SENTIMENT ANALYSIS OF ELECTRONIC PRODUCT TWEETS USING BIG DATA FRAMEWORK

Sunil Kumar, Vartika Koolwal and Krishna Kumar Mohbey

(Received: 8-Jan.-2019, Revised: 27-Feb.-2019, Accepted: 13-Mar.-2019)

ABSTRACT

Nowadays, social media has become more popular due to the advancement of Internet technologies and smartphone devices. Such platforms have generated interest among users to give their opinion. Social media-like Twitter- also plays an important role for business companies. Based on customer opinion about any product, business companies came to know more about customer choices. In the current scenario, millions of tweets are generated by people every year. But handling these huge unstructured tweets is not possible through the traditional platform. Therefore, big data framework, such as Hadoop and Spark, is used to handle such kind of large data.

In this paper, different sale tweets are used to analyze the sentiments of customers regarding electronic products. The experimental results of the proposed work will be useful for various business companies to take business decisions, which will further enhance the product sales.

KEYWORDS

Twitter, Spark, Big data, Flume, Sentiment analysis.

1. INTRODUCTION

Social media platforms, such as Twitter, Facebook and Instagram, have become vital constituents of daily life. People use these media to express their feelings, opinions, expressions, views and experiences about places or things [1]. Sentiment analysis is used to classify public opinion towards a particular topic or product. Various prominent categories of sentiment analysis, such as machine-learning [2], lexicon-based [3] and hybrid [4] categories, are worked upon. A progressive practice has grown to draw out the information from data available on social networks. This data has huge potential and can be harnessed for business-driven application [5], such as movie review [6], product advertisement, public election [9], brand endorsement and many more.

For real-time data analysis, Twitter is the rational choice due to a large amount of relevant data, compact and concise tweets up to 280 characters and simplicity to post an opinion. Real-time tweets are collected using hashtags (like #iphone, #OppoF9Pro). Opinion mining [7] approach was used to find polarity of tweets such as positive, negative and neutral. Knowing the collective sentimental affinity could help companies transform their strategies [5].

For many years, the problem of sentiment analysis has been studied and proposed solutions suffer from certain disadvantages. Constant problems with these approaches were centralized environment and time-consuming techniques, which scare many computational resources [8]. Furthermore, these standard approaches work on limited tweets and are not able to handle large size of tweets. Dubey et al. [9] proposed opinion-lexical approach in R platform to get insight about public opinion on political diplomats. However, the proposed approach works on a small dataset of approx. 3000 tweets. So, for enhancing the capability to handle a large number of tweets, we require distributed or parallel processing techniques, such as Spark.

Al-Saqqa et al. [10] collected 4 million Amazon customers' review dataset for large-scale sentiment analysis under Apache Spark framework. The dataset was tested for supervised machine-learning algorithm, where the model was trained using labeled training set. It applied classification techniques, where support vector outperforms Naïve Bayes and logistic regression, attaining an accuracy of 86%.

S. Kumar[1], V. Koolwal[2] and K. K. Mohbey[3] are with Department of Computer Science, Central University of Rajasthan, Ajmer, India. Emails: [1]sunil.cs@gmail.com, [2]vartikakoolwal14@gmail.com and [3]kmobhbey@gmail.com

In the age of Internet with such massive data, there is a need for faster computing and distributed storage, leading to a framework like Apache Spark, Apache Hadoop and Map Reduce techniques. Spark has emerged as the most popular big data processing engine. It improves over its predecessor, i.e., Hadoop MapReduce. MapReduce provides a simple model for writing programs that could execute in parallel in cluster. Spark improves MapReduce in three ways. Firstly, Spark engine can execute more general Directed Acyclic Graph (DAG) of operators than the rigid map-then-reduce format of MapReduce. Secondly, it has a rich set of transformation, which enables the output of one operation directly fed into another operation. Lastly, Spark extends with in-memory processing. Developers can instruct to cache any point in a processing pipeline, so future operations that need same data don't require to reload or recompute. It can be launched as a stand-alone or on cluster modes like Hadoop YARN, Apache Mesos and Kubernetes. It can integrate with distributed storage, such as HDFS, HBase and Cassandra. It is fast, much easy to use because of its high-level APIs in Java, Scala, Python and R. It has libraries, like MLlib for machine learning in Big data, GraphX for graph processing, Spark SQL and Spark Streaming [11].

In this paper, we do not propose any sentiment-prediction technique, but our aim is to analyze the eminent techniques regarding electronic products. We aim to perform sentiment analysis of data collected from Twitter using flume. These tweets are classified based on supervised learning approaches, such as Naive Bayes, SVM, Decision Tree, Random Forest and Logistic Regression classifier.

The remainder of the paper is arranged in the following manner. Section 2 represents related work. Section 3 is regarding big data processing using MapReduce, Spark and MLlib. Classification techniques are shown in section 4. In section 5, we present the sentiment analysis framework. Moreover, section 6 demonstrates the comprehensive experimental results. Conclusion and future work are presented in section 7.

2. RELATED WORK

Semantic analysis is the investigation of people's opinions, beliefs, attitudes and emotions towards an entity, such as products, services, events, issues and topics [1]. It is the field of machine learning which has gained the attention of researchers since the beginning of the century. Miller et al. [12] introduces WordNet, an online database for English language semantic processing using synonym sets (synsets) relationship. SentiWordNet [13] is an advancement of WordNet as a tool for knowledge-based word level processing *via* building a dictionary to find a score of each word.

Kim and Hovy [16] operated on a word granularity by using initially some seed words and using them to create a net; they proceeded further to sentence level by combining the strengths of the words, as they classify people's opinions. Moreover, Wilson et al. [17] operated on a phrase level, by running a supervised learning approach to determine the polarity or neutrality of phrases. Furthermore, document granularity [18] used word frequency and part of speech approach on Amazon reviews in categories, like books, DVDs, electronic and kitchen appliances to evaluate the response of people about the products.

Twitter streaming API[1] was used to gather data for product sentiment analysis [3]. The aim of using twitter data is to understand public opinion. Around 60,000 tweets were collected using Twitter API to analyze customer opinions on widely used smartphones in Korea [21]. Kumar et al. excavated opinions of the people about the quality of services provided by Airtel company [22]. For this purpose, they collected 80,000 tweets using the hashtag "#Airtel". They assessed them using Naïve Bayes approach with an accuracy of 80.9% on Mahout installed over Hadoop to classify them into different classes. They used term frequency and inverse document frequency for internal processing.

Various techniques, such as machine learning [2], entropy-based [24] and tree-kernel [25] techniques, are used for Twitter sentiment analysis. The hybrid algorithms presented in [26] for Twitter feed classification improve accuracy when compared with similar techniques. To increase accuracy, word sequence disambiguation [15] and negation handling [16] could be used. In [27], the authors mined tweets with emoticons and punctuations. They concluded that Naïve Bayes performance and accuracy

[1] Twitter Apps. Available online: http://www.tweepy.org/

are higher than those of SVM. Emoticons and hashtags [28] are employed as sentiment labels to carry out KNN classification of diverse sentiment types. Kaur et al. [28] have used Spark for processing large data. They have also used Bloom filter for inspecting element membership in any proposed set and space compaction.

Agarwal et al. [25] used unigram model to classify Twitter data into 4 classes: positive, negative, neutral and junk, where junk included tweets not understood by a human annotator. They investigated on tree kernel and feature-based models and reported that these models outperform the unigram baseline. They highlighted that for feature analysis, prominent features were a combination of the prior polarity of words and their parts-of-speech-tags. However, they used manually annotated Twitter data for the test.

Kaptein [29] studied what influence the tweets have on the reputation of the company. They explored the sentimental-bearing text (i.e. subjective text) for factual information to derive reputational polarity. For example, *Nokia Smartphone blasted while charging* has a negative reputation for Nokia Company. They suggested that developing a polarity lexicon for the specific domain will be cost-beneficial.

In [10], the authors retrieved 4 million tweets, which required bulk processing speed and distributed storage, signifying the need for Big Data frameworks, like Hadoop and Spark. These frameworks are required to meet up the shooting data generation demand. Many researchers are using similar frameworks for tweet analysis [30]. Baltas et al. [31] has used Twitter data with Spark platform. In the proposed approach, they have used binary and ternary classification. The result of F-measure of feature vector of logistic regression indicated 62.8% positive, 59.2% negative and 54.2% neutral. Chan and Thein [32] used sentiment analysis on 60k real-time tweets using Apache Flume on iphone mobile product. The results show that linear SVM performs better than NB by 10 % and better than logistic regression by 2%.

