EVALUATION OF A FACTUAL CLAIM CLASSIFIER WITH AND WITHOUT USING ENTITIES AS FEATURES

by

ABU AYUB ANSARI SYED

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

THE UNIVERSITY OF TEXAS AT ARLINGTON

JULY 2017

Acknowledgements

I am grateful to Dr. Chengkai Li, my supervising professor for thesis for giving me this opportunity to work with him. His guidance, continuous support and attention were the soul of this project.

I would like to thank Dr. Ramez Elmasri and Mr. David Levine for serving as my committee members and for their invaluable comments, suggestions, and support.

I would like to thank Dr. Naeemul Hassan for his guidance, comments, and suggestions and allowing me to contribute to his research work. I would like to thank the complete team of IDIR lab.

Finally, I would like to express my gratitude to my parents for their encouragement and invaluable support.

JULY 14, 2017

Abstract

EVALUATION OF A FACTUAL CLAIM CLASSIFIER WITH AND WITHOUT USING ENTITIES AS FEATURES

Abu Ayub Ansari Syed, MS

The University of Texas at Arlington, 2017

Supervising Professor: Chengkai Li

Fact-checking in real-time for events such as presidential debates is a challenging task. These fact-checking processes have a difficult and rigorous task in having the best accuracy in classifying facts, finding topics, etc. The first and foremost task in fact-checking is to find out whether a sentence is factually check-worthy. The UTA IDIR Lab has deployed an automated fact-checking system named ClaimBuster. ClaimBuster has a core functionality of identifying check-worthy factual sentences.

Named entities are essentially an important component of any textual data. To use these named entities, it is required to link them to labels such as a person, location, and organization. If we want the automated systems to read and understand the natural language like we do, the system must recognize the named entities that are mentioned in the text.

The ClaimBuster Project, in classifying the sentences of the presidential debates has categorized the sentences into three types, namely check-worthy factual sentences (CFS), non-factual sentences (NFS) and unimportant factual sentences (UFS). This categorization helps us in making the supervised classification problem as a three-class problem (or a two-class problem, by merging NFS and UFS). ClaimBuster, in the process

of identifying check-worthy factual claims, has employed named entities as a feature along with sentiment, length, words (W) and part-of-speech(POS) tags in the classification models. In this work, I have evaluated the classification algorithms such as Naïve Bayes Classifier (NBC), Support Vector Machine (SVM) and Random Forrest Classifier (RFC). The evaluation mainly constitutes the comparison of the performance of these classifiers with and without using named entities as a feature. We have also analyzed the mistakes that the classifiers have made by comparing two sets of features at a time. Therefore, the analysis consists of 18 experiments constituting three classifiers, two classification types and three sets of feature comparison. We see that the presence of named entities contributes very little to the classifier, but also that their presence is subdued by the presence of better performing features such as the part-of-speech (POS) tags.

Table of Contents

List of Illustrations

List of Tables

Chapter 1 INTRODUCTION

ClaimBuster is a factual claim classifier that helps journalist find claims to fact-check [1]. It is an outcome of research by a team in IDIR Lab at UT Arlington. This thesis employs the factual claim classifier of the complete automatic fact checking system: ClaimBuster (idir.uta.edu/claimbuster). Chapter 2 gives an overview of ClaimBuster and its functionality. Figure 1.1 below is a screenshot of a 2016 US presidential debate deployed on ClaimBuster.

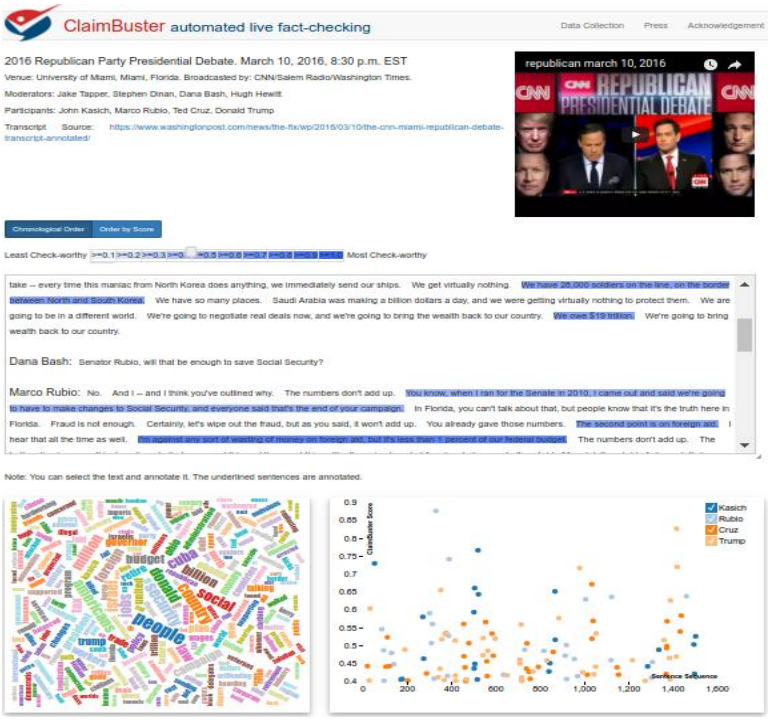

Figure 1.1: ClaimBuster platform [3]

The problem of identifying factual check worthy sentences can be modeled as a classification problem [1]. This classification task can be modeled as a supervised learning task. The dataset that is used in this classification task includes labeled sentences from presidential candidates during the general elections since 1960. A total of

30 debate episodes spanned from 1960-2012 [1]. The sentences that were labeled are used for training and testing the classifier to measure the classification performance. These sentences constitute of various features that are extracted and used in modeling. The data sets and the collection of labels for the sentences are explained further in Chapter 2.

Among the features that were extracted, named entities are one of them. These named entities of various entity types were extracted and collected as features for both training and testing. The tools that are used to extract entities and their entity types are explained in Chapter 3. After the extraction and feature building, we classify the sentences using three different classifiers, namely Naïve Bayes Classifier (NBC), Support Vector Machine (SVM) and Random Forrest Classifier (RFC). The performance of each of these classifiers is measured in the standard forms as precision, recall, and f-measure. Chapter 4 discusses the results of these classifiers.

Further, we analyzed each of these classifiers with various combinations of features. The analysis involves understanding incorrect predictions made by classifiers using the different feature sets. Each of this analysis is discussed regarding the classifier performance and their incorrectly predicted sentences. These analyses are explained in Chapter 5 and 6.

Overall, this project contributes to the ClaimBuster project as follows:

- The evaluation of named entities on ClaimBuster factual claim classifier.
- Extraction of named entities using Illinois NE tagger.
- Evaluating the performance of ClaimBuster using new features.
- Analysis of incorrect predictions of classifiers by comparing a classifier using various feature sets.
- Conclusion on named entities as features in classification.

Chapter 2 OVERVIEW OF CLIAMBUSTER

ClaimBuster is a tool that helps journalists find claims to fact-check [3]. ClaimBuster is explained in detail in [1]. In this section, a brief description of the system is given. ClaimBuster processes every sentence and uses classification and ranking model to determine how check worthy the sentence is [3].

ClaimBuster then assigns a score between 0 and 1 to each sentence. The larger score means that the sentence is check-worthy for facts and lower score means that the sentence is mostly non-factual or an opinion.

2.1 PROBLEM DESCRIPTION

The sentences from the presidential debates can be categorized into three types. This categorization helps us in modeling the classification task as a multi-class problem. The below mentioned are the classes used:

Non-Factual Sentence (*NFS*): [1] These sentences do not contain any factual claim.

Unimportant Factual Sentence (*UFS*): [1] These are factual claims but not check-worthy.

Check-worthy Factual Sentence (*CFS*): [1] These sentences contain factual claims.

Therefore, classes used in classification are *NFS(-1)*, *UFS(0)* and *CFS(-1).*

2.2 DATASET

To construct the dataset for developing and evaluating approaches to detect check-worthy factual claims, ClaimBuster has used presidential debate transcripts.

2.2.1 DATA USED

The datasets are prepared using the transcripts of presidential debates. There was a total of 14 debates that were held during the period starting from 1960 until 2012. There were no debates in a few elections. Therefore, there is a total of 30 debates. The transcripts all together provide a total of 28029 sentences. The system removes sentences that are too short i.e. less than five words and keeps only the sentences spoken by presidential candidates.

2.2.2 DATA COLLECTION

Each sentence of the transcript was labeled manually by users. Journalists, professors, and university students were invited to participate in the survey. The participants labeled each sentence with any one of the following options.

1. There is no factual claim in the sentence.
2. There is a factual claim but it is unimportant.
3. There is an important factual claim.

The details regarding the number of participants, domain experts and the mechanism to achieve top-quality participants are mentioned in [1]. The data collection process was still in progress during [1]. Currently, in this project, we have used 20617 labeled sentences.

2.3 FEATURE EXTRACTION

Different types of features were extracted from each sentence. 5 types of features are explained with an example in [1]. Briefly, they are:

1. Sentiment: Sentiment score that ranges from -1 (negative sentiment) to 1 (positive sentence).
2. Length: Word count in a sentence.
3. Word (W): Words in sentences was used to build a TF-IDF feature.

4. Parts of Speech (POS tags): Count of words that belong to a POS tag in a sentence is the value of the feature.
5. Entity Type (ET): Alchemy API extracted named entities that belong to 26 entity types.

Feature selection is performed to avoid over-fitting and to achieve a simpler model. The system trained a random forest classifier for which GINI index is used to measure the importance of the features [1].

2.4 CLASSIFICATION

ClaimBuster classifier used 4-fold cross validation in supervised learning methods namely, Naïve Bayes Classifier (NBC), Support Vector Machine (SVM) and Random Forrest Classifier (RFC). The system calculates the performance in terms of precision (p), recall (r), f-measure (f) and Cohen's kappa coefficient (k). Various feature sets were used in experiments using sentiment and length in all cases.

2.5 EVALUATION

SVM using words and POS tag features on the CFS class, ClaimBuster achieved 79% precision (i.e., it is accurate 79% of the time when it declares a CFS sentence) and 74% recall (i.e., 74% of true CFSs are classified as CFS).

Chapter 3 ILLINOIS NER TOOL

Identifying and categorizing strings from text into different classes is a process defined as Named Entity Recognition (NER) [4]. These classes that the NER process identifies are called the entity types. There are two main reasons for the extraction of entities. [4].

- They can be used directly in the many applied research domains.
- They are used as a preprocessing step in more advanced NLP tools.

The NER tools differ in many ways. [4] gives a detailed explanation of NER tools. We briefly include a few.

