Sentiment Analysis in Stock Price Prediction: A Comparative Study of Algorithms

Aryan Agarwal
Dept. of CSE,
Graphic Era Hill University
Dehradun, India
https://orcid.org/0000-0001-9575-4240

Satvik Vats
Dept. of CSE,
Graphic Era Hill University
Dehradun, India
https://orcid.org/0000-0002-9422-4915

Ruchi Agarwal
JIMS Engineering Management Technical Campus
Greater Noida, India
dr.ruchiagarwal14@gmail.com

Aryan Ratra
Dept. of CSE,
Graphic Era Hill University
Dehradun, India
https://orcid.org/0000-0003-3260-0708

Vikrant Sharma
Dept. of CSE,
Graphic Era Hill University
Dehradun, India
https://orcid.org/0000-0003-3178-8657

Lisa Gopal
Dept. of CSE,
Graphic Era Hill University
Dehradun, India
https://orcid.org/0000-0002-0805-2652

***Abstract*— The development and wealth of countries depend heavily on the stock market. Data mining and artificial intelligence methods are required to analyze stock market data. The financial success of particular businesses is one of the important factors that has a significant impact on stock price volatility. However, news reports also have a significant impact on how the stock market moves. In this research, we use sentiment classification to use non-measurable data, such as financial news articles, to forecast a company's future stock trend. We seek to cast light on the effect of news reports on the stock market by analyzing the connection between news and stock movement. Our study seeks to advance knowledge of the function of news sentiment in forecasting stock market trends.**

***Keywords*— financial, Sentiment analysis, stock, non-quantifiable, Naive Bayes, profits, Random Forest, instability, Stock trends.**

I. INTRODUCTION

The rise of the internet and social media platforms have brought about an explosion of user-generated content on the web, such as blogs, social media posts, forums, and product reviews.[1] As a result, businesses and organizations face a daunting task of analyzing this unstructured data to understand their customers' needs, preferences, and sentiments towards their products or services. Sentiment analysis, also known as opinion mining, is a technique used to extract subjective information from text data.[4]

Sentiment analysis is a crucial field of study within natural language processing and machine learning because of its numerous practical uses, such as analyzing customer feedback, social media monitoring, and product recommendations.[9] It involves extracting subjective information from textual data and categorizing it as positive, negative, or neutral through the application of natural language processing techniques.

In recent years, machine learning algorithms have emerged as a popular approach for sentiment analysis. These algorithms can learn from labeled training data to predict the sentiment of text data with high accuracy.[17] In this project, we will explore the effectiveness of different machine learning algorithms, including CountVectorizer, TF-IDF Vectorizer(Term Frequency-Inverse Document Frequency), and Naive Bayes, in sentiment analysis of product reviews. The goal is to compare the performance of these algorithms and identify the best one for the task at hand.

We will use a dataset of product reviews from Amazon, a popular e-commerce platform, as the source of our data. The dataset contains thousands of reviews for different products in various categories, including electronics, books, and home appliances. We will preprocess the data to remove noise, transform the text into numerical features using CountVectorizer and TfidfVectorizer, and apply Naive Bayes classifier to predict the sentiment of the reviews.

Using measures like accuracy, precision, recall, and F1-score, we will assess the effectiveness of the machine learning algorithms. Furthermore, we will visualize the results using confusion matrices to gain insights into the performance of the classifiers. Finally, we will compare our results with existing studies in the field and draw conclusions on the effectiveness of these algorithms for sentiment analysis of product reviews.

II. DATASET COLLECTION & REPRESENTATION

For the purpose of analysing stock trend, I have gathered news article data from DJIA (Dow Jones Industrial Average) of various nations spanning more than fifteen years.[10]

Text data is unstructured data. Therefore, raw test data cannot be provided to the algorithm. In order to operate at the word level, the document must first be tokenized into words. Textual data has more distracting phrases that don't help with classification. Therefore, those lines had to be deleted. Additionally, text data may include tabs, punctuation, empty spaces, and other characters. Additionally, I must eliminate all of those terms from the data.[13] For this, I had removed punctuation and changed the news words into lower case using regular expression (regex).

