CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY, ISLAMABAD

Propaganda Detection Model for News Articles Using Machine Learning Approaches

by

Muhammad Imran Tahir

A thesis submitted in partial fulfillment for the degree of Master of Science

in the

Faculty of Computing
Department of Computer Science

2021

I dedicate my dissertation work to my father and teachers. A special feeling of gratitude is for my supervisor and brother for their love, endless support and encouragement

CERTIFICATE OF APPROVAL

Propaganda Detection Model for News Articles Using Machine Learning Approaches

by

Muhammad Imran Tahir

(MCS181002)

THESIS EXAMINING COMMITTEE

S. No.	Examiner	Name	Organization
(a)	External Examiner		
(b)	Internal Examiner		
(c)	Supervisor	Dr. Muhammad Shahid Iqbal	CUST, Islamabad

Dr. Muhammad Shahid Iqbal
Thesis Supervisor
June, 2021

Dr. Nayyer Masood
Head
Dept. of Computer Science
June, 2021

Dr. Muhammad Abdul Qadir
Dean
Faculty of Computing
June, 2021

Author's Declaration

I, **Muhammad Imran Tahir** hereby state that my MS thesis titled "**Propaganda Detection Model for News Articles Using Machine Learning Approaches**" is my own work and has not been submitted previously by me for taking any degree from Capital University of Science and Technology, Islamabad or anywhere else in the country/abroad.

At any time if my statement is found to be incorrect even after my graduation, the University has the right to cancel my MS Degree.

Muhammad Imran Tahir
(MCS181002)

Plagiarism Undertaking

I solemnly declare that research work presented in this thesis titled " **Propaganda Detection Model for News Articles Using Machine Learning Approaches** " is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been dully acknowledged and that complete thesis has been written by me.

I understand the zero tolerance policy of the HEC and Capital University of Science and Technology towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MS Degree, the University reserves the right to withdraw/revoke my MS degree and that HEC and the University have the right to publish my name on the HEC/University website on which names of students are placed who submitted plagiarized work.

Muhammad Imran Tahir
(MCS181002)

Acknowledgements

In the name of Allah, the Most Merciful Alhamdulillah, all praises to Owner of the universe for the strengths and the blessing in completing this thesis. This study is nothing, but an effort to understand and articulate the principles of one of the several hundred thousand phenomena, with a tool, the brain, a precious gift from Almighty. I would like to express my sincerest appreciation to my enthusiastic supervisor, **Dr. Muhammad Shahid Iqbal** for his supervision, assistance, and immense knowledge. I am sincerely thankful to him for his constant support, motivation, and patience. His invaluable help of constructive comments and suggestions throughout the thesis work have contributed to the success of this research. It has been an amazing experience and I thank him wholeheartedly, not only for his tremendous support. My deepest gratitude goes to my beloved parents for tolerating my mood swings and being patient with me. I would also like to thank my younger brother, friends and family for encouraging me and motivating me for completion of this project.

Muhammad Imran Tahir
(MCS181002)

Abstract

Now a days, online platforms are popular sources for the propagation of news articles. This excessive amount of articles is increasing tremendously, because in every day millions of news articles are published on either news websites or posted on social media e.g. Facebook, Twitter. For a lay reader as well as for journalists, it is hard to detect manually that published article is either propaganda or non-propaganda, because propaganda is a communication technique which influence the general public perspective regarding any specific agenda by presenting selective information. Due to rapid evolution of propaganda, AI and NLP researchers are curious to propose such a methodology which helps the media community to detect the propaganda content automatically. In literature, variety of methodologies have been proposed for binary (BLC) or multi-label classification (MLC) by using semantic or linguistic features and their combinations. However, linguistic features performance is comparatively better than all remaining features, but hybrid set of features produce extraordinary results for propaganda content detection.

In this experimental analysis, we proposed a binary propaganda detection model consist upon stand alone as well as hybrid features which are extracted by wrapper method. These features are evaluated by ML model Random Forest with respect to four evaluation metrics precision, accuracy, recall, f-measure and Area Under Curve (AUC). Different linguistic factors e.g. representations, writing style and readability level against the certain number of words and characters, assist to detect an article as propagandistic or non-propagandistic. We examined a labeled news articles dataset which contains '0' for non-propaganda and '1' for propaganda class. This examination is based upon two types of different features set: (i) each stand-alone feature (ii) the combination of the two features, and deduced that character Tri-grams outperforms with precision 94.10 %, recall 72.0 %, f-measure 81.6 % , accuracy 96.40 % and AUC 85.70 % as a stand-alone as well as the combination of selected features, character Tri-grams and POS perform robust with precision 92.6 %, recall 91.3 %, f-measure 91.5 %, accuracy 98.10 %, and AUC 95.10 % than the existing alternatives for propaganda identification (Char n-gram, Word n-gram). Unlike previous work, our dataset is comprises on real-time based

news articles, which assures that proposed features set work absolutely for in-domain and out-of-domain news articles. It makes able the end users to inspect quickly about different aspects of the same story, and it also helps the mass media community to draw out further that in how many ways a media platform use such propagandistic stories to fulfil their agenda.

Contents

List of Figures

List of Tables

Abbreviations

ML	Machine Learning
RF	Random Forest
SVM	Support Vector Machine
MLP	Multi Perceptron
AUC	Area Under Curve
SVD	Single Value Decomposition
LIWC	Linguistic Inquiry and Word Count
LDA	Latent Dirichlet Allocation

Chapter 1

Introduction

From the couple of decades there is an immiscible development is noticed in field of artificial intelligence, big data analysis, and natural language processing, which prove them self a double-edged sword in the field of computer science. If we talk about one edge, the applications like automatically text summarization [1] (in which shortening a document and produced a condensed version without losing any information and content of the original document), chat bots [2] (are machine based agents which behave like a natural language user interfaces for data as well as for service providers), and automated journalism [3] (is a technique which generates news content automatically and assists in both ways, informs about useful working practices of journalists and how journalism is need to operate for better outcomes), are assisting the humans. On the other edge, such technologies having also negative impacts on society because these technologies are much effective tools for the generation and dissemination of misinformation. Modern societies are facing several critical challenges which are based upon misinformation and its amplification through different social media platforms. For instance, the notable topics such as democracy [4], journalism [5], health [6], economy [7], and climate change [8] are facing great threatening impacts due to fake news and various propaganda techniques. In general to describe the propaganda, the deliberately designed opinion or action by individuals or groups which place influential impact to the opinions or actions of other individuals or groups concerning predetermined

Table 1.1: Propaganda Spreading Techniques

Sr. #	Technique	Description
1	Card Stacking	Card Checking technique over/under emphasizes the facts by omitting or falsifying the truth.
2	Name Calling	Name Calling is labelling of individual or group with names, i.e., fascist, or terrorist on their beliefs, nations, races.
3	Glittering Generalities	Glittering Generalities use abstract words such as patriotism, freedom, or rights to appeal the emotions of audience.
4	Transfer	Transfer propaganda highlights the positive or negative qualities of one individual, group of companies to make second more acceptable.
5	Testimonial	Testimonial use quotes from celebrities to make an argument more strengthen.
6	Plain Folks	Speaker wins the confidence of audience by appearing as a common person.
7	Band wagon	Band wagon method group the targeted audience on a common point and take the action that everyone else is taking.

ends [9]. The term propaganda demonstrates very frequently with lies, distortion, and deceit [10] but any biased content which is published either intentional or unintentional is propaganda [11] . In literature, there are seven different techniques are discussed which are commonly used to spread propagandistic content [12]. For example, name calling labels the individuals or group with bad names and card stacking method falsifies the facts to overemphasize the agenda. Description of each propaganda spreading techniques is described in Table 1.1.

The ramifications of propaganda in the United States of America (USA) elections is a prime example of its impact on societies [13]. The propaganda disseminated by Cambridge Analytica (CA) [14] and Internet Research Agency (IRA) [15] through Facebook shaped the political attitude of citizens to manipulate election results. Similarly, online propaganda has affected the foreign policies of European countries [16]. The conspiracy theories linking 5G technology to coronavirus (COVID-19) pandemic have led to violent riots [17], [18]. This phenomenon is not confined by any specific language. Accordingly, propaganda in regional languages is disseminated by extremist groups to sway the local population towards violent crimes [19],

[20]. Such extremism and anti-state based propaganda content by these extremists has also alarming signals for national security [21] of any targeted country. Polio which is a very critical disease among different countries also effected by propaganda news. A more than 100 local Urdu newspapers published a rumor regarding polio vaccine without any authenticity, in resulting of this rumor a huge increase in polio cases was recorded in the vicinity of propaganda affected areas [22].

The evolution in the field of artificial intelligence make the computer scientists able to design and develop such automated tools and techniques which help for the detection of propaganda content. Lot of researchers proposed state-of-the-art classification algorithms for the detection of propaganda content automatically from any online platform. The neural architectures with Bidirectional Encoder Representations from Transformers (BERT) embedding's, sentences identification from propaganda news content by fine-grained analysis implemented by [23]. In another effort, Proppy [29] which is our baseline use different stylometric and representation features includes, readability NELA and LIWC along with textual features for the identification of propagandistic content from online news platforms.

Furthermore, to dig up a best solution for fake text analysis researchers counter with two main hurdles i) Fetch a suitable dataset and ii) Select a machine learning model for text analysis. To counter the first problem, we select a pre fetched dataset Qprop and use dev part of it for our research. Dev part consist upon 5135 English language news articles which is fetched from 104 different online news plat forms and labelled according to the biasness of the published content by Media Bias/Fact Check (MBFC) manually. To counter the second problem, we perform our experiments with following machine learning models:

1. Naïve-Bayes
2. Random Forest
3. Decision Tree

And use precision, recall, f-measure, AUC and accuracy as evaluation metrics.

1.1 What is Classification

Classification is a well knowned technique which is used to sort and organize a huge data in different types, form or classes which are already defined. Every data scientist must be familiar with classification technique, because directly or indirectly they implement classification in their research, so it is a very well renowned technique in the area of machine learning. The classification task is implemented in various Machine Learning problems, most commonly it is used in text classification, speech recognition etc. From previous literature review we observed that the text classification is used as baseline for news content detection, and classify that content either hoax, satire, trusted or propaganda [27]. Classification of News articles fetched from different online sources is very helpful to recognize an articles either fake (propaganda) or true (non-propaganda). Online news articles classification technique is mainly composed after extracting some most influential features from the news content which could help us to categorize that article either in fake (propaganda) or true (non-propaganda) class. Mainly, there are two types of classification. 1) Binary-label Classification (it involves only two class labels, one is normal class label and second is abnormal class label, news article must belongs to any one of defined binary classes either normal or abnormal), 2) Multi-label classification (unlike the binary classification, it involves more than two class labels, news article must belongs to any one of defined multi classes). In this research we use a binary classification model with two classes one is propaganda (yes) and second is non-propaganda (no). Every article is examined on the bases of its content rather than author, publishing platform or meta data.

1.2 Background

Propaganda inhabited in our lives as a key technique for destruction of democracies of societies. In this era of social media there are wide range of news platforms are available: which seemingly present either neutral articles or clearly biased. During the reading of such news article, every reader should be familiar that, at least to some extent, it inevitably reflects the bias of both the writer of the article and the news platform where the news article is published. However, which one is clearly biased, it is hard to depict. It could be that either the author itself may not be intentionally bias about any specific topic or it may be that it is part of author's agenda to capture the reader's thinking regarding any specific topic. Further discussion represents the definition of propaganda. [24] Perform a classical work and elaborate a very comprehensive definition for propaganda as follows:

Definition 1. Propaganda is expression of opinion or action by individuals or groups deliberately designed to influence opinions or actions of other individuals or groups with reference to predetermined ends.

Because of complex nature of this phenomenon, in different experimental setups propaganda appears in different dimensions, in psychology, sociology, history, and political science, each discipline illustrates the word propaganda by its own perspective. As a common definition by [25], all these disciplines elaborate the propaganda as following:

Definition 2. Propaganda is an organized attempt to influence a group of people, small or large.

Propaganda has very influential effect when reader overview the article with negligence. That is, if any person reads a broadcast writing text, in a formal or an informal news article platform (e.g., in a blog/news forum, social media) than it is difficult to identify that the reading content is propagandistic or not. In such case, the reader is revealed to the propagandistic content without any background knowledge and some of his opinions might change when it concludes the whole article. According to [26], 2016 US Presidential elections are very prominent example of the use of propaganda, in the result of all this every reader deduced the same observation which was depicted in propaganda. Given the numerous perspectives of news platforms which publish news articles in form of tabloids, broadsheets, printed and digital manners. It's obvious that both the news consumers and vendors might benefit from such automatic techniques which can detect propagandistic articles which published from different news platforms.

In this research, we perform a binary classification task. For the investigation of

our experimental results we use two classes, one is propaganda (yes) and other is non-propaganda (no). Our first traditional experimental setup in which each standalone feature are tested on Dev portion of proppy dataset [27], but later own we proposed such hybrid features which are extracted by using wrapper method. Hybrid features proved themselves as a best determinant comparatively standalone features. We examined these standalone as well as hybrid feature results by using different ML models including Naïve-Bayes, Random Forest, SVM, MLP, XGBoost and Decision Tree with 10-fold cross validation in python. In order to direct comparison with previous work [28] and our generated results of all ML models, we choose Random forest as binary classifier and precision, recall, f1-measue and AUC as evaluation matrices. Our main aim to propose such hybrid features among (POS, Word2Vec, LIWC, LSA, Word Uni Gram and Word Tri Gram) which significantly improve the propaganda news article detection techniques.
In previous works [28] introduced n-gram technique but author admitting by itself that n-gram shows decline in performance when out of domain articles are used as input. Similarly our baseline [29] share hybrid features as their significant determinate for propaganda detection model.

1.3 Problem Statement

Identify cation of propagandistic and non-propagandistic contents from news platform. Prior approaches utilized basic type of textual/content features to identify the propaganda from news texts. However, there is a need to utilize some contextual and influential set of significant features to improve the prediction accuracy.Second, there is a need to apply a more robust machine learning model for classification task to produce more promising results.

1.4 Purpose

There is a vast landscape for news article publisher, at very first that it is a bottleneck for news and article lover to identify that which article illustrating facts and which one is not. Secondly literature review acknowledged that previous researches were working on word n-gram and char n-gram. Under the consideration of this time wasteful activity the purpose of our study is quite clear. The purpose of this proposed study is describe as follow:

1. To provide an organized platform for news article readers, through which they can read propagandistic ad non-propagandistic content without any extra struggle.
2. To implement different machine learning models (SVM, Deep Neural Networks, Logistic Regression, Random Forest.) For screening of propagandistic and non-propagandistic content from news articles.

1.5 Scope

The dominant aspect of this study is to design and develop an extraction technique which make easier for journalist and news lover community to distinguish between propaganda and non-propaganda articles regarding their computed propaganda score. The coverage of this study covers the identification of propaganda and non-propaganda article on a dataset called Qprop. Which is already used in development of proppy web based portal for propaganda content identification. Our technique will be trained only on selected dataset and produce results according to that dataset of articles.

1.6 Research Objective

Objective of this experimental research is to assist the journalist community as well as general news readers to distinguish between propaganda vs. non-propaganda news content automatically. It is helpful for numerous online systems: information retrieval, author publisher ranking, recommender systems and search engines. It assist both authors publishers to evaluate their content regarding propaganda. It help the publisher to categorize the articles in two different categories (propaganda and no-propaganda) as well as readers to read content of desired category.

1.7 Research Question

On the bases of problems identified in the introduction section, this research provide a road map for both researchers as well as journalist community and general public to identify the propagandistic content before establishing a biased opinion about any targeted event or news. After under consideration of all defined scenario following research questions have been formulated in this thesis.

1.7.1 Research Question 01:

What is the impact of proposed feature for the detection of propaganda and non-propaganda news articles?

1.7.2 Research Question 02:

Which Machine learning model provide robust performance when applied with proposed feature set?

1.7.3 Research Question 03:

Is there exist an influential set of features which out performs?

1.8 Application of Proposed Approach

Proposed approach could be much beneficial for real time applications. Some applications are listed below:

1.8.1 Author Ranking Systems:

Such a news content analysis techniques helps the online publishers to rank the authors according to their articles, either they are identifying true news or disseminating the propaganda news.

1.8.2 Publisher Ranking Systems:

Such a news content analysis techniques helps the journalists as well as general public to rank the online news articles publishers according to their articles, either they are publishing non-propaganda news or disseminating the propaganda news.

1.8.3 Recommender Systems:

Such a news content analysis techniques could be integrated with different search engines e.g. google, yahoo and bing which identify any online news publisher or author, and recommend to other journalists as well as general public.

1.8.4 Propagandistic/Non-Propagandistic Search Engines:

Such a news content analysis techniques helps to develop news articles based search engines, which classify every new published article either Propagandistic or non – Propagandistic and update its repository.

1.9 Limitation

There are numerous articles which daily publish on different online news platforms across the world. These articles are highly unstructured and scattered due to which it is impossible to collect all those articles and perform such a critical analysis. So, we perform our analysis on news articles which are collected from 104 online sources, due to which our results could not be generalized for other news articles. Our methodology, mainly focused on articles related to politics and current affairs, so it is hard to extract useful information for other type of articles e.g. entertainment, sports, cooking, fashion design etc. We did not use networks and semantic analysis, so this research could be enhanced by this mean.

Chapter 2

Literature Review

Chapter 1 describes enough, which helps to understand about propaganda and propagation mediums of propaganda content in this modern era. In this chapter we focus on analysis of previous proposed approaches by the researches for detection of fake and propaganda content from online news sources. Every new invention in the field of science is based upon previous research work, and then modified as per advanced requirements to achieve better results. As the news articles are dependent upon current incidents, so they are increasing tremendously, so in parallel it is mandatory to detect that which article contains fake content or true content. So it is hard for research community to detect such type of articles manually. They proposed lot of new techniques for automatically detection of fake news from online web sources. In very first, text classification was introduced in 18th century but with the passage of time as classification techniques became more mature, then researchers start classification of documents regarding different categories e.g. News, Web pages etc.

There are many approaches proposed by the researchers in the past to detect fake and propaganda content from online web sources. Every research is basically depends upon analysis of stylometric, writing or readability features. We have also analyzed some linguistic features including char n-gram, part – of – speech, Latent Semantic Analysis (LSA), Word2vec, LIWC and word n-gram. For further advancement of this research we extract some features which outperformed on our dataset for detection of propaganda content from online web sources.

Section 2.1 provides a comprehensive literature review of the research conduct in this area and provides reviews of different proposed approaches used to detect propaganda content from social media platforms.

2.1 Propaganda Detection Using Social Media Data:

As we knows, now a days there are multiple social media platforms which are the best mediums for the propagation of any news content, regardless verifying their authenticity. So, social media platforms and news content propagation is directly proportional to each other. Recently, there are lot of researchers which are digging up new techniques and models for disinformation and biasness detection from news articles content and in social media platforms. Because news articles are increasing tremendously, which enhance the risk of dissemination of propagandistic content on the print media as well as online news platforms. The Online News Association (2000) [42] published a report in which they claimed that 55 % internet users of America think that traditional news outlets are more accurate than the web sites because they fulfill the journalism standards more strictly as compared to online news sources. Due to which the general public rate the online news sources much lower in credibility than did the traditional news sources. In 2000, I. Finberg L. Stone [43] conducted a survey about accuracy of information published by traditional news sources and online news sources. According to their survey report, about 69 % of internet users believed that there is no difference between the accuracy of information which make available either by traditional news sources or online news sources. Include all of this, in 2001 M. Brill [44] select 12 online newspapers and attempt to compare online media platforms with traditional media platforms e.g. print media, newspaper and detect the truthiness of reporting content on the bases of key roles defined professionally for online journalists as well as for traditional working journalists. Similarly, in 2007 Cassidy [45] design a systematic probability newspaper sample, drawn from the 1,191 different newspapers and use this sample for the comparison of perceptions about online and print newspaper journalists.

World Wide Web (www) is increasing with information in a tremendous way, which made it essential for information seekers to extract useful information while filtering out unwanted, fake and hoax contents. In 2019, Hashemi Hallb [46] put forward a binary classification and eight-way classification project for real-time detection of visual propaganda, so called dark material which is published by violent extremist organizations (VEOs). According to this project, visual propaganda is further classified in hard propaganda, soft propaganda, symbolic propaganda, landscape, and organizational communications based upon type of VEO and intent of extracted image. For this project, more than 1.2 million images from different online social networks and web pages were collected among them 120,000 images were classified manually for training dataset. An accuracy of 97.02 %, 86.08 % and F1 of 97.89 %, 85.76 % was generally achieved for a binary as well as for eight-way classification.

Propaganda is an influential mechanism which could create biasness among the general public opinion and it inherently present in extremely biased and fake news. There is a need for such a investigative model which help the general public as well as journalists to explore different perspective of same story, and how media platforms peruse their agenda by using different perspectives of stories. In 2019, Barrón-Cedeño et al. [47] Proposed a model which detect level of propagandistic content present in a news article. Propagandistic content detection is performed by different representation styles including writing and readability level of presence of certain keywords. They perform their experiments on an unseen dataset Qprop which consist upon 51.3k articles: 5.7k collected from propagandistic sources and 45.6k from trustworthy sources. After performing a set of experiments they conclude that, char n-gram outperform with f-measure 82.93 % as a standalone feature whereas it shows 83.21 % f-measure when combined with Nela features.

Hyper partisan is an influential mechanism to spread out extremely one-sided news which propagate among the general public very frequently. Such a hyper partisanship influence the general public opinion, either extremely in favor of any targeted topic or extremely against that topic. From the news publishing platforms in 2018, Potthast et al. [48] conducted a stylometric study for detection of hyper partisan and fake context. It is a viable alternative albeit not specifically for fake news but hyper partisanship could also be detected in a mannered way. They have conducted their experiments on a corpus of 1,627 articles which are collected from 9 different political publishers and organize these publisher in three classes one is the mainstream, second is the hyper partisan left, and third is the hyper partisan right. In this regard, their experimental setup consist upon (1) an annotated news corpus with respect to veracity and hyper partisanship, (2) a stylometric analysis which us based upon set of extensive experiments for the discrimination of fake news, hyper partisan news, and satire news (3) a novel way experiment for the verification and validation of findings and analyzed that the writing style of the left and the right have more in common rather than the mainstream. In the very first all the articles have been analyzed by professional journalists at BuzzFeed, which evaluated that 97 % among the 299 fake news articles are hyper partisan news articles. As a result they concluded that, stylometry with F1 = 0.46 is not a silver bullet for style-based fake news detection. Whereas style analysis perform well when differentiation is required among hyper partisan news from mainstream with F1 = 0.78 as well as satire from both hyper partisan and mainstream news with F1 = 0.81.

According to Williamson Scrofani in 2019 [49], computational propaganda has achieved a significant attention because it plays a key role during the US 2016 presidential elections, UK's Brexit referendum and the Catalan independence vote. It was widely reported that, all these events were campaigned by automated accounts (bots) on the twitter. Bots are quit new phenomena for propaganda dispersion and the tactics are still developing. In 2017, Varol et al. [50] Proposed a model for autonomous entities detection including social bots. They extracted thousands of features from public data and meta-data about users, friends, tweet content, sentiment, and network pattern and activity time series. For the experimental setup they use a publically available dataset which consist upon 15k manually verified twitter bots and 16k manually verified user accounts. As a whole 2.6 million tweets collected against bots and 3 million tweets collected against manually verified users, for features extraction. As a result, they attain best classification performance of 0.95 AUC by Random Forest machine leaning algorithm.

In many cases, the news outlets get labelled either propagandistic or non-propagandistic on the bases of their published content. Such labels are then imposed on each news article of that particular news outlet without verifying the content. Thus, labeling a news article regarding its published platform could introduce noise. To overcome this problem in 2019, Seunghak Yu et al. [51] Proposed a multi-granularity neural network. In this approach, they perform two type of classification i) Sentence Level Classification (SLC), ii) Fragment Level Classification (FLC) on a corpus of 293, 57, 101 articles. They also propose eighteen propaganda techniques *(Loaded language, Name calling or labeling, Repetition, Exaggeration or minimization, Doubt, Appeal to fear/prejudice, Flag-waving, Causal oversimplification, Slogans, Appeal to authority, Black-and-white fallacy, dictatorship, Thought-terminating cliché, Whataboutism, Reductio ad Hitlerum, Red herring, Bandwagon, Obfuscation, intentional vagueness, confusion and Straw man)* and an *ad hoc* evaluation measure with name entity recognition (NER) and plagiarism detection (PD). Each article is analyzed according to these propaganda techniques for both Sentence Level Classification and Fragment Level Classification. According to their experimental results sentence level propaganda detection outperformed yielding F1 = 60.98 % whereas fragment level propaganda detection performance is not very impactful with F1 = 22.58 %. Which shows sentence level propaganda detection is much effective for propaganda detection as compared to fragment level propaganda detection.

Similarly, In 2019 Giovanni Da San Martino at.el [53] presents a shared level task which is based upon Fine Grained Propaganda Detection which was organized as an integrated part of NLP4IF workshop at EMNLP-IJCNLP 2019. Two sub task was assigned for analysis, one is Fragment Level task in which propaganda technique needs to be identified and second is Sentence Level propaganda detection task in which sentences are need to be identified which contains propaganda content. Total 90 teams were participated for both task, but only 14 teams were succeeded to submit system description paper. As a result of Fragment Level task newspeak team attain highest score with F1=0.2422, recall=0.2084 and precision= 0.2893 whereas on the other hand for Sentence Level propaganda detection task, team Tha3aroon performed best with F1=0.6883, recall=0.7889, precision=0.6104.

Under the quest of false information dissemination about US Presidential elections of 2016, Ansgar Kellner at.el. [52] In 2019 developed an automated bot detector based upon tweets from Twitter. They implemented different classifiers with 10-fold cross validation but Gradient Boosting out performed with average F-Measure of 0.891 average AUC of 0.976.

Same like the other communities, online social media (Facebook, Twitter) platforms change the operational ways for extremist and terrorist. In 2019, Mariam Nouh .at.el.[54] Proposed an automated model for the detection of radicle content from social media. Performance of the proposed model was based upon textual, psychological or behavioral context of the published material. Their research comprises upon three fold: 1) Analysis of propaganda material, 2) build a model based upon psychological properties of the published material, 3) evaluation of proposed model on tweets.TF-IDF identifies the suspicious terms regarding radicle content, later on these suspicious terms were used to train Random Forest, Support Vector Machine and K-Nearest Neighbor classifier. Finally Random Forest produce robust results with accuracy=94 %, precision= 95 %, recall=94 % and F-Measure=94 %.