- The mechanism they employ. NER tools range from manually specified systems (e.g. grammar rules) to fully automatic machine-learning processes.
- They utilize different classes i.e. entity types.
- Outputted data are of different formats.

Moreover, their accuracies vary depending upon the entity types, class of text, etc. [4]. There are three main families in NER tools [4]. This categorization is mainly concerning the mechanism that the tools employed. The three families of NER tools are as follows:

1. Hand-made rule-based methods: This category uses manually constructed finite state patterns.
2. Machine-Learning based methods: This category treats NER as a classification process.
3. Hybrid methods: This category is a combination of the above two approaches.

In this project, we have used a hybrid NER tool named *Illinois Named Entity Tagger (INET).* Illinois NET relies on several supervised learning methods: hidden Markov models, multilayered neural networks and other statistical methods [4]. The details of

Illinois NE tagger are coming in the following section. These sections help us in understanding the extracted entities and performance of the tagger.

3.1 ENTITIES

Named entities refer to a phrase that is representing a specific class [5]. Ultimately, named entities are a group of consecutive words found in a sentence that represents concepts such as persons, locations, organizations, objects, etc. [4].

3.1.1. IMPORTANCE OF ENTITIES

In information retrieval tasks, structured information of organizations and companies are extracted from unstructured data such as news articles. Here it was noticed that it is essential to recognize the information units like names such as a person, organization, and locations as well as numeric information such as time, date, money and percentages [9]. This task was considered as one of the important sub tasks of Information retrieval.

3.1.2. ENTITY TYPES

The specific named entity classes called entity types aims to restrict the task of recognition to only those entities in the textual documents. The main factors discussed in [9] explain that language, genre or domain, and entity types are the most critical ones. Considering the domain of our dataset as politics and language used is English we have chosen the Illinois NE tagger. The tagger consists of the following entity types.

PERSON	Person	ORG	Organization
LOC	Location	TIME	Time
LAW	Law	NORP	Nationality
GPE	Geo-political Entity	LANGUAGE	Language
PERCENT	Percentage	FAC	Facility
PRODUCT	Product	ORDINAL	Ordinal Number
CARDINAL	Cardinal Number	WORK_OF_ART	Work of Art
MONEY	Money	DATE	Date
EVENT	Event	QUANTITY	Quantity

Figure 3.1: Illinois NER Entity Types

The entity types are self-explanatory. The following illustration is a sample text tagged by Illinois NER tagger with some of the entity types as mentioned above.

Helicopters will patrol the temporary no-fly zone around [LOC New Jersey's] [FAC MetLife Stadium] [DATE Sunday] , with F-16s based in [GPE Atlantic City] ready to be scrambled if an unauthorized aircraft does enter the restricted airspace.

Down below, bomb-sniffing dogs will patrol the trains and buses that are expected to take [CARDINAL approximately 30,000] of the 80,000-plus spectators to [DATE Sunday] 's [ORG Super Bowl] between [ORG the Denver Broncos] and [ORG Seattle Seahawks] .

[ORG The Transportation Security Administration] said it has added [CARDINAL about two dozen] dogs to monitor passengers coming in and out of the airport around [EVENT the Super Bowl] .

On [DATE Saturday] , [ORG TSA] agents demonstrated how the dogs can sniff out many different types of explosives. Once they do, they're trained to sit rather than attack, so as not to raise suspicion or create a panic.

[ORG TSA] spokeswoman [PERSON Lisa Farbstein] said the dogs undergo [DATE 12 weeks] of training, which costs about $200,000, factoring in food, vehicles and salaries for trainers.

Dogs have been used in cargo areas for some time, but have just been introduced recently in passenger areas at [GPE Newark] and [GPE JFK] airports. JFK has one dog and [GPE Newark] has a handful, [PERSON Farbstein] said.

Figure 3.2: Illinois NER Entities tagged to sample text

The Illinois NER tagger was used as a part of the complete toolkit from [8] called Curator. The Illinois toolkit provides a total of 18 distinct entity types. The above example is from the online demo using extended entity types. The NER tool was used to extract all

the 28029 sentences of our dataset. The following graph shows the entity count for each entity types in our dataset.

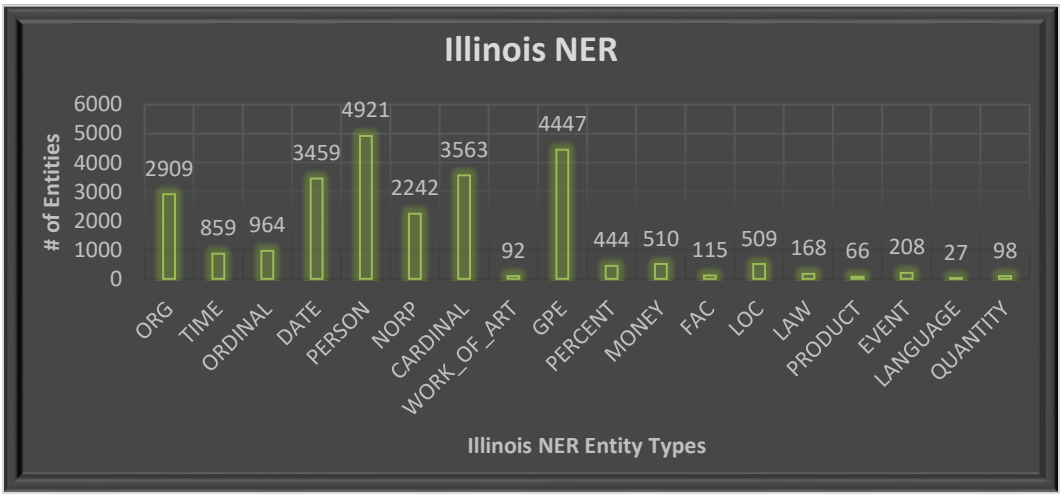

Figure 3.3: Illinois NER Entity Count

After extracting entities from all the sentences, we remove entities that belong to short sentences and sentences that are not spoken by presidential candidates. Below are some of the statistics of entities in classes which are used in classification.

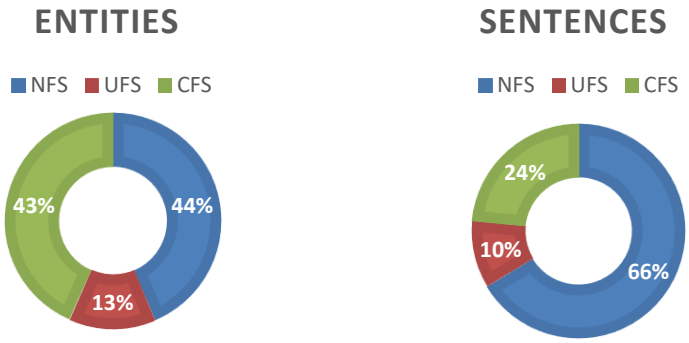

Figure 3.4: Sentences and Entities counts in classes

From the above two charts (Figure 3.4) we see that the number of NFSs in the dataset are the largest. NFS and UFS constitute 76% of the dataset and CFS are only 24%. Moreover, NFS (66%) contribute to only 44% of total entities. Whereas, CFS (24%) contribute to 43% of entities.

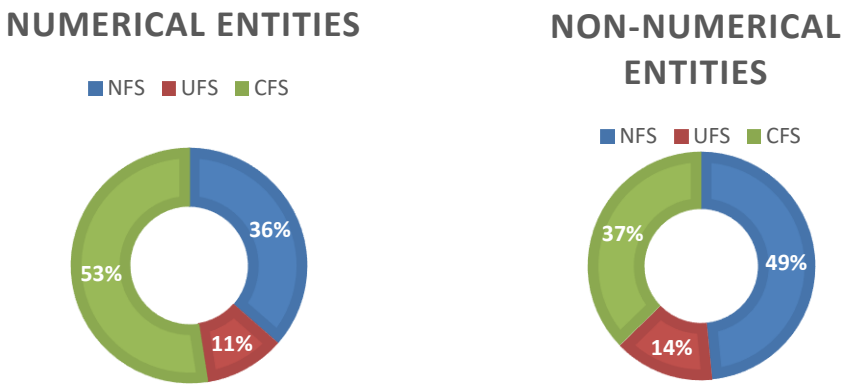

Figure 3.5: Numerical and Non-Numerical entities in classes

Above charts (Figure 3.5) show the distribution of numerical and non-numerical entities in classes. Numerical entities are date, time, money, cardinal, ordinal, percent and quantity. Non-numerical entities are a person, location, organization, GPE, NORP, facility, event, law, work-of-art, language, and product. We see that CFS have more numerical entities than UFS and NFS. Hence, we can say that CFS contains the most numeric data. Moreover, NFS has more non-numerical entities.

3.2 STANDALONE AND AS A SERVICE

The Illinois NE tagger software can run either programmatically or as a command line application or as a server called by clients. In this project, we have used the tagger as a standalone tool. Since the transcripts were already available, we tagged them by sentences and id them to be used as features in classification.

3.3 PERFORMANCE OF ILLINOIS NER

Many research papers online show the performance of NE taggers based on only few entity types. The most commonly found evaluations are based on the entity types such as a person, location, organization and misc. The papers [4], [5], [9] are evaluated with four entity types. Also, even the papers [6] and [7], which are the citations to base the Illinois Curator and Illinois NE tagger [8] also have evaluations based on the most widely used entity types namely, PER, LOC, and ORG. These evaluations use common entity types as taggers use different entity types. [8] shows Illinois NE tagger performance of 90.8 F_1 score and [12] shows Stanford NLP named entity recognition performance of 85.53 F_1 score for CoNLL 2003 corpus.

3.4 ILLINOIS NER VS. ALCHEMY API

Previously, ClaimBuster used entities from a third-party tool called Alchemy API. In this section, we briefly see the comparison between the two NE taggers. The following are the entity types of the Alchemy API:

#	**Alchemy API Entity Types**	#	**Alchemy API Entity Types**
1	Field Terminology	15	Health Condition
2	Person	16	Company
3	Print Media	17	Crime
4	Television Station	18	Holiday
5	Job Title	19	Geographic Feature
6	Country	20	Entertainment Award
7	Quantity	21	Television Show
8	Drug	22	Degree

9	City	23	Sport
10	Organization	24	Sporting Event
11	Facility	25	Technology
12	Continent	26	Operating System
13	State Or County	27	Movie
14	Region	28	Hashtag

Table 3.1: Alchemy API Entity Types

A few major reasons that we replaced the tagger are:

1. Independence of extraction: Illinois NE tagger can be used a standalone tool from which we can extract entities locally in the system. Alchemy API requires us to send the data over to a third party to extract the entities.
2. Cost: Illinois NE tagger is an open source tool which has zero cost for the software unlike Alchemy API, which is expensive for a large number of sentences. Academic institutions have a discount for 30,000 sentences.
3. Better Entity types: Illinois NE tagger entity types favors over the Alchemy API mainly due to entity types such as the DATE, TIME, CARDINAL, ORDINAL and PERCENT. These entities are not available in Alchemy API.

