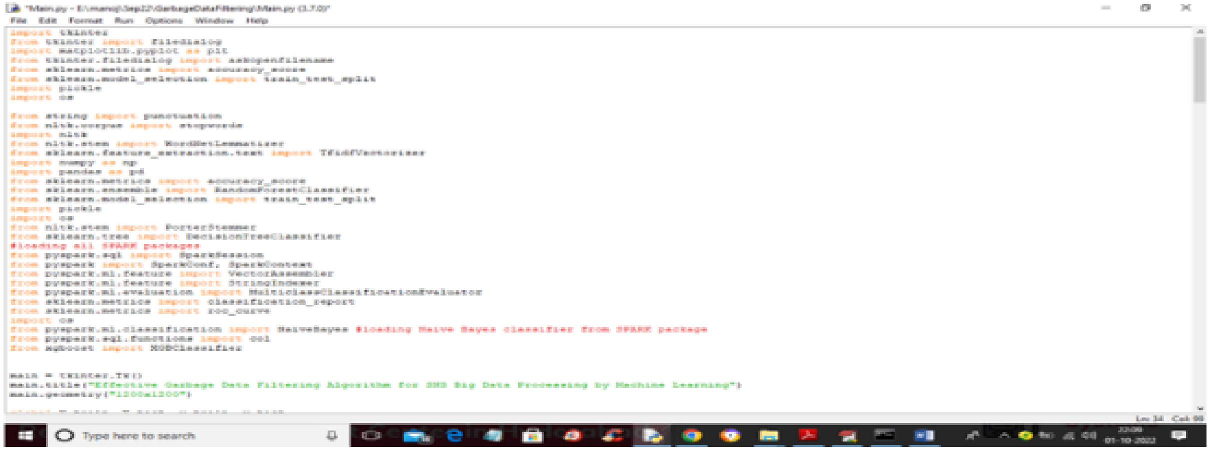
- DECISION TREE

The decision tree classifier creates the classification model by building a decision tree. Each node in the tree specifies a test on an attribute, each branch descending from that node corresponds to one of the possible values for that attribute.

- XG Boost Classifier

XG Boost Classifier It is a Machine learning algorithm that is applied for structured and tabular data. XGBoost is an implementation of gradient boosted decision trees designed for speed and performance.

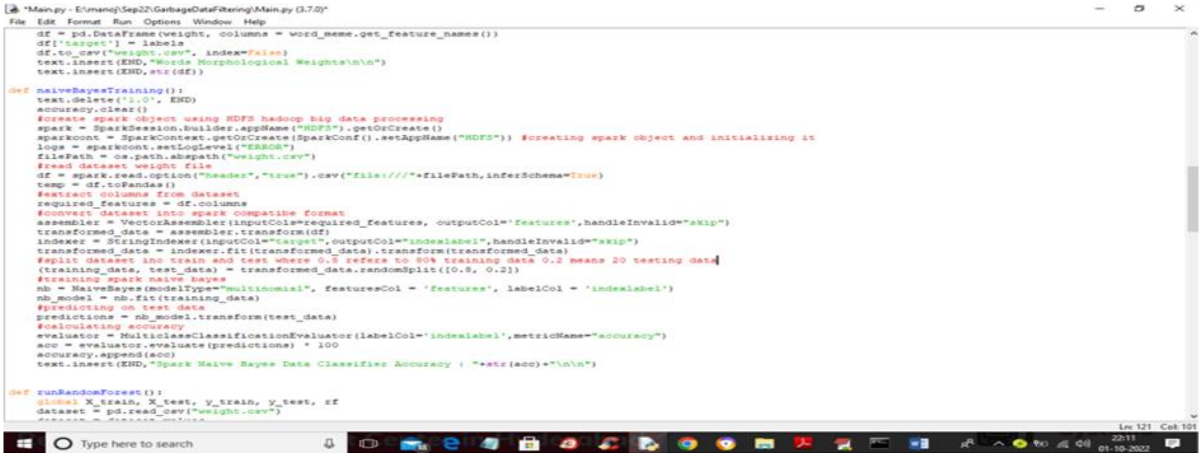
IV. RESULTS



```
import tkinter
from tkinter import filedialog
import matplotlib.pyplot as plt
from tkinter.filedialog import askopenfilename
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split
import pickle
import os

from string import punctuation
from nltk.corpus import stopwords
import nltk
from nltk.stem import WordNetLemmatizer
from sklearn.feature_extraction.text import TfidfVectorizer
import numpy as np
import pandas as pd
from sklearn.metrics import accuracy_score
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
import pickle
import os
from nltk.stem import PorterStemmer
from sklearn.tree import DecisionTreeClassifier
#loading all SPARK packages
from pyspark.sql import SparkSession
from pyspark import SparkConf, SparkContext
from pyspark.ml.feature import VectorAssembler
from pyspark.ml.feature import StringIndexer
from pyspark.ml.evaluation import MulticlassClassificationEvaluator
from sklearn.metrics import classification_report
from sklearn.metrics import roc_curve
import os
from pyspark.ml.classification import NaiveBayes #loading Naive Bayes classifier from SPARK package
from pyspark.sql.functions import col
from xgboost import XGBClassifier


main = tkinter.Tk()
main.title("Effective Garbage Data Filtering Algorithm for SNS Big Data Processing by Machine Learning")
main.geometry("1300x1200")
```