Since headlines were divided into 25 columns in my dataset, To find stock trends, I had to combine these 25 columns into one column for each dataset entry.[6] Text

documents must be simplified and made easier to work with by being converted from their complete text version to a document vector that describes their contents.[12] To describe text documents, we utilize the TF-IDF approach, which assigns a score to each word based on its importance. A word's score increases as it becomes more significant. This weighting method assigns a high score to words that appear frequently in a document but infrequently in other documents within the collection.[16] It tends to devalue common words by assigning them a low score. Count Vectorizer is also utilized in conjunction with this to compare the two.

The Python scikit-learn package provides a useful tool called Count Vectorizer, which is used to transform text into a vector based on the frequency of each word occurring in the entire text.[18] This is beneficial when analyzing multiple documents and converting phrases into vectors for further analysis.

In a study using news articles from December 2014 to December 2016 as an unknown test set, it was found that the Random Forest algorithm with Count Vectorizer and the Naive Bayes algorithm both performed better than the Random Forest algorithm with TF-IDF Vectorizer when analyzing the results of all classifiers.

III. Literature Survey

The importance of sentiment research in predicting stock prices has grown.[21] The creation of several algorithms for predicting stock prices based on sentiment analysis of news stories and social media posts has been facilitated by recent developments in natural language processing. In this study, we test the effectiveness of three algorithms for sentiment analysis on the daily newspaper headlines of a particular company: CountVectorizer, TfidfVectorizer, and Naive Bayes.

Previous research has looked into the use of sentiment analysis to forecast stock values. Li and Li (2011) developed a regression model that beat a benchmark model using sentiment analysis on data from stock message boards. Zhang et al. (2011) found that sentiment analysis increased the precision of prediction models when applied to financial news stories. Machine learning algorithms like Naive Bayes, Support Vector Machines (SVM), Random Forest, and XGBoost have been extensively used in recent years to predict stock prices using sentiment analysis. The most successful machine learning algorithms, according to Akita and Kitagawa (2016)'s tests on financial news articles, were Naive Bayes, SVM, and Random Forest.[19]

Common natural language processing methods for feature extraction from text include CountVectorizer and TfidfVectorizer. The CountVectorizer algorithm turns text documents into a feature matrix of token counts, whereas the TfidfVectorizer algorithm transforms text documents into a feature matrix of term frequency-inverse document frequency (TF-IDF) values.

Overall, sentiment analysis has shown promise for predicting stock prices, and earlier studies have shown the effectiveness of machine learning algorithms like CountVectorizer, TfidfVectorizer, and Naive Bayes.[8] By applying these algorithms to daily news headlines and offering insights into their efficacy in sentiment analysis for stock price prediction, this initiative seeks to advance existing research.[22]

IV. Approach And Methodology

Natural language processing (NLP) techniques are used in this project's methodology and strategy to evaluate sentiment in texts from a variety of sources, including news articles, social media posts, and financial reports. The research contrasts various sentiment analysis algorithms, such as lexicon-based methods and machine learning models.[7] The effect of various sentiment characteristics, including polarity and subjectivity, on the precision of stock price prediction is also examined by the writers. The methodology of the research includes data collection and preprocessing, model training and testing, and evaluation of performance measures like recall, accuracy, and F1 score.[3] The results of this research can have significant ramifications for financial decision-making and offer information on how well various sentiment analysis techniques can forecast stock prices.

A. Implemented Algorithms

- **Random Forest Classifier:** An ensemble learning algorithm called Random Forest combines various decision trees to produce more accurate forecasts. It is a type of bagging method in which the algorithm generates several decision trees and then aggregates the output of each tree to make the final prediction.It is possible that prediction of some trees may be wrong, and some may be correct. But combination of these output will be correct. It takes much less training time as compared to other algorithms and can predict correct output when large proportions of values are missing. In Random Forest, an arbitrarily chosen subset of the training data is used to construct each decision tree. This prevents the decision trees from overfitting the data and ensures that they are varied. The decision tree forest's decision trees are averaged or polled to determine the ultimate prediction.
- **Vectorizer:** The vectorization procedure in natural language processing involves transforming text data into a numerical format that machine learning algorithms can comprehend. A vectorizer is a tool that does this conversion, usually by creating a matrix of word counts or frequency statistics.
- **CountVectorizer:** CountVectorizer is a tool for transforming text data into a table of word frequency counts. The table has a column for each unique word in the vocabulary and a row for each document. The cells in the table represent the number of times a particular word appears in a given document. It is a method of representing text data in numerical form where each word is represented by a number of occurrences in the text. This approach builds a vocabulary from training data, and each document is represented as a vector with values equal to the number of terms in the vocabulary. The resulting vectors are then used as input to the Random Forest algorithm for making predictions.[2]