Earlier studies have shown that the traditional approach is suitable for limited data only. But, if we have a large amount of real-time tweets, we can’t process them with normal architecture and traditional approaches. Therefore, it is high time to develop a framework with distributed processing to improve accuracy and performance of the models. So, in this paper, we are working with Spark framework and have used Flume for fast data retrieval. We have demonstrated the results of semantic analyzers and their machine learning validation is shown in tabular formats and graphs to render a complete picture about accuracy gained. We have not formulated any semantic prediction technique, but have analyzed SVM, NB, logistic regression, decision tree and random forest techniques on unstructured real-time electronic product tweets using Big Data framework. We have attained the average accuracy of 91% in logistic regression that is outperforming all the competing techniques.

3. BIG DATA PROCESSING

Big data deals with large datasets which require complex processing and need huge storage. Big data frameworks are listed below.

3.1 Hadoop

Hadoop software library is an open source implementation of the MapReduce framework. It enables distributed and parallel processing of large datasets. It also provides distributed storage on cluster of computers [33]. Hadoop core contains MapReduce and Hadoop Distributed File System (HDFS). HDFS is responsible for storing large datasets on the cluster, which are partitioned into blocks and distributed into nodes.

3.2 MapReduce

MapReduce model allows distributed processing across multiple nodes in a cluster. It contains a map and a reduce function procedure, called mapper and reducer, respectively [34]. Input data is partitioned into the mapper phase and transferred to workers to execute the map function; each worker output is in key-value pairs after processing the data. Shuffle phase sorts the output and groups it by key. Reducer calls for every unique key and gets a set of values associated with key. MapReduce framework deals with the underlying parallelization, adjustment to internal failure, information distribution between nodes and load adjustment. Data is replicated and distributed across nodes to improve both accessibility and reliability.

3.3 Spark Framework

Apache Spark[2] is a fast and general framework for large-scale data processing. It is the improvement of Hadoop framework. Hadoop is ideal for large batch processing when we require to go through all data. However, its performance drops quickly for certain scenarios, e.g. when we have to deal with graph-based or iterative algorithms. Hadoop does not cache intermediate results but instead, it flushes the data to the dish in between each step. In contrast, Spark has a Directed Acyclic Graph (DAG) execution engine that allows cyclic data flow and in-memory computing. So, it can execute programs up to 100x times faster than Hadoop. It contains a set of libraries which combines streaming, SQL, graph processing and machine learning in a single engine. It provides many high-level APIs in Python, Scala, java and R and can run on Hadoop or standalone while using different data sources, such as, HDFS, Cassandra or HBase. It provides a programming model that hides the partitioning of dataset in cluster, using a new data structure called Resilient Distributed Dataset (RDD) [35]. RDD is an immutable distributed collection of records partitioned into different nodes of the cluster. Data-sharing abstraction property of RDD allows to run a wide range of APIs provided by Spark: MLlib, Spark streaming, Spark SQL and GraphX (graph processing). By default, RDDs are short-lived, so if they are used in an action, they need to be recomputed. However, they can persist in memory for frequent reuse.

3.4 MLlib

MLlib is Spark’s largest distributed learning library. It includes fast, scalable and easy implementation of common learning algorithms of machine learning, including classification, regression, clustering and collaborative filtering [36]. The library also has low-level primitives for convex optimization, statistical analysis tools, distributed linear algebra and feature extraction and provides various I/O formats, such as LIBSVM format, Spark SQL data integration[3] and MLlib’s internal format. It shows excellent performance and scalability to handle larger problems.

4. Classification Techniques

This section describes sentiment analysis phases. The complete process of sentiment analysis is shown in Figure 1. The following supervised classification approaches are used to predict the polarity of a tweet.

4.1 Naïve Bayes

Naïve Bayes is an easy probabilistic classifier, which uses Bayes Theorem with an assumption of high (naïve) independence between features. It had proven effective in many application domains, like system performance management [37], text classification, medical diagnosis and many more. It assigns the most favourable class to a given instance according to its feature vector which is given by:

$$P\,(CL \mid X) = \frac{P(CL) * P\,(X \mid CL\,)}{P(X)} \tag{1}$$

where, X= (x_1, x_2, …, $x_{n)}$, indicating some independent feature vectors.
CL : L possible outcomes (classes).
X : Tweet needing to be classified.
P (CL | X): Posterior probability.
P(CL) and P(X) : Prior probabilities.

4.2 Support Vector Machine

Support Vector Machine carries out classification by searching for the hyperplane (boundary dividing

[2] Spark https://spark.apache.org
[3] Spark SQL https://spark.apache.org/sql/

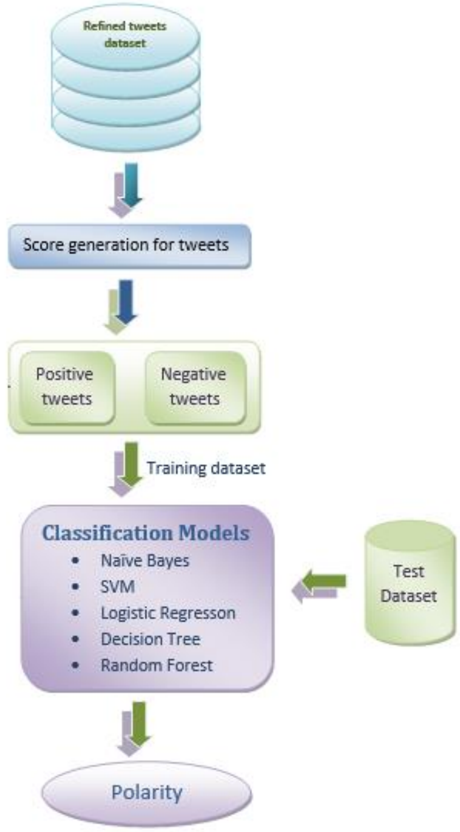


Figure 1. Sentiment analysis of tweet dataset.

one entity set from another) that maximizes the margin between two classes. Hyperplanes are explored using "important training tuples" (support vectors) along with margins [38]. SVM can be implemented on both linear and non-linear datasets. SVM as a supervised learning classifier is popular due to its high reliability, varied application usage and less vulnerability to overfitted model [39].

We traverse linearly separable class using two-class problems. We are given a dataset S as (P1, Q1), (P2, Q2),………..(P|S|, Q|S|), where Qj is the class label whose value is from +1 to -1(Qj ∈ (-1, +1)). Qj is associated with Pj set of training tuples.

Any hyperplane can be defined as P set of points satisfying

$$W.P - B = 0 \tag{2}$$

where, W is normal vector to the hyperplane. $\frac{B}{||W||}$ is the offset of the plane from the origin and normal vector W.

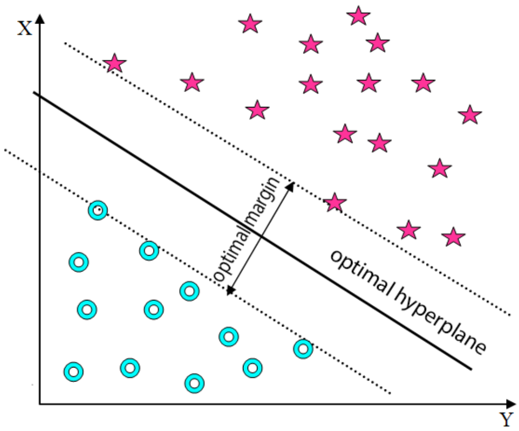


Figure 2. Support vector machine classifier.

We can plot multiple separating lines. We have to find the “best” line (least classification error), in general, best “hyperplane” by the maximum distance of the hyperplane to the closest negative instance and positive instance. Figure 2 shows SVM optimal hyperplane in training with sample tweets to classify positive tweets (star-shaped) and negative tweets (disk-shaped).

4.3 Decision Tree

Decision Tree is a flow-chart like structure, where each non-leaf node signifies test condition on the attribute; branches indicate the result of test and leaf node represents class label of entity set. First and topmost node is root node [25]. Tree is explored from top to bottom indicating classification rules. It is a decision support tool which is used to display the outcome of test condition, resource cost, utility along with an algorithm that contains a statement of conditional control.

Decision tree can be converted into decision rules by association rules with target variable on right-hand side. A decision tree can be used in temporal or causal relations [40]. Figure 3 shows decision tree classification processing based on test condition.