Finally, we are extracting entities using Illinois NE tagger in this project. Alchemy API entities were collected for experiments conducted in [1]. Below are some statistics of the extracted entities:

	Alchemy API	Illinois NE Tagger
Number of Sentences	28029	28029
Total number of Entities	18864	25601
Number of Unique Entities	3309	5499
Number of Sentences with at least 1 Entity	10918	13934

Table 3.2: Illinois NE tagger vs. Alchemy API stats

From above table, we see that Illinois NE tagger has extracted more entities and the number of sentences with at least one entity is larger, that is more sentences have entity feature for classification compared to Alchemy API.

Chapter 4 EVALUATION OF CLASSIFIERS

In this section, we explain classifier types used, performance measure and experiment results for the factual claim classifier.

4.1 PERFORMANCE OF NBC, SVM, AND RFC FOR CLAIMBUSTER

ClaimBuster uses Naïve Bayes Classifier (NBC), Support Vector Machine (SVM) and Random Forrest Classifier (RFC) for training and testing the labeled sentences spoken by the presidential candidates.

In our project, we have experimented the classification problem as two-class and three-class classification problem. The labeled sentences (collected: 20617), based on majority votes are categorized as follows:

#	Class Name	Class Label
1	Non-Factual Sentence (NFS)	-1
2	Unimportant Factual Sentence (UFS)	0
3	Check-worthy Factual Sentence (CFS)	1

Table 4.1: Three-class classification labels

In three class classification, we keep classes the same as above categories. In two-class classification, we have merged the NFS with UFS into a single class and keep CFS as an independent class. The two-class classification problem is considered in the following fashion:

#	Class Name	Class Label
1	Non-Factual and Unimportant Factual Sentence (N_UFS)	-1
2	Check-worthy Factual Sentence (CFS)	1

Table 4.2: Two-class classification labels

Classification experiments were conducted with four combinations of features:

1. Words (W),
2. Words + Illinois Entities (W_ET),
3. Words + POS Tags (W_P),
4. Words + POS Tags + Illinois Entities (W_P_ET)

Note: Sentiment and Length are included in all the combinations.

PRECISION RECALL AND F-MEASURE

The results are calculated in a standard Information retrieval measure for classification processes. Precision (p) is a measure of result relevancy while recall (r) is a measure of how many truly relevant results are returned. An ideal system with high precision and high recall will return many results, with all results labeled correctly. Precision and Recall are defined as follows:

$$Precision\ (p) = \frac{\#\ relevant\ found}{\#\ found} = \frac{tp}{tp + fp}$$

$$Recall\ (r) = \frac{\#\ relevant\ found}{\#\ relevant} = \frac{tp}{tp + fn}$$

These measurements are also related to F_1 score which is defined as the harmonic mean of precision and recall.

$$F_1 = 2\ \frac{P.R}{P + R}$$

wavg denotes weighted average of corresponding measure across all classes.

RESULTS FOR DIFFERENT FEATURES SETS

The table 4.3 shows the ClaimBuster classifier performance results in terms of the above-described measures. The results below are similar to the experiments

explained in [1]. However, there are changes in a few aspects such as the entities are modified from Alchemy API to Illinois NE tagger. The labeled sentences are more than they were in [1]. The complete dataset of 20617 sentences from all presidential debates is labeled and used in the classification task unlike in [1], where only the initial 12 debates sentences were labeled.

We see that SVM using W_P_ET achieved 76%,77% and 76% weighted average precision, recall, and f-measure respectively. It also yields the highest recall for CFS. All the classification models perform better on NFSs and CFSs compared to UFSs. SVM and NBC outperform RFC in most cases. RFC has worst performance in case of CFS. There is always a trade-off between precision and recall i.e. on increase of one the other decreases. The overall performance of the classifiers has not significantly improved by using additional features such as entities and POS tags, but the recall of CFS and UFS has improved due to the contribution of entities.

algorithm	features	p_NFS	p_UFS	p_CFS	p_wavg	r_NFS	r_UFS	r_CFS	r_wavg	f_NFS	f_UFS	f_CFS	f_wavg
SVM	W	0.83	0.43	0.69	0.76	0.90	0.30	0.62	0.77	0.86	0.34	0.65	**0.76**
NBC	W	0.74	**0.89**	**0.79**	0.77	**0.98**	0.01	0.39	0.74	0.84	0.02	0.52	0.68
RFC	W	0.74	0.74	0.76	0.75	0.97	0.04	0.40	0.74	0.84	0.08	0.52	0.69
SVM	W_P	0.83	0.49	0.68	0.76	0.91	0.25	0.63	**0.77**	**0.86**	0.33	0.65	0.76
NBC	W_P	0.81	0.48	0.63	0.74	0.91	0.13	0.62	0.76	0.86	0.21	0.62	0.74
RFC	W_P	0.76	0.73	0.76	0.75	0.97	0.05	0.45	0.76	0.85	0.09	0.57	0.71
SVM	W_P_ET	**0.85**	0.44	0.68	**0.77**	0.88	**0.34**	**0.64**	0.77	0.86	**0.37**	**0.66**	0.76
NBC	W_P_ET	0.82	0.47	0.63	0.74	0.90	0.17	0.62	0.76	0.86	0.25	0.63	0.74
RFC	W_P_ET	0.76	0.75	0.76	0.76	0.97	0.05	0.47	0.76	0.85	0.10	0.58	0.71

Table 4.3: ClaimBuster Classifier performance for three-class classification

algorithm	features	p_N_UFS	p_CFS	p_wavg	r_N_UFS	r_CFS	r_wavg	f_N_UFS	f_CFS	f_wavg
SVM	W	0.87	0.69	0.83	0.92	0.55	0.83	0.89	0.59	0.82
NBC	W	0.82	0.84	0.82	0.98	0.29	0.82	0.89	0.43	0.78
RFC	W	0.82	**0.86**	0.83	**0.99**	0.29	0.82	0.89	0.43	0.79
SVM	W_P	0.87	0.72	0.84	0.93	0.56	**0.84**	**0.90**	0.63	0.84
NBC	W_P	0.87	0.65	0.82	0.91	0.57	0.83	0.89	0.61	0.82
RFC	W_P	0.82	0.86	0.83	0.98	0.32	0.83	0.90	0.46	0.80
SVM	W_P_ET	**0.88**	0.70	**0.84**	0.92	**0.60**	0.84	0.90	**0.64**	**0.84**
RFC	W_P_ET	0.83	0.85	0.83	0.98	0.34	0.83	0.90	0.48	0.80
NBC	W_P_ET	0.88	0.64	0.82	0.90	0.58	0.83	0.89	0.61	0.82

Table 4.4: ClaimBuster Classifier performance for two-class classification

Chapter 5 ANALYSIS OF THREE CLASS CLASSIFIERS

In this section, we analyze the classifiers by using their performance and incorrectly predicted sentences. In this chapter, we discuss three-class classification experiments, and in the next chapter, we discuss the two-class classification.

To understand the contribution and influence of entities on classifiers, we have chosen three sets of comparisons using which the analysis of the classifier can be made reasonable.

Therefore, we compare the results of a classifier using the following features:

1. W vs. W_ET
2. W_P vs. W_ET
3. W_P vs. W_P_ET

All the classifiers are compared as above. Cumulatively, they form 9 experiments. Section 5.1 explains NBC three-class analysis and above mentioned three comparisons are subsections of 5.1. Similarly, 5.2 and 5.3 explain SVM three-class analysis and RFC three-class analysis respectively. Table 5.1 shows the performance results in terms of precision, recall, and f-measure for all the classifiers using different feature sets.

Further we extract sentences from both the classifiers which are being compared in which the predictions were made incorrectly with respect to verdicts (labels). This yields us with 27 cases, that is 3 classes (NFS, UFS, CFS) of verdicts, feature set 1 and feature set 2 each (3^3). This extraction helps us to make a deeper study on the sentences which are classified correctly and incorrectly by the contribution of entities.

algorithm	features	p_NFS	p_UFS	p_CFS	p_wavg	r_NFS	r_UFS	r_CFS	r_wavg	f_NFS	f_UFS	f_CFS	f_wavg
SVM	W	0.83	0.43	0.69	0.76	0.90	0.30	0.62	0.77	0.86	0.34	0.65	0.76
SVM	W_ET	0.82	0.48	0.68	0.75	0.91	0.22	0.60	0.76	0.86	0.31	0.61	0.74
SVM	W_P	0.83	0.49	0.68	0.76	0.91	0.25	0.63	0.77	0.86	0.33	0.65	0.76
SVM	W_P_ET	0.85	0.44	0.68	0.77	0.88	0.34	0.64	0.77	0.86	0.37	0.66	0.76
RFC	W	0.74	0.74	0.76	0.75	0.97	0.04	0.40	0.74	0.84	0.08	0.52	0.69
RFC	W_ET	0.75	0.81	0.76	0.76	0.97	0.03	0.44	0.75	0.85	0.06	0.55	0.70
RFC	W_P	0.76	0.73	0.76	0.75	0.97	0.05	0.45	0.76	0.85	0.09	0.57	0.71
RFC	W_P_ET	0.76	0.75	0.76	0.76	0.97	0.05	0.47	0.76	0.85	0.10	0.58	0.71
NBC	W	0.74	0.89	0.79	0.77	0.98	0.01	0.39	0.74	0.84	0.02	0.52	0.68
NBC	W_ET	0.77	0.71	0.71	0.75	0.96	0.03	0.50	0.75	0.85	0.05	0.58	0.71
NBC	W_P	0.81	0.48	0.63	0.74	0.91	0.13	0.62	0.76	0.86	0.21	0.62	0.74
NBC	W_P_ET	0.82	0.47	0.63	0.74	0.90	0.17	0.62	0.76	0.86	0.25	0.63	0.74

Table 5.1: Classifier performance of three-class classification for Analysis

5.1 NBC ANALYSIS FOR THREE CLASS

Case 1: W vs. W_ET

	Verdicts	NBC_W	NBC_W_ET
#CFS (1)	4843	2410	3426
#UFS (0)	2103	26	78
#NFS (-1)	13671	18181	17113
Total	20617	20617	20617

Table 5.1.1 (a): Classifier prediction count for CFS, UFS, and NFS in NBC: W vs. W_ET

Verdicts_NFS	631
Verdicts_UFS	2082
Verdicts_CFS	3114
Total	5827

Table 5.1.1 (b): Mistakes made by classifiers in NBC: W vs. W_ET

In this experiment, we have compared NBC using W and W_ET, i.e. Words as features and words coupled with entities as features. Table 5.1.1 (a) shows the count of CFS, UFS, and NFS in verdicts (labels), predictions in NBC_W and NBC_W_ET. From this, we see that both these classifiers have performed weakly in UFSs. The presence of entities has helped classify more CFSs.