Main.py - E:\manoj\Sep22\GarbageDataFiltering\Main.py (3.7.0)
File Edit Format Run Options Window Help
Type here to search

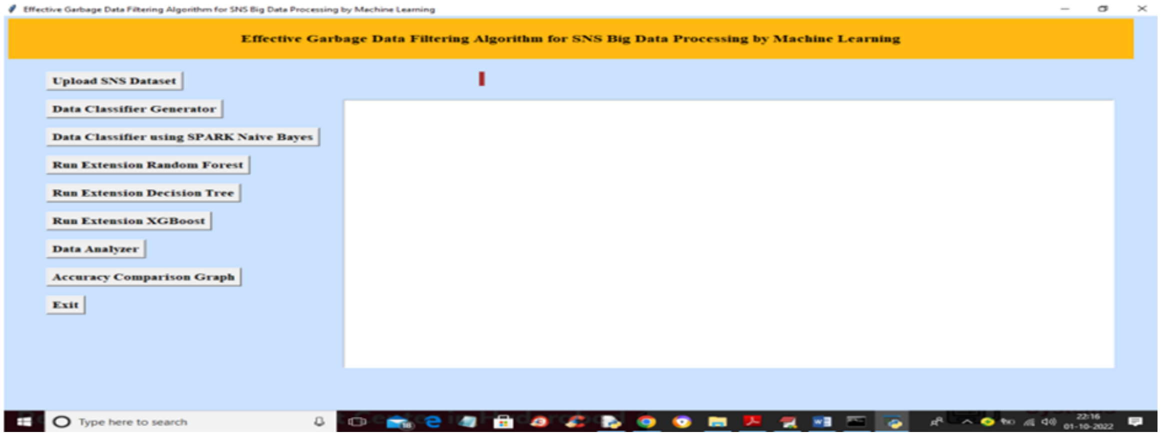
Fig.2 In above screen we are showing code for SPARK and Naïve Bayes processing



```
    df = pd.DataFrame(weight, columns = word_meme.get_feature_names())
    df['target'] = labels
    df.to_csv("weight.csv", index=False)
    text.insert(END,"Words Morphological Weights\n\n")
    text.insert(END,str(df))

def naiveBayesTraining():
    text.delete('1.0', END)
    accuracy.clear()
    #create spark object using HDFS hadoop big data processing
    spark = SparkSession.builder.appName("HDFS").getOrCreate()
    sparkcont = SparkContext.getOrCreate(SparkConf().setAppName("HDFS")) #creating spark object and initializing it
    logs = sparkcont.setLogLevel("ERROR")
    filePath = os.path.abspath("weight.csv")
    #read dataset weight file
    df = spark.read.option("header","true").csv("file:///"+filePath,inferSchema=True)
    temp = df.toPandas()
    #extract columns from dataset
    required_features = df.columns
    #convert dataset into spark compatibe format
    assembler = VectorAssembler(inputCols=required_features, outputCol='features',handleInvalid="skip")
    transformed_data = assembler.transform(df)
    indexer = StringIndexer(inputCol="target",outputCol="indexlabel",handleInvalid="skip")
    transformed_data = indexer.fit(transformed_data).transform(transformed_data)
    #split dataset ino train and test where 0.8 refers to 80% training data 0.2 means 20 testing data
    (training_data, test_data) = transformed_data.randomSplit([0.8, 0.2])
    #training spark naive bayes
    nb = NaiveBayes(modelType="multinomial", featuresCol = 'features', labelCol = 'indexlabel')
    nb_model = nb.fit(training_data)
    #predicting on test data
    predictions = nb_model.transform(test_data)
    #calculating accuracy
    evaluator = MulticlassClassificationEvaluator(labelCol='indexlabel',metricName="accuracy")
    acc = evaluator.evaluate(predictions) * 100
    accuracy.append(acc)
    text.insert(END,"Spark Naive Bayes Data Classifier Accuracy : "+str(acc)+"\n\n")


def runRandomForest():
    global X_train, X_test, y_train, y_test, rf
    dataset = pd.read_csv("weight.csv")
```