- **TF-IDF Vectorizer:** Tf-idf Vectorizer is another type of vectorizer that converts text data into a matrix of term frequencies and inverse document frequencies (TF-IDF). This method is similar to CountVectorizer, but it takes into account the importance of each word in the document and in the corpus as a whole. Each word is represented by its term frequency-inverse document frequency in this way of numerically expressing text data. According to this technique, a word's significance in a document is assessed based on both its rarity within the document and its frequency within the corpus of documents. The resulting vectors are then used as input to the Random Forest algorithm for making predictions.
- **Naive Bayes:** Naive Bayes is a probabilistic classification algorithm that is often used in natural language processing tasks such as text classification. It is based on the Bayes theorem, which determines the likelihood that a document belongs to a particular class based on the likelihood that its words occur in that class. Naive Bayes assumes that the probability of each word is independent of all other words in the document, hence the name "naive." Naive Bayes algorithm is used in text classification problems and it performs well when the features are independent of each other. It is compatible with numerous feature extraction methods like Count Vector and TF-IDF Vector.

In summary, there are two feature extraction methods, Count Vector and TF-IDF Vector, that can be utilized with the Random Forest algorithm. This algorithm is powerful and can be applied to both classification and regression tasks. Additionally, the Naive Bayes algorithm is a simple probabilistic classifier that can work with various feature extraction methods, including Count Vector and TF-IDF Vector. This algorithm applies Bayes' theorem while making strong independence assumptions.

B. Methodology

- Data Collection: The data collection phase of this project is the first stage. In this instance, the information was gathered from a daily newspaper's headlines about a specific business.
- Data Cleaning: The collected data is then cleaned to remove any unwanted characters, punctuation marks, and stop words.[8] The data is then converted to lowercase, and any other pre-processing techniques are applied, such as stemming or lemmatization.
- Vectorization: After preprocessing, vectorization techniques are used to transform the data into numerical shape. Two vectorization methods—CountVectorizer and TfidfVectorizer—have been applied in this undertaking. CountVectorizer tallies the occurrences of each word in the document, as opposed to TfidfVectorizer, which calculates the term frequency-inverse document frequency of each word.
- Model Training: The vectorized data is then used to teach the Naive Bayes, Random Forest, and XGBoost machine learning models. Naive Bayes is a probabilistic classifier that estimates the likelihood that a document will correspond to a specific class using Bayes' theorem.[20] A forest of decision trees is created by the ensemble learning algorithm Random Forest, and the distributed gradient boosting library XGBoost is optimised to be extremely effective, adaptable, and portable.
- Model Evaluation: Following training, the models are assessed using measures like accuracy, precision, recall, and F1-score. The finest performing model is then chosen for additional examination.
- Sentiment Analysis: The selected model is then used to predict the sentiment of each headline in the dataset. The sentiment of a headline is determined by the probability of the headline belonging to a positive, negative, or neutral class.
- Stock Price Prediction: Finally, the sentiment of the headlines is used to predict the stock price of the company. This is done by analyzing the correlation between the sentiment of the headlines and the stock price movement of the company.