From Table 5.1, we see that the precision of NFSs has also increased in the case of entities, but overall weighted average (wavg) of precision has reduced. This fall in precision is due to the decline of precision of UFS which is evident as the number of sentences predicted in UFS is very low. Overall recall (r_wavg) remains the same in both the cases, but the recall for CFS has improved about 10%. Overall, the performance of both the classifiers remains the same, but with very minimal improvement in the case of entities.

Verdicts	NBC_W	NBC_W_ET	Count
-1	-1	-1	0
-1	-1	0	11
-1	-1	1	364
-1	0	-1	0
-1	0	0	2
-1	0	1	0
-1	1	-1	54
-1	1	0	0
-1	1	1	200
0	-1	-1	1560
0	-1	0	26
0	-1	1	234
0	0	-1	2
0	0	0	0
0	0	1	1
0	1	-1	40
0	1	0	7
0	1	1	212
1	-1	-1	2253
1	-1	0	7
1	-1	1	686
1	0	-1	0
1	0	0	0
1	0	1	0
1	1	-1	164
1	1	0	4
1	1	1	0
		Total	5827

Table 5.1.1 (c): All possible combinations for prediction in NBC: W vs. W_ET

Table 5.1.1 (b) shows the incorrect predictions of the classifiers for the verdicts. We see that CFS made the majority of mistakes. When we look in detail into the errors made by classifiers, we look at all the possible combinations of prediction. Table 5.1.1 (c) shows all the combinations of predictions with the count of incorrectly predicted sentences in each combination. Figure 5.1.1 is the graph of Table 5.1.1 (c).

From the graph, it shows that the maximum errors were made in (1, -1, -1) i.e. both the classifiers predicted it to be NFS when the sentence was labeled CFS and in case of (0, -1, -1) i.e. both the classifiers predicted it to be NFS when the sentence was labeled UFS. The combinations where entities have helped are (-1, 0, -1), (-1, 1, -1), (0, -1, 0), (0, 1, 0), (1, -1, 1) and (1, 0, 1) which sums up to 773

sentences. The combinations where the entities have harmed the classifier are (-1, -1, 0), (-1, -1, 1), (0, 0, -1), (0, 0, 1), (1, ,1 -1) and (1, 1, 0) which sums up to 546 sentences.

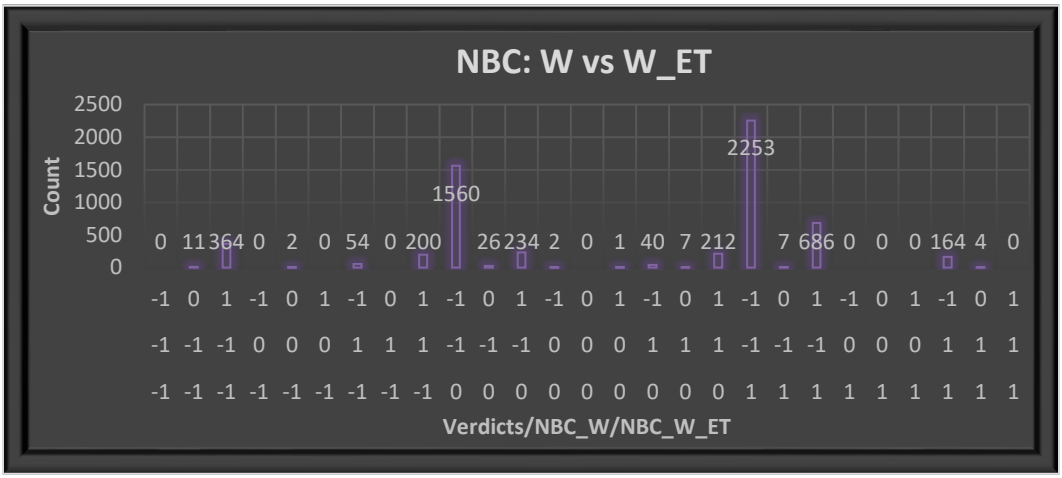

Figure 5.1.1: Incorrect predictions in NBC: W vs. W_ET

From the above graph, we see that NBC predicts more sentences towards NFS regardless of whether it has entities as features. 65% of errors are in this category.

Finally, we see here that entities improve the recall of CFS but the classifier performance remains almost the same. The overall performance (f_wavg) improves only by 3%.

Case 2: W_P vs. W_ET

	Verdicts	NBC_W_P	NBC_W_ET
#CFS (1)	4843	4792	3426
#UFS (0)	2103	587	78
#NFS (-1)	13671	15238	17113
Total	20617	20617	20617

Table 5.1.2 (a): Classifier prediction count for CFS, UFS, and NFS in NBC: W_P vs. W_ET

Verdicts_NFS	1428
Verdicts_UFS	2054
Verdicts_CFS	2669
Total	6151

Table 5.1.2 (b): Mistakes made by classifiers in NBC: W_P vs. W_ET

In this experiment, we have compared NBC using W_P and W_ET, i.e. Words coupled with POS tags as features and words coupled with entities as features. Table 5.1.2 (a) shows the counts of CFS, UFS, and NFS in verdicts (labels), predictions in NBC_W_P and NBC_W_ET. From this, we see that W_P has classified more sentences in CFS compared to W_ET as features. We also see that W_P has classified more UFS than W_ET.

From Table 5.1, we see that precision (p_wavg) and recall (r_wavg) remains the same in both the cases, but the recall for CFS is better in the case of POS tags by 10%.

Table 5.1.2 (b) shows the mistakes made by both the classifiers concerning the verdicts. We see that CFS made the majority of errors. Figure 5.1.2 illustrates the graph of all the combinations of predictions with the count of incorrectly predicted sentences in each category.

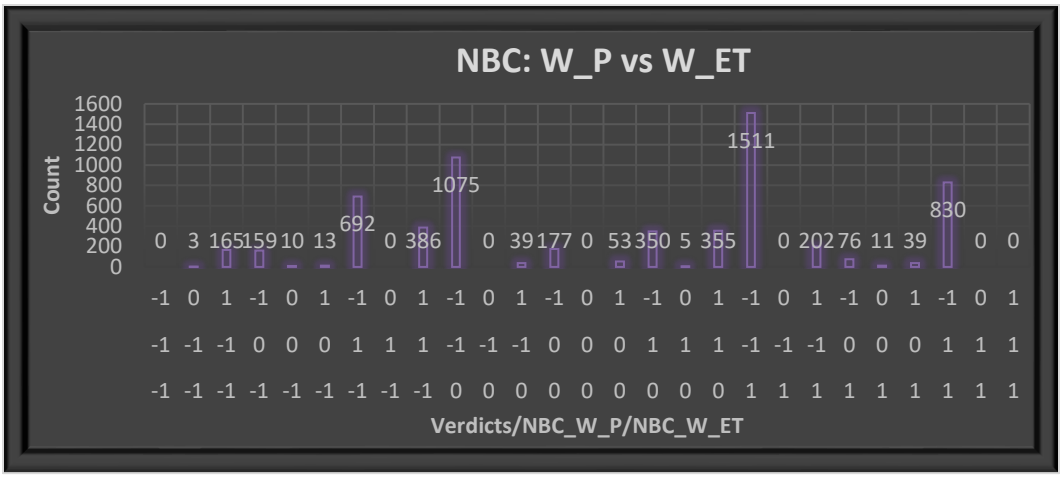

Figure 5.1.2: Incorrect predictions in NBC: W_P vs. W_ET

From the above graph, we see that NBC predicts more sentences towards NFS regardless of whether entities are features or POS tags are features.

Finally, we see here that entities and POS tags features have the same impact on classification as to each other. The overall performance (f_wavg) is very close to each other in both the cases.

Case 3: W_P vs. W_P_ET

	Verdicts	NBC_W_P_ET	NBC_W_P
#CFS (1)	4843	4794	4792
#UFS (0)	2103	772	587
#NFS (-1)	13671	15051	15238
Total	20617	20617	20617

Table 5.1.3 (a): Classifier prediction count for CFS, UFS, and NFS in NBC: W_P vs. W_P_ET

Verdicts_NFS	1525
Verdicts_UFS	1836
Verdicts_CFS	2012
Total	5373

Table 5.1.3 (b): Mistakes made by classifiers in NBC: W_P vs. W_P_ET

In this experiment, we have compared NBC using W_P and W_P_ET, i.e. Words combined with POS tags as features and words coupled with entities and POS tags as

features. Table 5.1.3 (a) shows the counts of CFS, UFS, and NFS in verdicts (labels), predictions in NBC_W_P and NBC_W_P_ET. From this, we see that W_P and W_P_ET have classified almost the same number of sentences in all classes.

From Table 5.1, we see that precision (p_wavg) and recall (r_wavg) and f-measure (f_wavg) remains the same in both the cases.

Table 5.1.1 (b) shows the mistakes made by both the classifiers concerning the verdicts. Figure 5.1.3 illustrates the graph of all the combinations of predictions with the count of incorrectly predicted sentences in each category.

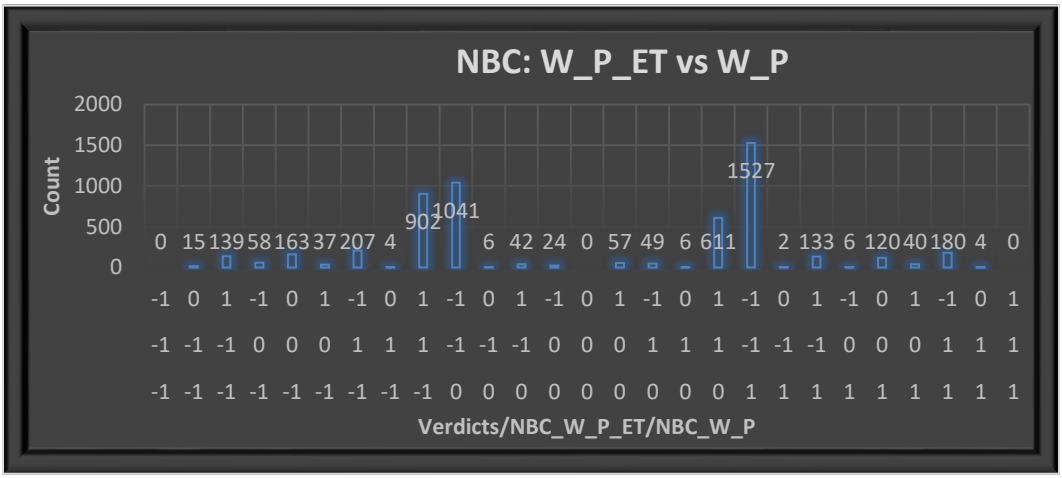

Figure 5.1.3: Incorrect predictions in NBC: W_P vs. W_P_ET

From above, we see here that entities have not contributed any further in this case. The overall performance (f_wavg) remains same (74%).