2.1.1 Research Gap:

On the bases of previous work, it is concluded that majority of the researchers applied stylometric and readability features as a standalone features set for propaganda detection from online news platforms. Whereas hybrid features which consist upon combination of standalone features are also very impactful for propaganda detection. In this research we conducted three set of experiments, first two

Table 2.1: Critical Analysis of Existing Approaches

Ref	**Problem Statement**	**Machine Learning Models**	**Features**	**Performance**	**Limitations**
Varol et al. [50] 2017	Binary Classification	Random Forest	User-Based - Features, Friends-Feature, Network-Features, Content & Language Features, Sentiment-Features,	AUC = 95 %	Rather than individual, only cluster of accounts and bots can be identified .
J. Kiesel et al. [48] 2018	Three-way Classification	Weka 's Random Forest	Char n-gram, Part-of-Speech. Readability, Dictionary-Features, Domain-Specific-Features	Hyper-Partisan vs Mainstream $F_1 = 78\%$, Stair vs (Hyper-Partisan & Mainstream) $F_1 = 81\%$	Large-Scale-Fact Checking can not handled by Stylometric Analysis. .
M. Hallb et al. [46] 2019	Binary Classification, Eight-way Classification	5-Layer AlexNet	Pooling, Activation, Function, Resizing Images	Binary Classification, ($F_1 = 97.89\%$ Acc = 97.02 %) Eight-way Classification ($F_1 = 85.76\%$ Acc = 86.08 %)	Work only on fixed resolution images. .

Table 2.1 - Continued from Previous Page

Ref	**Problem Statement**	**Machine Learning Models**	**Features**	**Performance**	**Limitations**
Alberto et al. [29] 2019	Binary Classification	Support Vector Machine, Maximum Entropy Classifier	NELA, LIWC, Readability, Char n-gram, Word n-gram,	Standalone Features ($F_1 = 82.93\%$) Hybrid-Features Features ($F_1 = 83.21\%$)	Work only on large piece of content, by which propaganda technique can not be detected.
S. Yu et al. [51] 2019	Sentence Level Classification, Fragment Level Classification,	BERT, Multi Granularity	Spans, Full-Stack	Sentence Level Classification $F_1 = 60.98\%$, Fragment Level Classification $F_1 = 22.58\%$	Sentence Level analysis and Fragment Level analysis is not applicable for free text.
Giovanni et al. [53] 2019	Sentence Level Classification, Fragment Level Classification,	Logistic regression, Convolutional Neural Networks, BERT	Readability, Sentiment, Emotions, Linguistic Features	Sentence Level Classification ($F_1 = 68.83\%$ Recall = 78.89 % Precision = 61.04 %) , Fragment Level Classification ($F_1 = 24.22\%$ Recall = 20.84 % Precision = 28.93 %)	Sentence Level analysis and Fragment Level analysis is not applicable for free text. . .

experiments consist upon standalone features set *(Part-of-Speech, Word2vec,Latent Semantic Analysis, LIWC,Char Tri-Gram, Word-Uni-Gram)* and combination of these features. Third experiment give us a significant edge upon all pervious approaches, we extract most influential features among these standalone features *(Part-of-Speech, Word2vec, Latent Semantic Analysis, LIWC, Char Tri-Gram)* using wrapper method which give us *14 word2vec, 19 LSA, 2 LIWC, 1 POS and 14 Char Tri Gram* based features and evaluated these features with three machine learning models Random Forest, Decision Tree and Naïve Bayes.

Chapter 3

Proposed Methodology

The literature review section delineates that there are multiple approaches have been proposed in the past for binary classification. The key observation from previous proposed approaches which make motivated and significant our proposed methodology is described as follows: 1)To the best of our knowledge, there does not exist any study which implement these features (Word2Vec,Latent Semantic Analysis, Part – of – Speech , Word Uni Gram , Word Tri Gram, LIWC) and their two and three-set combinations, 2) there does not exist any study which implement these machine learning models (Naïve-Bayes, Random Forest, and Decision Tree), 3) for a better and an accurate result analysis we consider four different evaluation metrics (Area Under Curve, F1-Measure, Recall, Precision and Accuracy) which we does not found in any previous study.

In this chapter, we will discuss on the development methodology of our experimental scheme. This scheme outperformed for fake and propaganda content detection from online web sources. We have divided this chapter into different sections. Section 3.1 describes the Block Diagram of proposed research methodology, section 3.2 Dataset Description, section 3.3 Pre-Processing of data, section 3.4 Feature Reference, section 3.5 Normalization of extracted features, section 3.6 Machine Learning Models, section 3.7 Evaluation Metrics.

3.1 Propaganda Detection Model

Figure 3.1 is a graphical representation of our proposed research methodology which consist upon following modules:

1. Pre-Processing
2. Feature Extraction
3. Normalization
4. Machine Learning Models
5. Evaluation Metrics
6. Binary output (Propaganda or non-propaganda class label)

As stated by an abstract elaboration of block diagram, pre-processing consist upon data cleaning and tokenization. In data cleaning phase every type of noise has been removed including stop words and special characters. After removal of noise, stemming is performed in which every inflected word is reduced toward its stem from which it us originated and then all unlabeled articles have been removed from data set to make out our data more precise and accurate. Now, all data go through from a tokenization process in which each sentence is divided in to comma separated words, those words are called tokens. As a result of features extraction a numeric value is obtained from these tokens which is highly un-normalized. To make these numeric values between 0 and 1, all these extracted results are passed through a normalization process. Normalized results are used as an input for the evaluation of machine learning models (Naïve-Bayes, Random Forest and Decision Tree). For the evaluation of these machine learning models on the bases of obtained results, we examine four evaluation metrics (Area Under Curve, F1-Measure, Recall, and Precision) and predict results for yes class as well as for two –way classification which depicts average result of each evaluation metric.

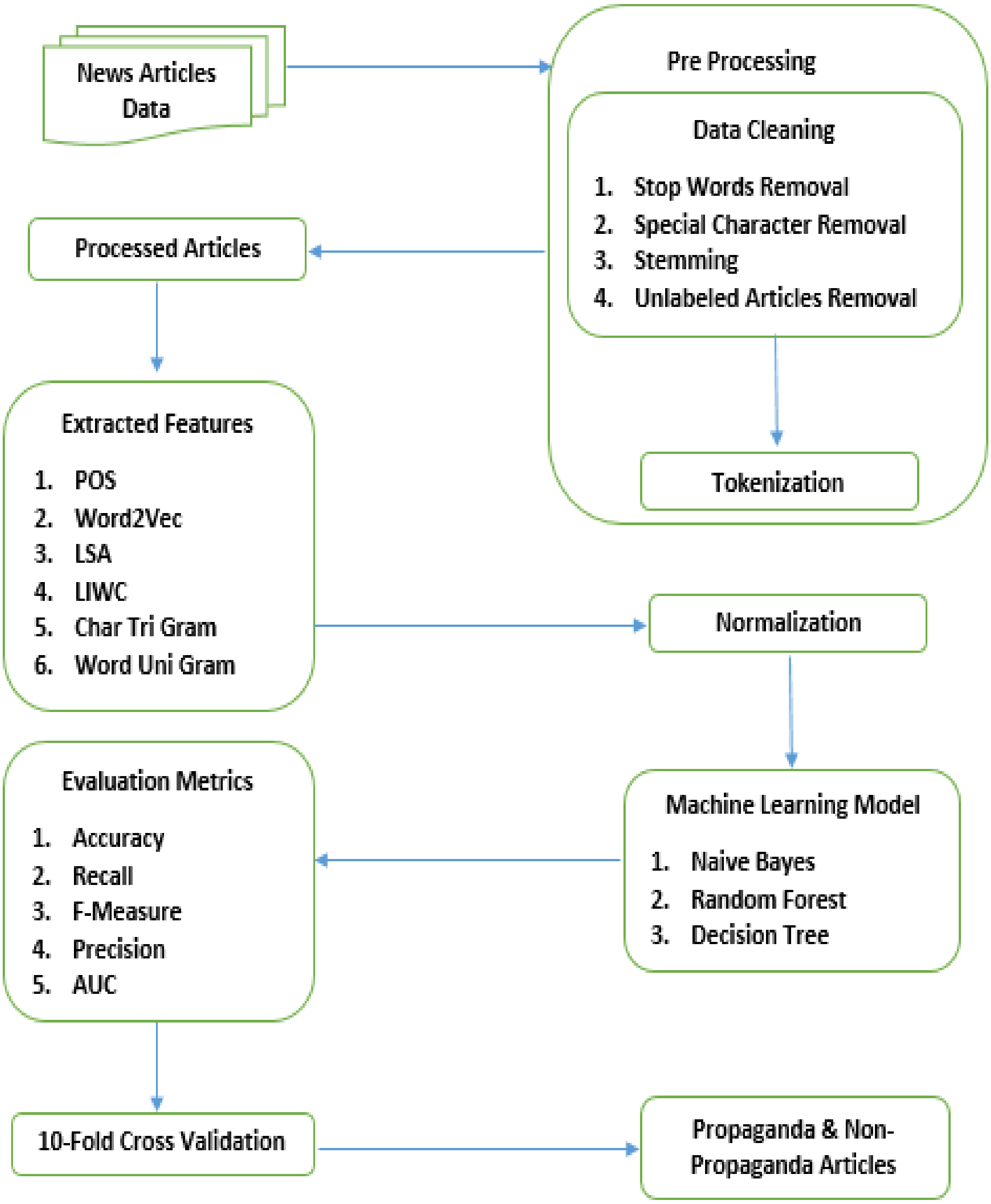
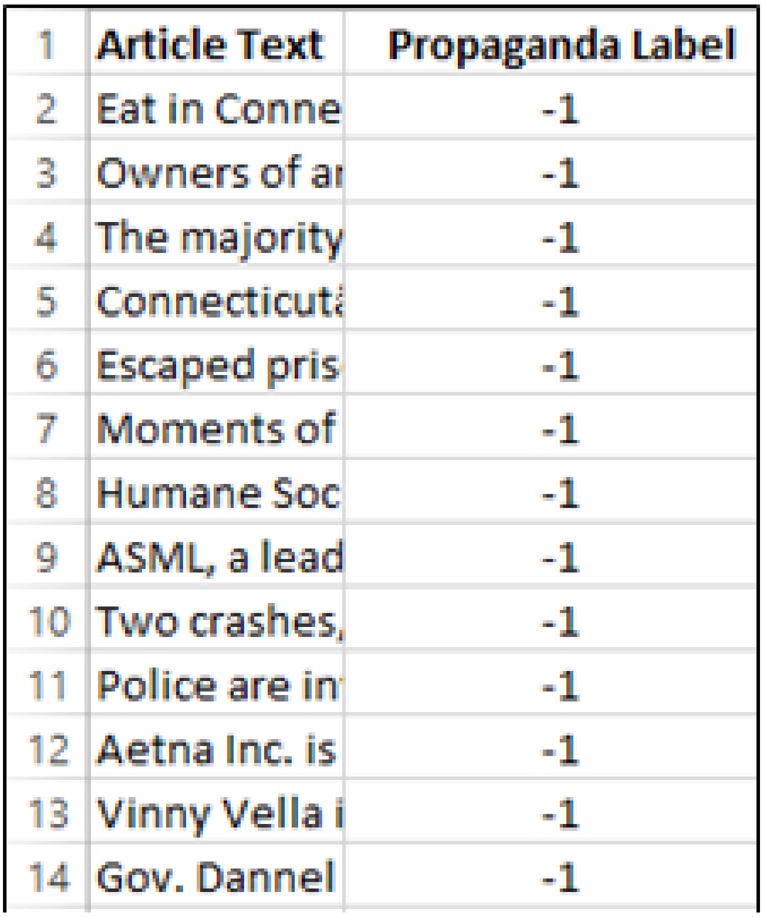


FIGURE 3.1: Block Diagram of proposed methodology



1	Article Text	Propaganda Label
2	Eat in Conne	-1
3	Owners of ar	-1
4	The majority	-1
5	Connecticut	-1
6	Escaped pris	-1
7	Moments of	-1
8	Humane Soc	-1
9	ASML, a lead	-1
10	Two crashes,	-1
11	Police are in	-1
12	Aetna Inc. is	-1
13	Vinny Vella i	-1
14	Gov. Dannel	-1

FIGURE 3.2: Extracted Attributes for Binary Propaganda Detection Model

3.1.1 Dataset Description

For our experimental setup of binary propaganda detection model we use proppy [30] dataset. This dataset as whole consist upon 52 thousand news articles which are collected from 104 different news outlets. These articles are divided in three different parts, 1) Test Dataset, 2) Train Dataset, 3) Dev Dataset. We used Dev partition of proppy dataset which consists of 5135 news articles. Every part of proppy dataset is consist upon 15 columns which contains information about each specific news article.

Dev partition of proppy dataset also consist upon same columns, for our experimental setup, as shown in the following figure 3.2 we extract only two columns (Article Text and Propaganda Label)

Table 3.1: Proppy Dataset Attributes

Sr.#	**Label**	**Description**
1	article text	Text of the article retrieved from news outlets.
2	event location	Geographical location which is collected from GDELT.
3	average tone	Impact of the event which is collected from GDELT.
4	article date	Article publish date which is collected from GDELT.
5	article ID	GDELT ID, unique among the dataset's articles.
6	article URL	Source website direct URL for the published article.
7	MBFC Factuality Label	Factuality label for the source from MBFC.
8	Article Title	Title of the article
9	Article Author	Name of author of the published article.
10	URL to MBFC page	Article url for MBFC page.
11	source name	Source from which article is published.
12	MBFC notes about source	Notes about source originated by MBFC.
13	MBFC biased label	Biasness label assigned by MBFC.
14	Source URL	Home URL for source publisher.
15	Propaganda Label	1 for propaganda and -1 for non-propaganda.

3.1.1.1 Pre-Processing

To convert the news articles in the form of a suitable input for machine learning models, all the data passed from following data pre-processing steps. **Special Characters Removal:**

Special Characters are usually such type of characters which are used for some abbreviations e.g. it's, doesn't, don't etc. and symbols e.g.!, @ etc. Removal of such characters does not place any effective impact on results calculations. In our data set we remove all special characters which improve our investigation results significantly.

Stop Words Removal:

Stop words are usually considered such type of extra words which does not place any effective impact on results calculations. In our data set we remove all stop words which improve our investigation results significantly.

Stemming:

Porter's Stemming algorithm is designed to stem English language based texts, which was one of the most popular stemming methods proposed in 1979. It is used in data normalization process that is usually done in Natural language processing. To get morphological variants of searched terms stemming algorithm's such as Porter Stemmer used following rules:

1. To avoid the plurals and -ed or -ing suffixes.
2. Replace terminal y with i if no other vowel exist in the stem
3. Mapping of double suffix to single ones: -ization, -ational, etc
4. Deals with suffixes, -full, -ness etc
5. Takes of -ant, -ence, etc
6. Removes a final -e

Unlabeled Articles Removal: Dataset was comprises upon 5139 news articles, 4 of them was unlabeled. So we remove unlabeled articles to improve our investigation results. **Tokenization:** It is a process in which all text is splitted in single comma separated words. Each single comma separated word is known as token. It is necessary in Natural language processing.

3.2 Feature Extraction

3.2.1 POS Tagging

Part-Of-Speech (POS) is well known technique for English language based text, in which a specific part of the speech is assigned to each tokenized word in the text. In 2008 E.Atwell [31] develop a verity of tag set for POS-Tagging because traditional English grammar generally provide only 8 part – of – speech tags which are derived from Latin grammar. It make easy to identify the linguistic features. These are total 35 features comprise of adjective, adverb and their distinct forms. Natural language tool kit built in methods for POS tagging.

3.2.2 LIWC

In 2007, J. W. Pennebaker et al. [32] proposed a very efficient and affective framework for analysis of verbal and written context based samples which is called Linguistic Inquiry and Word Count or LIWC. LIWC program perform computerized text analysis and categorize the words which are used in our everyday language. These words are then mapped against the specific category according to physiological thoughts e.g. feelings, personality and motivation.

3.2.3 Word2vec

Word2vec is an advance technique which is used for natural language processing, proposed by google [33] which is not an individual algorithm, but it comprises upon two different learning models, one is Continuous Bag of Words (CBOW) and second is Skip-gram. These models take text data as an input and generate word vectors as an output. These word vectors that can be represented as a large piece of text. To learn word associations, word2vec algorithm uses a neural network model. Once it trained, it can detect synonymous words or suggest additional words for a sentence.

3.2.4 Latent Semantic Analysis

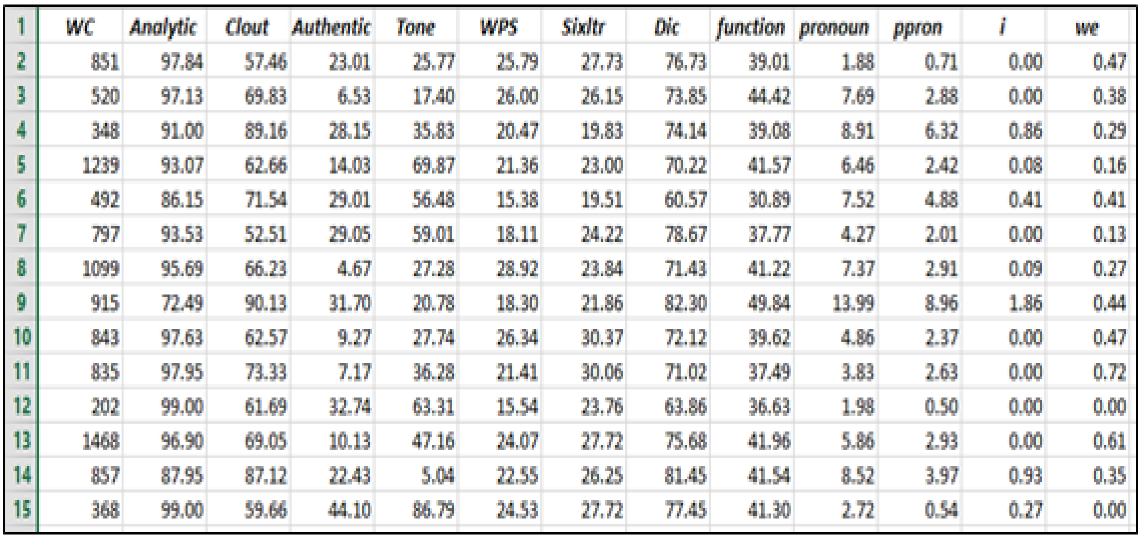
Latent semantic analysis (LSA) is a statistical model for natural language processing that permits comparisons of the semantic similarity between the information extracted by textual data [34]. To analyze the relationships between a set of instances and the topic a matrix is constructed which contains word counts per instance, and then reduce that matrix order by singular value decomposition (SVD). Similarity or dissimilarity of vectors depends upon cosine of the angle between the two vectors. Cosine value close to 1 represent more similar instances while values close to 0 represent more dissimilar instances.

3.2.5 Word Uni Gram

Word n-gram model is used to embed textual sequence which is based upon uni, bi or tri gram words. It predicts the probability of next word occurrence as well as sequence of a given sentence [35]. Probability can be calculated for word uni-gram e.g. thank, you , very, much etc. word bi-gram e.g. thank you, you very, very much etc., word tri-gram e.g. thank you very, you very much etc. and so on. For our research methodology we used only word uni-gram as a standalone feature.

3.2.6 Char Tri Gram

Character n-gram model is used to embed textual sequence which is based upon uni, bi or tri gram characters [36]. It predicts the occurrence of next character as well as sequence in a given sentence. Probability can be calculated for character uni-gram e.g. t,h,a,n,k etc. char bi-gram e.g. th,ha,an,nk etc., char tri-gram e.g.



1	WC	Analytic	Clout	Authentic	Tone	WPS	Sixltr	Dic	function	pronoun	ppron	i	we
2	851	97.84	57.46	23.01	25.77	25.79	27.73	76.73	39.01	1.88	0.71	0.00	0.47
3	520	97.13	69.83	6.53	17.40	26.00	26.15	73.85	44.42	7.69	2.88	0.00	0.38
4	348	91.00	89.16	28.15	35.83	20.47	19.83	74.14	39.08	8.91	6.32	0.86	0.29
5	1239	93.07	62.66	14.03	69.87	21.36	23.00	70.22	41.57	6.46	2.42	0.08	0.16
6	492	86.15	71.54	29.01	56.48	15.38	19.51	60.57	30.89	7.52	4.88	0.41	0.41
7	797	93.53	52.51	29.05	59.01	18.11	24.22	78.67	37.77	4.27	2.01	0.00	0.13
8	1099	95.69	66.23	4.67	27.28	28.92	23.84	71.43	41.22	7.37	2.91	0.09	0.27
9	915	72.49	90.13	31.70	20.78	18.30	21.86	82.30	49.84	13.99	8.96	1.86	0.44
10	843	97.63	62.57	9.27	27.74	26.34	30.37	72.12	39.62	4.86	2.37	0.00	0.47
11	835	97.95	73.33	7.17	36.28	21.41	30.06	71.02	37.49	3.83	2.63	0.00	0.72
12	202	99.00	61.69	32.74	63.31	15.54	23.76	63.86	36.63	1.98	0.50	0.00	0.00
13	1468	96.90	69.05	10.13	47.16	24.07	27.72	75.68	41.96	5.86	2.93	0.00	0.61
14	857	87.95	87.12	22.43	5.04	22.55	26.25	81.45	41.54	8.52	3.97	0.93	0.35
15	368	99.00	59.66	44.10	86.79	24.53	27.72	77.45	41.30	2.72	0.54	0.27	0.00

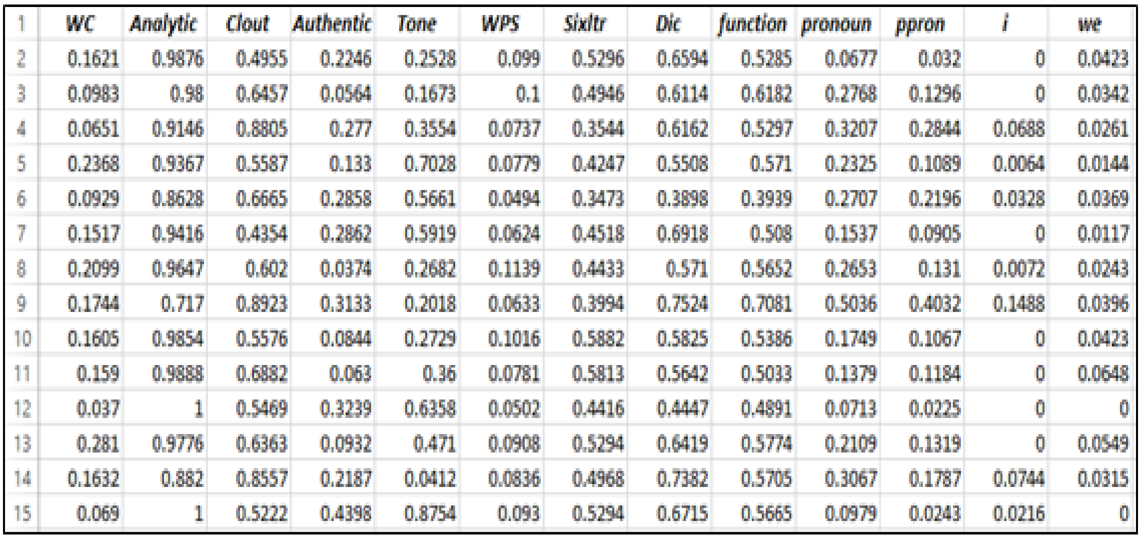
FIGURE 3.3: Un-Normalized form of extracted features

tha,han,ank etc. and so on. For our research methodology we used only char tri-gram as a standalone feature.

3.3 Normalization

Extracted features have different numeric values which are highly disperssed. To minimize this disperssion, generally normalization is the only technique which is often applid as part of data preparation for machine learning models. The main aim of normalization is to scale down all dissperssed numeric values in a defined scale without distorting the differences in ranges of orignal values . In this research methodolgy, all the numeric results of extracted features are scaled down between 0 and 1, which reduce the deviation and shows a consistency among the results of machine learning models.Figure 3.3 is illustarting the un-normalized results of extracted features.

Figure 3.4 is illustarting the normalized results of extracted features which are scaled down between 0 and 1.



1	WC	Analytic	Clout	Authentic	Tone	WPS	Sixltr	Dic	function	pronoun	ppron	i	we
2	0.1621	0.9876	0.4955	0.2246	0.2528	0.099	0.5296	0.6594	0.5285	0.0677	0.032	0	0.0423
3	0.0983	0.98	0.6457	0.0564	0.1673	0.1	0.4946	0.6114	0.6182	0.2768	0.1296	0	0.0342
4	0.0651	0.9146	0.8805	0.277	0.3554	0.0737	0.3544	0.6162	0.5297	0.3207	0.2844	0.0688	0.0261
5	0.2368	0.9367	0.5587	0.133	0.7028	0.0779	0.4247	0.5508	0.571	0.2325	0.1089	0.0064	0.0144
6	0.0929	0.8628	0.6665	0.2858	0.5661	0.0494	0.3473	0.3898	0.3939	0.2707	0.2196	0.0328	0.0369
7	0.1517	0.9416	0.4354	0.2862	0.5919	0.0624	0.4518	0.6918	0.508	0.1537	0.0905	0	0.0117
8	0.2099	0.9647	0.602	0.0374	0.2682	0.1139	0.4433	0.571	0.5652	0.2653	0.131	0.0072	0.0243
9	0.1744	0.717	0.8923	0.3133	0.2018	0.0633	0.3994	0.7524	0.7081	0.5036	0.4032	0.1488	0.0396
10	0.1605	0.9854	0.5576	0.0844	0.2729	0.1016	0.5882	0.5825	0.5386	0.1749	0.1067	0	0.0423
11	0.159	0.9888	0.6882	0.063	0.36	0.0781	0.5813	0.5642	0.5033	0.1379	0.1184	0	0.0648
12	0.037	1	0.5469	0.3239	0.6358	0.0502	0.4416	0.4447	0.4891	0.0713	0.0225	0	0
13	0.281	0.9776	0.6363	0.0932	0.471	0.0908	0.5294	0.6419	0.5774	0.2109	0.1319	0	0.0549
14	0.1632	0.882	0.8557	0.2187	0.0412	0.0836	0.4968	0.7382	0.5705	0.3067	0.1787	0.0744	0.0315
15	0.069	1	0.5222	0.4398	0.8754	0.093	0.5294	0.6715	0.5665	0.0979	0.0243	0.0216	0

Figure 3.4: Normalized form of extracted features

3.4 Machine Learning Models

3.4.1 Naïve-Bayes

Naïve-Bayes is a classification algorithm which is subset of Bayesian decision theory. It simplifies the learning regarding any given data by assuming that features are independent of given class [37]. Text classification, Spam filtration, Sentiment analysis, and Recommendation System are some of the important applications of Naïve-Bayes algorithm.