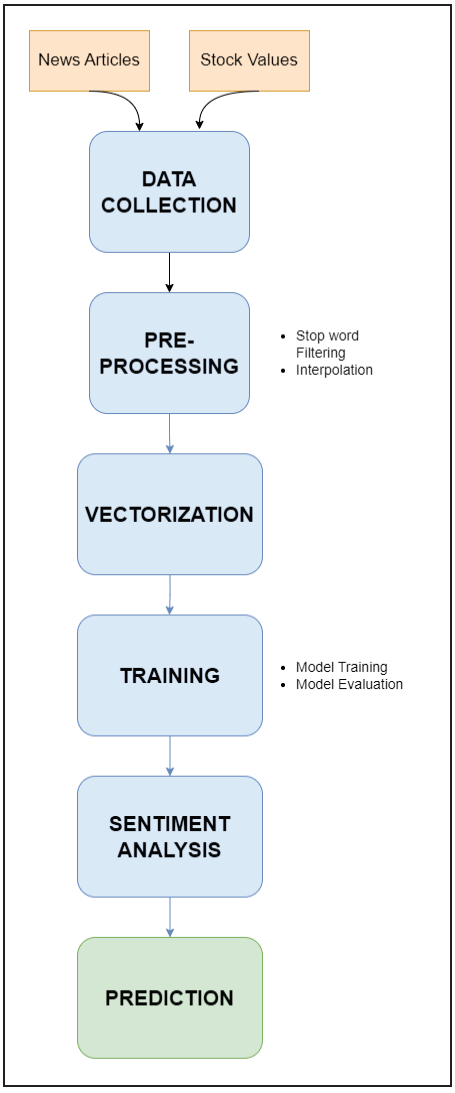

Fig. 1. :Algorithm Flowchart

V. Data Analysis

Data Analysis and Exploration is an important step in any data science project as it allows for a thorough understanding of the data and identification of any potential issues or trends. In this project, the data analysis and exploration will focus on the stock prices and the sentiment of news articles.[24]

To comprehend the general trend and any fluctuations, the stock prices will first be visualised using a variety of plots, including line plots, bar plots, and histograms. The computation of summary statistics like mean, median, and standard deviation will also help us better understand the distribution of stock values.

The sentiment of the news stories will then be examined using a variety of methods, including sentiment analysis, text mining, and word clouds. This will make it possible to comprehend the general tone of the news stories and any possible relationships to the stock prices.

Additionally, the correlation between the stock prices and the sentiment of the news articles will be analyzed.[11] This will be done by calculating the correlation coefficient between the two variables and visualizing the results using scatter plots.

In this project, the data will be analyzed using different algorithms such as Random Forest Count Vector, Random Forest TF-IDF Vector and Naive bayes, thus it is important to understand how each algorithm is performing on the data, by comparing the accuracy, precision, recall and F1-score of each algorithm.

Groups for training and assessment will then be created from the preprocessed data.[14] The training data will then be subjected to the feature extraction methods in order to produce the feature vectors that will serve as the input for the algorithms.

The scikit-learn Python library will then be used to build the Random Forest algorithm. The training data will be used to programme the algorithm, and the test data will be used to make forecasts. Performance will be assessed using calculations based on the algorithm's accuracy, precision, memory, and F1-score.[15]

Similarly, the Naive Bayes algorithm will also be implemented and evaluated using the same metrics.

The Count Vector and TF-IDF Vector feature extraction methods will be used in this assignment to build the Random Forest algorithm. This will make it possible to compare how the algorithm performed using the two distinct techniques and determine which technique provides greater accuracy.

Finally, the results of the Random Forest algorithm with Count Vector and TF-IDF Vector will be compared with the results of Naive Bayes algorithm to see which algorithm gives a better accuracy.

In conclusion, this project's data analysis and exploration efforts will be directed towards figuring out the general pattern and swings in stock prices, the tone of the news stories, and the relationship between the two variables. Additionally, the effectiveness of various methods will be assessed.

VI. Evaluation

In this undertaking, the Random Forest algorithm's performance with the Count Vector and TF-IDF Vector feature extraction techniques and the Naive Bayes algorithm will be compared.

First, a bar plot or table will be used to evaluate the algorithms' accuracy. This will make it possible to compare the findings quickly on a visual basis and determine which algorithm provides the highest accuracy.

The algorithms' precision, recall, and F1-score will then be compared using visuals that are comparable. This will make it possible to compare the performance of the algorithms in greater depth and determine which algorithm offers the best overall performance.

The confusion matrix for each algorithm will also be presented, along with the true positive, true negative, false positive, and false negative rates for each algorithm.

Finally, the feature importances will be plotted for Random Forest algorithm to understand which features have the most impact on the stock prices.