Finally, the presence of entities along with POS tags has subdued its presence. Use of entities has helped in classifying more CFS. Thus using entities in the case of NBC is not effective as the best performance can be achieved using POS tags as features.

5.2 SVM ANALYSIS FOR THREE CLASS

Case 1: W vs. W_ET

	Verdicts	SVM_W	SVM_W_ET
#CFS (1)	4843	4019	4442
#UFS (0)	2103	1359	986
#NFS (-1)	13671	15239	15189
Total	20617	20617	20617

Table 5.2.1 (a): Classifier prediction count for CFS, UFS, and NFS in SVM: W vs. W_ET

Verdicts_NFS	1554
Verdicts_UFS	1668
Verdicts_CFS	2373
Total	5595

Table 5.2.1 (b): Mistakes made by classifiers in SVM: W vs. W_ET

In this experiment, we have compared SVM using W and W_ET, i.e. Words as features and words combined with entities as features. Table 5.2.1 (a) shows the counts of CFS, UFS, and NFS in verdicts (labels), predictions in SVM_W and SVM_W_ET. From this, we see that presence of entities has helped classify more CFSs.

From Table 5.1, we see that precision (p_wavg) and recall (r_wavg) and f-measure (f_wavg) remains the same in both the cases.

Table 5.2.1 (b) shows the mistakes made by both the classifiers concerning the verdicts. Figure 5.2.1 illustrates the graph of all the combination of predictions with the count of incorrectly predicted sentences in each category. The overall improvement made by entities is nullified by the harm done by the entities themselves.

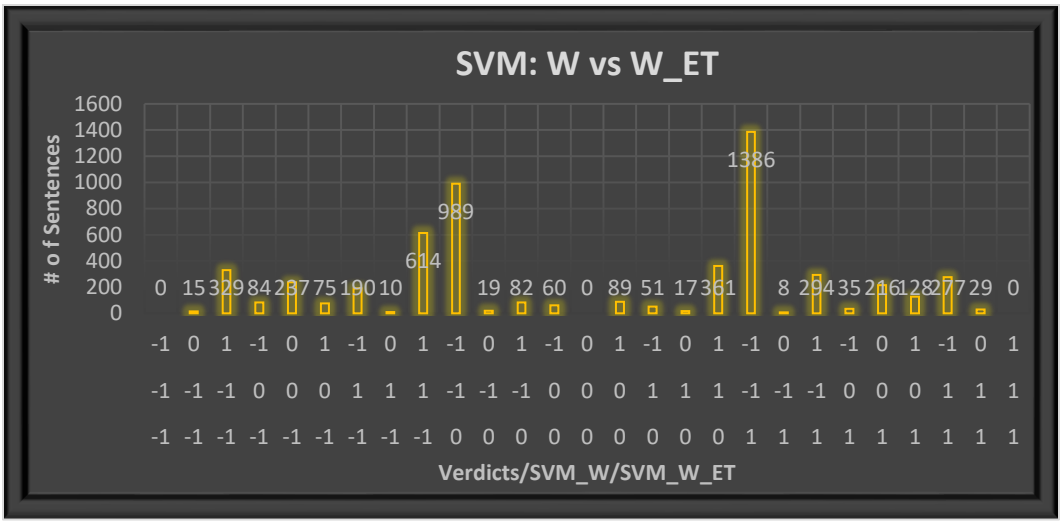

Figure 5.2.1: Incorrect predictions in SVM: W vs. W_ET

From above, we see here that entities, in this case, have not improved the performance. The overall performance (f_wavg) has reduced by 2%.

Finally, the presence of entities does not contribute to the classifier. Thus using entities alone is not effective as features in this case.

Case 2: W_P vs. W_ET

	Verdicts	SVM_W_P	SVM_W_ET
#CFS (1)	4843	4501	4442
#UFS (0)	2103	1064	986
#NFS (-1)	13671	15052	15189
Total	20617	20617	20617

Table 5.2.2 (a): Classifier prediction count for CFS, UFS, and NFS in SVM: W_P vs. W_ET

Verdicts_NFS	1621
Verdicts_UFS	1694
Verdicts_CFS	2223
Total	5538

Table 5.2.2 (b): Mistakes made by classifiers in SVM: W_P vs. W_ET

In this experiment, we have compared SVM using W_P and W_ET, i.e. Words combined with POS tags as features and words coupled with entities as features. Table 5.2.2 (a) shows the counts of CFS, UFS, and NFS in verdicts (labels), predictions in SVM_W_P and SVM_W_ET.

From this, we see that W_P and W_ET have classified almost the same number of sentences in all classes. From Table 5.1, we see that precision (p_wavg) and recall (r_wavg) and f-measure (f_wavg) remains almost the same in both the cases.

Table 5.2.2 (b) shows the mistakes made by both the classifiers concerning the verdicts. Figure 5.2.2 illustrates the graph of all the combinations of predictions with the count of incorrectly predicted sentences in each category.

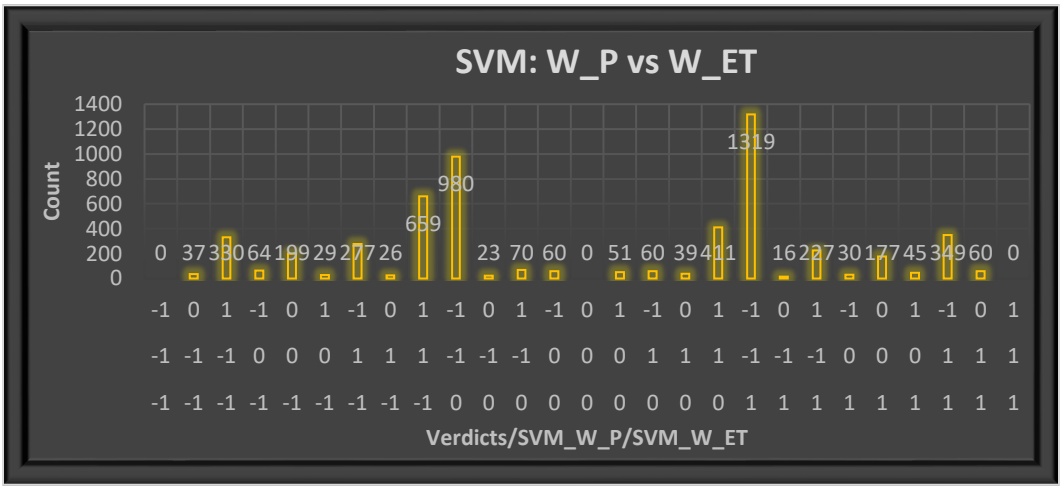

Figure 5.2.2: Incorrect predictions in SVM: W_P vs. W_ET

From above, we see that entities and POS tags have the same influence on the classifier. The overall performance (f_wavg) is almost the same in both cases (74%). Finally, the influence of POS tags is as good as entities. Entities alone as a feature do not contribute significantly to the classification performance.

Case 3: W_P vs. W_P_ET

	Verdicts	SVM_W_P_ET	SVM_W_P
#CFS (1)	4843	4427	4501
#UFS (0)	2103	1174	1064
#NFS (-1)	13671	15016	15052
Total	20617	20617	20617

Table 5.2.3 (a): Classifier prediction count for CFS, UFS, and NFS in SVM: W_P vs. W_P_ET

Verdicts_NFS	1447
Verdicts_UFS	1623
Verdicts_CFS	1979
Total	5049

Table 5.2.3 (b): Mistakes made by classifiers in SVM: W_P vs. W_P_ET

In this experiment, we have compared SVM using W_P and SVM_W_P_ET, i.e. Words combined with POS tags as features and words coupled with entities and POS tags as features. Table 5.2.3 (a) shows the counts of CFS, UFS, and NFS in verdicts (labels), predictions in SVM_W_P and SVM_W_P_ET.

From this, we see that W_P and W_P_ET have classified almost the same number of sentences in all classes. From Table 5.1, we see that precision (p_wavg) and recall (r_wavg) and f-measure (f_wavg) remains the almost the same in both the cases. Interestingly, we see that recall for UFS was low in the event of entities and POS tags as features individually. However, when the all the features have combined the recall improved by 12%.

Table 5.2.3 (b) shows the mistakes made by both the classifiers concerning the verdicts. Figure 5.2.3 illustrates the graph of all the combinations of predictions with the count of incorrectly predicted sentences in each category.

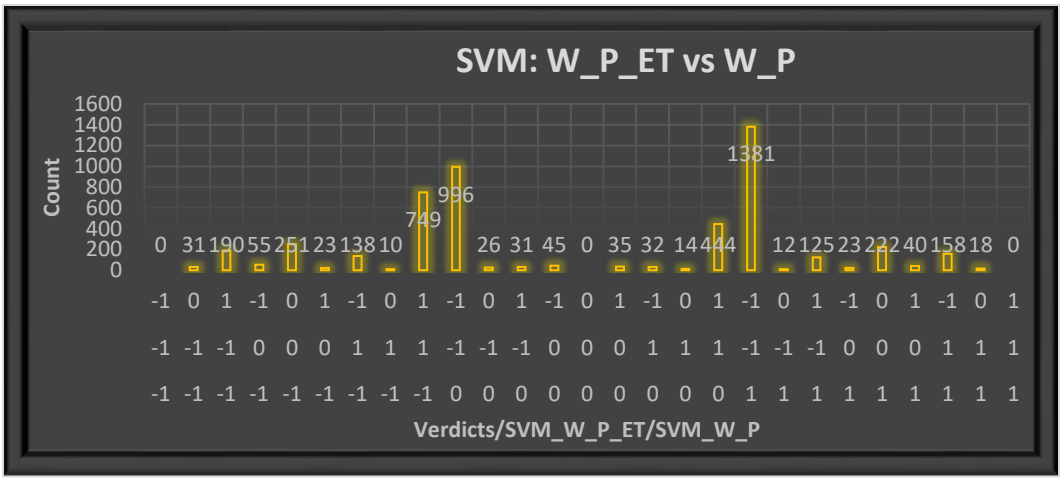

Figure 5.2.3: Incorrect predictions in SVM: W_P vs. W_P_ET

From above, we see that presence of entities does not influence the classification. The overall performance (f_wavg) is almost the same in both the cases (76%).