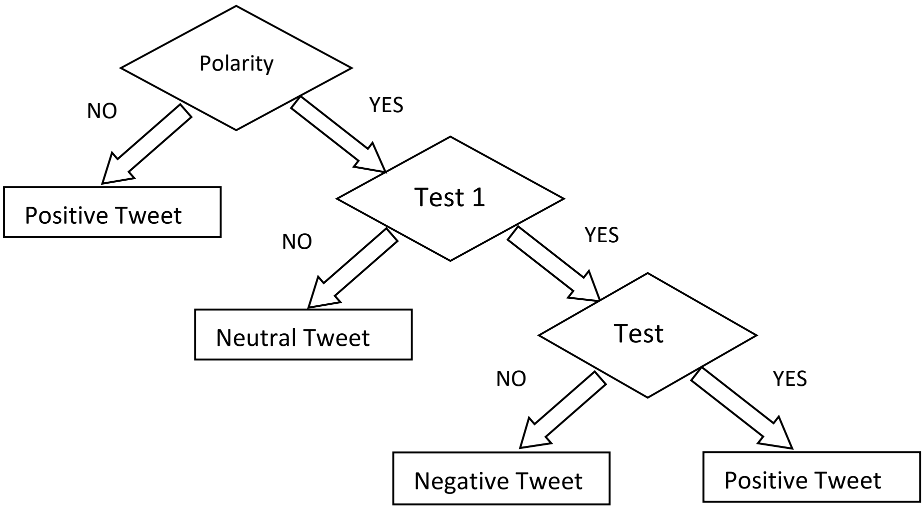


Figure 3. Decision tree classifier.

4.4 Random Forest

Random forest classifier is a tree classifier which is generated using independently selected random vector from input dataset. Each tree for most favourable class casts one vote to classify input vector [41]. It uses one or more combinations of features at every node to expand a tree. Bagging is a method to make training set *via* randomly drawing N replacement examples (N is the size of original training set used for feature selection). Every input instance can be classified by exploring most desirable voted class by all forest trees. We can use GINI index as a measure of attribute selection, which weights attribute impurity of all classes. For a given training dataset D, choosing one cast and ascertaining that it belongs to a class C_i, could be written as:

$$\sum\sum_{j\neq i}\left(\frac{f(c_i,D)}{|D|}\right)\left(\frac{f(c_j,D)}{|D|}\right) \tag{3}$$

where, $\frac{f(c_i,D)}{|D|}$ is the probability of that labelled class belongs to class C_i.

4.5 Logistic Regression

Logistic regression is a predictive classifier that is used to a model-dependent variable using logistic function. Dependent variable is a categorical value having two categories labelled as “0” and “1” like (loose or win, sick or not sick, true or false, tea or coffee). Independent variable is numerical or categorical value. It is used to classify observations, in terms of whether an observation belongs to a particular category or not (positive tweet or negative tweet in our problem).

Types of Logistic Regression:

- Binary Logistic Regression: models binary outcome (yes/no).
- Ordinal Logistic Regression: models an ordered response (completely disagree, disagree, somewhat agree, agree).
- Nominal Logistic Regression: models a multilevel outcome which is insensitive to ordering (choice of a transport mode such as bus, car, train).

Logit (log-odds) is a function which is equivalent to log odds of variables. If p is a probability of occurrence of an event (E= 1), then $\frac{p}{1-p}$ represents the corresponding odds. Logit (E) is given by:

$$logit(E) = log\left(\frac{p}{1-p}\right) \tag{4}$$

A logistic curve is obtained by a logistic function. Logistic curve is just like a sigmoid curve the input of which is as any real value k (k € R), while the output value falls between (0, 1). Logistic curve is shown in Figure 4. Logistic function (k) is given by:

$$p(k) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 k)}} \tag{5}$$

where, p (k) is the probability of dependent variable.
β_0 : intercept from the linear regression equation.
$\beta_1 k$: Regression coefficient multiplied by some predictor value.
e : Base e indicates the exponential function.

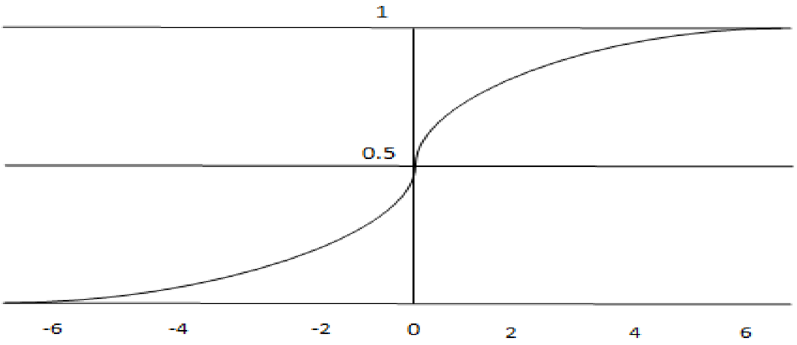


Figure 4. Logistic regression.

5. Sentiment Analysis Framework

We present a framework for sentiment analysis which includes data collection, pre-processing, sentiment score calculation for tweets, classification and polarity prediction.

5.1 Data Collection Using Twitter API by Flume

Twitter is a corpus of 500 million published tweets by 321 million active monthly users[4]. This real-time data provides immense opportunities to study social trends. Crawling data from Twitter was collected using Flume. Flume links Flume agent with web servers. This is done with API keys extracted from Twitter developer's account. Twitter delivers Rest API and Streaming API to different client systems to absorb tweets. Figure 5 shows the process of data retrieval using Flume agent. Tweets are collected from source to channel and then from channel to HDFS sink. Different hashtags are used to collect live-stream data from Twitter. Description of used hashtags and collected tweets is shown in Table 1.

[4] Statista 2019, February 2019, Number of monthly active Twitter users worldwide from 1st quarter 2010 to 4th quarter 2018 (in millions). Available: https://www.statista.com/statistics/282087/number-of-monthly-active-twitter-users/

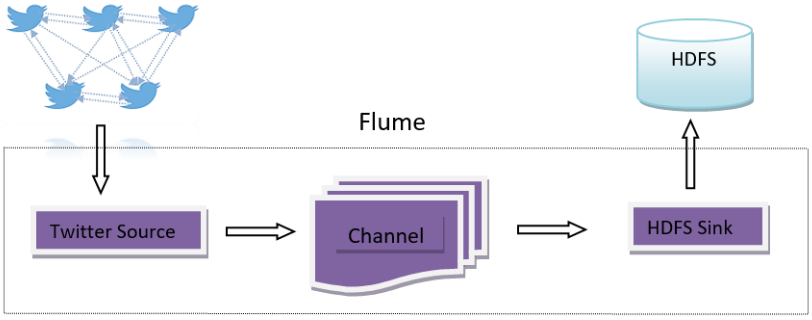
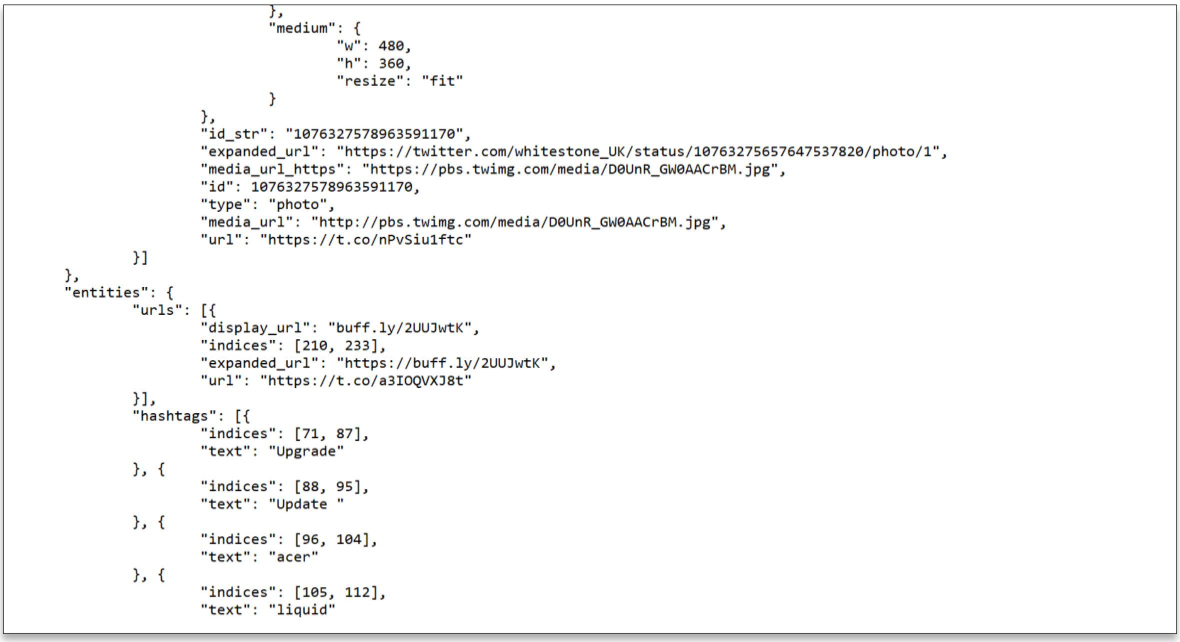


Figure 5. Twitter data collection.