3.4.2 Random Forest

Random forest is one of the most flexible and popular machine learning model. It is a supervised machine learning model which usually trained with 'Bagging' method. Bagging means its produce multiple random decision trees and merge them to comprise a more accurate, precise and sable prediction for any input data. It can be used both for regression as well as classification task [38].

Table 3.2: HyperPerameter Tuning of Machine Learning Models

Model	**estimator**	**oob_score**	**random state**	**criterion**	**Cross-Validation**
Random Forest	RandomForest Classifier()	True	42	mse	10
Decision Tree	DecisionTree Classifier()	True	100	gini	10
Naive Bays	GaussianNB()				10

3.4.3 Decision Tree

Decision tree is one of the most popular classification technique which is used in different machine learning approaches. It consist upon three parts, i) Root Node, ii) Branch and iii) Leaf Nodes. Root node is the topmost node of the tree from where the tree begins. All the testing features are placed on internal nodes, every decision is displayed on branch and each leaf node represents an outcome which might be a categorical or continue value [39].

3.5 Evaluation Metrics:

The performance of proposed standalone as well as two-set features have been evaluated on the base of four evaluation metrics AUC, F1-Measure, Recall, Precision and Accuracy. We used Naïve-Bayes, Random Forest and Decision Tree as ML models with 10-fold cross validation.

3.5.1 Area Under Curve

AUC is our first evaluation metric which is two-dimensional area underneath the entire ROC curve - graphical representation of a classification model at all classification thresholds.

3.5.2 F1-Measure

F1-Measure is our second evaluation metric which is used when we need to seek relation between precision and recall an uneven class distribution. The standard formula used for assessment of results is given below:

$$\mathbf{F_1} = 2 * \frac{\mathbf{precision} * \mathbf{recall}}{\mathbf{precision{+}recall}} \tag{3.1}$$

3.5.3 Recall

Recall is our third evaluation metric which talks about how much instances our applied ML model captured as actual positive (True Positive) .The standard formula used for assessment of results is given below:

$$\mathbf{Precision} = \frac{\mathbf{TP}}{\mathbf{TP{+}FN}} \tag{3.2}$$

3.5.4 Precision

Precision is our fourth evaluation metric which talks about how much our applied ML model produced accurate results. The standard formula used for assessment of results is given below:

$$\mathbf{Precision} = \frac{\mathbf{TP}}{\mathbf{TP{+}FP}} \tag{3.3}$$

3.5.5 Accuracy

Accuracy is our fifth evaluation metric which talks about how much proportion of our selected data have been identified correctly. It shows the correct proportion of predicted outcomes either true positive or true negative. The standard formula used for assessment of results is given below:

$$\textbf{Precision} = \frac{\textbf{TP + TN}}{\textbf{TP+TN+FP+FN}} \tag{3.4}$$

3.6 Tools and Language :

For the evaluation of our experimental results we use following tools and techniques:

1. Natural Language Tool Kit (NLTK) is used for POS tagging.
2. Porter Stemmer is used to get the root word.
3. Python – is used for the implementation of all algorithms.
4. Microsoft Excel – is used to store all calculated results.
5. Weka – a well-known data mining tool is used for features selection.

Chapter 4

Experiments and Results

This chapter contains a comprehensive description of all results which are collected from our set of experiments. we conducted a set of experiments to classify the propaganda articles using Dev partition of proppy dataset. It consist of 5135 news articles which are collected from different 104 online web sources. As a binary classification problem here, we have defined two classes: i.e. propaganda (yes) and non-propaganda (no). For experimental setup, we used Part-of-Speech, Word2vec based features, LIWC, Word Uni Gram, Latent Semantic Analysis based features, and Char Tri Gram as the features to investigate their performance as a standalone model as well as combination of two features for propaganda detection. In addition, we select the most influential features by forward-feature selection method. All these selected features are evaluated using following machine learning models:

1. Naïve-Bayes
2. Random Forest
3. Decision Tree

Every machine learning model is evaluated with 10-fold cross validation and for the evaluation of results which are collected from these machine learning models, we utilized precision, recall, f-measure, area under curve (AUC), and accuracy as the evaluation metrics. After the organization of all results which are collected by our selected machine learning models, we examine that Random Forest is the only one ML model which performed best for all set of features, so we choose it for further examinations. From two available class labels (yes vs. no), we examined only yes class (Propaganda) here to compare the performance of our methodology with state of the art baseline. The baseline evaluated its proposed methodology using only yes class label and with a standard performance metric, i.e. f-measure. We conducted our experiments using two types of features set: (i) each stand-alone feature, (ii) the two – features set. First, we take propaganda (yes) class under consideration for performance analysis using standalone features type with each evaluation metric (Precision, Recall, F-Measure, AUC and Accuracy) separately.

4.1 Experimental Setup

Following hardware and software is used for the analysis.

Hardware Requirements:

Following hardware is used for features selection.

1. Processor Intel® Core™ i5-5200U Processor
2. 16 GB RAM
3. 500 GB Hard disk

Operating System and Development Software Following software is used for features selection.

1. Windows 10 or above
2. Python 3.7
3. Idle 3.7
4. Microsoft Excel 2013

4.2 Experiment 1: Performance analysis for propaganda class

In our first experiment, we are interested to examine the influence of proposed features with their evaluated results for propaganda (yes) class only. Our experimental setup consist upon two types of features set: (i) each stand-alone feature, (ii) the combination of the two features. Stand-alone features set includes Char-Tri-Gram, Word-Uni-Gram, Latent Semantic Analysis, Word2vec, LIWC and Part-of-Speech, whereas combination of the two features consist upon Char-Tri-Gram POS, Char-Tri-Gram LIWC, Char-Tri-Gram Word2vec and Char-Tri-Gram LSA. Further, we implement Random forest, Decision Tree and Neive Bayes as a machine learning models with 10-fold cross validation and consider precision, recall, F1 measure, AUC and accuracy as evaluation metrics.

4.2.1 Standalone Feature Performance Using Accuracy:

Figure 4.1 illustrates the results of all features with respect to accuracy. With random forest as a machine learning model, it is obvious from following graphical representation that Char-Tri-Gram presents superior result among all other features with 96.40 % accuracy and LSA shows second best results with 92.30 % accuracy score. Whereas all remaining features including Word-Uni-Gram, LIWC, Word2vec and Part-of-Speech show 92.20 % , 91.60 % , 90.60 % and 88.80 % accuracy score respectively. Similarly with the decision tree as a machine learning model it is obvious from figure 1 that, Char-Tri-Gram shows robust result among all other features with 95 % accuracy and Word-Uni-Gram shows second best results with 91 % accuracy score. Whereas all remaining features including LSA, LIWC, Word2vec and Part-of-Speech show 90 % , 88 % , 87 % and 85 % accuracy score respectively. Unlike the other selected machine learning models, Neive Bayes shows highest score of 81 % when applied on Part-of-Speech and second highest score of 80 % when applied on LSA. Whereas all remaining features including

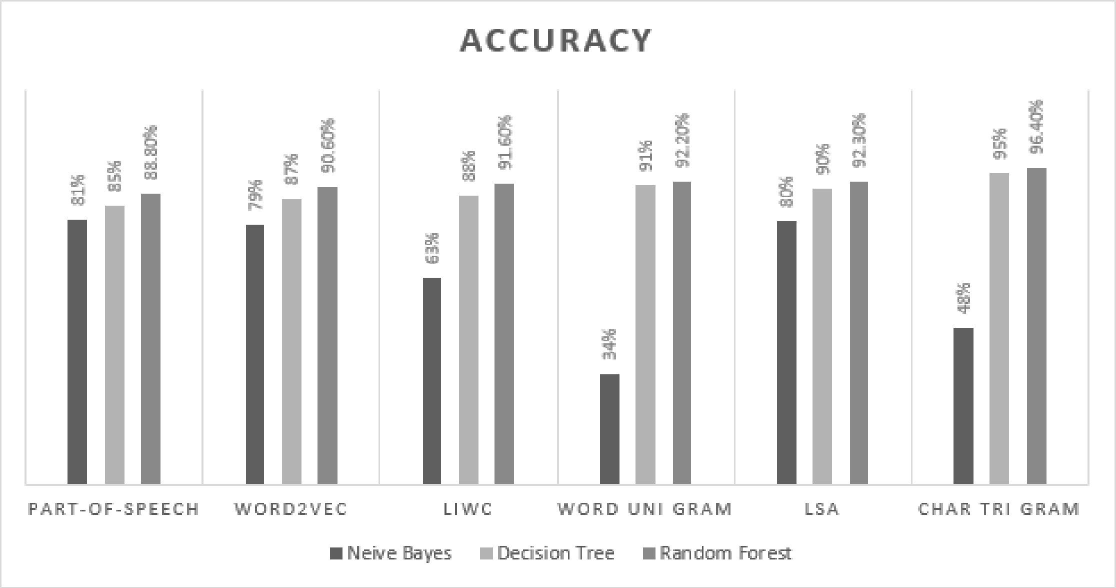


Figure 4.1: Standalone Feature Performance Using Accuracy

Char-Tri-Gram, Word-Uni-Gram , LIWC and Word2vec show 48 % , 34 % , 63 % and 79 % accuracy score respectively. All these representations depicting that, cumulatively random forest out performed among all other selected machine learning models whereas Char-Tri-Gram is best features set among all other features.

4.2.2 Standalone Feature Performance Using Area Under Curve (AUC):

Figure 4.2 illustrates the results of all features with respect to Area under Curve (AUC). With random forest as a machine learning model, it is obvious from following graphical representation that Char-Tri-Gram presents superior result among all other features with 85.70 % AUC and LSA shows second best results with 66.80 % AUC score. Whereas all remaining features including Word-Uni-Gram, LIWC, Word2vec and Part-of-Speech show 65.40 % , 64.50 % , 60.20 % and 53 % AUC score respectively. Similarly with the decision tree as a machine learning model it is obvious from figure 2 that, Char-Tri-Gram shows robust result among all other features with 87.50 % AUC and LSA shows second best results with 73.50 % AUC score. Whereas all remaining features including Word-Uni-Gram, LIWC,

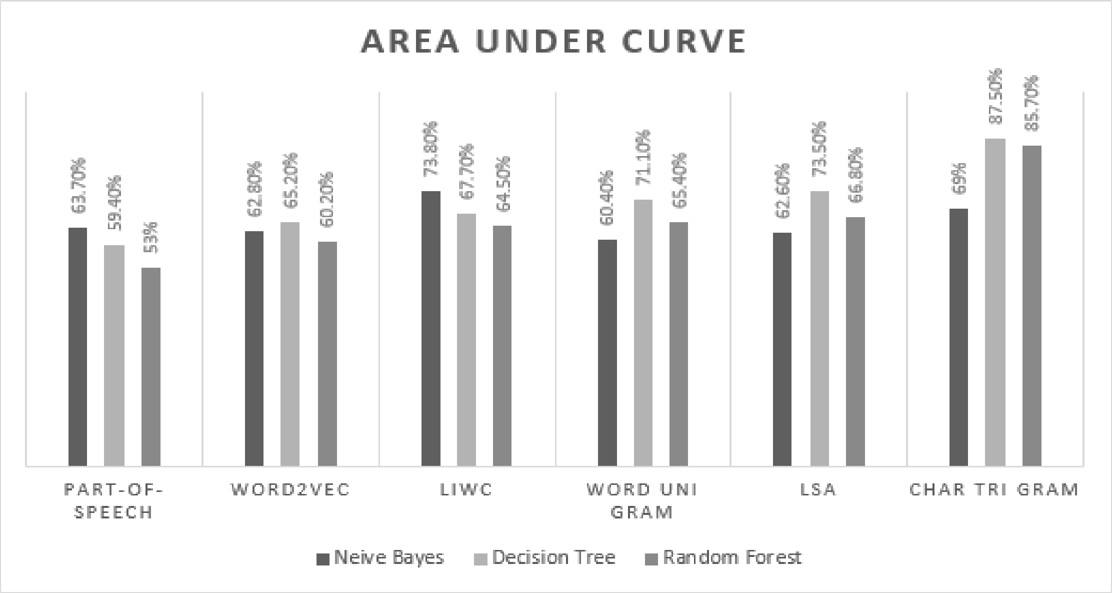


Figure 4.2: Standalone Feature Performance Using Area under Curve (AUC)

Word2vec and Part-of-Speech show 71.10 % , 67.70 % , 65.20 % and 59.40 % AUC score respectively. Unlike the other selected machine learning models, Neive Bayes shows highest score of 73.80 % when applied on LIWC and second highest score of 80 % when applied on Char-Tri-Gram. Whereas all remaining features including LSA, Word-Uni-Gram , Word2vec and Part-of-Speech show 62.60 % , 60.40 % , 62.80 % and 63.70 % AUC score respectively. All these representations depicting that, cumulatively random forest out performed among all other selected machine learning models whereas Char-Tri-Gram is best features set among all other features.

4.2.3 Standalone Feature Performance Using F-Measure:

Figure 4.3 illustrates the results of all features with respect to F-Measure. With random forest as a machine learning model, it is obvious from following graphical representation that Char-Tri-Gram presents superior result among all other features with 81.60 % F-Measure and LSA shows second best results with 49.90 % F-Measure score. Whereas all remaining features including Word-Uni-Gram, LIWC, Word2vec and Part-of-Speech show 47 % , 44.10 % , 33.50 % and 12.20 %

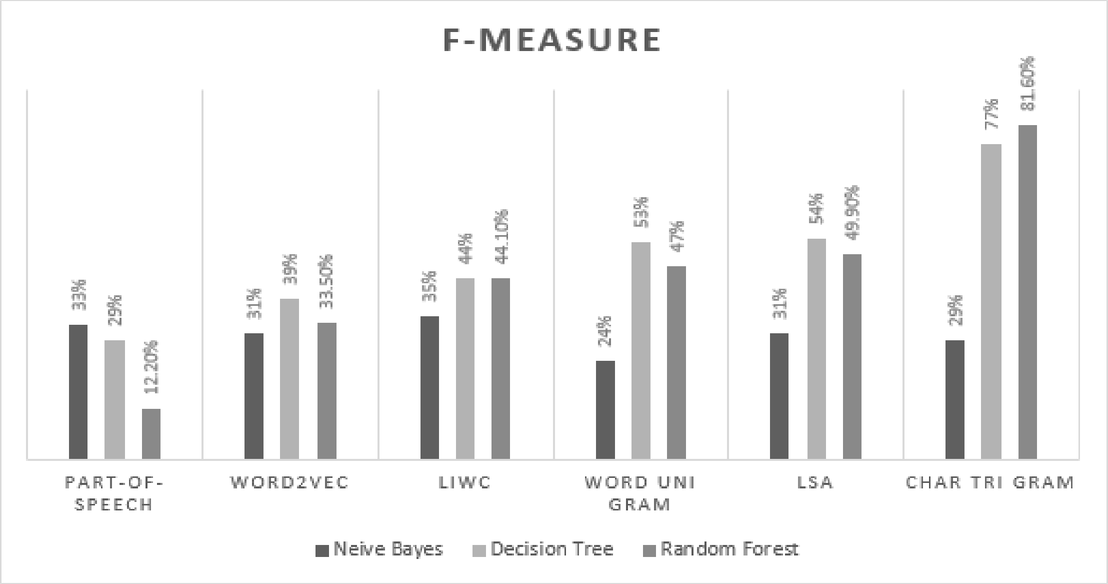


FIGURE 4.3: Standalone Feature Performance Using F-Measure

F-Measure score respectively. Similarly with the decision tree as a machine learning model it is obvious from figure 4.3 that, Char-Tri-Gram shows robust result among all other features with 77 % F-Measure and LSA shows second best results with 54 % F-Measure score. Whereas all remaining features including Word-Uni-Gram, LIWC, Word2vec and Part-of-Speech show 53 % , 44 % , 39 % and 29 % F-Measure score respectively. Unlike the other selected machine learning models, Neive Bayes shows highest score of 35 % when applied on LIWC and second highest score of 33 % when applied on part-of-speech. Whereas all remaining features including LSA, Word-Uni-Gram , Word2vec and Char-Tri-Gram show 31 % , 24 % , 31 % and 29 % F-Measure score respectively. All these representations depicting that, cumulatively random forest out performed among all other selected machine learning models whereas Char-Tri-Gram is best features set among all other features.

4.2.4 Standalone Feature Performance Using Recall:

Figure 4.4 illustrates the results of all features with respect to Recall. With Random Forest as a machine learning model, it is obvious from following graphical

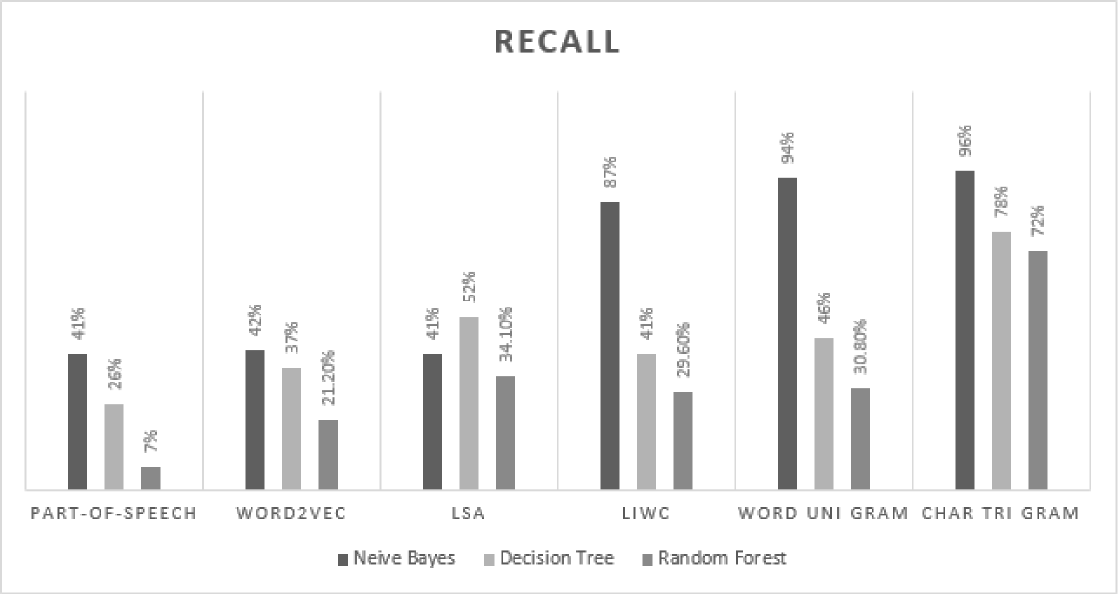


Figure 4.4: Standalone Feature Performance Using Recall

representation that Char-Tri-Gram presented superior result among all other features with 72 % Recall and LSA show second best results with 34.10 % Recall score. Whereas all remaining features including Word-Uni-Gram, LIWC, Word2vec and Part-of-Speech show 30.80 % , 29.60 % , 21.20 % and 7 % Recall score respectively. Similarly with the decision tree as a machine learning model it is obvious from figure 2 that, Char-Tri-Gram shows robust result among all other features with 78 % Recall and LSA shows second best results with 52 % Recall score. Whereas all remaining features including Word-Uni-Gram, LIWC, Word2vec and Part-of-Speech show 46 % , 41 % , 37 % and 26 % Recall score respectively. Unlike the other selected machine learning models, Neive Bayes shows highest score of 96 % when applied on Char-Tri-Gram and second highest score of 94 % when applied on Word-Uni-Gram. Whereas all remaining features including LSA, LIWC, Word2vec and Part-of-Speech show 41 % , 87 % , 42 % and 41 % Recall score respectively. All these representations depicting that, cumulatively Neive Bayes out performed among all other selected machine learning models whereas Char-Tri-Gram is best features set among all other features.

4.2.5 Standalone Feature Performance Using Precision:

Figure 4.5 illustrates the results of all features with respect to Precision. With Random Forest as a machine learning model, it is obvious from following graphical representation that Word-Uni-Gram presented superior result among all other features with 99.40 % Precision and Char-Tri-Gram show second best results with 94.10 % Precision score. Whereas all remaining features including LSA, LIWC, Word2vec and Part-of-Speech show 92.90 % , 86.70 % , 79.20 % and 50 % Precision score respectively. Unalike the Random Forest, the decision tree as a machine learning model show significant precision score of 77 % on Char-Tri-Gram among all other selected features and Word-Uni-Gram placed at second position with 62 % Precision score. Whereas all remaining features including LSA, LIWC, Word2vec and Part-of-Speech show 56 % , 47 % , 42 % and 32 % Precision score respectively. Unlike the other selected machine learning models, Neive Bayes shows highest score of 27 % when applied on Part-of-Speech and present similar precision score of 25 % when applied on Word2vec and LSA. Whereas all remaining features including Word-Uni-Gram, LIWC show 14 % and 22 % Precision score respectively. All these representations depicting that, cumulatively Random Forest out performed among all other selected machine learning models whereas Word-Uni-Gram is best features set among all other features.

4.2.6 Two-Feature Set Performance Using Accuracy:

Figure 4.6 illustrates the results of all selected two-features set with respect to Accuracy. With Random Forest as a machine learning model, it is obvious from following graphical representation that Char-Tri-Gram + Word2vec outperformed among all other selected two-features set with 96.70 % Accuracy and Char-Tri-Gram + POS show second best performance with 96.60 % Accuracy score. Whereas all remaining two-features set including Char-Tri-Gram + LIWC and Char-Tri-Gram + LSA yielding alike accuracy score of 96.30 %. Unalike the Random Forest, the decision tree as a machine learning model presents similar performance for all

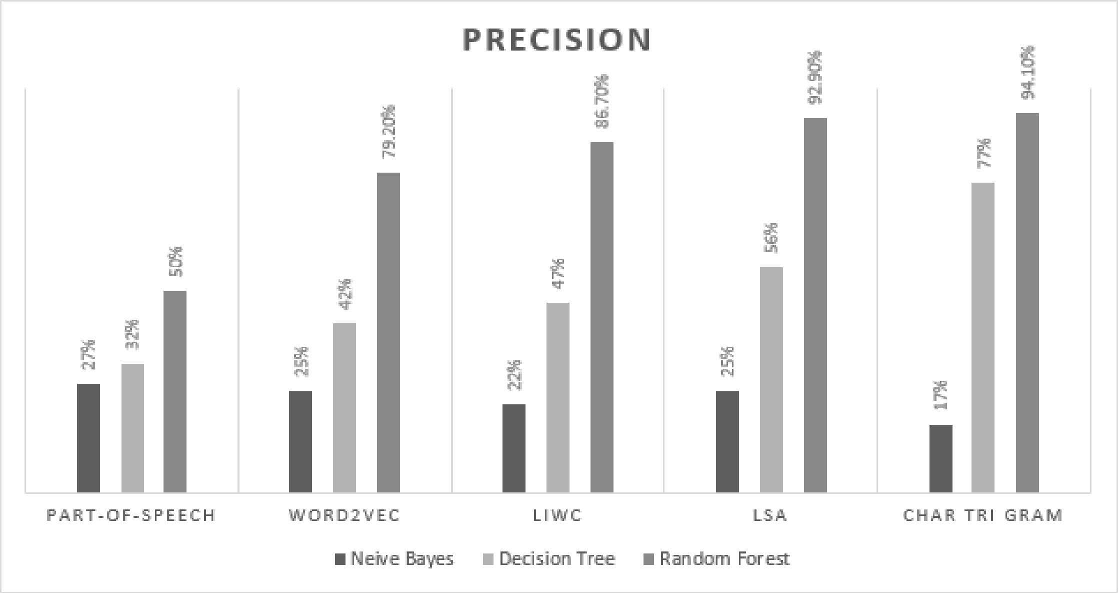


FIGURE 4.5: Standalone Feature Performance Using Precision

selected two-features set with 95 % Accuracy score. Similarly, Neive Bayes also have same representations for all selected two-features set with 48 % Accuracy score. Decision tree and Neive Bayes results depict that, both these machine learning models does not place an influential impact on different combination of selected two-features set. All these representations depicting that, cumulatively Random Forest out performed among all other selected machine learning models whereas Char-Tri-Gram + Word2vec is a best features among all other selected two-features set.

4.2.7 Two-Feature Set Performance Using AUC:

Figure 4.7 illustrates the results of all selected two-features set with respect to Area under Curve (AUC). With Random Forest as a machine learning model, it is obvious from following graphical representation that Char-Tri-Gram + Word2vec outperformed among all other selected two-features set with 86.60 % AUC and Char-Tri-Gram + POS show second best performance with 86.10 % AUC score. Whereas all remaining two-features set including Char-Tri-Gram + LIWC and Char-Tri-Gram + LSA yielding 85.30 % and 84.90 % AUC score respectively. Unalike the Random Forest, the decision tree as a machine learning model presents

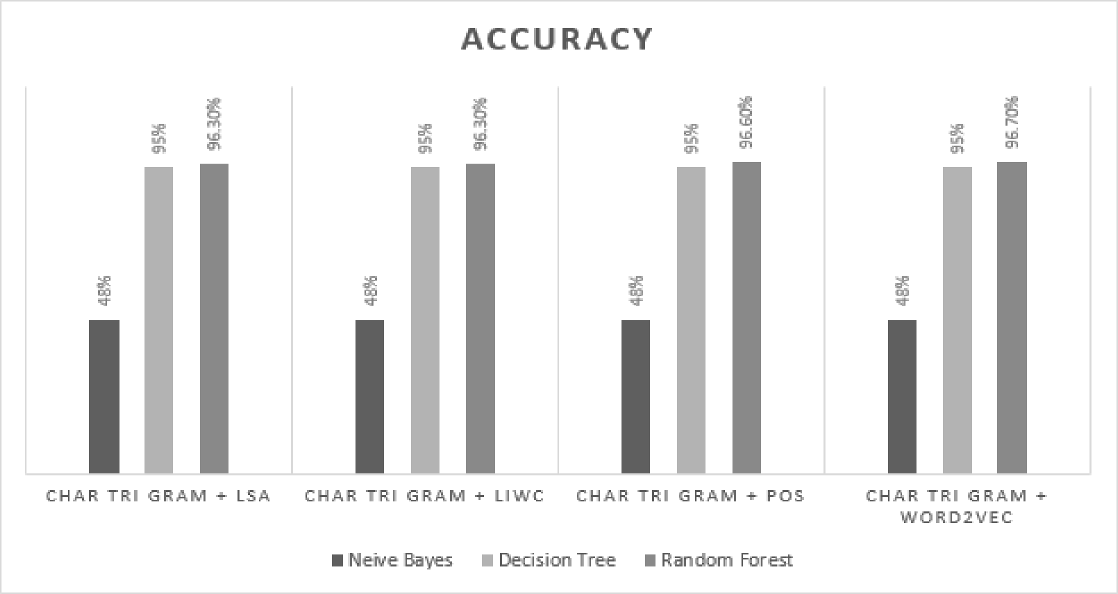


Figure 4.6: Two-Feature Set Performance Using Accuracy

a supreme performance for Char-Tri-Gram + LSA with 88.10 % AUC score and Char-Tri-Gram + Word2vec show second best result with 87.90 % AUC score among all other selected two-features set. Whereas all other selected two-features set including Char-Tri-Gram + LIWC and Char-Tri-Gram + POS show a slight difference with 87.10 % and 87.70 % AUC score. On the other hand Neive Bayes have same representations for all selected two-features set with 69 % AUC score. Which depicts that, Neive Bayes machine learning model does not place an influential impact on different combination of selected two-features set. All these representations depicting that, cumulatively Decision Tree out performed among all other selected machine learning models whereas Char-Tri-Gram + LSA is a best feature among all other selected two-features set.