Finally, the influence of POS tags is as good as entities. Entities alone as a feature does not make a significant improvement in the classification. However, overall performance increases in case of UFS when all the features are combined. The maximum recall concerning CFS (64%) as well as the maximum overall recall (r_wavg) which is 77% in three class classification is achieved in W_P_ET using SVM.

5.3 RFC ANALYSIS FOR THREE CLASS

Case 1: W vs. W_ET

	Verdicts	RFC_W	RFC_W_ET
#CFS (1)	4843	2525	2794
#UFS (0)	2103	128	88
#NFS (-1)	13671	17964	17735
Total	20617	20617	20617

Table 5.3.1 (a): Classifier prediction count for CFS, UFS, and NFS in *RFC: W vs. W_ET*

Verdicts_NFS	462
Verdicts_UFS	2040
Verdicts_CFS	3074
Total	5576

Table 5.3.1 (b): Mistakes made by classifiers in RFC: W vs. W_ET

In this experiment, we have compared RFC using W and W_ET, i.e. Words as features and words combined with entities as features. Table 5.3.1 (a) shows the counts of CFS, UFS, and NFS in verdicts (labels), predictions in RFC _W and RFC _W_ET. From this, we see that both these classifiers have performed weakly in UFSs.

From Table 5.1, we see that the recall of NFSs is very high because the classifier predicts a significant number of sentences as NFS. We also see that the overall precision is lowest in these two classifiers. Recall (r_wavg) remains the same in both the cases, but the recall concerning CFS has improved about 4%. Overall, the performance of both the classifiers remains the same, but with minimal improvement in the case of entities.

Table 5.3.1 (b) shows the mistakes made by both the classifiers concerning the verdicts. When we look in detail into the errors made by classifiers, we see all the possible combinations of prediction. Figure 5.3.1 shows the graph of all the combinations of predictions with the count of incorrectly predicted sentences in each category.

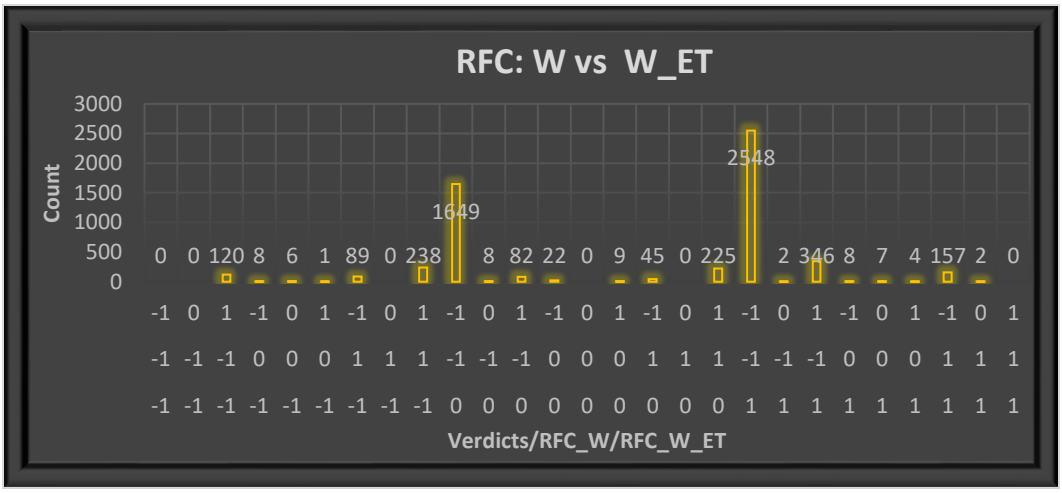

Figure 5.3.1: Incorrect predictions in RFC: W vs. W_ET

From the above graph, we see that RFC incorrectly predicted a significant number of sentences from CFS and UFS as NFS. This behavior of RFC is regardless of whether entities are features.

Finally, we see here that entities have not contributed significantly in this case. The overall performance (f_wavg) remains the same, but the recall of CFS improves by 4%.

Case 2: W_P vs. W_ET

	Verdicts	RFC_W_P	RFC_W_ET
#CFS (1)	4843	2919	2794
#UFS (0)	2103	134	88
#NFS (-1)	13671	17564	17735
Total	20617	20617	20617

Table 5.3.2 (a): Classifier prediction count for CFS, UFS, and NFS in RFC: W_P vs. W_ET

Verdicts_NFS	516
Verdicts_UFS	2055
Verdicts_CFS	3000
Total	5571

Table 5.3.2 (b): Mistakes made by classifiers in RFC: W_P vs. W_ET

In this experiment, we have compared RFC using W_P and W_ET, i.e. Words combined with POS tags as features and words combined with entities as features. Table 5.3.2 (a) shows the counts of CFS, UFS, and NFS in verdicts (labels), predictions in RFC_W_P and RFC_W_ET. From this, we see that both these feature sets contribute to the same extent.

From Table 5.1, precision(p_wavg), recall (r_wavg) and f-measure (f_wavg) remains the same in both the cases.

Table 5.3.2 (b) shows the mistakes made by both the classifiers concerning the verdicts. Figure 5.3.2 illustrates the graph with all the combination of predictions with the count of incorrectly predicted sentences in each category.

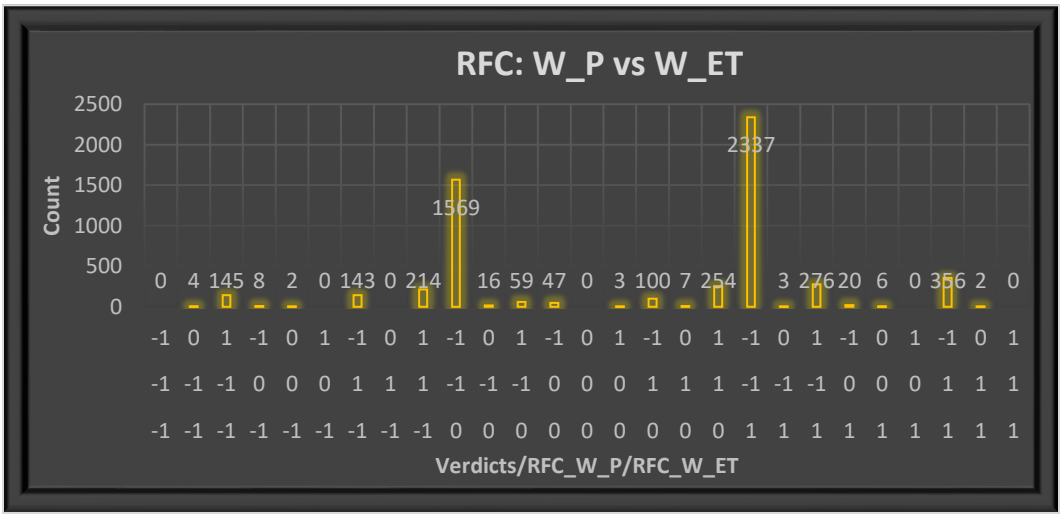

Figure 5.3.2: Incorrect predictions in RFC: W_P vs. W_ET

Finally, we see here both POS tags and entities have contributed to the same amount. The overall performance (f_wavg) remains the same (71%).

Case 3: W_P vs. W_P_ET

	Verdicts	RFC_W_P_ET	RFC_W_P
#CFS (1)	4843	3002	2919
#UFS (0)	2103	127	134
#NFS (-1)	13671	17488	17564
Total	20617	20617	20617

Table 5.3.3 (a): Classifier prediction count for CFS, UFS, and NFS in RFC: W_P vs. W_P_ET

Verdicts_NFS	441
Verdicts_UFS	2026
Verdicts_CFS	2778
Total	5245

Table 5.3.3 (b): Mistakes made by classifiers in RFC: W_P vs. W_P_ET

In this experiment, we have compared RFC using W_P and W_P_ET, i.e. Words combined with POS tags as features and Words combined with POS tags and entities as features. Table 5.3.3 (a) shows the counts of CFS, UFS, and NFS in verdicts (labels), predictions of RFC_W_P and RFC_W_ET.

From this, we see that presence of entities has not further improved the classifier from the performance of POS tags. From Table 5.1, Precision(p_wavg), Recall (r_wavg) and f-measure (f_wavg) remains the same in both the cases.

Table 5.3.3 (b) shows the mistakes made by both the classifiers concerning the verdicts. When we look in detail into the errors made by classifiers, we see all the possible combinations of prediction. Figure 5.3.3 concerning the graph with all the combinations of predictions with the count of incorrectly predicted sentences in each category.

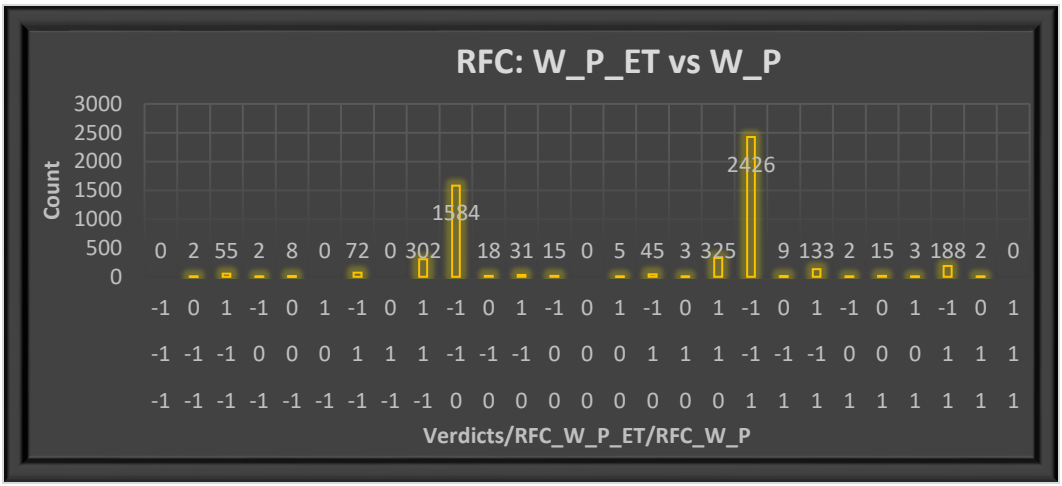

Figure 5.3.3: Incorrect predictions in RFC: W_P vs. W_P_ET

Finally, we see here that the entities helped in CFS when used individually. Otherwise, they do not influence RFC. RFC has the worst performance in all of the three-class classification. It also has the worst recall for UFS due to its bias towards NFS. Moreover, the best in RFC can be achieved using POS tags itself. The use of entities does not improve in the presence of POS tags.