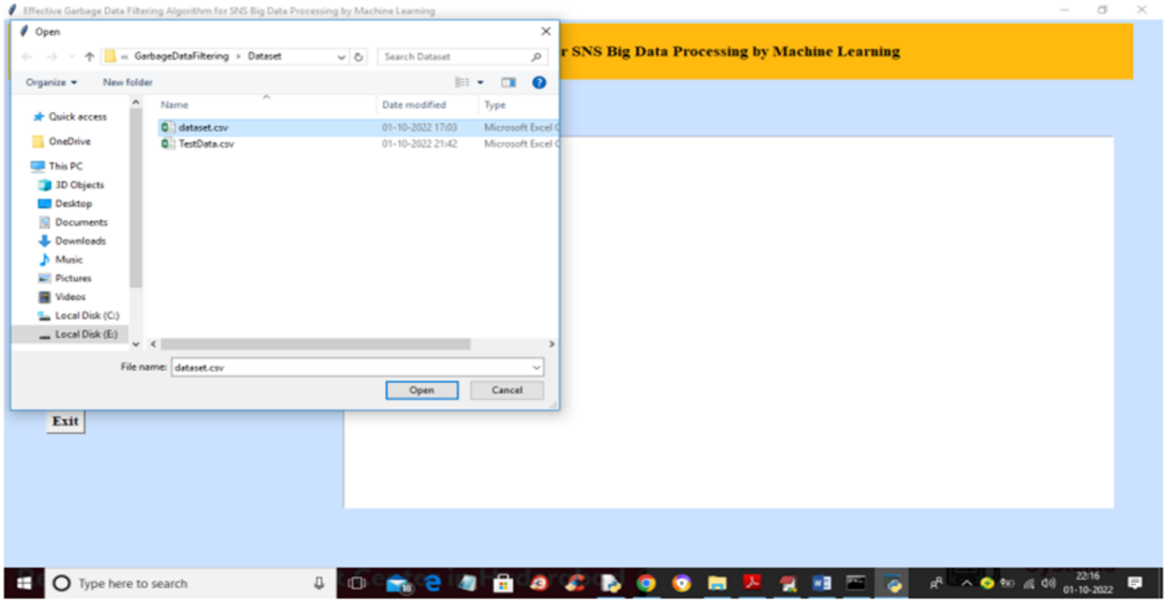
Main.py - E:\manoj\Sep22\GarbageDataFiltering\Main.py (3.7.0)
File Edit Format Run Options Window Help
Ln: 121 Col: 101
Type here to search

Fig.3 In above screen we are processing dataset weight using SPARK and then using Naïve Bayes algorithm for training



Effective Garbage Data Filtering Algorithm for SNS Big Data Processing by Machine Learning
Effective Garbage Data Filtering Algorithm for SNS Big Data Processing by Machine Learning
Upload SNS Dataset
Data Classifier Generator
Data Classifier using SPARK Naive Bayes
Run Extension Random Forest
Run Extension Decision Tree
Run Extension XGBoost
Data Analyzer
Accuracy Comparison Graph
Exit
Type here to search