4.2.8 Two-Feature Set Performance Using F-Measure:

Figure 4.8 illustrates the results of all selected two-features set with respect to F-Measure. With Random Forest as a machine learning model, it is obvious from following graphical representation that Char-Tri-Gram + Word2vec outperformed among all other selected two-features set with 83.30 % F-Measure and Char-Tri-Gram + POS show second best performance with 82.50 % F-Measure

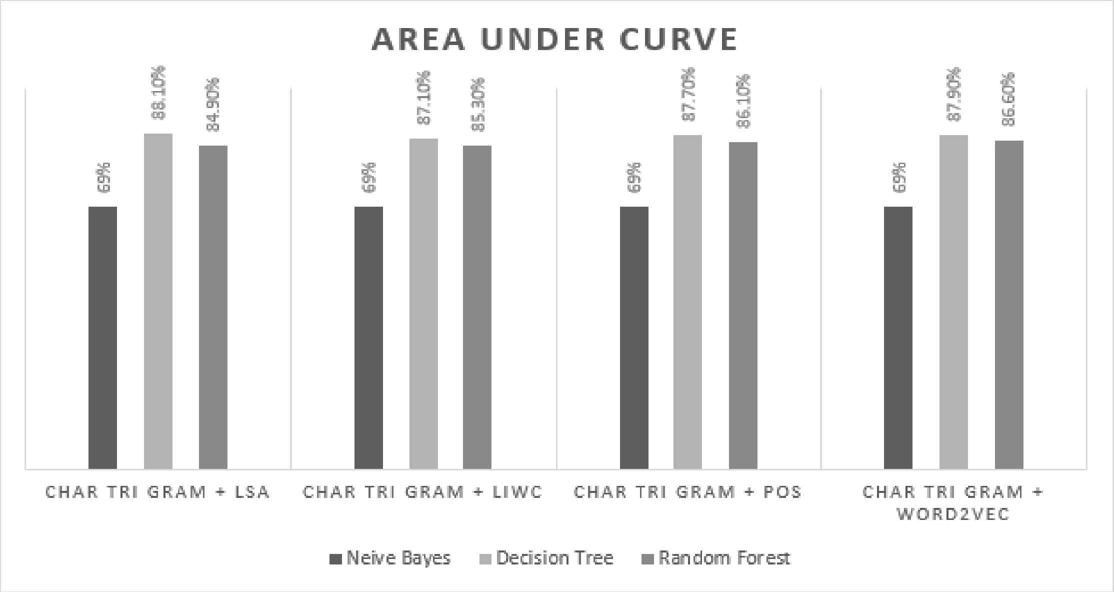


Figure 4.7: Two-Feature Set Performance Using Area Under Curve (AUC)

score. Whereas all remaining two-features set including Char-Tri-Gram + LIWC and Char-Tri-Gram + LSA yielding 81.20 % and 80.90 % F-Measure score respectively. Unalike the Random Forest, the decision tree as a machine learning model presents a supreme performance for both Char-Tri-Gram + Word2vec and Char-Tri-Gram + LSA with 78 % F-Measure score and Char-Tri-Gram + POS show second best result with 77 % F-Measure score among all other selected two-features set. Whereas Char-Tri-Gram + LIWC show least performance with 76 % F-Measure score. On the other hand Neive Bayes have same representations for all selected two-features set with 29 % F-Measure score. Which depicts that, Neive Bayes machine learning model does not place an influential impact on different combination of selected two-features set. All these representations depicting that, cumulatively Random Forest out performed among all other selected machine learning models whereas Char-Tri-Gram + Word2vec is a best feature among all other selected two-features set.

4.2.9 Two-Feature Set Performance Using Recall:

Figure 4.9 illustrates the results of all selected two-features set with respect to Recall. With Random Forest as a machine learning model, it is obvious from following

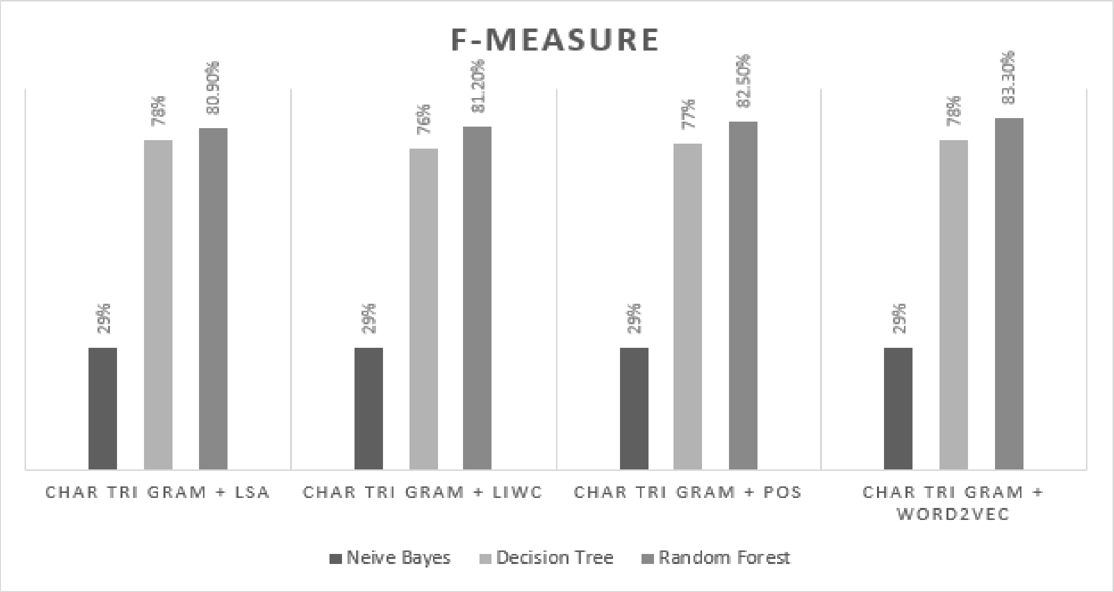


Figure 4.8: Two-Feature Set Performance Using F-Measure

graphical representation that Char-Tri-Gram + Word2vec outperformed among all other selected two-features set with 73.70 % Recall and Char-Tri-Gram + POS show second best performance with 72.50 % Recall score. Whereas all remaining two-features set including Char-Tri-Gram + LIWC and Char-Tri-Gram + LSA yielding 71.10 % and 70.30 % recall score respectively. Unalike the Random Forest, the decision tree as a machine learning model presents a similar performance for Char-Tri-Gram + Word2vec, Char-Tri-Gram + LSA and Char-Tri-Gram + POS with 79 % Recall score. Whereas Char-Tri-Gram + LIWC show second best result with 77 % Recall score. On the other hand Neive Bayes have same representations for all selected two-features set with 96 % Recall score. Which depicts that, Neive Bayes machine learning model does not place an influential impact on different combination of selected two-features set. All these representations depicting that, cumulatively Neive Bayes out performed among all other selected machine learning models.

4.2.10 Two-Feature Set Performance Using Precision:

Figure 4.10 illustrates the results of all selected two-features set with respect to Precision. With Random Forest as a machine learning model, it is obvious from

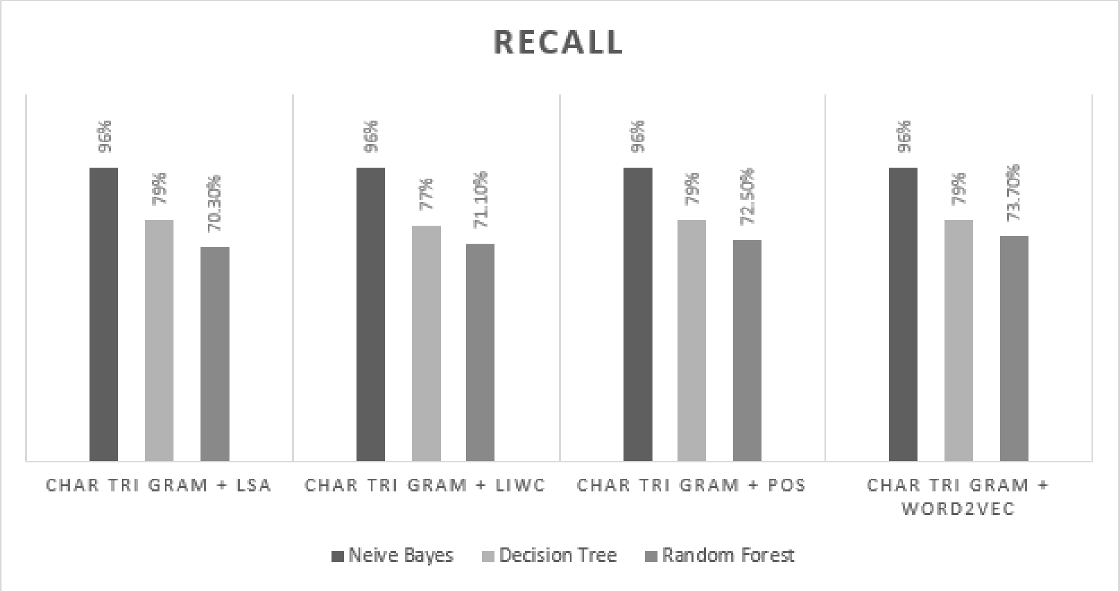


Figure 4.9: Two-Feature Set Performance Using Recall

following graphical representation that Char-Tri-Gram + Word2vec outperformed among all other selected two-features set with 95.70 % Precision and Char-Tri-Gram + POS show second best performance with 95.60 % Recall score. Whereas all remaining two-features set including Char-Tri-Gram + LIWC and Char-Tri-Gram + LSA yielding 94.70 % and 95.30 % recall score respectively. Unalike the Random Forest, the decision tree as a machine learning model presents a supreme performance for Char-Tri-Gram + Word2vec with 78 % Precision score. Whereas Char-Tri-Gram + POS and Char-Tri-Gram + LSA present second best performance with 76 % Recall score among all other selected two-features set. Char-Tri-Gram + LIWC show least performance with 75 % Precision score which does not place any significant impact on experimental results. On the other hand Neive Bayes have same representations for all selected two-features set with 17 % Precision score. Which depicts that, Neive Bayes machine learning model does not place an influential impact on different combination of selected two-features set. All these representations depicting that, cumulatively Random Forest out performed among all other selected machine learning models whereas Char-Tri-Gram + Word2vec is a best feature among all other selected two-features set.

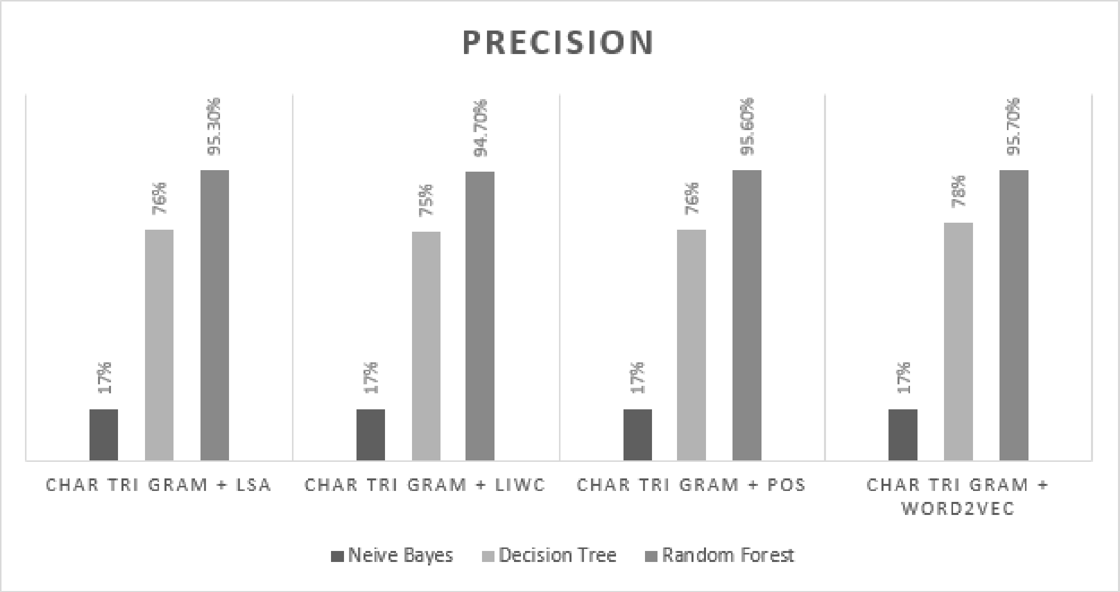


FIGURE 4.10: Two-Feature Set Performance Using Precision

4.3 Experiment 2: Two-way classification

In our second experiment, we are interested to examine the influence of proposed features with their macro average on binary classification task. Our experimental setup consist upon two types of features set: (i) each stand-alone feature, (ii) the combination of the two features. Stand-alone features set includes Char-Tri-Gram, Word-Uni-Gram, Latent Semantic Analysis, Word2vec, LIWC and Part-of-Speech, whereas combination of the two features consist upon Char-Tri-Gram POS, Char-Tri-Gram LIWC, Char-Tri-Gram Word2vec and Char-Tri-Gram LSA. Experiment 1 results depict that Random Forest outperformed among all remaining selected machine learning models. So, for further implementation we use only Random Forest as a machine learning model with 10-fold cross validation and consider precision, recall, F1 measure, AUC and accuracy as evaluation metrics.

4.3.1 Standalone Feature Performance Using Accuracy:

Figure 4.11 illustrates the results of all selected features with respect to accuracy. As it is discussed earlier that only Random Forest is implemented as a machine

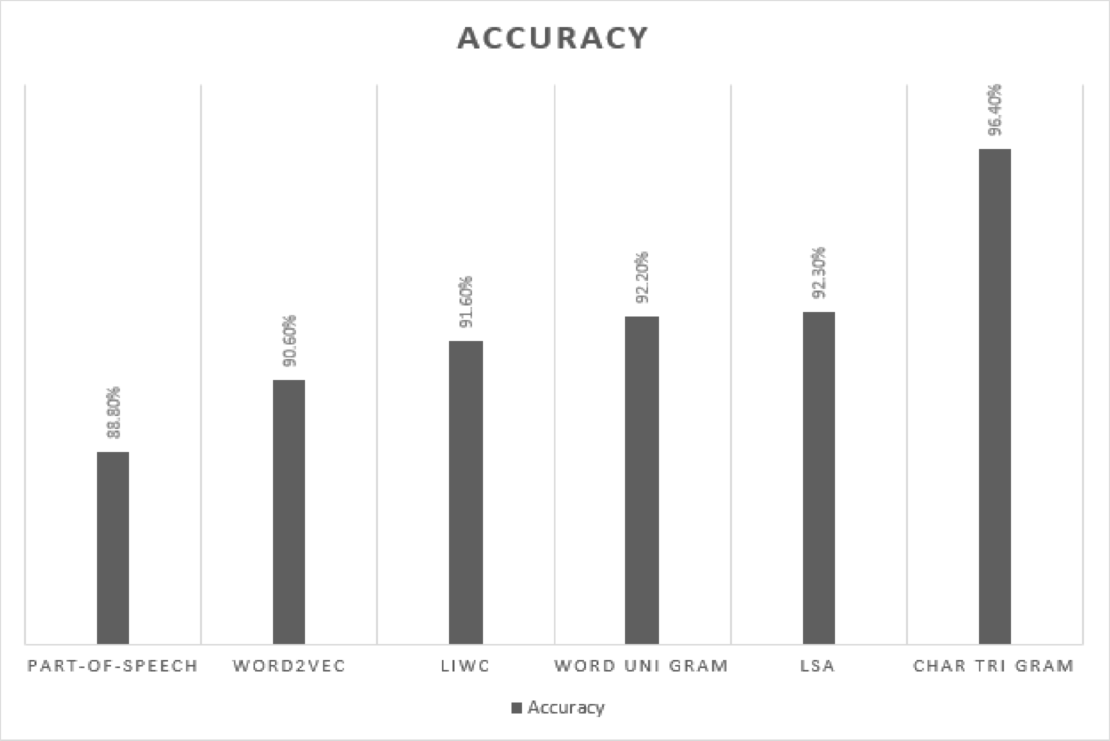


Figure 4.11: Standalone Feature Performance Using Accuracy

learning model which depicts a gradual decrease in the performance of features from Char-Tri-Gram towards Part-of-Speech. It is obvious from the following graphical representation that Char-Tri-Gram presents supreme result among all other selected features with 96.40 % Accuracy score and LSA shows second best influential impact with 92.30 % Accuracy score. Word-Uni-Gram presents a slight difference of 0.1 % by LSA, which placed it on third position with 92.20 % Accuracy score. Whereas all remaining features including LIWC, Word2vec and Part-of-Speech show 91.60 % , 90.60 % and 88.80 % Accuracy score respectively. All these representations depicting that, cumulatively Char-Tri-Gram out performed among all other selected features.

4.3.2 Standalone Feature Performance Using AUC:

Figure 4.12 illustrates the results of all selected features with respect to Area under Curve (AUC). As it is discussed earlier that only Random Forest is implemented as a machine learning model which depicts a gradual decrease in the

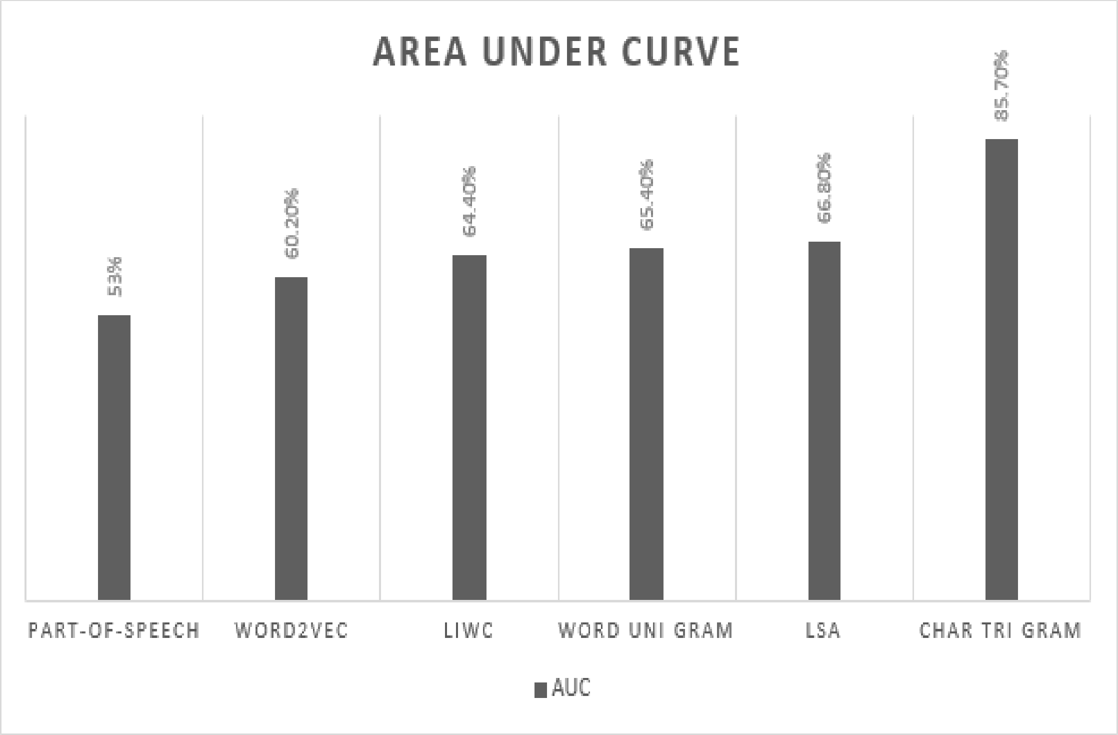


FIGURE 4.12: Standalone Feature Performance Using Area Under Curve

performance of features from Char-Tri-Gram towards Part-of-Speech. It is obvious from the following graphical representation that Char-Tri-Gram presents supreme result among all other selected features with 85.70 % AUC score and LSA shows second best influential impact with 66.80 % AUC score. Word-Uni-Gram presents a difference of 1.4 % by LSA, which placed it on third position with 65.40 % AUC score. Whereas all remaining features including LIWC, Word2vec and Part-of-Speech show 64.40 % , 60.20 % and 53 % AUC score respectively. All these representations depicting that, cumulatively Char-Tri-Gram out performed among all other selected features.

4.3.3 Standalone Feature Performance Using F-Measure:

Figure 4.13 illustrates the results of all selected features with respect to F-Measure. As it is discussed earlier that only Random Forest is implemented as a machine learning model which depicts a gradual decrease in the performance of features from Char-Tri-Gram towards Part-of-Speech. It is obvious from the following

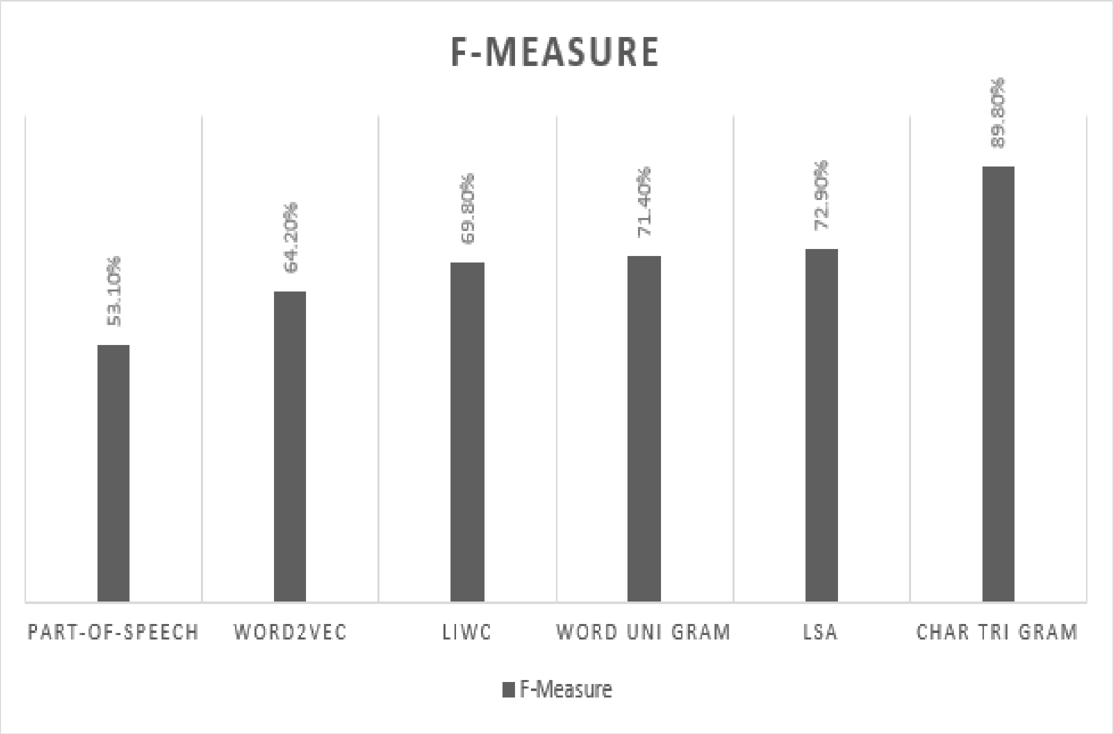


Figure 4.13: Standalone Feature Performance Using F-Measure

graphical representation that Char-Tri-Gram presents supreme result among all other selected features with 89.80 % F-Measure score and LSA shows second best influential impact with 72.90 % F-Measure score. Word-Uni-Gram presents a difference of 1.5 % by LSA, which placed it on third position with 71.40 % F-Measure score. Whereas all remaining features including LIWC, Word2vec and Part-of-Speech show 69.80 % , 64.20 % and 53.10 % F-Measure score respectively. All these representations depicting that, cumulatively Char-Tri-Gram out performed among all other selected features.