Chapter 6 ANALYSIS OF TWO CLASS CLASSIFIERS

In this section, we analyze the classifiers by their performance and incorrect predictions of sentences. To understand the contribution and influence of entities on classifiers, we have chosen four sets of features using which the comparison of classification results can be made reasonable.

In this analysis, we compare the results of classifiers in 3 ways:

1. W vs. W_ET
2. W_P vs. W_ET
3. W_P vs. W_P_ET

All the classifiers are compared as above. Cumulatively, they form 9 experiments. Section 6.1 explains NBC two-class analysis and above mentioned three comparisons are subsections in 6.1. Similarly, 6.2 and 6.3 explain SVM two-class analysis and RFC two-class analysis respectively. The performance results in terms of precision, recall, and f-measure for all the classifiers and feature sets are shown together in Table 6.1.

Further we extract sentences from classifiers in comparison where the predictions were made incorrectly with respect to verdicts (labels). This yields us with 8 cases, that is 2 classes (N_UFS, CFS) of verdicts, feature set 1 and feature set 2 each. This extraction helps us to make a deeper study on sentences which are classified correctly and incorrectly due to presence of entities.

algorithm	features	p_N_UFS	p_CFS	p_wavg	r_N_UFS	r_CFS	r_wavg	f_N_UFS	f_CFS	f_wavg
SVM	W	0.87	0.69	0.83	0.92	0.55	0.83	0.89	0.59	0.82
SVM	W_ET	0.85	0.75	0.83	0.95	0.46	0.83	0.90	0.55	0.82
SVM	W_P	0.87	0.72	0.84	0.93	0.56	0.84	0.90	0.63	0.84
SVM	W_P_ET	0.88	0.70	0.84	0.92	0.60	0.84	0.90	0.64	0.84
RFC	W	0.82	0.86	0.83	0.99	0.29	0.82	0.89	0.43	0.79
RFC	W_ET	0.82	0.85	0.83	0.98	0.31	0.82	0.90	0.45	0.79
RFC	W_P	0.82	0.86	0.83	0.98	0.32	0.83	0.90	0.46	0.80
RFC	W_P_ET	0.83	0.85	0.83	0.98	0.34	0.83	0.90	0.48	0.80
NBC	W	0.82	0.84	0.82	0.98	0.29	0.82	0.89	0.43	0.78
NBC	W_ET	0.84	0.76	0.82	0.96	0.42	0.83	0.90	0.54	0.81
NBC	W_P	0.87	0.65	0.82	0.91	0.57	0.83	0.89	0.61	0.82
NBC	W_P_ET	0.88	0.64	0.82	0.90	0.58	0.83	0.89	0.61	0.82

Table 6.1: Classifier performance of two-class classification for Analysis

6.1 NBC ANALYSIS FOR TWO CLASS

Case 1: W vs. W_ET

	Verdicts	NBC_W	NBC_W_ET
#CFS (1)	4843	1671	2666
#N_UFS (-1)	15774	18946	17951
Total	20617	20617	20617

Table 6.1.1 (a): Classifier prediction count for CFS and N_UFS in Two-Class NBC: W vs. W_ET

Verdicts_N_UFS	698
Verdicts_CFS	3542
Total	4240

Table 6.1.1 (b): Mistakes made by classifiers in Two-Class NBC: W vs. W_ET

In this experiment, we have compared two-class NBC using W and W_ET, i.e. Words as features and words combined with entities as features. Table 6.1.1 (a) shows the counts of CFS, N_UFS in verdicts (labels), predictions of NBC_W and NBC_W_ET. From this, we see that the presence of entities has helped classify more CFSs.

From Table 6.1, recall (r_wavg) remains the same in both the cases, but the recall concerning CFS has improved about 13%. Overall, the performance of both the classifiers remains the same, but with minimal improvement in the case of entities.

Table 6.1.1 (b) shows the mistakes made by both the classifiers concerning the verdicts. We see that majority of errors are made in CFS. When we look in detail into the errors made by classifiers, we have to look at all the possible combinations of prediction. Table 6.1.1 (c) shows all the combinations of predictions with the count of incorrectly predicted sentences in each category. Figure 6.1.1 illustrates the graph of Table 6.1.1 (c).

Verdicts	NBC_W	NBC_W_ET	Count
-1	-1	-1	0
-1	-1	1	435
-1	1	-1	49
-1	1	1	214
1	-1	-1	2719
1	-1	1	716
1	1	-1	107
1	1	1	0
		Total	4240

Table 6.1.1 (c): All possible combinations for prediction in Two-Class NBC: W vs. W_ET

From above table we see that the maximum errors were made in (1, -1, -1) i.e. both the classifiers predicted it to be N_UFS whereas the sentence was labeled CFS. The combinations where entities have helped are (-1, 1, -1) and (1, -1, 1) which sums up to 765 sentences. The combinations where entities have harmed the classifier are (-1, -1, 1) and (1, 1, -1) which sums up to 542 sentences.

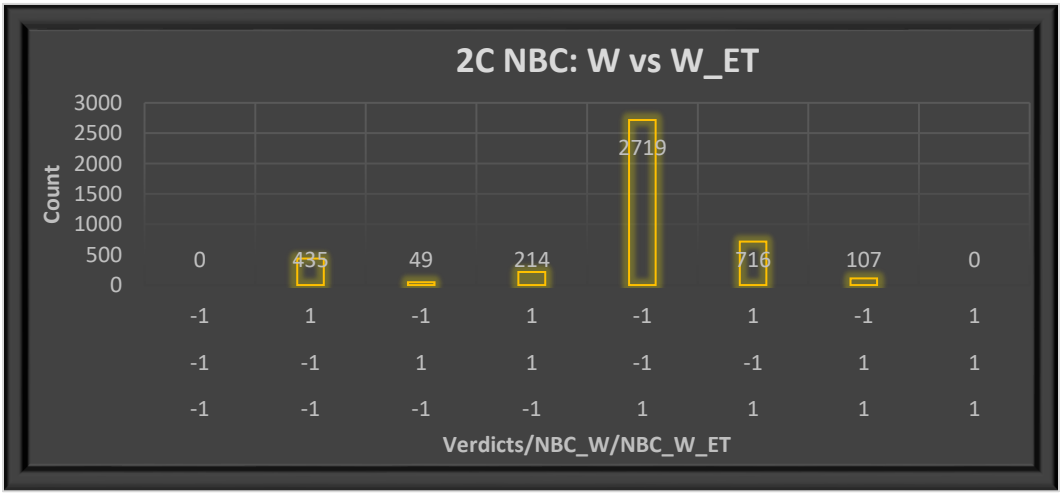

Figure 6.1.1: Incorrect predictions in 2C NBC: W vs. W_ET

From the above graph, we see that NBC predicts more sentences towards NFS regardless of whether it has entities as features. 65% of errors are in this category.

Finally, we see that entities have contributed very little in this case. They have helped classify more CFS. The overall performance (f_wavg) improves only by 3%.

Case 2: W_P vs. W_ET

	Verdicts	NBC_W_P	NBC_W_ET
#CFS (1)	4843	4235	2666
#N_UFS (-1)	15774	16382	17951
Total	20617	20617	20617

Table 6.1.2 (a): Classifier prediction count for CFS and N_UFS in two-class NBC: W_P vs. W_ET

Verdicts_NFS	1636
Verdicts_CFS	3013
Total	4649

Table 6.1.2 (b): Mistakes made by classifiers in two-class NBC: W_P vs. W_ET

In this experiment, we have compared two-class NBC using W_P and W_ET, i.e. Words combined with POS tags as features and words combined with entities as features. Table 6.1.2 (a) shows the counts of CFS, N_UFS in verdicts (labels), predictions of NBC_W_P and NBC_W_ET.

From this, we see that POS tags make a better impact on CFS than entities. From Table 6.1, recall of CFS in the case of POS tags is 15% more than that of entities. The recall (r_wavg) remains the same in both the cases, as it is the weighted average and N_UFS influence more. Overall, the weighted performance of both the classifiers remains the same.

Table 6.1.2 (b) shows the mistakes done by both the classifiers concerning the verdicts. Figure 6.1.2 illustrates the graph with all the combination of predictions with the count of incorrectly predicted sentences in each category.

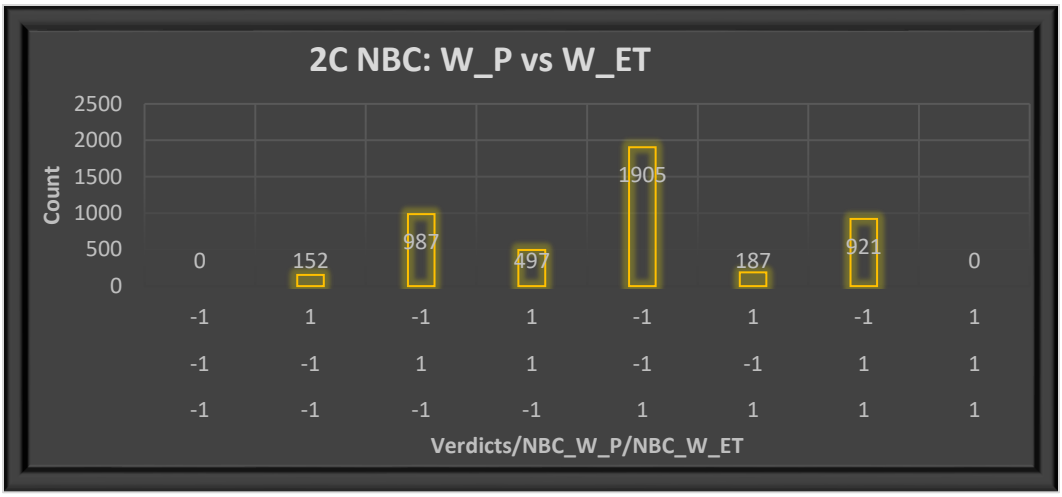

Figure 6.1.2: Incorrect predictions in 2C NBC: W_P vs. W_ET

Also, the combination (-1, 1, -1) which shows the assistance of entities is almost the same amount as (1, 1, -1) in which they harm.

Finally, we see that entities and POS tags have not improved the classifier in performance compared with only words as features (Table 6.1). Even though the overall performance (f_wavg) remains close to each other, POS tags have performed better than entities in classifying more CFSs (r_CFS in Table 6.1).