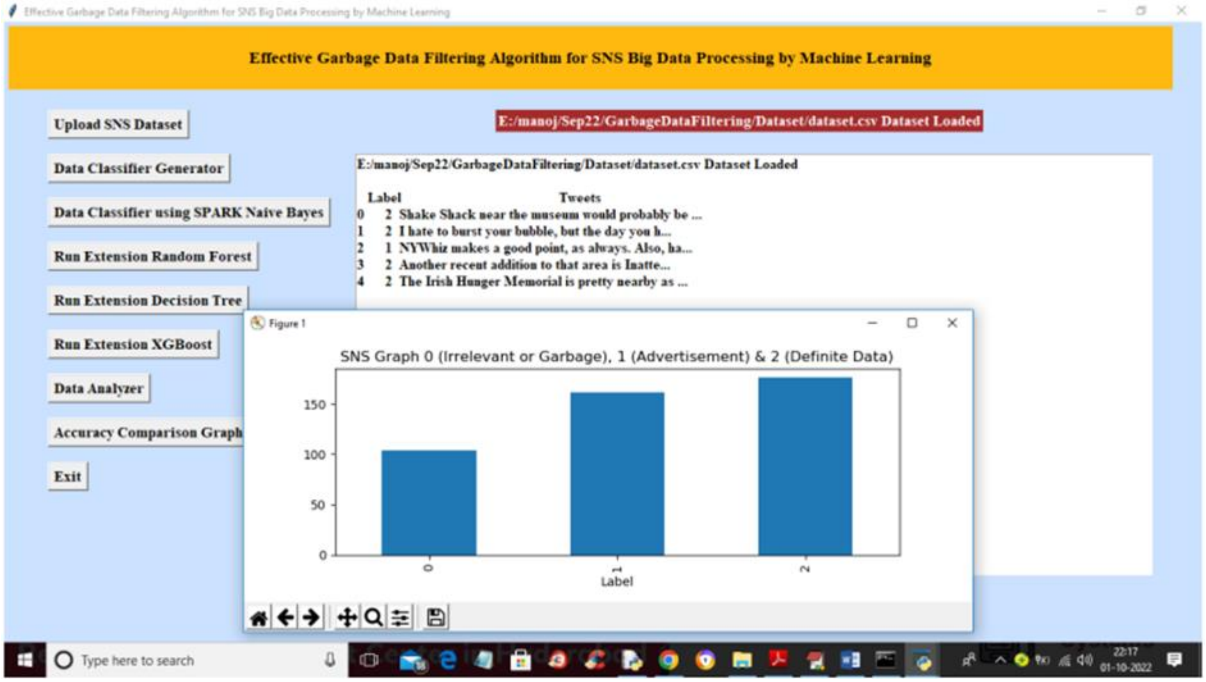
Fig.4 In above screen click on ‘Upload SNS Dataset’ button to load dataset and get below screen



Name	Date modified	Type
dataset.csv	01-10-2022 17:03	Microsoft Excel C
TestData.csv	01-10-2022 21:42	Microsoft Excel C

File name: dataset.csv

Fig.5 In above screen selecting and uploading ‘dataset.csv’ file and then click on ‘Open’ button to load dataset and get below output