4.3.4 Standalone Feature Performance Using Recall:

Figure 4.14 illustrates the results of all selected features with respect to Recall. As it is discussed earlier that only Random Forest is implemented as a machine learning model which depicts a gradual decrease in the performance of features from Char-Tri-Gram towards Part-of-Speech. It is obvious from the following graphical representation that Char-Tri-Gram presents supreme result among all other

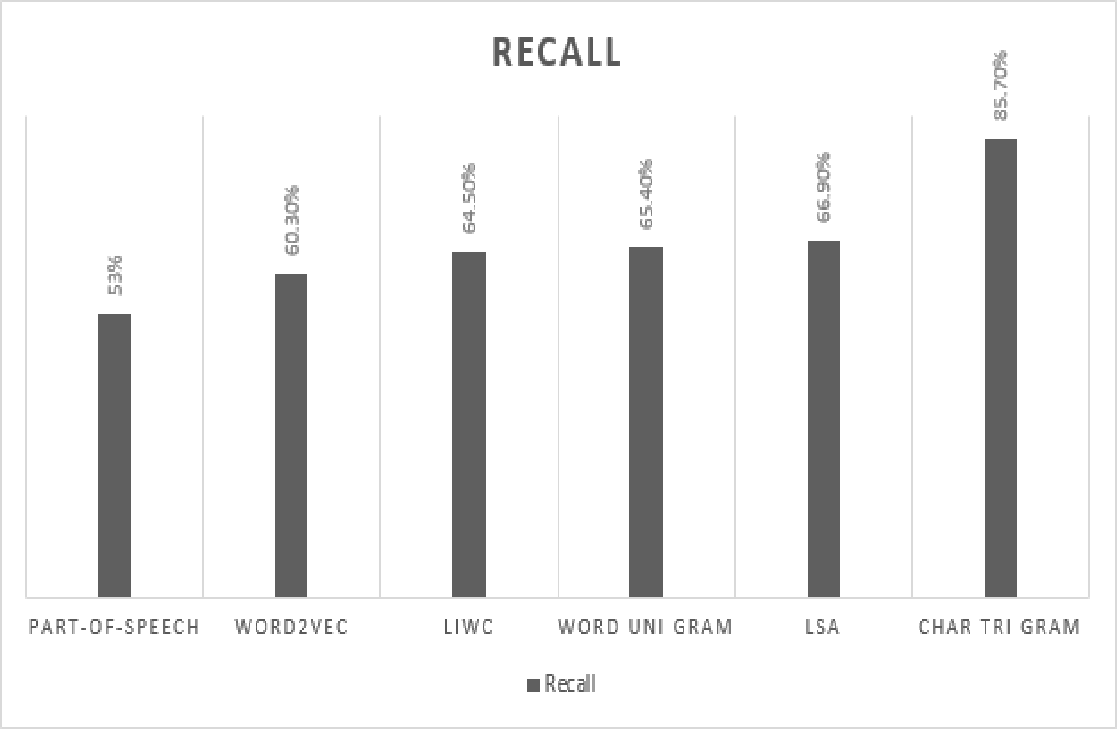


FIGURE 4.14: Standalone Feature Performance Using Recall

selected features with 85.70 % Recall score and LSA shows second best influential impact with 66.90 % Recall score. Word-Uni-Gram presents a difference of 1.4 % by LSA, which placed it on third position with 65.40 % Recall score. Whereas all remaining features including LIWC, Word2vec and Part-of-Speech show 64.50 % , 60.30 % and 53 % Recall score respectively. All these representations depicting that, cumulatively Char-Tri-Gram out performed among all other selected features.

4.3.5 Standalone Feature Performance Using Precision:

Figure 4.15 illustrates the results of all selected features with respect to Precision. As it is discussed earlier that only Random Forest is implemented as a machine learning model which depicts a slight difference of 0.40 % between Char-Tri-Gram and Word-Uni-Gram. Due to this slight difference Char-Tri-Gram placed on second place with 95.30 % Precision score, whereas Word-Uni-Gram leads all the features yielding 95.70 % Precision score. All remaining features including LSA,

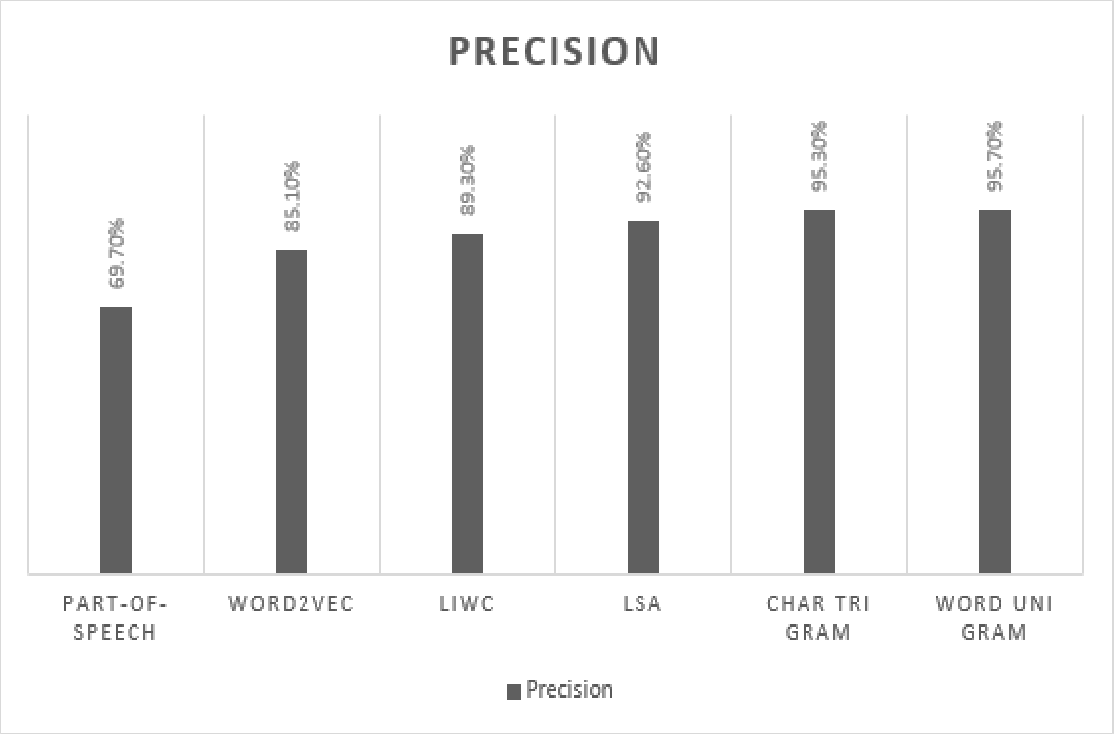


FIGURE 4.15: Standalone Feature Performance Using Recall

LIWC, Word2vec and Part-of-Speech show a gradual decrease in the performance with 92.60 %, 89.30 %, 85.10 % and 69.70 % Precision score respectively. All these representations depicting that, cumulatively Word-Uni-Gram out performed among all other selected features.

4.3.6 Two-Feature Set Performance Using Accuracy:

Figure 4.16 illustrates the results of all selected two-features set with respect to Accuracy. Our baseline did not consider accuracy as their evaluation metric. As we consider macro average, it is obvious from following figure 4.16 that all two-feature sets have very impressive impact when we applied Random Forest as a machine learning model. But if we discussed individually then, Char-Tri-Gram + Word2vec outperformed among all other selected two-features set with 96.70 % Accuracy and Char-Tri-Gram + POS show second best performance with 96.60 % Accuracy score. Whereas Char-Tri-Gram + LSA and Char-Tri-Gram + LIWC have same representations of 96.30 % Precision score. All these representations

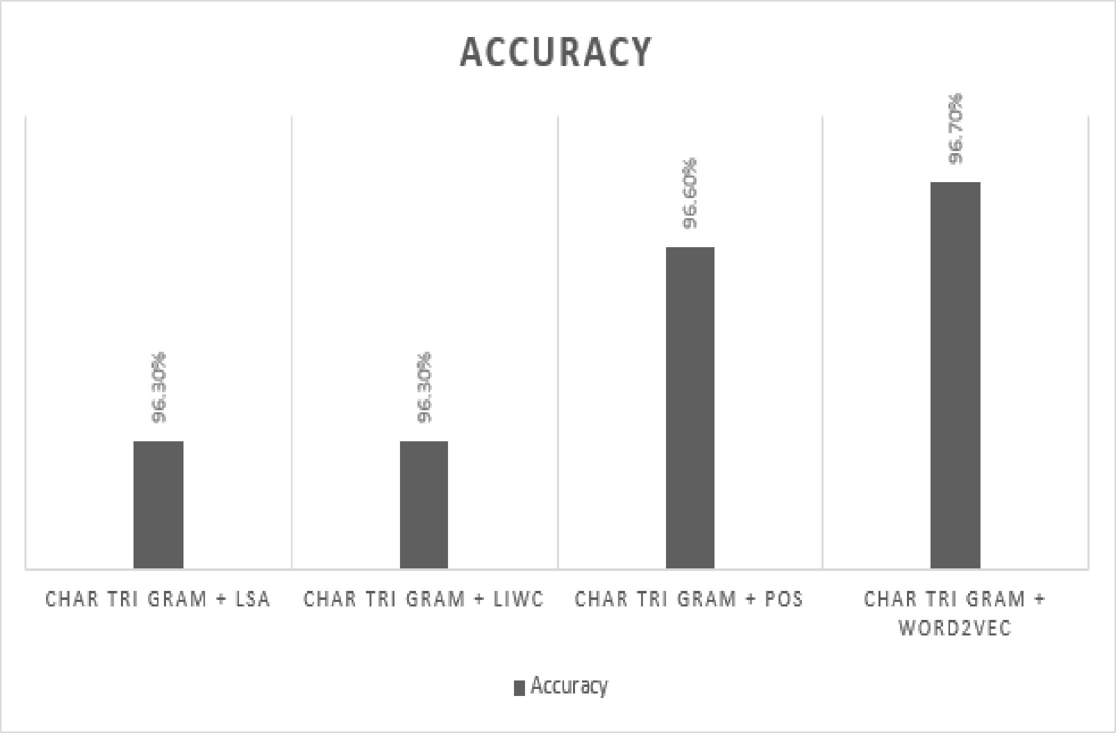


Figure 4.16: Two-Feature Set Performance Using Accuracy

depicting that, cumulatively Char-Tri-Gram + Word2vec is a best feature among all other selected two-features set.

4.3.7 Two-Feature Set Performance Using AUC:

Figure 4.17 illustrates the results of all selected two-features set with respect to Area under Curve (AUC). As we consider macro average, it is obvious from following figure 4.17 that all two-feature sets have very impressive impact when we applied Random Forest as a machine learning model. But if we discussed individually then, Char-Tri-Gram + Word2vec outperformed among all other selected two-features set with 86.60 % AUC and Char-Tri-Gram + POS show second best performance with 86.10 % AUC score. Whereas Char-Tri-Gram + LIWC and Char-Tri-Gram + LSA show 85.30 % and 84.90 % AUC score respectively. However, all these representations depicts that our selected two-feature sets show

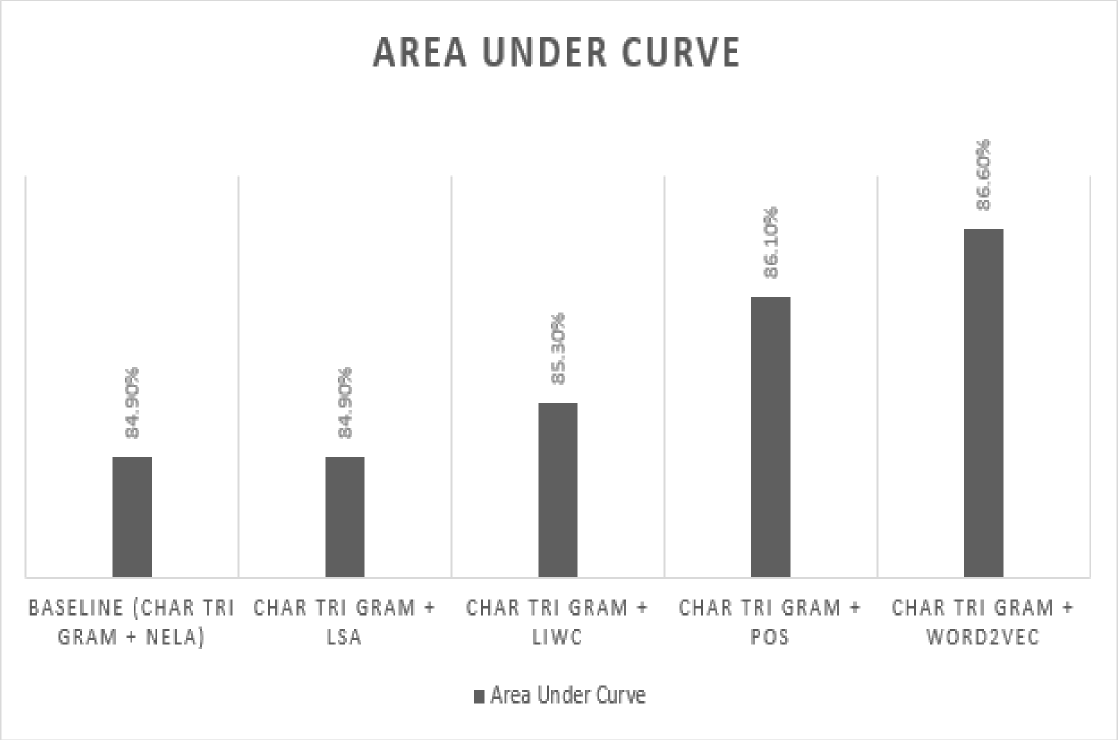


Figure 4.17: Two-Feature Set Performance Using Area Under Curve

higher performance then baseline except Char-Tri-Gram + LSA, which have exactly the same result of 84.90 % AUC score as our baseline. All these representations depicting that, cumulatively Char-Tri-Gram + Word2vec is a best feature among all other selected two-features set.

4.3.8 Two-Feature Set Performance Using F-Measure:

Figure 4.18 illustrates the results of all selected two-features set with respect to F-Measure. As we consider macro average, it is obvious from following figure 4.18 that all two-feature sets have very impressive impact when we applied Random Forest as a machine learning model. But if we discussed individually then, Char-Tri-Gram + Word2vec outperformed among all other selected two-features set with 90.70 % F-Measure and Char-Tri-Gram + POS show second best performance with a slight difference of 0.4 % and have 90.30 % F-Measure score. Whereas Char-Tri-Gram + LIWC and Char-Tri-Gram + LSA slightly differ by each other with 89.60 % and 89.40 % F-Measure score respectively. However, all

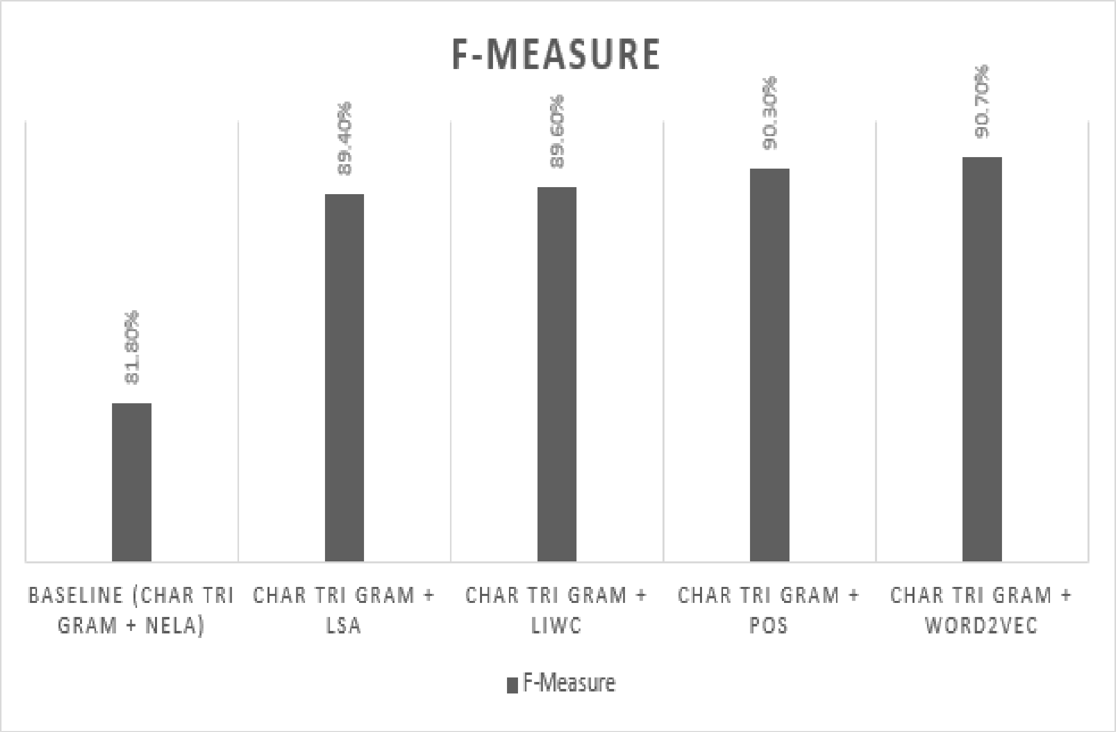


FIGURE 4.18: Two-Feature Set Performance Using F-Measure

these representations depicts that our selected two-feature sets show higher performance then baseline, which have 81.80 % F-Measure. All these representations depicting that, cumulatively Char-Tri-Gram + Word2vec is a best feature among all other selected two-features set.

4.3.9 Two-Feature Set Performance Using Recall:

Figure 4.19 illustrates the results of all selected two-features set with respect to Recall. As we consider macro average, it is obvious from following figure 19 that all two-feature sets have very impressive impact when we applied Random Forest as a machine learning model. But if we discussed individually then, Char-Tri-Gram + Word2vec outperformed among all other selected two-features set with 86.70 % Recall and Char-Tri-Gram + POS show second best performance with a slight difference of 0.6 % and have 86.10 % Recall score. Whereas Char-Tri-Gram + LIWC and Char-Tri-Gram + LSA slightly differ by each other with 85.30 % and 84.90 % recall score respectively. However, all these representations depicts

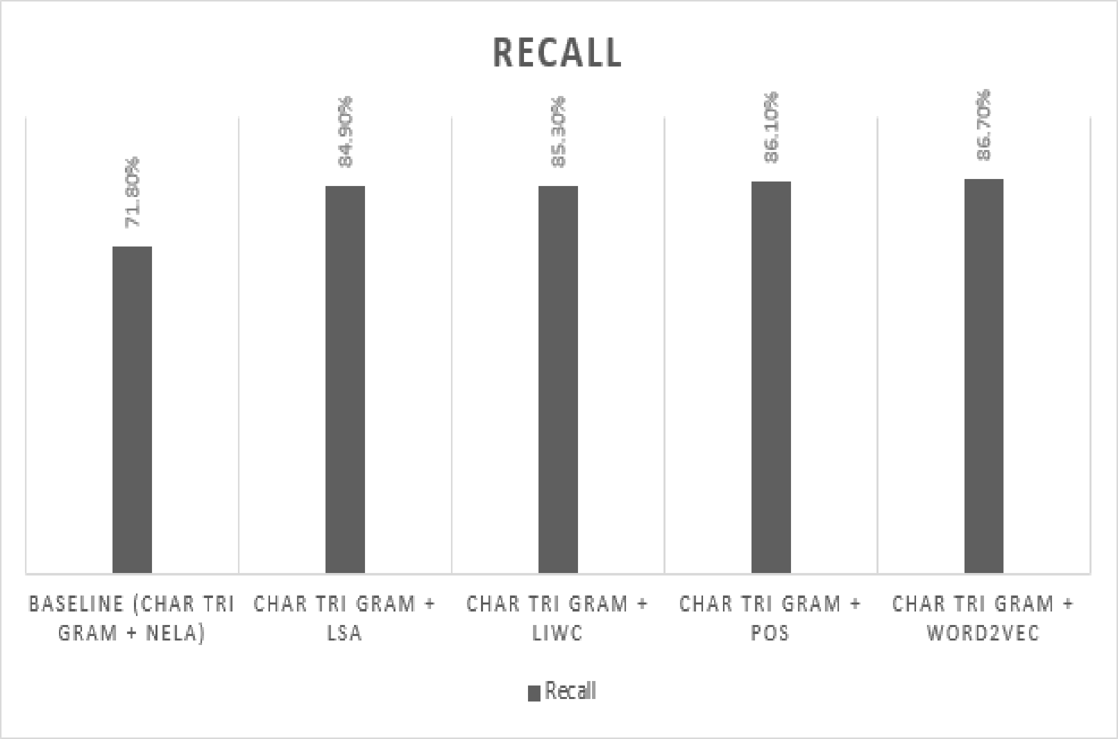


FIGURE 4.19: Two-Feature Set Performance Using Recall

that our selected two-feature sets show higher performance then baseline, which have 71.80 % Recall. All these representations depicting that, cumulatively Char-Tri-Gram + Word2vec is a best feature among all other selected two-features set.

4.3.10 Two-Feature Set Performance Using Precision:

Figure 4.20 illustrates the results of all selected two-features set with respect to Precision. As we consider macro average, it is obvious from following figure 4.20 that all two-feature sets have very impressive impact when we applied Random Forest as a machine learning model. But if we discussed individually then, Char-Tri-Gram + Word2vec outperformed among all other selected two-features set with 96.20 % Precision and Char-Tri-Gram + POS show second best performance with a slight difference of 0.1 % and have 96.10 % Precision score. Whereas Char-Tri-Gram + LIWC and Char-Tri-Gram + LSA slightly differ by each other with

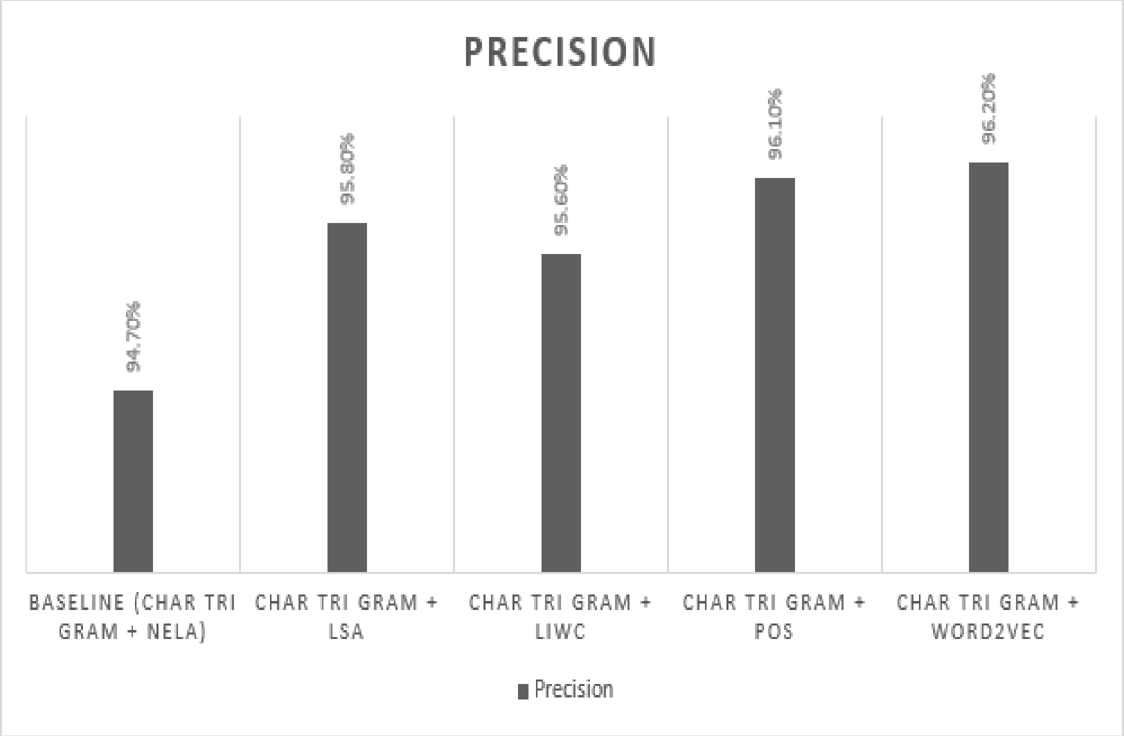


FIGURE 4.20: Two-Feature Set Performance Using Precision

95.60 % and 95.80 % Precision score respectively. However, all these representations depicts that our selected two-feature sets show higher performance then baseline, which have 94.70 % Precision. All these representations depicting that, cumulatively Char-Tri-Gram + Word2vec is a best feature among all other selected two-features set.

4.4 Experiment 3: Impact of Feature Selection

Our third set of experiments is conducted to examine the binary classification task of distinguishing propaganda vs. non-propaganda news articles. In this experimental setup we are interested to examine the influence of proposed features with their evaluated results for propaganda (yes) class only. It consists upon two types of features set: (i) each stand-alone feature (ii) the combination of the two features. Stand-alone features set includes Char-Tri-Gram, Word-Uni-Gram, Latent Semantic Analysis, Word2vec, LIWC and Part-of-Speech, whereas combination of the two features consist upon Char-Tri-Gram POS, Char-Tri-Gram LIWC, Char-Tri-Gram Word2vec and Char-Tri-Gram LSA. In addition, these proposed features are selected by two different techniques i) Top 20 filter based features ii) Forward feature selection technique using wrapper method . Both of these filter selection methods are available in a well-known data mining tool Weka. We use Info Gain as a filter for selection of features whereas wrapper method extract 14 word2vec, 19 LSA, 2 LIWC, 1 POS and 14 Char Tri Gram based features. For evaluation of results for these standalone and hybrid features we implement the Random forest as a machine learning model with 10-fold cross-validation and consider precision, recall, F1 measure, AUC and Accuracy as evaluation metrics.

4.4.1 Filter Based Features Analysis:

In the first part of this experiment we use Info Gain in Weka for the ranking of all proposed features. To achieve best result we selected only top 20 features among all ranked features by filter Info Gain.

4.4.1.1 Standalone Feature Performance Using Accuracy:

Figure 4.21 illustrates the results of all selected features with respect to Accuracy. As it is discussed earlier that only Random Forest is implemented as a machine learning model. Following graphical representation depicts that the selected features of Char-Tri-Gram outperformed among all other selected features set with 58.90 % Accuracy score and LSA show second best performance with 41.50 % Accuracy score. Due to a difference of 6.6 % between LIWC and Word-Uni-Gram features, LIWC score third place among all other selected features with 34.90 % Accuracy score. Whereas all remaining selected features including Word-Uni-Gram, Word2vec and Part-of-Speech show a gradual decrease in the performance with 28.30 %, 20.50 % and 6.90 % Accuracy score respectively. All these representations depicting that, cumulatively Char-Tri-Gram out performed among all other selected features.