Table 1. Hashtag description.

CATEGORY	HASHTAGS/KEYWORDS	# TWEETS
TWEETS FOR MOBILE PHONES	#Samsung # vivo #iPhone #htc #OppoF9Pro #Samsung # GooglePixel3XL #iPhone #htc #MiNote4 #motoG	1,00,000
TWEETS FOR LAPTOPS	#MacBookPro # iMac #HpEliteBook #ThinkPadLenovo #MSIGaming #chromebook # DellXPS #HPEnvy #AcerSwitch	70,000
TWEETS FOR TELEVISION	#SonyBraviaKLV # AndriodTv #SamsungQLED #TCL #LGLED #PanasonicSmartTv # VizioLcd #rokuTv #OLEDTV	50,000

Data extracted from Twitter using Twitter API comes in JSON format. Figure 6 is a snapshot of raw tweets in JSON format. However, JSON structure is not understood by user completely. Therefore, JSON Validator was used to validate data into a particular structure. Figure 7 shows the refined structural tweets after processing raw tweets in JSON format.

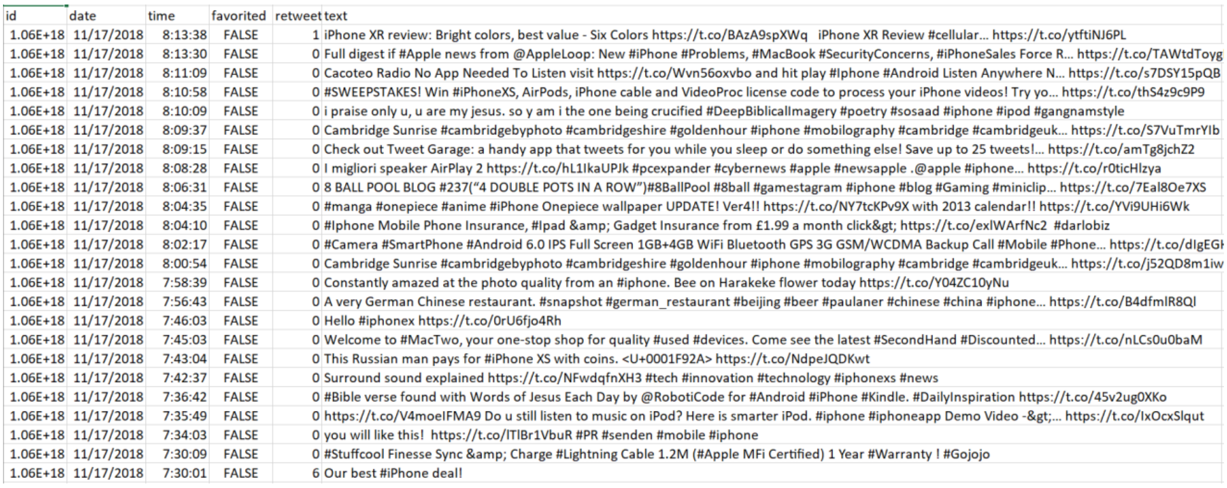


```
                                },
                                "medium": {
                                        "w": 480,
                                        "h": 360,
                                        "resize": "fit"
                                }
                        },
                        "id_str": "1076327578963591170",
                        "expanded_url": "https://twitter.com/whitestone_UK/status/1076327565764753782O/photo/1",
                        "media_url_https": "https://pbs.twimg.com/media/D0UnR_GW0AACrBM.jpg",
                        "id": 1076327578963591170,
                        "type": "photo",
                        "media_url": "http://pbs.twimg.com/media/D0UnR_GW0AACrBM.jpg",
                        "url": "https://t.co/nPvSiu1ftc"
                }]
        },
        "entities": {
                "urls": [{
                        "display_url": "buff.ly/2UUJwtK",
                        "indices": [210, 233],
                        "expanded_url": "https://buff.ly/2UUJwtK",
                        "url": "https://t.co/a3IOQVXJ8t"
                }],
                "hashtags": [{
                        "indices": [71, 87],
                        "text": "Upgrade"
                }, {
                        "indices": [88, 95],
                        "text": "Update "
                }, {
                        "indices": [96, 104],
                        "text": "acer"
                }, {
                        "indices": [105, 112],
                        "text": "liquid"
```

Figure 6. Sample of raw tweets in JSON format collected from Twitter.

5.2 Pre-processing of Tweets

One of the major tasks of semantic analysis is data filtering. It helps improve the efficiency of the classifier. Following are the pre-processing steps:



id	date	time	favorited	retweet	text
1.06E+18	11/17/2018	8:13:38	FALSE	1	iPhone XR review: Bright colors, best value - Six Colors https://t.co/BAzA9spXWq iPhone XR Review #cellular… https://t.co/ytftiNJ6PL
1.06E+18	11/17/2018	8:13:30	FALSE	0	Full digest if #Apple news from @AppleLoop: New #iPhone #Problems, #MacBook #SecurityConcerns, #iPhoneSales Force R… https://t.co/TAWtdToyg
1.06E+18	11/17/2018	8:11:09	FALSE	0	Cacoteo Radio No App Needed To Listen visit https://t.co/Wvn56oxvbo and hit play #Iphone #Android Listen Anywhere N… https://t.co/s7DSY15pQB
1.06E+18	11/17/2018	8:10:58	FALSE	0	#SWEEPSTAKES! Win #iPhoneXS, AirPods, iPhone cable and VideoProc license code to process your iPhone videos! Try yo… https://t.co/thS4z9c9P9
1.06E+18	11/17/2018	8:10:09	FALSE	0	i praise only u, u are my jesus. so y am i the one being crucified #DeepBiblicalImagery #poetry #sosaad #iphone #ipod #gangnamstyle
1.06E+18	11/17/2018	8:09:37	FALSE	0	Cambridge Sunrise #cambridgebyphoto #cambridgeshire #goldenhour #iphone #mobilography #cambridge #cambridgeuk… https://t.co/S7VuTmrYIb
1.06E+18	11/17/2018	8:09:15	FALSE	0	Check out Tweet Garage: a handy app that tweets for you while you sleep or do something else! Save up to 25 tweets!… https://t.co/amTg8jchZ2
1.06E+18	11/17/2018	8:08:28	FALSE	0	I migliori speaker AirPlay 2 https://t.co/hL1IkaUPJk #pcexpander #cybernews #apple #newsapple .@apple #iphone… https://t.co/r0ticHlzya
1.06E+18	11/17/2018	8:06:31	FALSE	0	8 BALL POOL BLOG #237(“4 DOUBLE POTS IN A ROW”)#8BallPool #8ball #gamestagram #iphone #blog #Gaming #miniclip… https://t.co/7Eal8Oe7XS
1.06E+18	11/17/2018	8:04:35	FALSE	0	#manga #onepiece #anime #iPhone Onepiece wallpaper UPDATE! Ver4!! https://t.co/NY7tcKPv9X with 2013 calendar!! https://t.co/YVi9UHi6Wk
1.06E+18	11/17/2018	8:04:10	FALSE	0	#Iphone Mobile Phone Insurance, #Ipad &amp; Gadget Insurance from £1.99 a month click&gt; https://t.co/exlWArfNc2 #darlobiz
1.06E+18	11/17/2018	8:02:17	FALSE	0	#Camera #SmartPhone #Android 6.0 IPS Full Screen 1GB+4GB WiFi Bluetooth GPS 3G GSM/WCDMA Backup Call #Mobile #Phone… https://t.co/dIgEGI
1.06E+18	11/17/2018	8:00:54	FALSE	0	Cambridge Sunrise #cambridgebyphoto #cambridgeshire #goldenhour #iphone #mobilography #cambridge #cambridgeuk… https://t.co/j52QD8m1iw
1.06E+18	11/17/2018	7:58:39	FALSE	0	Constantly amazed at the photo quality from an #iphone. Bee on Harakeke flower today https://t.co/Y04ZC10yNu
1.06E+18	11/17/2018	7:56:43	FALSE	0	A very German Chinese restaurant. #snapshot #german_restaurant #beijing #beer #paulaner #chinese #china #iphone… https://t.co/B4dfmlR8Ql
1.06E+18	11/17/2018	7:46:03	FALSE	0	Hello #iphonex https://t.co/0rU6fjo4Rh
1.06E+18	11/17/2018	7:45:03	FALSE	0	Welcome to #MacTwo, your one-stop shop for quality #used #devices. Come see the latest #SecondHand #Discounted… https://t.co/nLCs0u0baM
1.06E+18	11/17/2018	7:43:04	FALSE	0	This Russian man pays for #iPhone XS with coins. <U+0001F92A> https://t.co/NdpeJQDKwt
1.06E+18	11/17/2018	7:42:37	FALSE	0	Surround sound explained https://t.co/NFwdqfnXH3 #tech #innovation #technology #iphonexs #news
1.06E+18	11/17/2018	7:36:42	FALSE	0	#Bible verse found with Words of Jesus Each Day by @RobotiCode for #Android #iPhone #Kindle. #DailyInspiration https://t.co/45v2ug0XKo
1.06E+18	11/17/2018	7:35:49	FALSE	0	https://t.co/V4moeIFMA9 Do u still listen to music on iPod? Here is smarter iPod. #iphone #iphoneapp Demo Video -&gt;… https://t.co/IxOcxSlqut
1.06E+18	11/17/2018	7:34:03	FALSE	0	you will like this! https://t.co/lTlBr1VbuR #PR #senden #mobile #iphone
1.06E+18	11/17/2018	7:30:09	FALSE	0	#Stuffcool Finesse Sync &amp; Charge #Lightning Cable 1.2M (#Apple MFi Certified) 1 Year #Warranty ! #Gojojo
1.06E+18	11/17/2018	7:30:01	FALSE	6	Our best #iPhone deal!