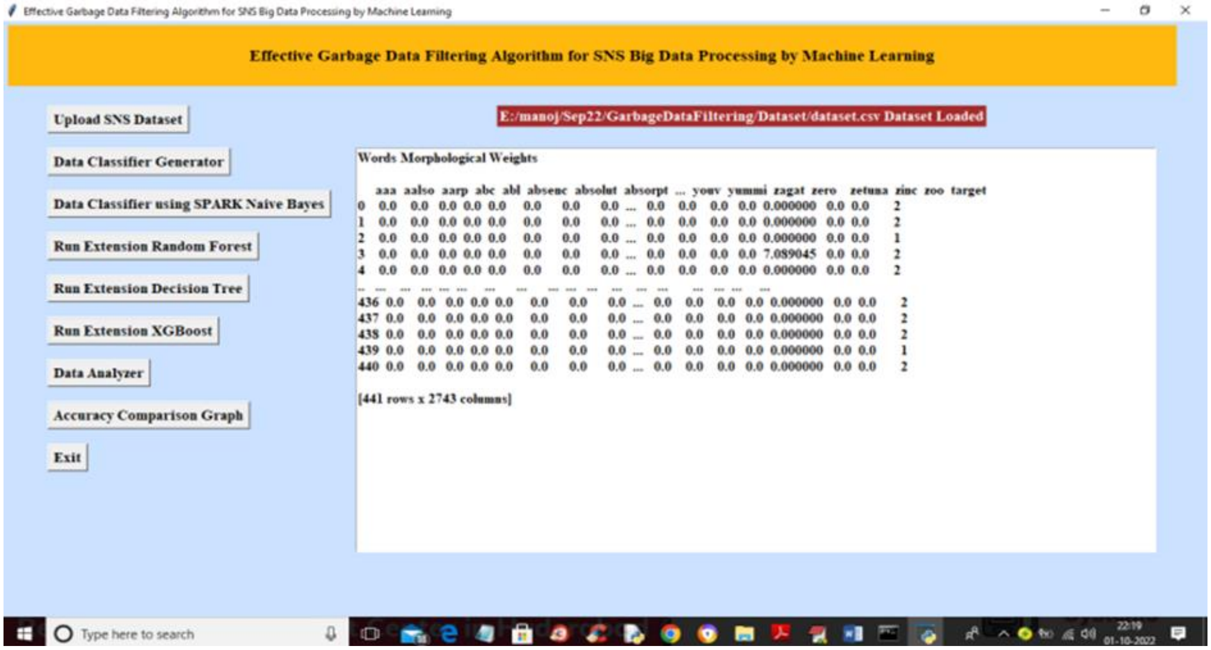


Effective Garbage Data Filtering Algorithm for SNS Big Data Processing by Machine Learning
Effective Garbage Data Filtering Algorithm for SNS Big Data Processing by Machine Learning
Upload SNS Dataset
Data Classifier Generator
Data Classifier using SPARK Naive Bayes
Run Extension Random Forest
Run Extension Decision Tree
Run Extension XGBoost
Data Analyzer
Accuracy Comparison Graph
Exit
Figure 1
SNS Graph 0 (Irrelevant or Garbage), 1 (Advertisement) & 2 (Definite Data)
150
100
50
0
0
1
2
Label
Type here to search
22:17
01-10-2022

E:/manoj/Sep22/GarbageDataFiltering/Dataset/dataset.csv Dataset Loaded

```
E:/manoj/Sep22/GarbageDataFiltering/Dataset/dataset.csv Dataset Loaded

  Label                                    Tweets
0     2  Shake Shack near the museum would probably be ...
1     2  I hate to burst your bubble, but the day you h...
2     1  NYWhiz makes a good point, as always. Also, ha...
3     2  Another recent addition to that area is Inatte...
4     2  The Irish Hunger Memorial is pretty nearby as ...
```

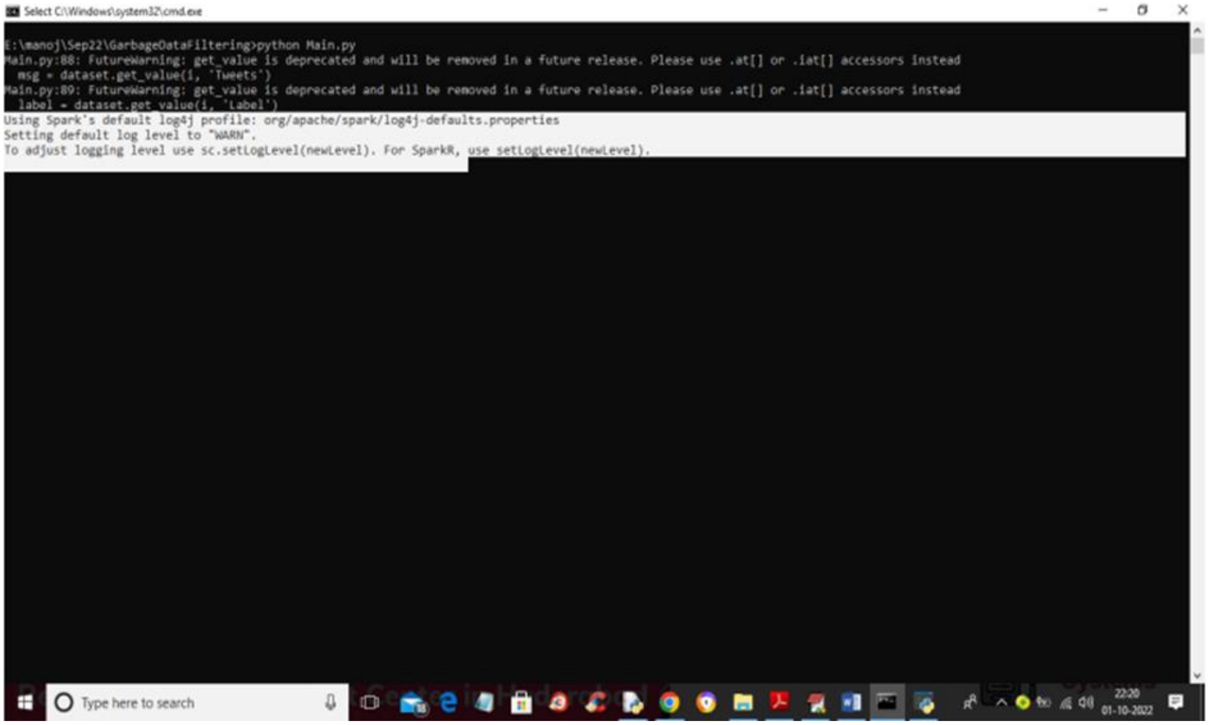
Fig.6 In above screen dataset loaded and in graph x-axis represents types of data as 0, 1 or 2 and y-axis represents number of records found in dataset in that group and now click on ‘Dataset Classifier Generator’ to convert dataset tweets into morphologic weights and get below output.