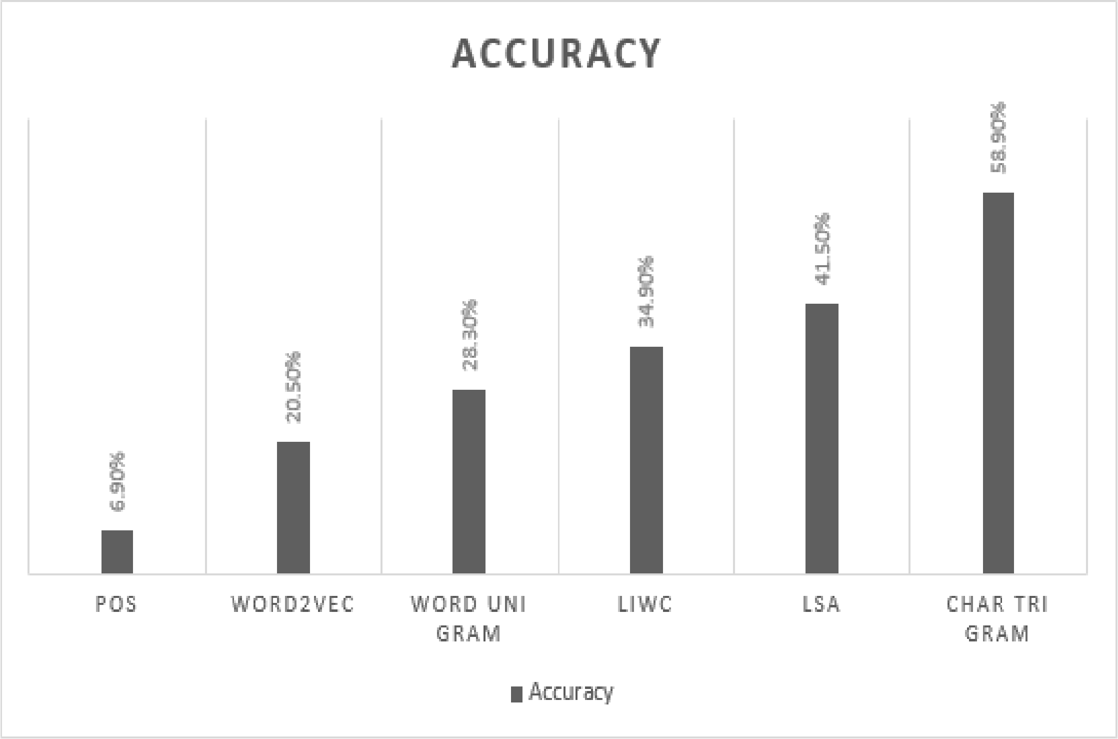


FIGURE 4.21: Standalone Feature Performance Using Accuracy

4.4.1.2 Standalone Feature Performance Using Area Under Curve (AUC):

Figure 4.22 illustrates the results of all selected features with respect to Area under Curve (AUC). As it is discussed earlier that only Random Forest is implemented as a machine learning model. Following graphical representation depicts that the selected features of Char-Tri-Gram outperformed among all other selected features set with 78.40 % AUC score and LSA show second best performance with 70.30 % AUC score. Due to a slight difference of 2.7 % between LIWC and Word-Uni-Gram features, LIWC score third place among all other selected features with 67 % AUC score. Whereas all remaining selected features including Word-Uni-Gram, Word2vec and Part-of-Speech show a gradual decrease in the performance with 61.70 %, 59.70 % and 52.90 % AUC score respectively. All these representations depicting that, cumulatively Char-Tri-Gram out performed among all other selected features.

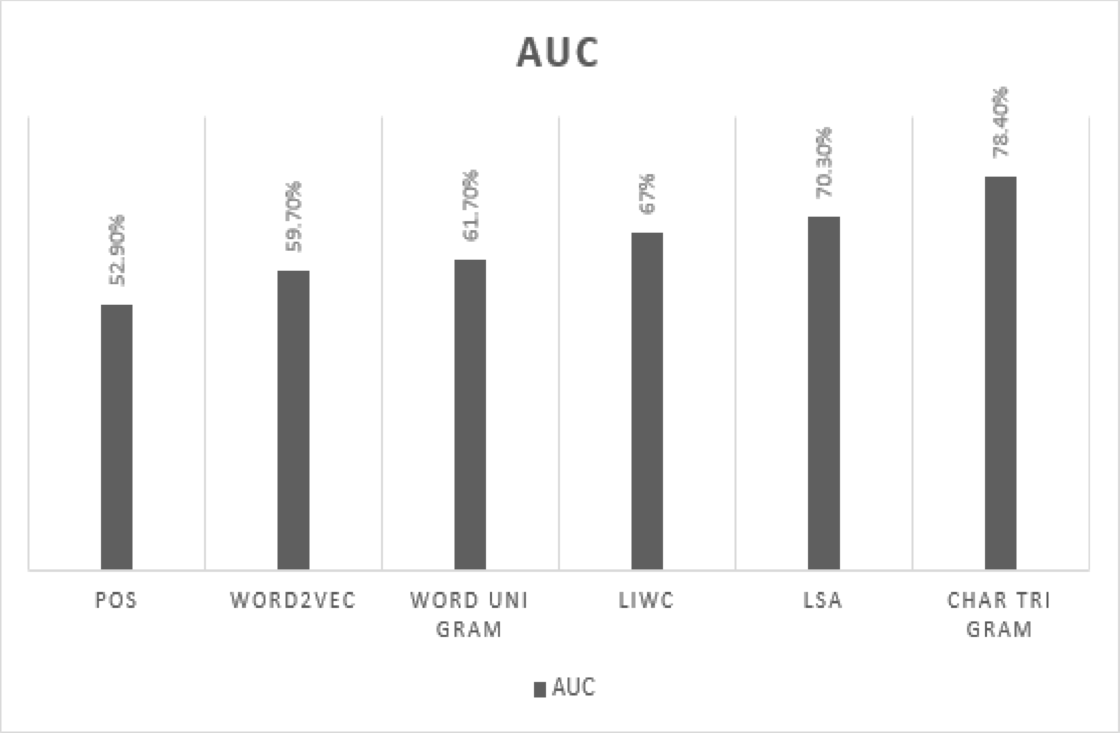


FIGURE 4.22: Standalone Feature Performance Using AUC

4.4.1.3 Standalone Feature Performance Using F-Measure:

Figure 4.23 illustrates the results of all selected features with respect to F-Measure. As it is discussed earlier that only Random Forest is implemented as a machine learning model. Following graphical representation depicts that the selected features of Char-Tri-Gram outperformed among all other selected features set with 67.10 % F-Measure score and LSA show second best performance with 56.10 % f-Measure score. Due to a difference of 15.50 % between LIWC and Word-Uni-Gram features, LIWC score third place among all other selected features with 49.50 % F-Measure score. Whereas all remaining selected features including Word-Uni-Gram, Word2vec and Part-of-Speech show a gradual decrease in the performance with 34 %, 31.70 % and 12 % F-Measure score respectively. All these representations depicting that, cumulatively Char-Tri-Gram out performed among all other selected features.

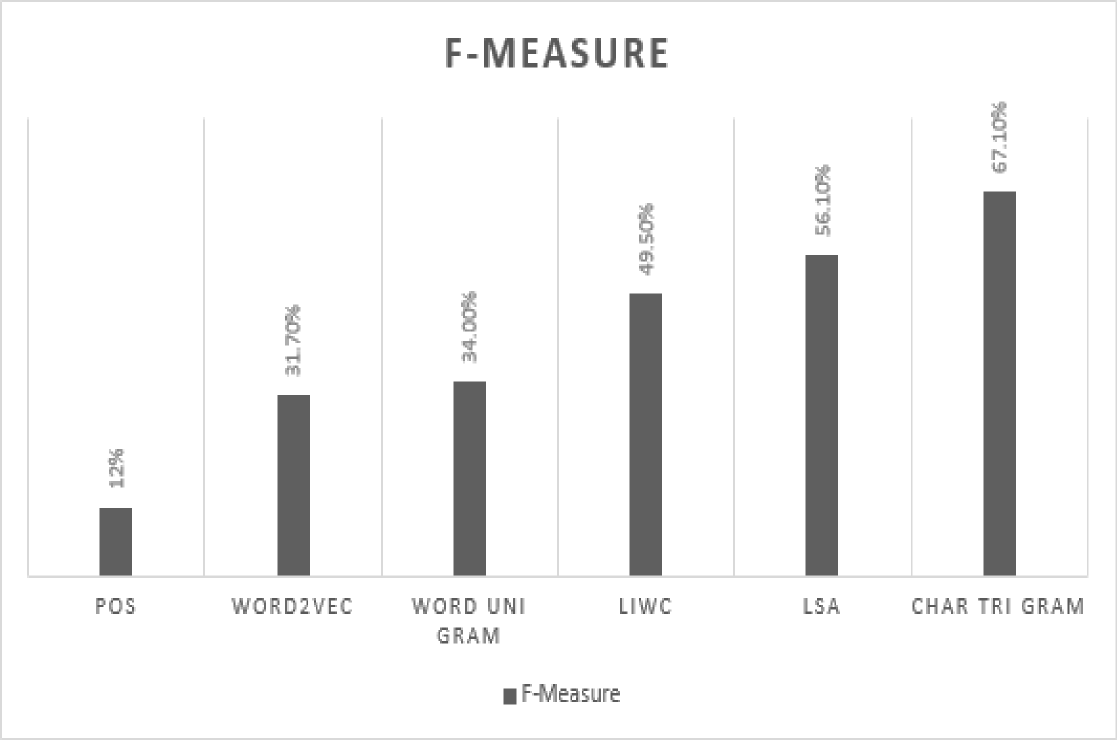


FIGURE 4.23: Standalone Feature Performance Using F-Measure

4.4.1.4 Standalone Feature Performance Using Recall:

Figure 4.24 illustrates the results of all selected features with respect to Recall. As it is discussed earlier that only Random Forest is implemented as a machine learning model. Following graphical representation depicts that the selected features of Char-Tri-Gram outperformed among all other selected features set with 59 % recall score and LSA show second best performance with 41.6 % recall score. Due to a difference of 7.7 % between LIWC and Word-Uni-Gram features, LIWC score third place among all other selected features with 35 % recall score. Whereas all remaining selected features including Word-Uni-Gram, Word2vec and Part-of-Speech show a gradual decrease in the performance with 28.30 %, 20.50 % and 7 % recall score respectively. All these representations depicting that, cumulatively Char-Tri-Gram out performed among all other selected features.

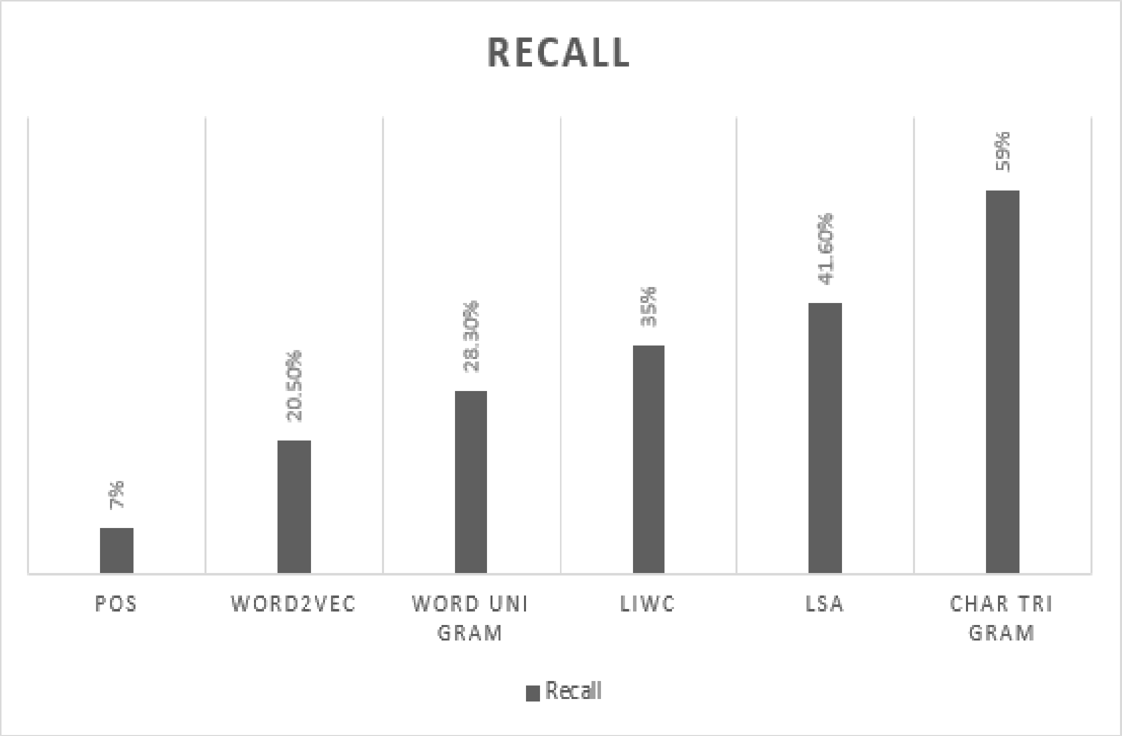


Figure 4.24: Standalone Feature Performance Using Recall

4.4.1.5 Standalone Feature Performance Using Precision:

Figure 4.25 illustrates the results of all selected features with respect to Precision. As it is discussed earlier that only Random Forest is implemented as a machine learning model. Following graphical representation depicts that the selected features of LSA outperformed among all other selected features set with 86.30 % precision score and LIWC show second best performance with 84.80 % precision score. Due to a difference of 8 % between Char-Tri-Gram and Word2vec features, Char-Tri-Gram score third place among all other selected features with 77.80 % precision score. Whereas all remaining selected features including Word2vec, Word-Uni-Gram, and Part-of-Speech show a gradual decrease in the performance with 69.80 %, 43.50 % and 42.40 % precision score respectively. All these representations depicting that, cumulatively Char-Tri-Gram out performed among all other selected features.

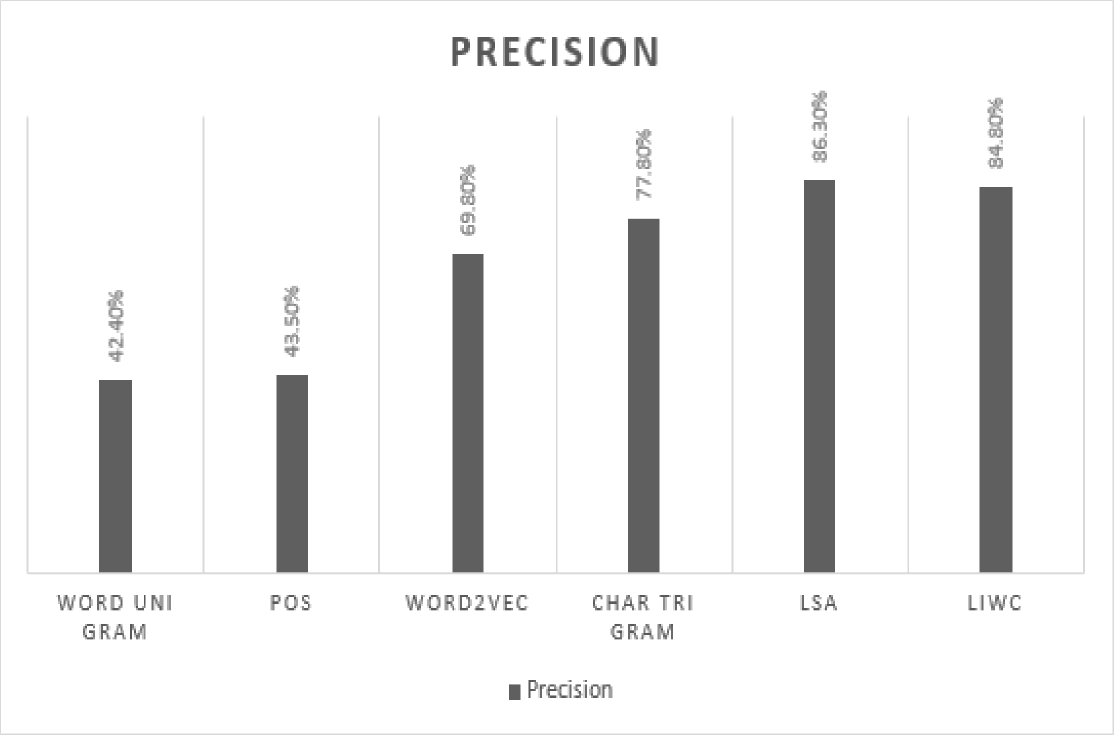


FIGURE 4.25: Standalone Feature Performance Using Precision

4.4.1.6 Two-Feature Set Performance Using Accuracy:

Figure 4.26 illustrates the results of all selected two-features set with respect to Accuracy. It is obvious from following representation that all selected two-feature sets have very impressive impact when we applied Random Forest as a machine learning model. But if we discussed individually then, Char-Tri-Gram + POS outperformed among all other selected two-features set with 62.40 % Accuracy, whereas Char-Tri-Gram + LIWC and Char-Tri-Gram + LSA show second and third best performance yielding 61.30 % and 60.60 % Accuracy score. Whereas Char-Tri-Gram + Word2vec shows a least impact with 60.10 % Accuracy score. All these representations depicting that, cumulatively Char-Tri-Gram + POS is a best feature among all other selected two-features set.

4.4.1.7 Two-Feature Set Performance Using Area Under Curve (AUC):

Figure 4.27 illustrates the results of all selected two-features set with respect to Area under Curve (AUC). It is obvious from following representation that all

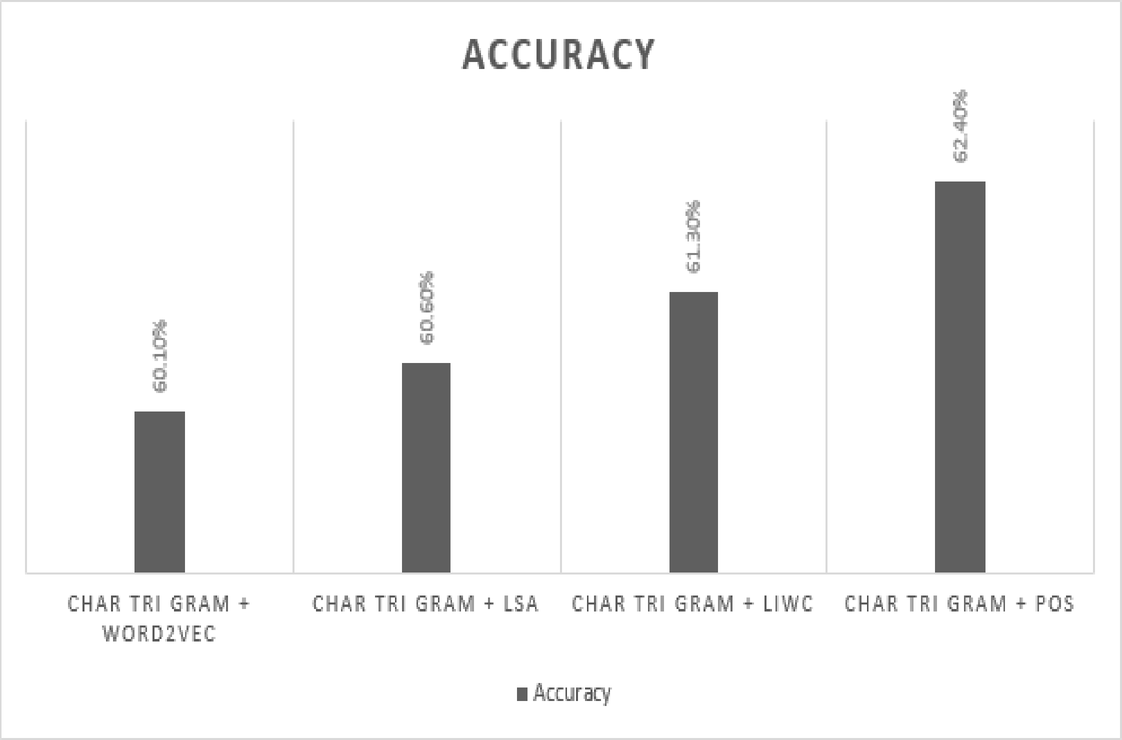


Figure 4.26: Two-Feature Set Performance Using Accuracy

selected two-feature sets have very impressive impact when we applied Random Forest as a machine learning model. But if we discussed individually then, Char-Tri-Gram + POS outperformed among all other selected two-features set with 80.50 % AUC , whereas Char-Tri-Gram + LIWC and Char-Tri-Gram + LSA show second and third best performance yielding 80.10 % and 79.80 % AUC score. Whereas Char-Tri-Gram + Word2vec shows a least impact with 79.30 % AUC score. All these representations depicting that, cumulatively Char-Tri-Gram + POS is a best feature among all other selected two-features set.

4.4.1.8 Two-Feature Set Performance Using F-Measure:

Figure 4.28 illustrates the results of all selected two-features set with respect to F-Measure. It is obvious from following representation that all selected two-feature sets have very impressive impact when we applied Random Forest as a machine learning model. But if we discussed individually then, Char-Tri-Gram + POS and Char-Tri-Gram + LSA outperformed among all other selected two-features set

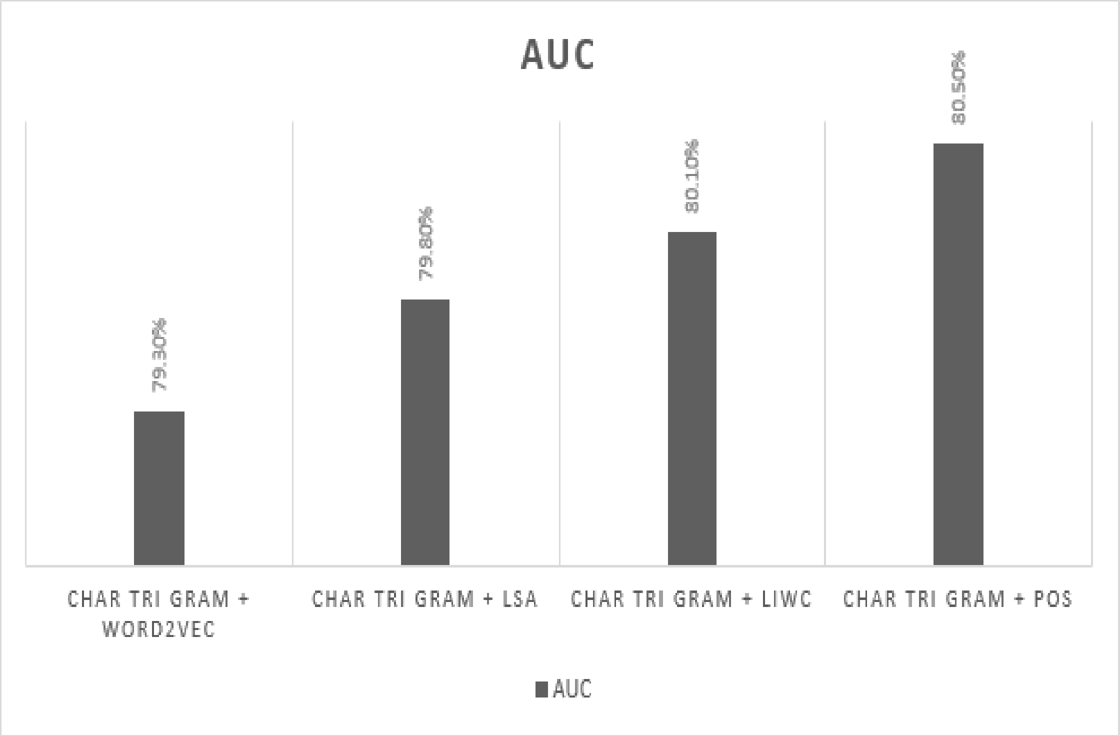


Figure 4.27: Two-Feature Set Performance Using AUC

with 71.90 % F-Measure, whereas Char-Tri-Gram + LIWC show second best performance yielding 71.70 % F-Measure score. Whereas Char-Tri-Gram + Word2vec shows a least impact with 69.90 % F-Measure score. All these representations depicting that, cumulatively Char-Tri-Gram + POS and Char-Tri-Gram + LSA are best features among all other selected two-features set.

4.4.1.9 Two-Feature Set Performance Using Recall:

Figure 4.29 illustrates the results of all selected two-features set with respect to Recall. It is obvious from following representation that all selected two-feature sets have very impressive impact when we applied Random Forest as a machine learning model. But if we discussed individually then, Char-Tri-Gram + POS outperformed among all other selected two-features set with 62.40 % recall , whereas Char-Tri-Gram + LIWC and Char-Tri-Gram + LSA show second and third best performance yielding 61.40 % and 60.70 % recall score. Whereas Char-Tri-Gram

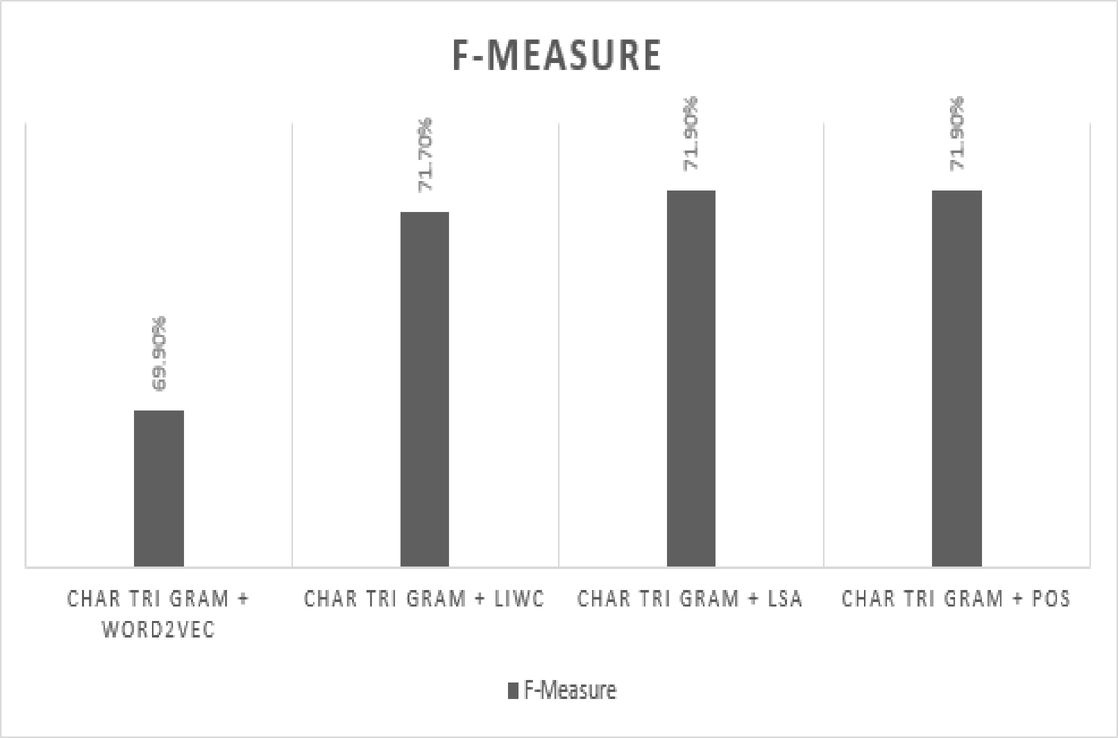


Figure 4.28: Two-Feature Set Performance Using F-Measure

+ Word2vec shows a least impact with 60.20 % recall score. All these representations depicting that, cumulatively Char-Tri-Gram + POS is a best feature among all other selected two-features set.