Figure 7. Sample of tweets in structured format.

- Filtering – we eliminate useless parts of tweets, such as URL links, Twitter usernames, punctuations, hashtags, Twitter special words (such as “RT”), special characters and symbols.
- Stop words removal –some words, such as pronouns (he, she, it), articles (a, an, the), don’t give any information for classification. Moreover, having these bags of words can lead to less accurate prediction. So, it’s better to eliminate these stop words [43].
- Stemming – it is a process of conversion of words in different forms into their single root word like “amuse”, “amused”, “amusement” and “amusing” have same root: “amus”. Result of stemming is less intuitive to humans, but more comparable across observations. Stemming decreases entropy and increases relevance of root words like “amus” [43].

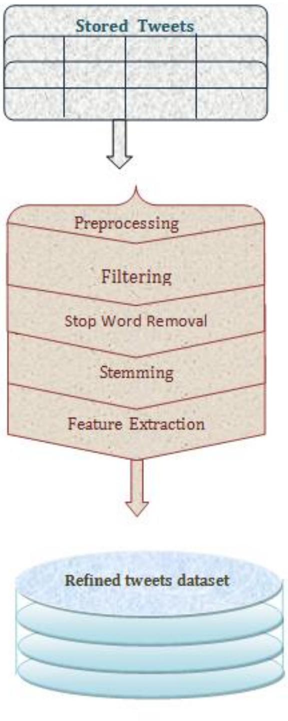
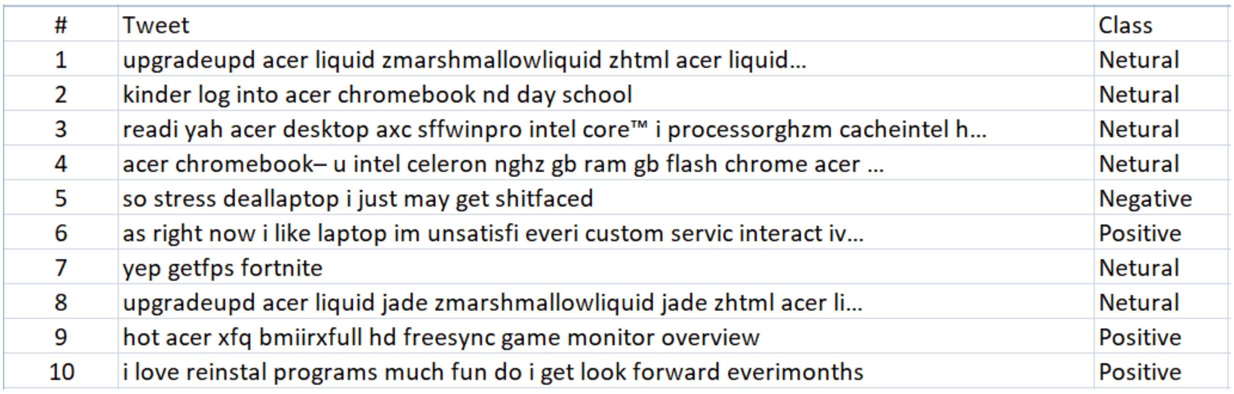


Figure 8. Pre-processing of Tweets.

- Feature extraction - Tokenization is a process of segmenting text by spaces and punctuation marks into tokens to form bags of words. Feature transformation function, like StringIndexer, OneHotEncoder and VectorIndexer, is used to transform categorical terms into vectors. TF-IDF is used to generate feature vectors from tweets. In TF-IDF, we compute TF (term frequency), which is the occurrence frequency of a term in that document and IDF (inverse document frequency) measuring how infrequent a word is present across all the document. TF-IDF shows relevancy of a word into a specific document. Spark MLlib library has HashingTF and IDF algorithms to calculate TF-IDF [44]. Figure 8 shows the execution of pre-processing steps. After completion of data filtering steps, we get refined tweets with their labels. A sample of tweets with their polarity is shown in Figure 9.



#	Tweet	Class
1	upgradeupd acer liquid zmarshmallowliquid zhtml acer liquid...	Netural
2	kinder log into acer chromebook nd day school	Netural
3	readi yah acer desktop axc sffwinpro intel core™ i processorghzm cacheintel h...	Netural
4	acer chromebook– u intel celeron nghz gb ram gb flash chrome acer ...	Netural
5	so stress deallaptop i just may get shitfaced	Negative
6	as right now i like laptop im unsatisfi everi custom servic interact iv...	Positive
7	yep getfps fortnite	Netural
8	upgradeupd acer liquid jade zmarshmallowliquid jade zhtml acer li...	Netural
9	hot acer xfq bmiirxfull hd freesync game monitor overview	Positive
10	i love reinstal programs much fun do i get look forward everimonths	Positive

Figure 9. Sample of tweets with labels.

5.3 Tweet Score Calculation

This approach uses a standard list of positive and negative words to detect the polarity of a tweet. Based on availability of positive or negative words within tweets, a sentiment score is generated. Polarity of a tweet, such as p(t) can be represented as {-1,0,1} referring to a negative, neutral and positive tweet, respectively [45].

A score of a tweet S(t) can be calculated as:

$$S(t) = \sum_{i \in t}^{n} p(i) \tag{7}$$

where, p(i) is the polarity of term i in tweet. Polarity of a tweet can be determined as follows:

1, if St > 0 (positive)
P(t) = -1, if St < 0 (negative)
0, otherwise (neutral)

After score calculation for each tweet, we have training datasets with their polarities, such as positive, negative and neutral.

5.4 Model Implementation

ML is a dataframe package API, introduced in Spark 2.0. From start, spark framework has MLlib as an RDD-based API. To carry out the implementation in Spark, we need to follow some steps.

Firstly, import data into DataFrames. these are a distributed collection of data organized into named columns, which makes Spark programming easier and simpler to develop.

Secondly, transforms, such as Tokenizer (), StopWordRemover (), HashingTF (), Tf-Idf, are used. Transformer is an algorithm which can change one dataframe to another.

Thirdly, estimators are used to implement method fit(), which accept dataframe and make a model, such as logistic regression, Naïve Bayes, random forest, linear SVM and decision tree.

```
val Estimator = new LinearSVC()
val Estimator = new NaiveBayes().setLabelCol("label").setFeaturesCol("features")
val Estimator = new LogisticRegression()
```

Lastly, to combine ML algorithms into a single pipeline, we use Spark ML standardize APIs. Pipeline chains multiple transformers and estimators together in order to specify an ML workflow.

```
    val pipeline = new Pipeline().setStages(Array(labelIndexer, tokenizer, remover, hashingTF, idf, Estimator))
    val model = pipeline.fit(training)
```

In this classification step, to train the model, 70% of the dataset is randomly selected for training and 30% for testing.

```
  val predictions = model.transform(test)
```

6. Results and Discussion

This section describes the details of experiments conducted on the Spark framework.