Effective Garbage Data Filtering Algorithm for SNS Big Data Processing by Machine Learning
Effective Garbage Data Filtering Algorithm for SNS Big Data Processing by Machine Learning
Upload SNS Dataset
Data Classifier Generator
Data Classifier using SPARK Naive Bayes
Run Extension Random Forest
Run Extension Decision Tree
Run Extension XGBoost
Data Analyzer
Accuracy Comparison Graph
Exit
Type here to search

E:/manoj/Sep22/GarbageDataFiltering/Dataset/dataset.csv Dataset Loaded

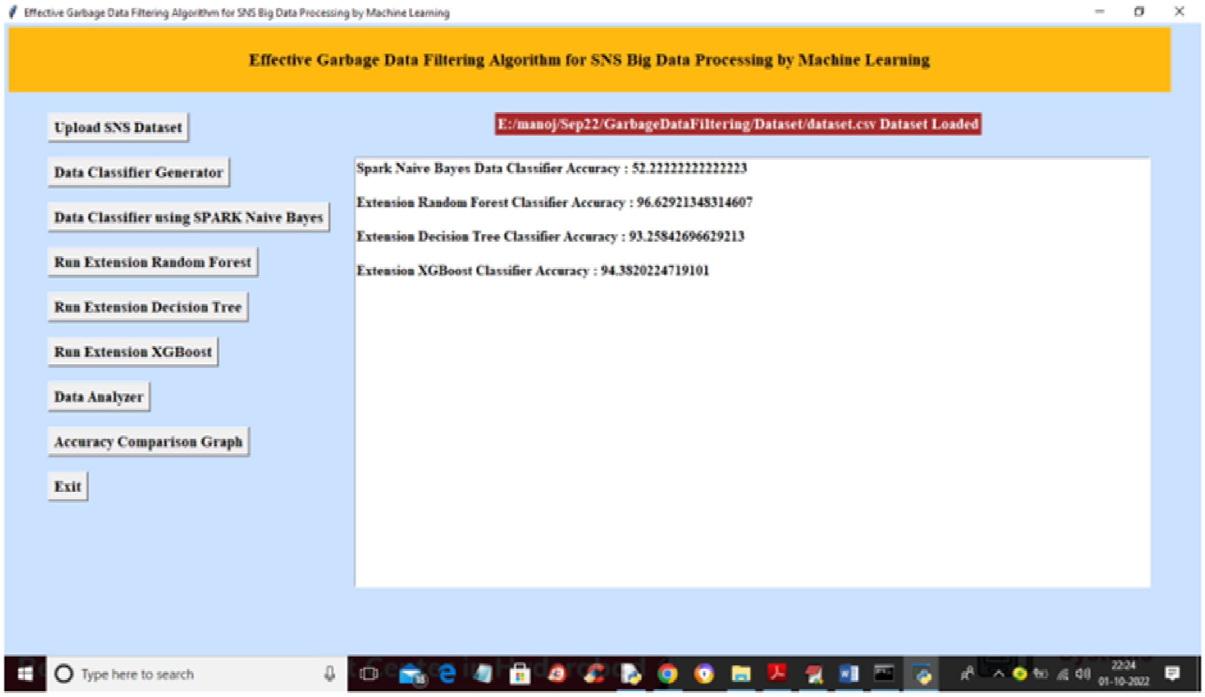
```
Words Morphological Weights

   aaa  aalso  aarp  abc  abl  absenc  absolut  absorpt  ...  youv  yummi  zagat  zero    zetuna  zinc  zoo  target
0   0.0   0.0   0.0  0.0  0.0     0.0      0.0      0.0  ...   0.0    0.0    0.0   0.0  0.000000   0.0  0.0      2
1   0.0   0.0   0.0  0.0  0.0     0.0      0.0      0.0  ...   0.0    0.0    0.0   0.0  0.000000   0.0  0.0      2
2   0.0   0.0   0.0  0.0  0.0     0.0      0.0      0.0  ...   0.0    0.0    0.0   0.0  0.000000   0.0  0.0      1
3   0.0   0.0   0.0  0.0  0.0     0.0      0.0      0.0  ...   0.0    0.0    0.0   0.0  7.089045   0.0  0.0      2
4   0.0   0.0   0.0  0.0  0.0     0.0      0.0      0.0  ...   0.0    0.0    0.0   0.0  0.000000   0.0  0.0      2
..  ...   ...   ...  ...  ...     ...      ...      ...  ...   ...    ...    ...   ...       ...   ...  ...    ...
436 0.0   0.0   0.0  0.0  0.0     0.0      0.0      0.0  ...   0.0    0.0    0.0   0.0  0.000000   0.0  0.0      2
437 0.0   0.0   0.0  0.0  0.0     0.0      0.0      0.0  ...   0.0    0.0    0.0   0.0  0.000000   0.0  0.0      2
438 0.0   0.0   0.0  0.0  0.0     0.0      0.0      0.0  ...   0.0    0.0    0.0   0.0  0.000000   0.0  0.0      2
439 0.0   0.0   0.0  0.0  0.0     0.0      0.0      0.0  ...   0.0    0.0    0.0   0.0  0.000000   0.0  0.0      1
440 0.0   0.0   0.0  0.0  0.0     0.0      0.0      0.0  ...   0.0    0.0    0.0   0.0  0.000000   0.0  0.0      2

[441 rows x 2743 columns]
```