4.4.1.10 Two-Feature Set Performance Using Precision:

Figure 4.30 illustrates the results of all selected two-features set with respect to precision. It is obvious from following representation that all selected two-feature sets have very impressive impact when we applied Random Forest as a machine learning model. But if we discussed individually then, Char-Tri-Gram + LSA outperformed among all other selected two-features set with 88.10 % precision , whereas Char-Tri-Gram + LIWC and Char-Tri-Gram + POS show second and third best performance yielding 86.30 % and 84.70 % precision score. Whereas Char-Tri-Gram + Word2vec shows a least impact with 83.40 % precision score. All these representations depicting that, cumulatively Char-Tri-Gram + LSA is a best feature among all other selected two-features set.

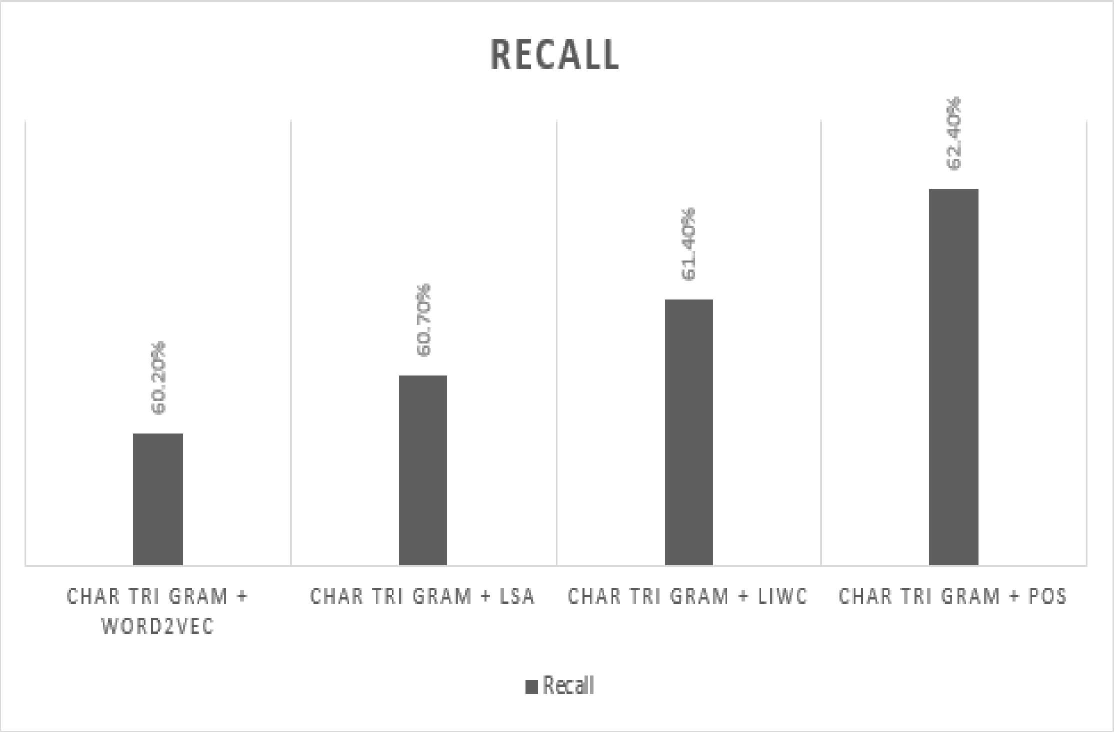


Figure 4.29: Two-Feature Set Performance Using Recall

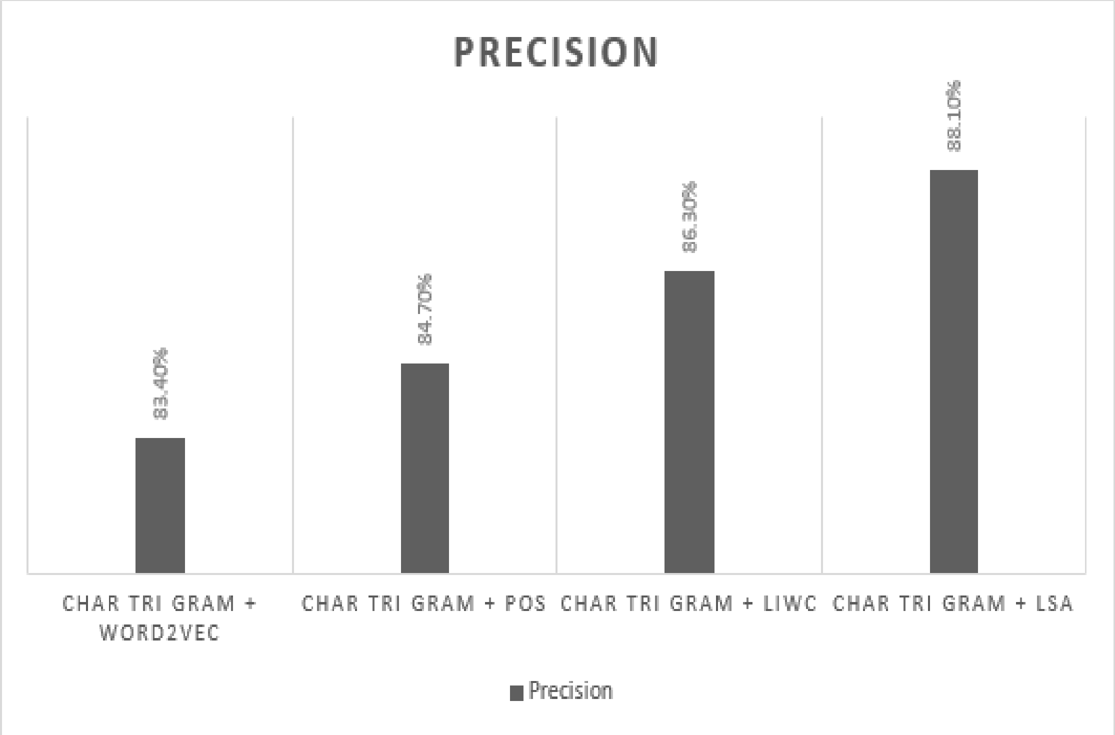


Figure 4.30: Two-Feature Set Performance Using Precision

After the analysis of both features (filter based features wrapper based features), It is obvious that filter based features are lacking from wrapper based features with respect to all evaluation metrics. This analysis represents that wrapper based features are more impactful than filter based features. So for further analysis we use only wrapper based features.

4.4.2 Wrapper Based Features Analysis:

In the second part of this experiment, features are extracted by wrapper method using forward feature selection technique. We implement this technique in a very well know data mining tool Weka which extracted 14 word2vec, 19 LSA, 2 LIWC, 1 POS and 14 Char Tri Gram based features for further analysis.

4.4.2.1 Standalone Feature Performance Using Accuracy:

Figure 4.31 illustrates the results of all selected features with respect to Accuracy. As it is discussed earlier that only Random Forest is implemented as a machine learning model. Following graphical representation depicts that the selected features of Char-Tri-Gram outperformed among all other selected features set with 94.60 % Accuracy score and LSA show second best performance with 93.40 % Accuracy score. Due to a slight difference of 1.4 % between LIWC and Word2vec features, LIWC score third place among all other selected features with 91.20 % Accuracy score. Whereas all remaining selected features including Word2vec and Part-of-Speech show a gradual decrease in the performance with 90.80 % and 89 % Precision score respectively. All these representations depicting that, cumulatively Char-Tri-Gram out performed among all other selected features.

4.4.2.2 Standalone Feature Performance Using AUC:

Figure 4.32 illustrates the results of all selected features with respect to Area under Curve (AUC). As it is discussed earlier that only Random Forest is implemented

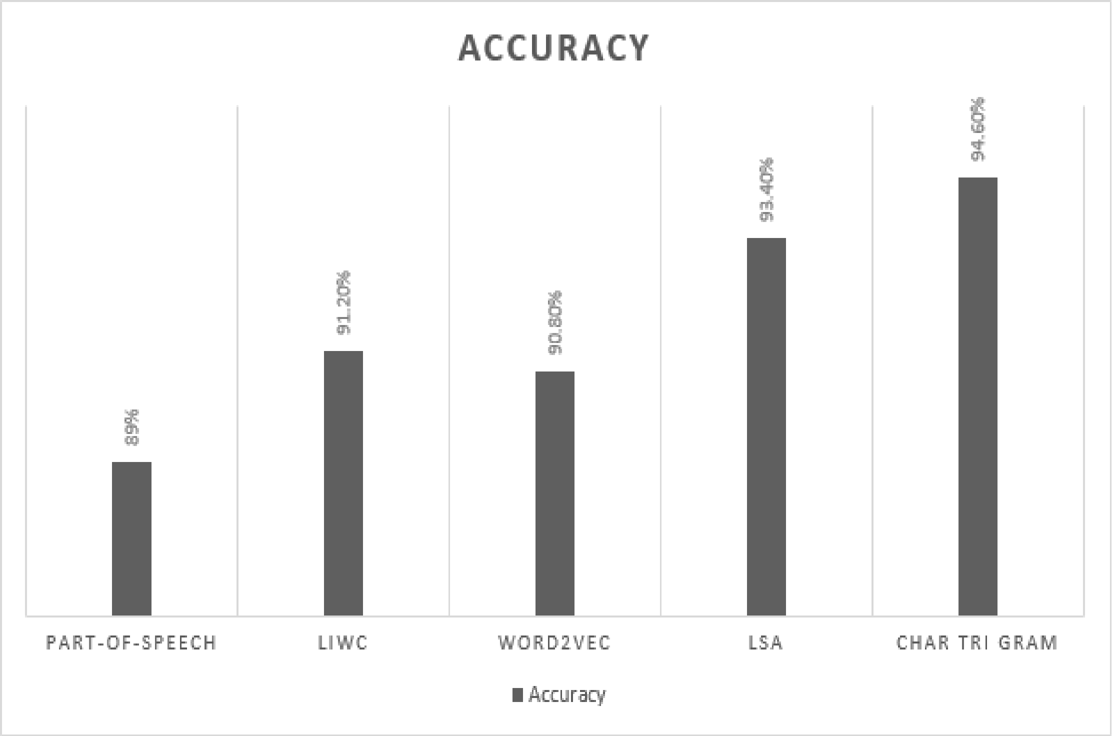


FIGURE 4.31: Standalone Feature Performance Using Accuracy

as a machine learning model. Following graphical representation depicts that the selected features of Char-Tri-Gram outperformed among all other selected features set with 92.40 % AUC score and LSA show second best performance with 72.60 % AUC score. Due to a slight difference of 1.2 % between LIWC and Word2vec features, LIWC score third place among all other selected features with 63.70 % AUC score. Whereas all remaining selected features including Word2vec and Part-of-Speech show a gradual decrease in the performance with 62.50 % and 53.90 % AUC score respectively. All these representations depicting that, cumulatively Char-Tri-Gram out performed among all other selected features.

4.4.2.3 Standalone Feature Performance Using F-Measure:

Figure 4.33 illustrates the results of all selected features with respect to F-Measure. As it is discussed earlier that only Random Forest is implemented as a machine learning model. Following graphical representation depicts that the selected features of Char-Tri-Gram outperformed among all other selected features set with

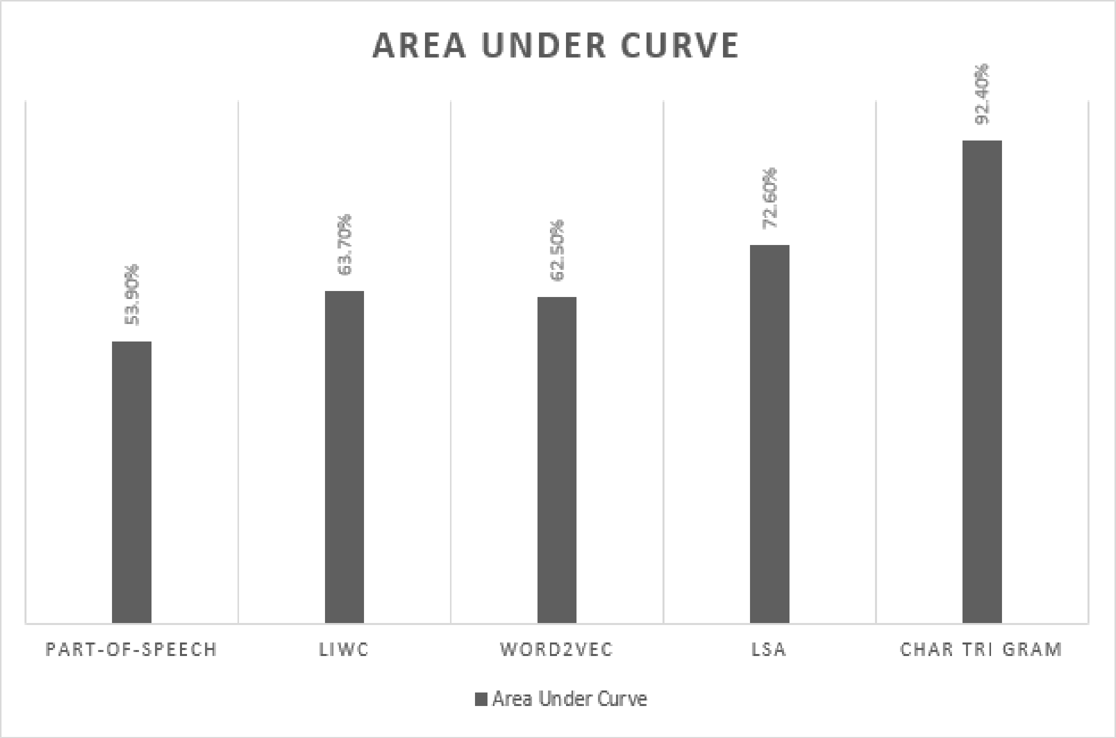


FIGURE 4.32: Standalone Feature Performance Using Area Under Curve

87.40 % F-Measure score and LSA show second best performance with 61 % F-Measure score. Whereas all remaining selected features including LIWC, Word2vec and Part-of-Speech show a gradual decrease in the performance with 46.80 %, 38.90 % and 15.30 % F-Measure score respectively. All these representations depicting that, cumulatively Char-Tri-Gram out performed among all other selected features.

4.4.2.4 Standalone Feature Performance Using Recall:

Figure 4.34 illustrates the results of all selected features with respect to Recall. As it is discussed earlier that only Random Forest is implemented as a machine learning model. Following graphical representation depicts that the selected features of Char-Tri-Gram outperformed among all other selected features set with 86.80 % Recall score and LSA show second best performance with 45.90 % Recall score. Whereas all remaining selected features including LIWC, Word2vec and Part-of-Speech show a gradual decrease in the performance with 30.30 %, 26.30

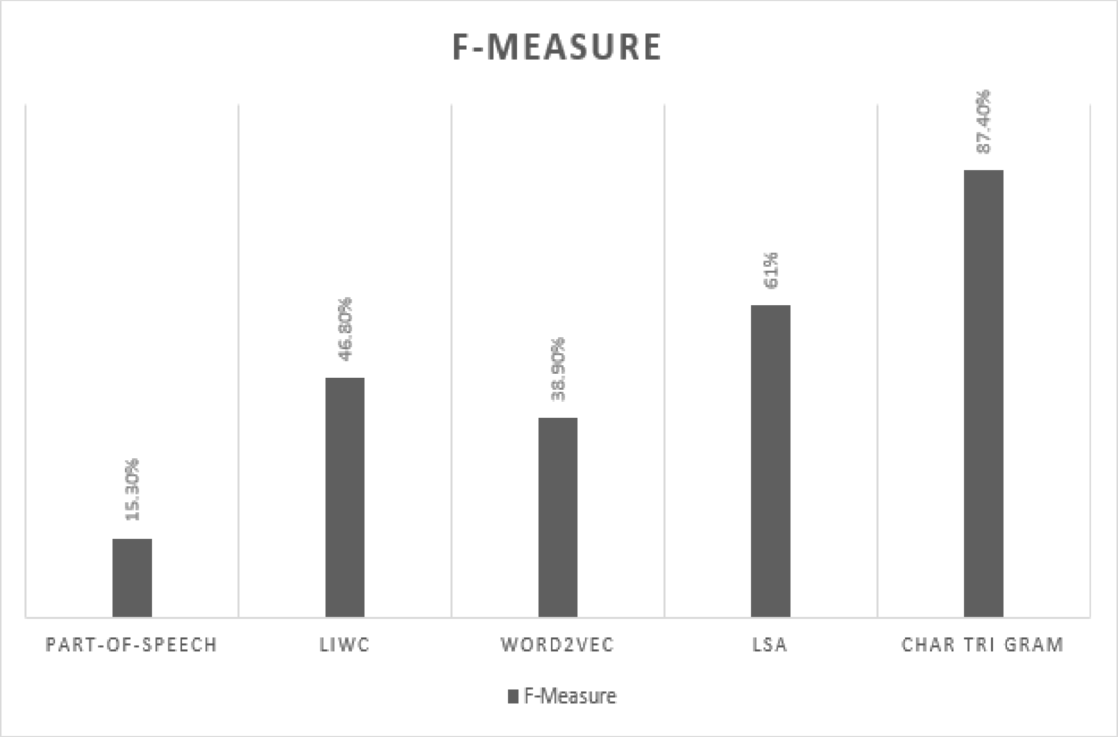


FIGURE 4.33: Standalone Feature Performance Using F-Measure

% and 8.9 % Recall score respectively. All these representations depicting that, cumulatively Char-Tri-Gram out performed among all other selected features.

4.4.2.5 Standalone Feature Performance Using Precision:

Figure 4.35 illustrates the results of all selected features with respect to Precision. As it is discussed earlier that only Random Forest is implemented as a machine learning model. Following graphical representation depicts that the selected features of LSA outperformed among all other selected features set with 91 % Precision score and Char-Tri-Gram show second best performance with 89.10 % Precision score. Whereas all remaining selected features including LIWC, Word2vec and Part-of-Speech show a gradual decrease in the performance with 87.50 %, 75.10 % and 55.40 % Precision score respectively. All these representations depicting that, cumulatively Char-Tri-Gram out performed among all other selected features.

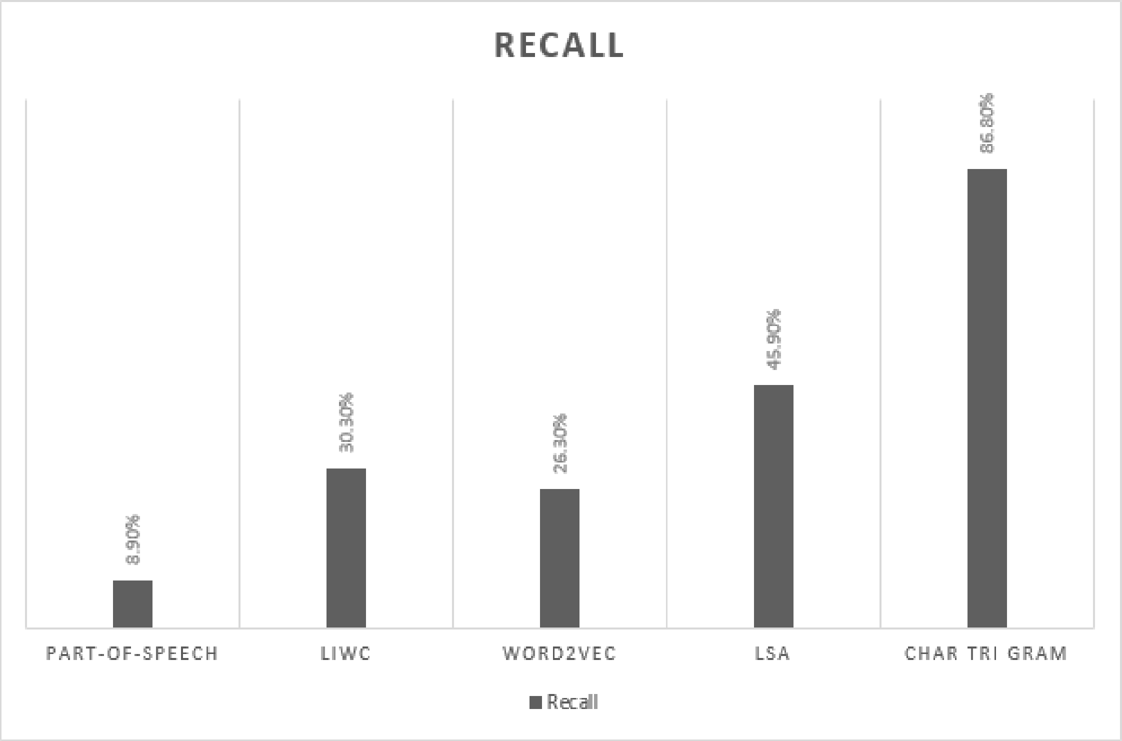


FIGURE 4.34: Standalone Feature Performance Using Recall

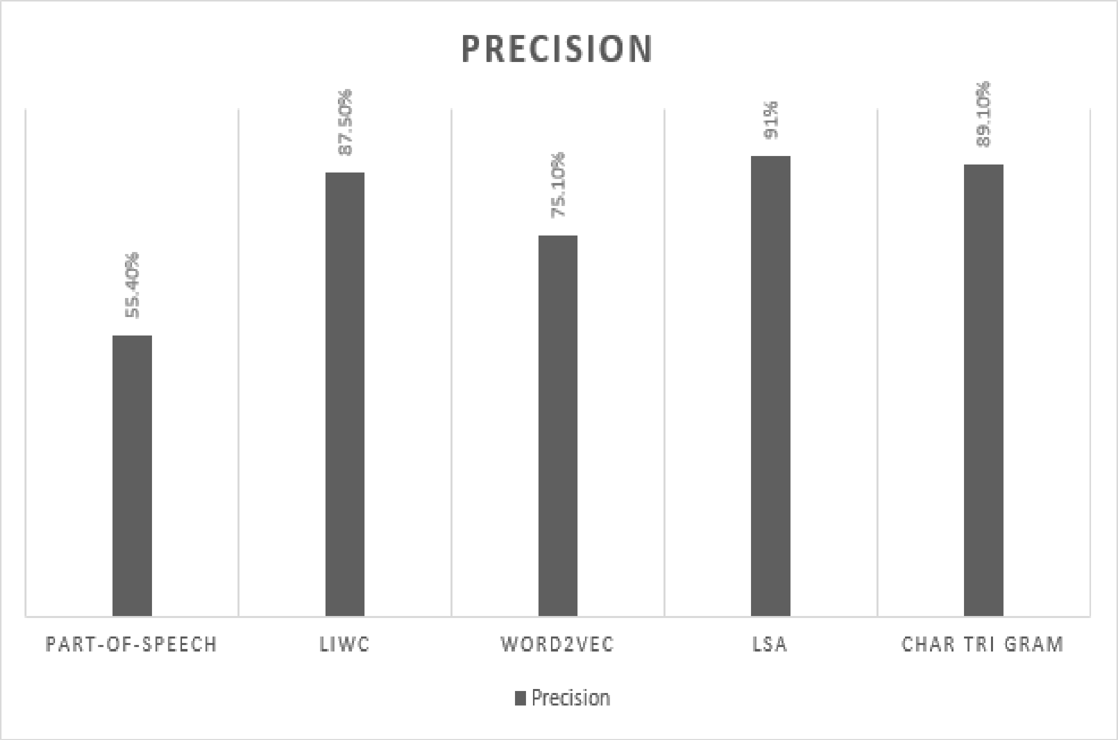


FIGURE 4.35: Standalone Feature Performance Using Precision

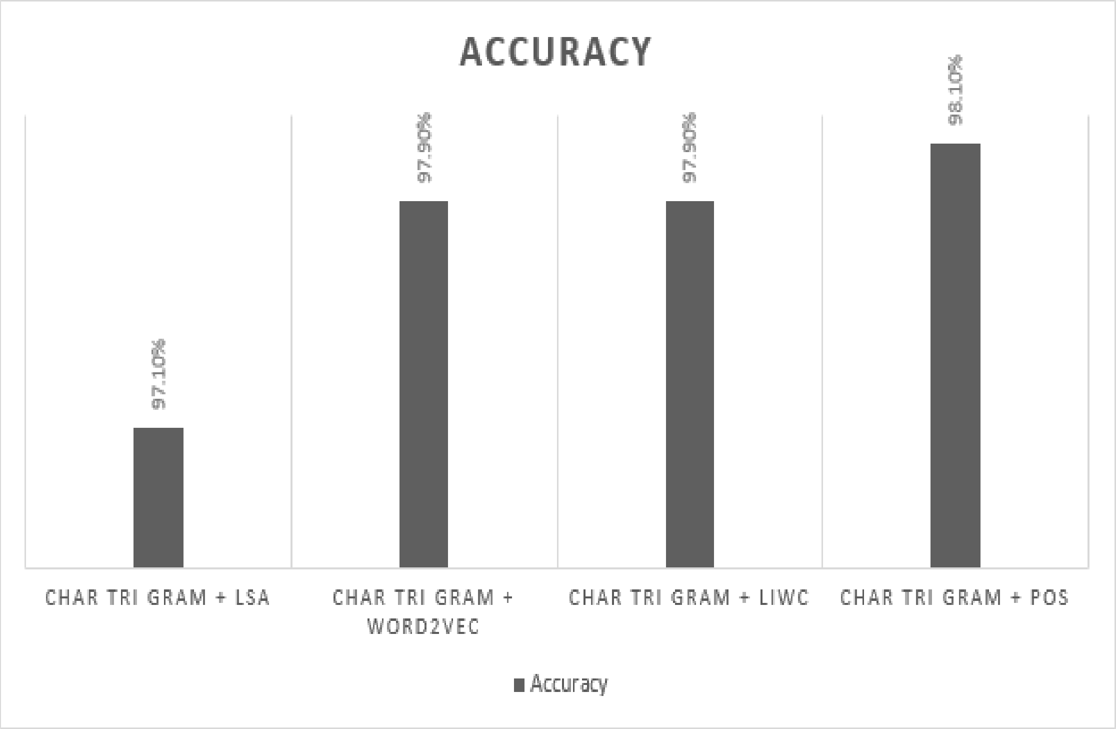


FIGURE 4.36: Standalone Feature Performance Using Accuracy

4.4.2.6 Two-Feature Set Performance Using Accuracy:

Figure 4.36 illustrates the results of all selected two-features set with respect to Accuracy. It is obvious from following representation that all selected two-feature sets have very impressive impact when we applied Random Forest as a machine learning model. But if we discussed individually then, Char-Tri-Gram + POS outperformed among all other selected two-features set with 98.10 % Accuracy, whereas Char-Tri-Gram + Word2vec and Char-Tri-Gram + LIWC show second best performance yielding 97.90 % Accuracy score. Whereas Char-Tri-Gram + LSA shows a least impact with 97.10 % Accuracy score. All these representations depicting that, cumulatively Char-Tri-Gram + POS is a best feature among all other selected two-features set.