6.1 Environment Description

We conducted experimental tests on Spark framework using a single node configuration. To achieve the desired performance, we have operated on Intel quad-core 3.0 GHz processor with a RAM of 8 GB and a storage capacity of 1 TB on Ubuntu 18.0.1 operating system. We configured Spark version 2.3.0, Scala version 2.11.6, Hadoop version 2.8.4, Flume 1.7.0, Hive 2.1.1 and Java-8.

We have used three different types of dataset related to electronic products; i.e., mobile phones, laptops and televisions, corresponding to 100 K, 70K and 50K tweets.

6.2 Polarity of Datasets

In this section, we have a pictorial representation of polarity in relation with phone, laptop and television tweets. Figures 10, 11 and 12 show the polarity of datasets indicating the ratio of positive, neutral and negative tweets, respectively.

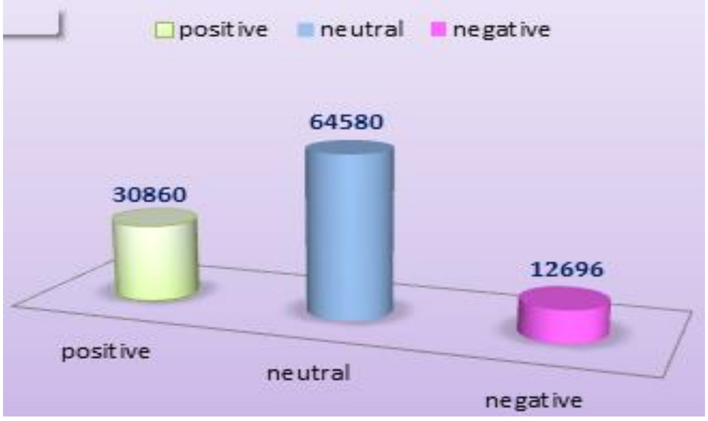


Figure 10. Polarity of phone dataset.

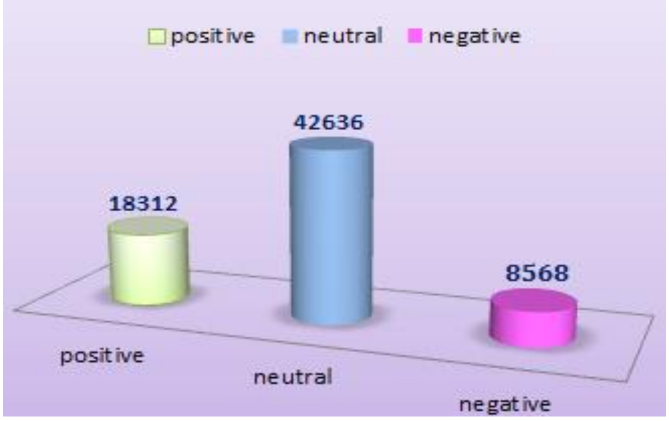


Figure 11. Polarity of Laptop Datasets.

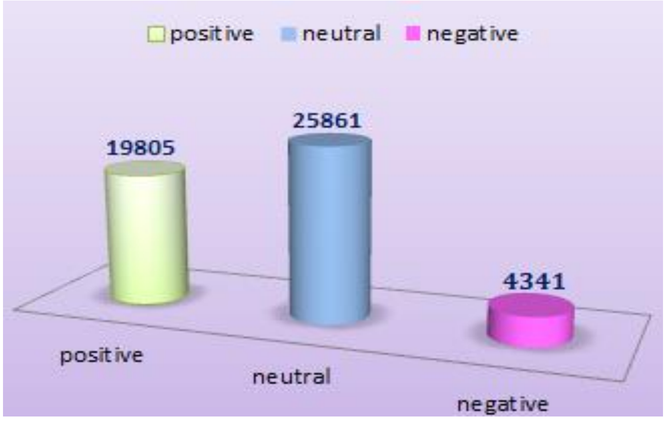


Figure 12. Polarity of television dataset.

6.3 Performance Evaluation

Before the model can be used to classify new data, evaluation of model on test dataset is important. To measure the effectiveness or quality of models, different metrics are being used.

The simplest model of evaluation metric is precision. It measures the exactness of the model. It calculates what fraction of positive classified data is actually positive. Recall is another simple measurement. It measures the completeness of the model. It calculates what percentage of positive data is classified as positive. Accuracy measures what fraction of data is accurately classified. F-measure and AUC are commonly used metrics for model evaluation. F-measure is the weighted harmonic mean of precision and recall. It is the trade-off between precision and recall, whose score lies between 0 and 1. F-measure with score 1 states the best model whereas 0 states the worst model.

AUC (area under ROC): It is a binary classifier generally evaluated using AUC evaluation metric. It measures the aggregate performance with every classification parameter. It plots true positive rate and false positive rate at random positive or negative observations. Table 3 shows the confusion matrix, which is a specific table layout that allows visualization of the effectiveness of a model.

Table 3. Confusion matrix.

		Pridicted Class	
		positive	negative
Actual Class	positive	TP	FP
	negative	FN	TN

- TP : True Positive
- FP : False Positive
- TN : True Negative
- FN : False Negative

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (7)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (8)$$

$$\text{Accuracy} = \frac{\text{TP+FN}}{\text{TP+FP+FP+TN}} \quad (9)$$

$$\text{F1} = \frac{\text{2.Precision.Recall}}{\text{Precision + Recall}} \quad (10)$$

Furthermore, the performance of different machine learning classification approaches is shown in Table 4, Table 5 and Table 6, respectively.

Table 4. Performance comparison (phone dataset).

Classification Approach	Accuracy	Recall	F1-measure	Precision
Naïve Bayes	0.82277	0.82277	0.82639	0.83386
SVM	0.85200	0.85200	0.84441	0.85011
Logistic Regression	**0.86358**	**0.86358**	**0.86006**	**0.86054**
Decision Tree	0.74882	0.74882	0.67685	0.79132
Random Forest	0.73647	0.73647	0.64792	0.79835

Table 5. Performance comparison (television dataset).

Classification Approach	Accuracy	Recall	F1-measure	Precision
Naïve Bayes	0.81973	0.81973	0.83528	0.87702
SVM	0.89777	0.89771	0.89098	0.89178
Logistic Regression	**0.91084**	**0.91084**	**0.90813**	**0.90724**
Decision Tree	0.81713	0.81713	0.73632	0.75132
Random Forest	0.81713	0.81713	0.73632	0.75132

Table 6. Performance comparison (laptop dataset).

Classification Approach	Accuracy	Recall	F1-measure	Precision
Naïve Bayes	0.81027	0.81027	0.81552	0.83448
SVM	0.86609	0.86609	0.86328	0.86434
Logistic Regression	**0.91084**	**0.91084**	**0.90813**	**0.90724**
Decision Tree	0.70493	0.70493	0.60479	0.76341
Random Forest	0.68892	0.68892	0.57204	0.77147

6.4 Comparison of Different Machine Learning Approaches

In this subsection, we have performed a series of tests using different machine learning classification approaches under the big data framework on our dataset. This comparison is carried out under different parameters. Figures 13 and 14 show the comparison of varied approaches in relation to training and prediction time on different datasets.

Figure 13 shows that for training the model, Naïve Bayes classifier takes less time related to all three categories. Similarly, to prepare the model, random forest classifier takes more time. It also informs that there is a direct relation between tweet size and training time; i.e., as tweet size increases, training time also increases.

Prediction time comparison using all approaches is shown in Figure 14. We can further conclude that logistic regression takes more prediction time in all three cases, while all the remaining approaches take approximately the same prediction time. Figure 15 shows accuracy comparison of all the approaches. This figure illustrates that logistic regression performs better for larger data sizes with an accuracy of 86% in the phone, 91% in the laptop and 91% in the television classes.

Another comparison measure is AUC (Area under the curve). The comparative result set value is shown in Table 7. It determines which approach best predicts the classes. Based on this view, Figure 16 shows that both SVM and logistic regression classification approaches perform good, compared to the other approaches.