Fig.7 In above screen first row represents word and remaining rows contains weight of that word and now click on 'Data Classifier using SPARK Naive Bayes' button to train Naïve Bayes algorithm and get below prediction accuracy



```
E:\manoj\Sep22\GarbageDataFiltering>python Main.py
Main.py:88: FutureWarning: get_value is deprecated and will be removed in a future release. Please use .at[] or .iat[] accessors instead
  msg = dataset.get_value(i, 'Tweets')
Main.py:89: FutureWarning: get_value is deprecated and will be removed in a future release. Please use .at[] or .iat[] accessors instead
  label = dataset.get_value(i, 'Label')
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
```

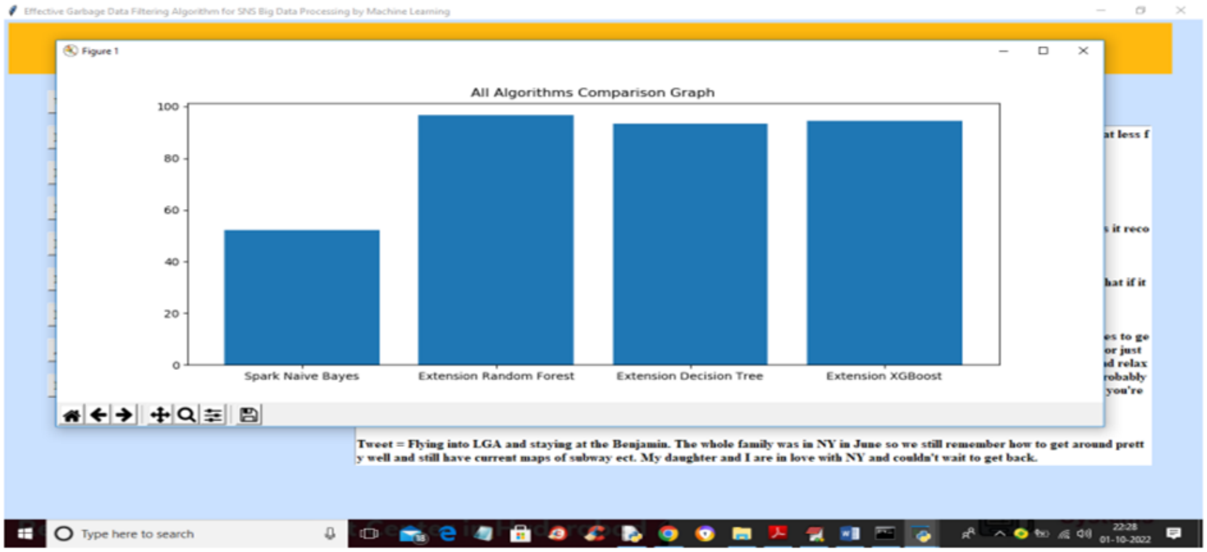
Fig.8 In above screen SPARK processing and naïve Bayes training started and after some time will get below output


Effective Garbage Data Filtering Algorithm for SNS Big Data Processing by Machine Learning
Effective Garbage Data Filtering Algorithm for SNS Big Data Processing by Machine Learning
Upload SNS Dataset
Data Classifier Generator
Data Classifier using SPARK Naive Bayes
Run Extension Random Forest
Run Extension Decision Tree
Run Extension XGBoost
Data Analyzer
Accuracy Comparison Graph
Exit
Type here to search
22:24
01-10-2022