4.4.2.7 Two-Feature Set Performance Using AUC:

Figure 4.37 illustrates the results of all selected two-features set with respect to Area under Curve (AUC). It is obvious from following representation that all

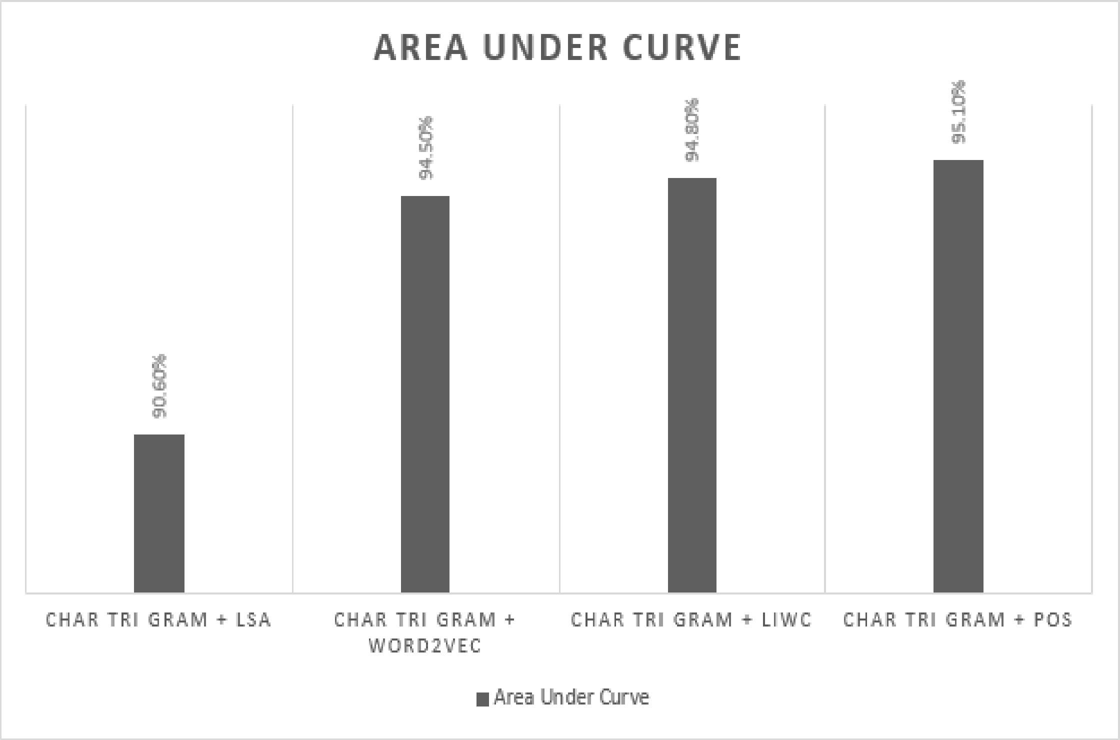


FIGURE 4.37: Standalone Feature Performance Using Area Under Curve

selected two-feature sets have very impressive impact when we applied Random Forest as a machine learning model. But if we discussed individually then, Char-Tri-Gram + POS outperformed among all other selected two-features set with 95.10 % AUC and Char-Tri-Gram + LIWC shows second best performance with 94.80 % AUC score. Due to a slight difference of 0.30 % between Char-Tri-Gram + LIWC and Char-Tri-Gram + Word2vec, Char-Tri-Gram + Word2vec score third place among all other selected two-features set with 94.50 % AUC score. Whereas Char-Tri-Gram + LSA shows a least impact with 90.60 % Accuracy score. All these representations depicting that, cumulatively Char-Tri-Gram + POS is a best feature among all other selected two-features set.

4.4.2.8 Two-Feature Set Performance Using F-Measure:

Figure 4.38 illustrates the results of all selected two-features set with respect to F-Measure. It is obvious from following representation that all selected two-feature sets have very impressive impact when we applied Random Forest as a machine

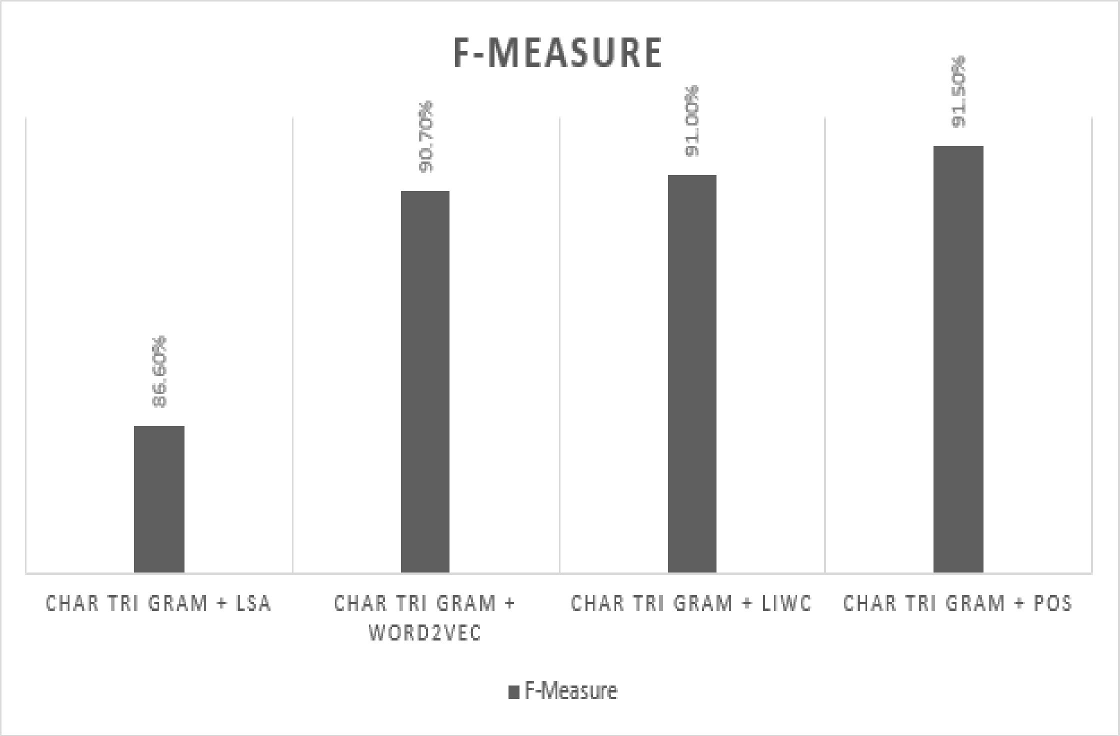


FIGURE 4.38: Standalone Feature Performance Using F-Measure

learning model. But if we discussed individually then, Char-Tri-Gram + POS outperformed among all other selected two-features set with 91.50 % F-Measure and Char-Tri-Gram + LIWC shows second best performance with 91 % F-Measure score. Due to a slight difference of 0.70 % between Char-Tri-Gram + LIWC and Char-Tri-Gram + Word2vec, Char-Tri-Gram + Word2vec score third place among all other selected two-features set with 90.70 % F-Measure score. Whereas Char-Tri-Gram + LSA shows a least impact with 86.60 % F-Measure score. All these representations depicting that, cumulatively Char-Tri-Gram + POS is a best feature among all other selected two-features set.

4.4.2.9 Two-Feature Set Performance Using Recall:

Figure 4.39 illustrates the results of all selected two-features set with respect to Recall. It is obvious from following representation that all selected two-feature sets have very impressive impact when we applied Random Forest as a machine learning model. But if we discussed individually then, both Char-Tri-Gram +

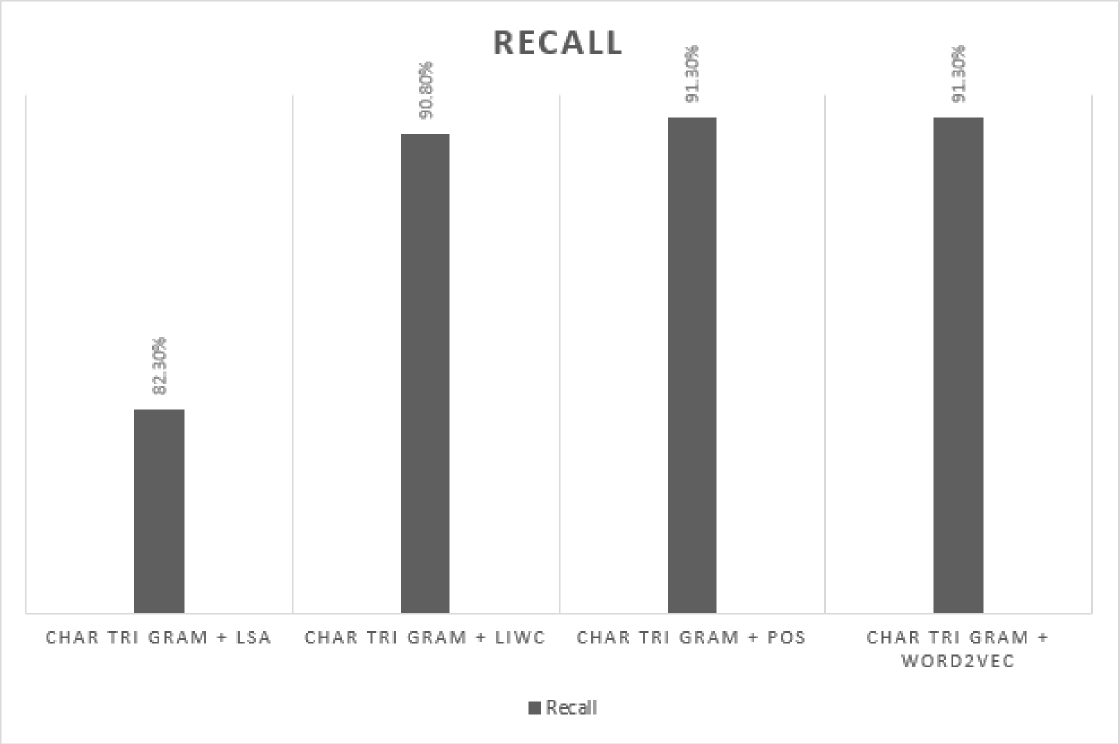


FIGURE 4.39: Standalone Feature Performance Using Recall

POS and Char-Tri-Gram + Word2vec outperformed among all other selected two-features set with 91.30 % Recall score and Char-Tri-Gram + LIWC shows second best performance with 90.80 % Recall score. Whereas Char-Tri-Gram + LSA shows a least impact with 82.30 % F-Measure score. All these representations depicting that, cumulatively Char-Tri-Gram + POS is a best feature among all other selected two-features set.

4.4.2.10 Two-Feature Set Performance Using Precision:

Figure 4.40 illustrates the results of all selected two-features set with respect to Precision. It is obvious from following representation that all selected two-feature sets have very impressive impact when we applied Random Forest as a machine learning model. But if we discussed individually then, Char-Tri-Gram + POS outperformed among all other selected two-features set with 92.60 % Precision and Char-Tri-Gram + Word2vec shows second best performance with 91.40 % Precision score. Due to a slight difference of 0.10 % between Char-Tri-Gram +

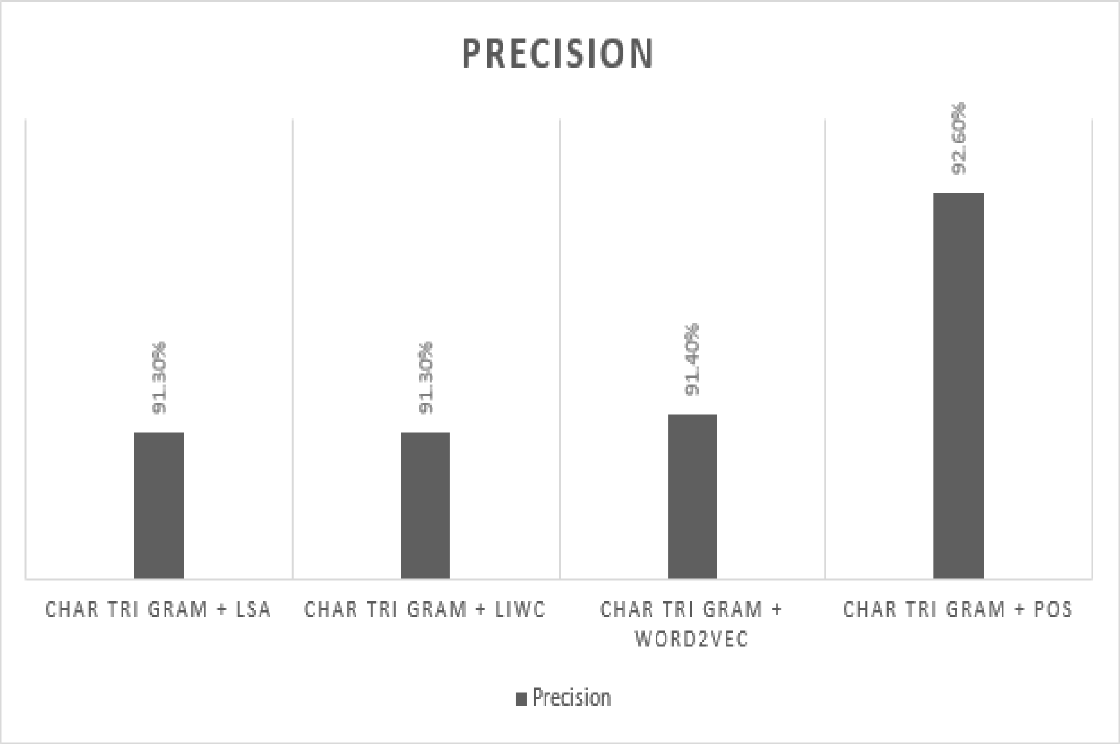


Figure 4.40: Standalone Feature Performance Using Recall

Word2vec, Char-Tri-Gram + LIWC and Char-Tri-Gram + LSA, Char-Tri-Gram + LSA and Char-Tri-Gram + LIWC score third place among all other selected two-features set with the same representation of 91.30 % Precision score. All these representations depicting that, cumulatively Char-Tri-Gram + POS is a best feature among all other selected two-features set.

Chapter 5

Conclusion and Future Work

Distinguishing task between propaganda and non-propaganda news articles was evolved from 20th century, but it emerges prominently present era because now every person have access to social media very easily. Now it is very interesting field for researchers because it has lot of dimensions. To evaluate this task different ML models and techniques are implemented by researchers but it's hard to convey that which one is robust and effective. In our research, we perform same analysis by implementing linguistic and stylometric features and their combinations, and eventually out results demonstrates that hybrid features outperform for this task.

5.1 Conclusion

In this thesis we performed a critical analysis on our experimental results for propaganda detection at the news article level obtained from Dev partition of proppy dataset. Our experimental results show that different representations modeling techniques are more effective than word n-grams. Our experimental results depicted contradiction against existing models and corroborates this hypothesis: models that consider stylistic features, such as character Tri-grams, LSA, Word2Vec and their combinations always outperform alternative representations, which are typically used in topic-related tasks. Different from previous approaches, we use wrapper method for best features extraction and then present two type of features set (i) each stand-alone feature, (ii) the hybrid features. After analysis on these stand alone as well as hybrid features extracted by wrapper method, We believe that this wrapper method technique is valuable for further research on propaganda detection, and that it will be also considered by the research community. Thus our proposed features are the effective indicators for the propaganda detection at the news article level. As and aspect of cost analysis for both classes (propaganda vs non-propaganda) from prior discussion in this research it is obvious that, a propaganda instance is detected as non-propaganda is much costly rather than a non-propaganda instance is detected as propaganda.Which reveals that, propaganda detection needs more attention rather than a non-propaganda detection

5.2 Future Work

This research can be further extended in multiple levels. Researchers will focus for the implementation of advance algorithms Fast Text and LDA to investigate their impacts on propaganda detection at the news article level. This research could be enhanced by implementation of semantic and network features.

Bibliography

[1] Mehdi Allahyari , Seyedamin Pouriyeh , Mehdi Assef, Saeid Safaei , Elizabeth D. Trippe , Juan B. Gutierrez , Krys Kochut. "Text Summarization Techniques: A Brief Survey" , (2017).

[2] Petter Bae Brandtzaeg , Asbjørn Følstad . "Why people use chatbots" , (2017).

[3] Matt Carlson. "The robotic reporter: Automated journalism and the redefinition of labor, compositional forms, and journalistic authority," Digit. Journalism" , (2015).

[4] V. Lysenko , C. Brooks. "Russian information troops, disinformation,and democracy." https://firstmonday.org/ojs/index.php/fm/article/view/8176 , (2015).

[5] D. Flick. "Combatting fake news: Alternatives to limiting social media misinformation and rehabilitating quality journalism." https://scholar.smu.edu/scitech/vol20/iss2/17/ , (2017).

[6] G.L.Freed , S.J.Clark, A.T.Butchart, D.C.Singer, and M.M.Davis. "Parental vaccine safety concerns in 2009." https://pubmed.ncbi.nlm.nih.gov/20194286/, (2010).

[7] D.J.Collison. "Corporate propaganda: Its implications for accounting and accountability."https://discovery.dundee.ac.uk/en/publications/corporate-propaganda-its-implications-for-accounting-and-accounta, (2003).

[8] S.D.Benegal , L.A.Scruggs. "Correcting misinformation about climate change: The impact of partisanship in an experimental

setting.”https://ideas.repec.org/a/spr/climat/v148y2018i1d10.1007$_{s}10584 - 018 - 2192 - 4.html$, (2018).

[9] H. Cantril. ”Propaganda analysis.”https://doi.org/10.2307/806063, (1938).

[10] G.S.Jowett , V.O’donnell. ”Propaganda and Persuasion.”https://www.amazon.com/Propaganda-Persuasion-Garth-S-Jowett/dp/1412977827, (2018).

[11] J.Ellul , K.Kellen. ”Propaganda: The Formation Men’s Attitudes. ”https://en.wikipedia.org/wiki/Propaganda:$_{T}he_{F}ormation_{o}f_{M}en27s_{A}ttitudes$, (1973).

[12] W.J.Severin , J.W.Tankard. ”Communication Theories: Origins, Methods, and Uses in the Mass Media.”https://www.amazon.com/Communication-Theories-Origins-Methods-Media/dp/0801333350, (1997).

[13] J. Mayer. ”How Russia Helped Swing the Election for Trump.” ”https://www.newyorker.com/magazine/2018/10/01/how-russia-helped-to-swing-the-election-for-trump, (2018).

[14] C. Cadwalladr , E. Graham-Harrison. ”Revealed: 50 Million Facebook Profiles Harvested for Cambridge Analytica in Major Data Breach.” ”https://www.theguardian.com/technology/2018/mar/17/facebook-cambridge-analytica-kogan-data-algorithm, (2018).

[15] R. DiResta. ”The tactics and tropes of the Internet research agency” https://digitalcommons.unl.edu/senatedocs/2/, (2019).

[16] F. Carmichael , A. Hussain. ”Pro-Indian ’Fake Websites Targeted Decision Makers in Europe’-BBC News”. https://www.bbc.com/news/world-asia-india-50749764, (2019).

[17] W. Ahmed, J. Vidal-Alaball, J. Downing, F. López Seguí. ”COVID-19 and the 5G Conspiracy Theory: Social Network Analysis of Twitter Data”. https://www.jmir.org/2020/5/e19458/, (2020).

[18] T. Warren. "British 5G Towers are Being Set on Fire Because of Coronavirus Conspiracy Theories". https://www.theverge.com/2020/4/4/21207927/5g-towers-burning-uk-coronavirus-conspiracy-theory-link, (2020).

[19] A. Aly, S. Macdonald, L. Jarvis, T. M. Chen. "Introduction to the Special Issue: Terrorist Online Propaganda and Radicalization". https://www.tandfonline.com/doi/abs/10.1080/1057610X.2016.1157402, (2016).

[20] J. Qin, Y. Zhou, H. Chen. "A multi-region empirical study on the internet presence of global extremist organizations". https://arizona.pure.elsevier.com/en/publications/a-multi-region-empirical-study-on-the-internet-presence-of-global, (2011).

[21] R. A. Jawad. "RAW boosts funding in Balochistan". https://nation.com.pk/05-Sep-2015/raw-boosts-funding-in-balochistan, (2015).

[22] Z. Ebrahim. "World Health Day-Pakistan: Anti-Polio Drive Hits Resistance." http://www.ipsnews.net/2007/04/world-health-day-pakistan-anti-poliodrive-hits-resistance/, (2007).

[23] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." https://www.aclweb.org/anthology/N19-1423/, (2019).

[24] Analysis, Institute for Propaganda. "Propaganda Analysis: Volume I of the Publications of the Institute for Propaganda Analysis."https://archive.org/stream/IPAVol1/IPA$_vol1_djvu.txt$, (1938).

[25] Diana Tal , Avishag Gordon. "Propaganda as a research field: a bibliometric study." https://link.springer.com/article/10.1007/s11192-019-03298-3 , 2019.

[26] R.Mueller. "Indictment of Internet Research Agency ."https://www.justice.gov/file/1035477/download , 2018.

[27] "https://zenodo.org/record/3271522.YAVDMnZKjIV"

[28] Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, Yejin Choi. "Truth of Varying Shades: Analyzing Language in Fake News and Political Fact-Checking" https://homes.cs.washington.edu/ eunsol/papers/factcheck$_e mnlp17.pdf$, 2017.

[29] Alberto Barrón-Cedeño, Israa Jaradat, Giovanni Da San Martino, Preslav Nakov. "Proppy: Organizing the news based on their propagandistic content." https://www.sciencedirect.com/science/article/abs/pii/S0306457318306058 (2019).

[30] A. Barrón-Cedeño, I. Jaradat, G. Da San Martino and P. Nakov, https://zenodo.org/record/3271522.YGyhyx9KjIV , 2019.

[31] E. Atwell, "Development of tag sets part-of-speech tagging," https://www.researchgate.net/publication/268523010, 2008.

[32] J. W. Pennebaker, C. K. Chung, M. Ireland, A. Gonzales and R. J. Booth, "The Development and Psychometric Properties of," https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.600.7227rep=rep1type=pdf, p. 22, 2007.

[33] L. Ma and Y. Zhang , "Using Word2Vec to Process Big Text Data," https://www.researchgate.net/publication/291153115, 2015.

[34] P. W. FOLTZ, "Latent semantic analysis for text-based research," https://link.springer.com/content/pdf/10.3758/BF03204765.pdf, p. 6, 1996.

[35] "Language Modeling," https://web.stanford.edu/jurafsky/slp3/slides/LM$_4.pdf$.

[36] J. Wieting, M. Bansal, K. Gimpel and K. Livescu, "CHARAGRAM: Embedding Words and Sentences via Character n-grams," https://www.aclweb.org/anthology/D16-1157.pdf, p. 12, 2016.

[37] P. Kaviani and S. Dhotre, "Short Survey on Naive Bayes Algorithm," https://www.researchgate.net/publication/323946641, 2017.

[38] N. Donges, "A COMPLETE GUIDE TO THE RANDOM FOREST ALGORITHM," https://builtin.com/data-science/random-forest-algorithm, 2019.

[39] S. RAY, ”Understanding Support Vector Machine(SVM) algorithm from examples (along with code),” https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code, 2017.

[40] J. Brownlee, ”Crash Course On Multi-Layer Perceptron Neural Networks,” https://machinelearningmastery.com/neural-networks-crash-course/, 2016.

[41] R. Santhanam and N. Uzir, ”Experimenting XGBoost Algorithm for Prediction and Classification of Different Datasets,” https://www.researchgate.net/publication/ 318132203, 2017.

[42] H. Patel and P. Prajapati, ”Study and Analysis of Decision Tree Based Classification Algorithms,” https://www.researchgate.net/publication/330138092, 2018.

[43] H. I. Finberg and M. L. Stone, ”Digital Journalism Credibility Study,” https://digitalfuturist.com/wp-content/uploads/2019/09/ona$_credibilityreport_v14.pdf, p.142, 2000.$

[44] A. M. Brill, ”Online Journalists Embrace New Marketing Function,” https://journals.sagepub.com/doi/10.1177/073953290102200203, 2001.

[45] W. P. Cassidy, ”Online News Credibility: An Examination of the Perceptions of Newspaper Journalists,” https://academic.oup.com/jcmc/article/12/2/478/4583013, 2007.

[46] M. Hashemi and M. Hallb, ”Detecting and classifying online dark visual propaganda,” https://www.sciencedirect.com/science/article/abs/pii/S0262885619300848, p. 11, 2019.

[47] A. Barrón-Cedeño, G. Da San Martino, P. Nakov and I. Jaradat, ”Proppy: Organizing the news based on their propagandistic content,” https://www.sciencedirect.com/science/article/abs/pii/S0306457318306058, 2019.

[48] M. Potthast, J. Kiesel, K. Reinartz, . J. Bevendorff and B. Stein, ”A Stylometric Inquiry into Hyperpartisan and Fake News,” https://www.aclweb.org/anthology/P18-1022.pdf, p. 10, 2018.

[49] W. Williamson and . J. Scrofani, ”Trends in Detection and Characterization of Propaganda Bots,” https://scholarspace.manoa.hawaii.edu/bitstream/10125/60148/0708.pdf, p. 6, 2019.

[50] O. Varol, E. Ferrara, C. A. Davis, F. Menczer and A. Flammin, ”Online Human-Bot Interactions: Detection, Estimation, and Characterization,” https://arxiv.org/pdf/1703.03107.pdf, p. 11, 2017.

[51] G. Da San Martino, S. Yu, A. Barron-Cede no, R. Petrov and P. Nakov, ”Fine-Grained Analysis of Propaganda in News Articles,” https://www.aclweb.org/anthology/D19-1565.pdf, p. 11, 2019.

[52] Ansgar Kellner, Lisa Rangosch, Christian Wressnegger, and Konrad Rieck , ”Political Elections Under (Social) Fire? Analysis and Detection of Propaganda on Twitter” , ”https://arxiv.org/pdf/1912.04143.pdf” , 2019

[53] Giovanni Da San Martino, Alberto Barron-Cedeno˜, Preslav Nakov , ”Findings of the NLP4IF-2019 Shared Task on Fine-Grained Propaganda Detection”, ”https://arxiv.org/pdf/1910.09982.pdf” , 2019

[54] Mariam Nouh, Jason R.C. Nurse,Michael Goldsmith, ”Understanding the Radical Mind: Identifying Signals to Detect Extremist Content on Twitter” , ”https://arxiv.org/pdf/1905.08067.pdf” , 2019