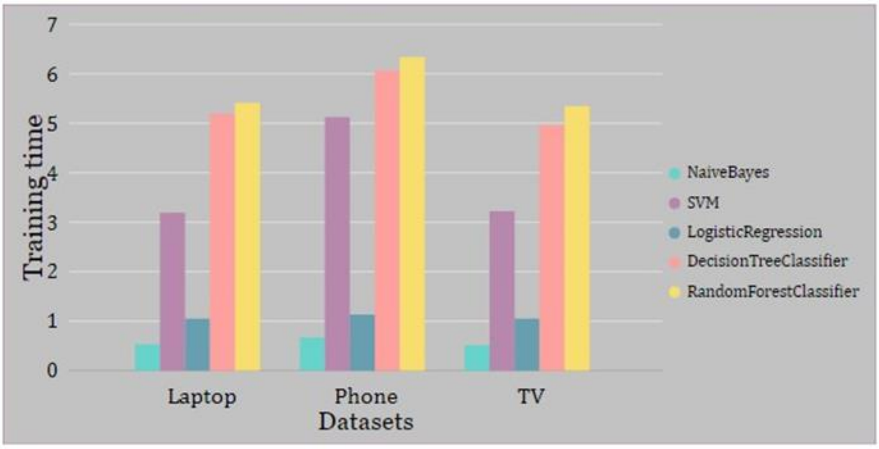


Figure 13. Training time comparison.

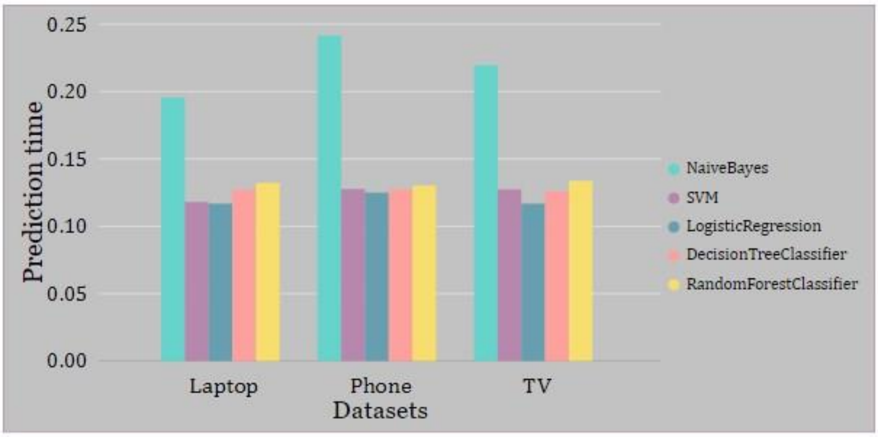


Figure 14. Prediction time comparison.

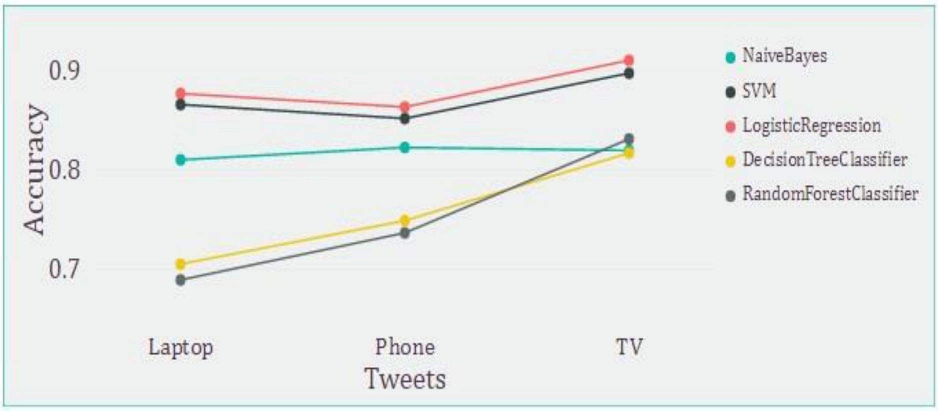


Figure 15. Accuracy comparison.

7. Conclusion and Future Work

In this paper, we analyze sentiments of different electronic product tweets. For this, real-time tweets are collected from the Twitter platform using different hashtags. Additionally, Flume was used to consume real-time tweets in big data framework. After pre-processing of collected tweets, sentimental

Table 7. AUC results.

Classification Approach	Laptop	TV	Phone
Naïve Bayes	0.5028828	0.3904879	0.5012804
SVM	0.9277823	0.9281304	0.9100904
Logistic Regression	**0.9357323**	**0.9475399**	**0.9200218**
Decision Tree Classifier	0.6800121	0.7552956	0.5545057
Random Forest Classifier	0.8290897	0.8136763	0.8095101

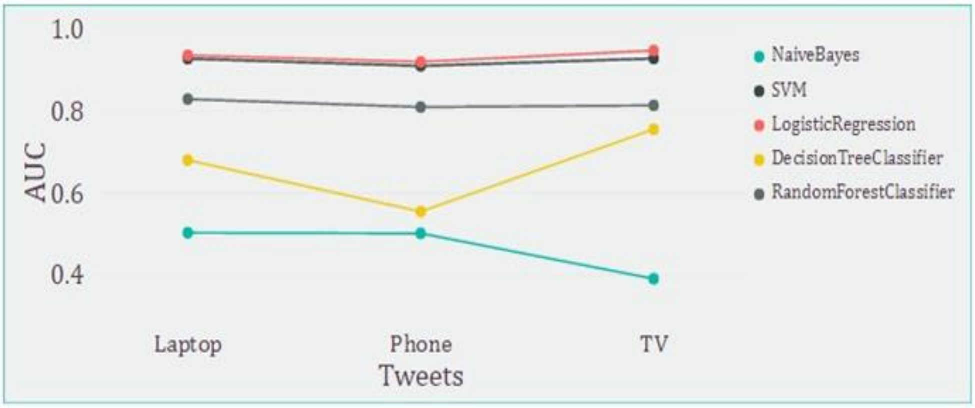


Figure 16. AUC comparison.

analysis has been performed by different supervised classification approaches. The experimental results show that the logistic regression approach has higher accuracy for all used datasets. Sentimental analysis comparison was carried out on the basis of Accuracy, F-measure and AUC.

Due to enhancement and popularity of social media platforms, such comparative results are more useful for business companies. They can easily help identify people's sentiment towards any specific electronic product or item. Based on sentiments, various decisions can be made.

In our future work, we intend to work on multiclass approaches to identify the exact polarity of tweets instead of positive, negative and neutral. In addition, we will work to enhance the accuracy of the approaches under big data technologies.

REFERENCES

[1] B. Liu, "Sentiment Analysis and Opinion Mining," Synthesis Lectures on Human Language Technologies, vol. 5, no. 1, pp. 1–167, 2012.

[2] A. Hasan, S. Moin, A. Karim and S. Shamshirband, "Machine Learning-based Sentiment Analysis for Twitter Accounts," Mathematical and Computational Applications, vol. 23, no. 1, p. 11, 2018.

[3] C. S. Khoo and S. B. Johnkhan, "Lexicon-based Sentiment Analysis: Comparative Evaluation of Six Sentiment Lexicons," Journal of Information Science, vol. 44, no. 4, pp. 491–511, 2017.

[4] F. Iqbal, J. Maqbool, B. C. M. Fung, R. Batool, A. M. Khattak, S. Aleem and P. C. K. Hung, "A Hybrid Framework for Sentiment Analysis Using Genetic Algorithm-based Feature Reduction," IEEE Access, pp. 1–1, 2019.

[5] F. Atefeh and D. Inkpen, Proceedings of the Workshop on Semantic Analysis in Social Media, Association for Computational Linguistics, France, 2012.

[6] A. Tyagi and S. Naresh, "Sentiments Analysis of Twitter Data Using K-Nearest Neighbour Classifier," International Journal of Engineering Science, vol. 17258, 2018.

[7] T. White, Hadoop: The Definitive Guide, 3rd Edition, O'Reilly Media, Inc., May 2012.

[8] M. V. Banerveld, N.-A. Le-Khac and M.-T. Kechadi, "Performance Evaluation of a Natural Language Processing Approach Applied in White Collar Crime Investigation," Future Data and Security Engineering Lecture Notes in Computer Science, pp. 29–43, 2014.

[9] G. Dubey, S. Chawla and K. Kaur, "Social Media Opinion Analysis for Indian Political Diplomats," Proc. of the IEEE 7th International Conference on Cloud Computing, Data Science and Engineering-Confluence, pp. 681-686, 2017.

[10] S. Al-Saqqa, G. Al-Naymat and A. Awajan, "A Large-Scale Sentiment Data Classification for Online Reviews Under Apache Spark," Procedia Computer Science, vol. 141, pp. 183–189, 2018.