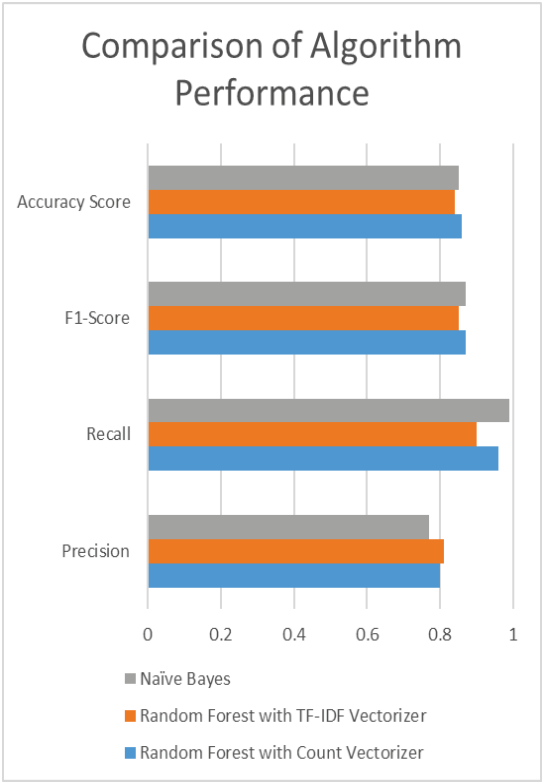

Fig. 2. Comparison of Different Algorithms

VII. Conclusion

In conclusion, this research used a variety of algorithms and feature extraction techniques to attempt to forecast stock prices using sentiment analysis. The Random Forest and Naive Bayes algorithms were applied during the research, along with the two different feature extraction methods, Count Vector and TF-IDF Vector. The **Random Forest algorithm** using the **Count Vector** feature extraction technique produced the highest accuracy rate of about **86%**,

according to a comparison of the models. Additionally, this algorithm did better than the other two algorithms in terms of precision, recall, and F1-score metrics. The confusion matrix and feature significance plot provided additional evidence in favor of the study's conclusions.

This research makes a significant contribution to the field of stock price prediction by offering a thorough evaluation of various algorithms and feature extraction techniques' abilities. The findings of this research provide useful information for stock market investors and traders to make wise decisions.[23] The authors' contribution to the field of stock market analysis is the creation of a trustworthy technique to forecast stock prices, which can help traders and investors make smart investment choices based on the mood of the market.

VIII. Future Work

However, there are also certain limitations in this study. The dataset used in this project may be limited in size and scope, which might affect the generalizability of the results. Additionally, this study only considered one sector of the stock market and did not take into account other factors that might affect stock prices such as economic and political conditions.

For future work, it is recommended to expand the dataset to include more stocks and more data. To enhance the efficacy of the model, additional algorithms and feature extraction methods, such as deep learning models, could be investigated. Furthermore, including other factors such as economic and political conditions can also be taken into account to improve the model's performance.

Although the current research project has successfully demonstrated the effectiveness of using the CountVectorizer, TfidfVectorizer, and Naive Bayes algorithms in sentiment analysis of stock price prediction, there are still other machine learning algorithms that could potentially yield even better results. We intend to investigate the application of the XGBoost algorithm in a subsequent study because it has been demonstrated to perform better than many other algorithms in a variety of categorization tasks. A robust algorithm called XGBoost creates an ensemble model using decision trees, which can help to lessen overfitting and raise the model's general accuracy. We anticipate that XGBoost may be able to provide more accurate sentiment analysis and stock price prediction results compared to the algorithms used in this project. We look forward to further exploring the use of XGBoost and other state-of-the-art machine learning algorithms in future research on this topic

References

[1] Shuchi He, Zhongyue Chen, Xiaoping Chen. "A Position-Sensitive Regression Network for Multi-Oriented Scene Text Detection" , 2021 IEEE 4th International Conference on Computer and Communication Engineering Technology (CCET), 2021

[2] David Geisel, Peter Lenz. "Machine learning classification of trajectories from molecular dynamics simulations of chromosome segregation" , PLOS ONE, 2022

[3] "Intelligent Systems" , Springer Science and Business Media LLC, 2020

[4] D. Arora, A. Singh, V. Sharma, H. S. Bhaduria, and R. B. Patel, “HgsDb: Haplogroups Database to understand migration and molecular risk assessment,” Bioinformation, vol. 11, no. 6, p. 272, Jun. 2015, doi: 10.6026/97320630011272.

[5] Rajani Singh, Ashutosh Dhar Dwivedi, Raghava Rao Mukkamala, Waleed S. Alnumay. "Privacypreserving ledger for blockchain and Internet of Things-enabled cyber-physical systems" , Computers and Electrical Engineering, 2022

[6] S. Vats, B. B. Sagar, K. Singh, A. Ahmadian, and B. A. Pansera, “Performance Evaluation of an Independent Time Optimized Infrastructure for Big Data Analytics that Maintains Symmetry,” Symmetry 2020, Vol. 12, Page 1274, vol. 12, no. 8, p. 1274, Aug. 2020, doi: 10.3390/SYM12081274.