E:/manoj/Sep22/GarbageDataFiltering/Dataset/dataset.csv Dataset Loaded

```
Spark Naive Bayes Data Classifier Accuracy : 52.22222222222223

Extension Random Forest Classifier Accuracy : 96.62921348314607

Extension Decision Tree Classifier Accuracy : 93.25842696629213

Extension XGBoost Classifier Accuracy : 94.3820224719101
```

Fig.9 In above screen with XGBOOST we got 94% accuracy and now click on 'Data Analyzer' button to upload test data and then classifier algorithm will predict group of test data



Tweet = Flying into LGA and staying at the Benjamin. The whole family was in NY in June so we still remember how to get around prett y well and still have current maps of subway ect. My daughter and I are in love with NY and couldn't wait to get back.

Fig.10 In above graph x-axis represents algorithm names and y-axis represents accuracy of those algorithms and in above graph we can see all extension algorithms got high accuracy compare to propose algorithms.

CONCLUSION

In this paper, we proposed and implemented an effective SNS garbage data filtering system through repetitive machine learning. We assume that the proposed system can improve the accuracy of the analysis of unstructured data in SNS by separating it into garbage, advertisement, and definite data through machine learning. Concerning the accuracy experiment, data filtering showed an accuracy of up to

74.45% following a comparison with the correct answer set. Therefore, it is found that it may be advantageous in a big data processing environment where a large amount of data must be processed quickly. Based on this, the contribution of this study is summarized as follows. First, this study proposed an effective garbage and advertisement data filtering system that can be used in big data processing system. It is designed to enhance the efficiency of big data processing by selecting and processing only data that is worth processing from a large amount of data generated in daily life such as SNS big data. Second, we introduced a recursive machine learning method for data filtering. We made initial learning data from SNS big data and used it for data filtering, and we improved the accuracy of filtering by using the filtered data as learning data through the proposed system.

REFERENCES

[1] J. Qiu, Q. Wu, G. Ding, Y. Xu, S. Feng, “A survey of machine learning for big data processing,” EURASIP J. Adv. Signal Process. vol. 2016, pp. 1-16, 2016.

[2] S. Suthanharan, “Big data classification: Problems and challenges in network intrusion prediction with machine learning,” ACM SIGMETRICS Perf. Eval. Rev. vol. 41, pp. 70-73, 2014. [

[3] O. Jarrah, P. Yoo, S. Muhaidat, G. Karagiannidis, K. Taha, “Efficient Machine Learning for Big Data: A Review,” Big Data Res. vol. 2, pp. 87-93, 2015.

[4] S. Landset, T. Khoshgoftaar, A. Richter, T. Hasanin, “A survey of open source tools for machine learning with big data in the Hadoop ecosystem,” J. Big Data, vol. 2, pp. 1-36, 2015.

[5] E. Xing, Q. Ho, W. Dai, J. Kim, Y. Yu, “Petuum: A New Platform for Distributed Machine Learning on Big Data,” IEEE Trans. Big Data, vol. 1, pp. 49-67, 2015.

[6] M. Chen, Y. Hao, K. Hwang, L. Wang, L. Wang, “Disease Prediction by Machine Learning Over Big Data from Healthcare Communities,” IEEE Access, vol. 5, pp. 8869–8879, 2017.

[7] M. Gunasekaran, V. Vijayakumar, R. Varatharajan, K. Priyan S. Revathi, H. Ching-Hsien, “Machine Learning Based Big Data Processing Framework for Cancer Diagnosis Using Hidden Markov Model and GM Clustering,” Wireless Personal Communications, vol. 102, pp. 2099-2116, 2018.

[8] W. Xiaofei, Z. Yuhua, L. Victor, G. Nadra, J. Tianpeng, “D2D Big Data: Content Deliveries over Wireless Device-to-Device Sharing in Large-Scale Mobile Networks,” IEEE Wireless Communications. vol. 25, pp. 32-38, 2018.

[9] Z. Zhenhua, H. Qing, G. Jing, N. Ming, “A deep learning approach for detecting traffic accidents from social media data,” Transportation Research Part C: Emerging Technologies, vol. 86, pp. 580-596, 2017.

[10] S. Ou; J. Lee, “Implementation of a Spam Message Filtering System using Sentence Similarity Measurements,” KIISE Trans. Comput. Pract. (KTCP), vol. 23, pp. 57-64, 2017.