[11] R. Sandy, U. Laserson, S. Owen and J. Wills, Advanced Analytics with Spark: Patterns for Learning from Data at Scale, O'Reilly Media, Inc., 2017.

[12] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross and K. J. Miller, "Introduction to WordNet: An On-line Lexical Database," International Journal of Lexicography, vol. 3, no. 4, pp. 235–244, 1990.

[13] A. Esuli and F. Sebastiani, "SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining," [Online], Available: http://nmis.isti.cnr.it/sebastiani/Publications/LREC06.pdf, 2006.

[14] S. Seifollahi and M. Shajari, "Word Sense Disambiguation Application in Sentiment Analysis of News Headlines: An Applied Approach to FOREX Market Prediction," Journal of Intelligent Information Systems, vol. 52, no. 1, pp. 57–83, 2018.

[15] M. Bhuiyan, A. Misra, S. Tripathy, J. Mahmud and R. Akkiraju, "Don't Get Lost in Negation: An Effective Negation Handled Dialogue Acts Prediction Algorithm for Twitter Customer Service Conversations," arXiv preprint arXiv:1807.06107, 2018.

[16] S. -M. Kim and E. Hovy, "Determining the Sentiment of Opinions," Proceedings of the 20th International Conference on Computational Linguistics (COLING 04), [Online], Available: http://aclweb.org/anthology/C04-1200, 2004.

[17] T. Wilson, J. Wiebe and P. Hoffmann, "Recognizing Contextual Polarity in Phrase-level Sentiment Analysis," Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT 05), [Online], Available: https://people.cs.pitt.edu/~wiebe/pubs/papers/emnlp05polarity.pdf, 2005.

[18] J. Blitzez, M. Dredze and F. Pereira, "Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification," Proc. of the 45th Annual Meeting of the Association of Computational Linguistics, pp. 440–447, 2007.

[21] J. Kim, M. Yang, Y. Hwang, S. Jeon, K. Kim, I. Jung, C. Choi, W. Cho and J. Na, "Customer Preference Analysis Based on SNS Data," Proc. of the IEEE 2nd International Conference on Cloud and Green Computing, pp. 609-613, 2012.

[22] M. Kumar and A. Bala, "Analyzing Twitter Sentiments through Big Data," Proc. of the IEEE 3rd International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, pp. 2628-2631, 2016.

[24] A. L. Berger, V. J. D. Pietra and S. A. D. Pietra, "A Maximum Entropy Approach to Natural Language Processing," Computational Linguist, vol. 22, no. 1, pp. 39–71, 1996.

[25] A. Agarwal, B. Xie, I. Vovsha, O. Rambow and R. Passonneau. "Sentiment Analysis of Twitter Data," Proceedings of the Workshop on Languages in Social Media, pp. 30-38, 2011.

[26] F. H. Khan, S. Bashir and U. Qamar, "TOM: Twitter Opinion Mining Framework Using Hybrid Classification Scheme," Decision Support Systems, vol. 57, pp. 245–257, 2014.

[27] S. Geetha and K. V. Kumar, "Tweet Analysis Based on Distinct Opinion of Social Media Users," Advances in Intelligent Systems and Computing Advances in Big Data and Cloud Computing, pp. 251–261, 2018.

[28] A. Kaur, D. Khaneja, K. Vyas and R. S. Saini, Sentiment Analysis on Twitter Using Apache Spark, [Online], Available: https://www.researchgate.net/profile/Deepesh_Khaneja/publication/320625064 _project_report_sentiment_analysis_on_twitter_using_apache_spark/links/59f24420aca272cdc7d0169a /project-report-sentiment-analysis-on-twitter-using-apache-spark.pdf, 2016.

[29] R. Kaptein, "Learning to Analyze Relevancy and Polarity of Tweets," CLEF (Online Working Notes/Labs/Workshop), [Online], Available: http://ceur-ws.org/Vol-1178/CLEF2012wn-RepLab-Kaptein2012.pdf, 2012.

[30] A. Kanavos, N. Nodarakis, S. Sioutas, A. Tsakalidis, D. Tsolis and G. Tzimas, "Large Scale Implementations for Twitter Sentiment Classification," Algorithms, vol. 10, no. 1, p. 33, 2017.

[31] A. Baltas, A. Kanavos and A. K. Tsakalidis, "An Apache Spark Implementation for Sentiment Analysis

on Twitter Data," Algorithmic Aspects of Cloud Computing Lecture Notes in Computer Science, pp. 15–25, 2017.

[32] W. N. Chan and T. Thein, "A Comparative Study of Machine Learning Techniques for Real-time Multi-tier Sentiment Analysis," Proc. of the IEEE 1st International Conference on Knowledge, Innovation and Invention (ICKII), 2018.

[33] N. Deshai, S. Venkataramana and G. P. S. Varma, "Performance and Cost Evolution of Dynamic Increase Hadoop Workloads of Various Data Centers," Smart Intelligent Computing and Applications Smart Innovation, Systems and Technologies, pp. 505–516, 2018.

[34] J. Dean and S. Ghemawat, "MapReduce," Communications of the ACM, vol. 51, no. 1, p. 107, 2008.

[35] M. Zaharia et al., "Resilient Distributed Datasets: A Fault-tolerant Abstraction for In-memory Cluster Computing, " Proc. of the 9th USENIX Conference on Networked Systems Design and Implementation, pp. 2-2, 2012.

[36] X. Meng, J. Bradley, B. Yavuz, E. Sparks, S. Venkataraman, D. Liu and D. Xin, "MLlib: Machine Learning in Apache Spark," The Journal of Machine Learning Research, vol. 17, no. 1, pp. 1235-1241, 2016.

[37] J. Hellerstein, J. Thathachar and I. Rish, "Recognizing End-user Transactions in Performance Management," Proc. AAAI-2000, pp. 596–602, 2000.

[38] J. Han, M. Kamber and J. Pei, "Data Mining: Concepts and Techniques," Elsevier, pp. 279–325, 2012.

[39] V. Vapnik, Estimation of Dependencies Based on Empirical Data, ISBN 978-0-387-34239-9, Springer, 1995.

[40] K. Karimi and J. H. Howard, "Generation and Interpretation of Temporal Decision Rules," arXiv preprint arXiv:1004.3334, 2010.

[41] L. Breiman, "Random Forests," UC Berkeley TR567, 1999.

[43] G. Angiani, L. Ferrari, T. Fontanini, P. Fornacciari, E. Iotti, F. Magliani and S. Manicardi, "A Comparison between Pre-processing Techniques for Sentiment Analysis in Twitter," In: KDWeb, 2016.

[44] H. Karau, A. Konwinski, P. Wendell and M. Zaharia, Learning Spark: Lightning-fast Big Data Analysis, O'Reilly Media, Inc., Jan 2015.

[45] A. Giachanou, J. Gonzalo, I. Mele and F. Crestani, "Sentiment Propagation for Predicting Reputation Polarity," Lecture Notes in Computer Science Advances in Information Retrieval, pp. 226–238, 2017.

ملخص البحث:

في وقتنا الحاضر، أصبحت وسائل التواصل الاجتماعي شائعةً جداً في ظل التقدم الذي طرأ على تقنيات الانترنت وأجهزة الهاتف الذكية. وقد أحدثت منصات التواصل الاجتماعي اهتماماً لافتاً بين المستخدمين لإبداء آرائهم. كذلك تلعب وسائل التواصل الاجتماعي، مثل تويتر، دوراً مهماً لشركات الأعمال. وبناءً على آراء الزبائن حول منتج ما، تصبح الشركات على اطلاع على اختيارات الزبائن. وفي الوقت الراهن، تتولد ملايين التغريدات من قبل الناس كل سنة. إلا أن التعامل مع هذا الكم الهائل من التغريدات غير المهيكلة ليس ممكناً عبر المنصة التقليدية. لذا، فإن أحد الأُطر الخاصة بمعالجة البيانات الضخمة – مثل هادوب وسبارك – يستخدم من أجل التعامل مع هذا النوع من البيانات الضخمة.

في هذه الورقة، يتم استخدام تغريدات تتعلق بالمبيعات لتحليل آراء الزبائن بشأن المنتجات الالكترونية. وتجدر الإشارة الى أن النتائج التجريبية للعمل المقترح ستكون ذات فائدة للعديد من شركات الأعمال بحيث تساعدها في اتخاذ القرارات المتعلقة بأعمالها، الأمر الذي من شأنه أن يعمل على تحسين مبيعاتها من المنتجات.