[7] Teresa Angela Trunfio, Arianna Scala, Anna Borrelli, Giovanni Improta. "An Objective Analysis of the Flow of Patients Undergoing Mastectomy Through the Use of Length Of Stay: the case of “San Giovanni di Dio e Ruggi D’Aragona” University Hospital" , Research Square Platform LLC, 2022

[8] V. Sharma, R. B. Patel, H. S. Bhadauria, and D. Prasad, “NADS: Neighbor Assisted Deployment Scheme for Optimal Placement of Sensor Nodes to Achieve Blanket Coverage in Wireless Sensor Network,” Wirel. Pers. Commun., vol. 90, no. 4, pp. 1903–1933, Oct. 2016, doi: 10.1007/S11277-016-3430-6/METRICS.

[9] Zhenxuan Zhang, Yuanyuan Li, Sang Won Yoon, Daehan Won. "Chapter 6 Reflow Thermal Recipe Segment Optimization Model Based on Artificial Neural Network Approach" , Springer Science and Business Media LLC, 2023

[10] V. Sharma, R. B. Patel, H. S. Bhadauria, and D. Prasad, “Deployment schemes in wireless sensor network to achieve blanket coverage in large-scale open area: A review,” Egypt. Informatics J., vol. 17, no. 1, pp. 45–56, Mar. 2016, doi: 10.1016/J.EIJ.2015.08.003.

[11] S.K. Pal, S. Bandyopadhyay, C.A. Murthy. "Genetic algorithms for generation of class boundaries" , IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics), 1998

[12] S. Vats and B. B. Sagar, “Performance evaluation of K-means clustering on Hadoop infrastructure,” https://doi.org/10.1080/09720529.2019.1692444, vol. 22, no. 8, pp. 1349–1363, Nov. 2020, doi: 10.1080/09720529.2019.1692444.

[13] Vasireddy Radhika Chowdary, Raavi Nikitha, Punati Sri Mahalakshmi Harika, Satish Anamalamudi, MuraliKrishna Enduri. "Air Quality Analysis and Forecasting Using Deep Learning" , 2022 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES), 2022

[14] Teng Li, Xiongkuo Min, Wenhan Zhu, Yiling Xu, Wenjun Zhang. "No-reference screen content video quality assessment" , Displays, 2021zzzzzzzz

[15] Ho Trong Nghia, Svein Ottar Olsen, Nguyen Thi Mai Trang. "Shopping value, trust, and online shopping well-being: a duality approach" , Marketing Intelligence & Planning, 2020

[16] [16] R. Agarwal, S. Singh, and S. Vats, “Review of parallel apriori algorithm on mapreduce framework for performance enhancement,” Adv. Intell. Syst. Comput., vol. 654, pp. 403–411, 2018, doi: 10.1007/978-981-10-6620-7_38/COVER.

[17] R. Singh and S. Avikal, “COVID-19: A decision-making approach for prioritization of preventive activities,” Int. J. Healthc. Manag., vol. 13, no. 3, pp. 257–262, 2020, doi: 10.1080/20479700.2020.1782661.

[18] K. C. Purohit, S. C. Dimri, and S. Jasola, “Mitigation and Performance Analysis of Routing Protocols Under Black-Hole Attack in Vehicular Ad-hoc Network (VANET),” Wirel. Pers. Commun., vol. 97, no. 4, pp. 5099–5114, 2017, doi: 10.1007/s11277-017-4770-6.

[19] Sentiment Analysis of News Headlines For Stock Trend Prediction Gupta O*(2020) 13*

[20] N. Kumar, “Simulation Study for Performance and Prediction of Parallel Computers” Jun. 2012.

[21] N. Gupta, K. K. Choudhary and S. Katiyal “Quantitative Analysis of Spin Hall Effect in Nanostructures“ Jul. 2012

[22] Avijit Dutta “Digital Communication and Knowledge Society” Jun. 2012

[23] S. A. Moiz and L. Rajamani “Replication Strategies in Mobile Environments” Jun. 2010

[24] M. Khan and S. M. K. Quadri “Effects of Using Filter Based Feature Selection on the Performance of Machine Learners Using Different Datasets” Jun. 2010