Case 3: W_P vs. W_P_ET

	Verdicts	NBC_W_P	NBC_W_P_ET
#CFS (1)	4843	4235	4399
#N_UFS (-1)	15774	16382	16218
Total	20617	20617	20617

Table 6.1.3 (a): Classifier prediction count for CFS and N_UFS in two-class NBC: W_P vs. W_P_ET

Verdicts_NFS	1752
Verdicts_CFS	2226
Total	3978

Table 6.1.3 (b): Mistakes made by classifiers in two-class NBC: W_P vs. W_P_ET

In this experiment, we have compared two-class NBC using W_P and W_P_ET, i.e. Words combined with POS tags as features and Words coupled with entities and POS tags as features. Table 6.1.3 (a) shows the counts of CFS, N_UFS in verdicts (labels), predictions of NBC_W_P and NBC_W_P_ET.

From Table 6.1, precision(p_wavg), recall (r_wavg) and f-measure (f_wavg) remains the same in both the cases. Table 6.1.3 (b) shows the mistakes made by both the classifiers concerning the verdicts. Figure 6.1.3 illustrates the graph with all the combinations of predictions with the count of sentences in each category.

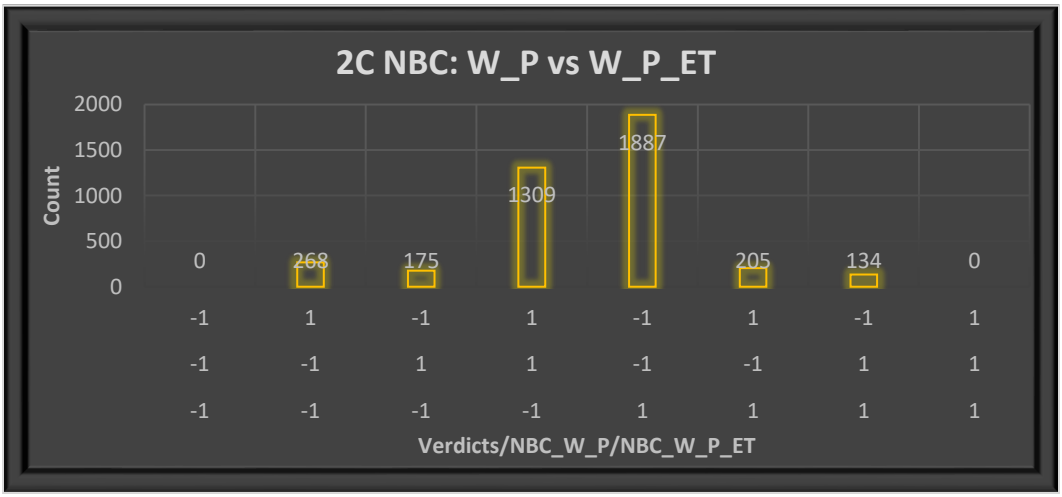

Figure 6.1.3: Incorrect predictions in 2C NBC: W_P vs. W_P_ET

Finally, we see that entities have not improved the classifier in the performance with W_P as features (Table 6.1). The overall performance (f_wavg) remains the same in both the case. The best result for two-class NBC is achieved in these two cases. The use of entities is not essential in achieving the best performance in NBC two-class.

6.2 SVM ANALYSIS FOR TWO CLASS

Case 1: W vs. W_ET

	Verdicts	SVM_W	SVM_W_ET
#CFS (1)	4843	3961	3073
#N_UFS (-1)	15774	16656	17544
Total	20617	20617	20617

Table 6.2.1 (a): Classifier prediction count for CFS and N_UFS in Two-Class SVM: W vs. W_ET

Verdicts_NFS	1553
Verdicts_CFS	2869
Total	4422

Table 6.2.1 (b): Mistakes made by classifiers in Two-Class SVM: W vs. W_ET

In this experiment, we have compared two-class SVM using W and W_ET, i.e. Words as features and words combined with entities as features. Table 6.2.1 (a) shows the counts of CFS, N_UFS in verdicts (labels), predictions of SVM_W and SVM_W _ET. We see that the use of entity feature has classified a lesser number of CFS.

From Table 6.1, recall of CFS (r_CFS) has decreased in case of entities. Precision(p_wavg), recall (r_wavg) and f-measure (f_wavg) remains the same in both the cases. Table 6.2.1 (b) shows the mistakes made by both the classifiers concerning the verdicts. Figure 6.2.1 illustrates the graph with all the combinations of predictions with the count of incorrectly predicted sentences in each category.

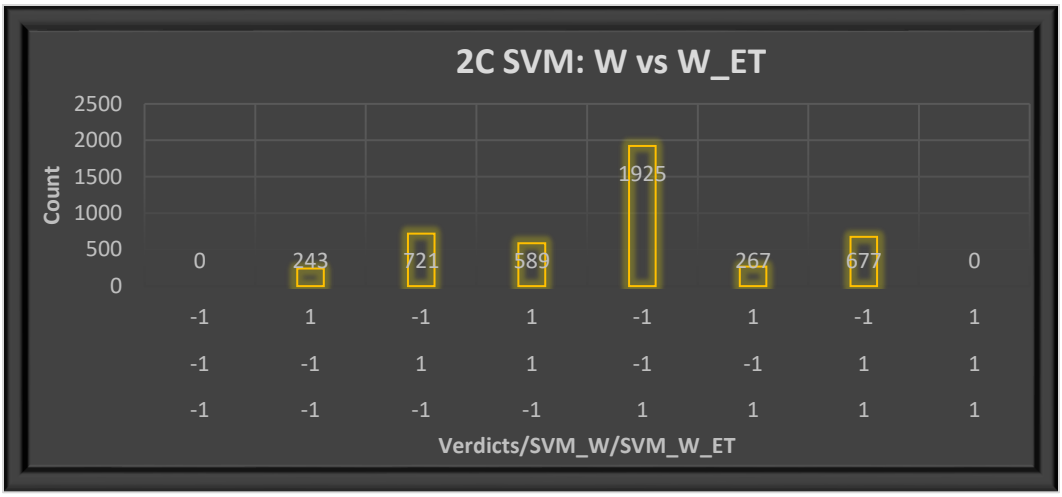

Figure 6.2.1: Incorrect predictions in 2C SVM: W vs. W_ET

Finally, we see that entities do not contribute to the classifier. They classify lesser CFS than words as features. They harm the classification of CFS as the recall of CFS (r_CFS) falls. The overall performance (f_wavg) remains the same in both the case.

Case 2: W_P vs. W_ET

	Verdicts	SVM_W_P	SVM_W_ET
#CFS (1)	4843	3829	3073
#N_UFS (-1)	15774	16788	17544
Total	20617	20617	20617

Table 6.2.2 (a): Classifier prediction count for CFS and N_UFS in two-Class SVM: W_P vs. W_ET

Verdicts_NFS	1326
Verdicts_CFS	2787
Total	4113

Table 6.2.2 (b): Mistakes made by classifiers in two-Class SVM: W_P vs. W_ET

In this experiment, we have compared two-class SVM using W_P and W_ET, i.e. Words combined with POS tags as features and words combined with entities as features. Table 6.2.2 (a) shows the counts of CFS, N_UFS in verdicts (labels),

predictions of SVM_W_P and SVM_W _ET. We see that the use of POS tags helps classify more CFS.

From Table 6.1, recall of CFS (r_CFS) in POS tags is better than that of entities. Table 6.2.2 (b) shows the mistakes made by both the classifiers concerning the verdicts. Figure 6.2.2 illustrates the graph with all the combinations of predictions with the count of incorrectly predicted sentences in each category.

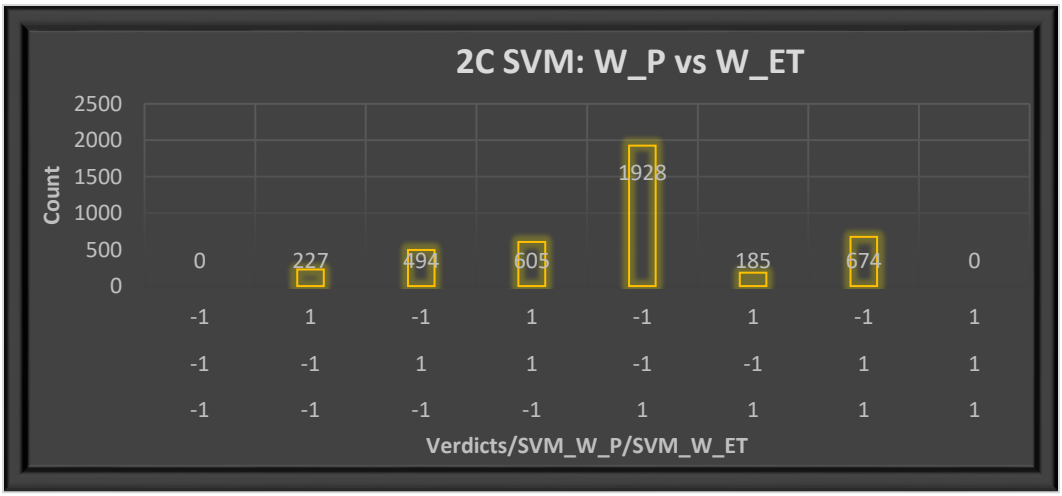

Figure 6.2.2: Incorrect predictions in 2C SVM: W_P vs. W_ET

Finally, neither POS tags or entities help the classifier perform better. The POS tags have almost the same result as only words as features.

Case 3: W_P vs. W_P_ET

	Verdicts	SVM_W_P	SVM_W_P_ET
#CFS (1)	4843	3829	4186
#N_UFS (-1)	15774	16788	16431
Total	20617	20617	20617

Table 6.2.3 (a): Classifier prediction count for CFS and N_UFS in Two-Class SVM: W_P vs. W_ET

Verdicts_NFS	1460
Verdicts_CFS	2252
Total	3712

Table 6.2.3 (b): Mistakes made by classifiers in Two-Class SVM: W_P vs. W_ET

In this experiment, we have compared two-class SVM using W_P and W_P_ET, i.e. Words combined with POS tags as features and words combined with entities and POS tags as features. Table 6.2.3 (a) shows the counts of CFS, N_UFS in verdicts (labels), predictions of SVM_W_P and SVM_W _P_ET.

From Table 6.1, precision(p_wavg), recall (r_wavg) and f-measure (f_wavg) remains the same in both the cases. Recall of CFS (r_CFS) in W_P_ET is better than that of POS tags. Table 6.2.3 (b) shows the mistakes done by either of the classifiers concerning the verdicts. Figure 6.2.3 illustrates the graph with all the combinations of predictions with the count of incorrectly predicted sentences in each category.

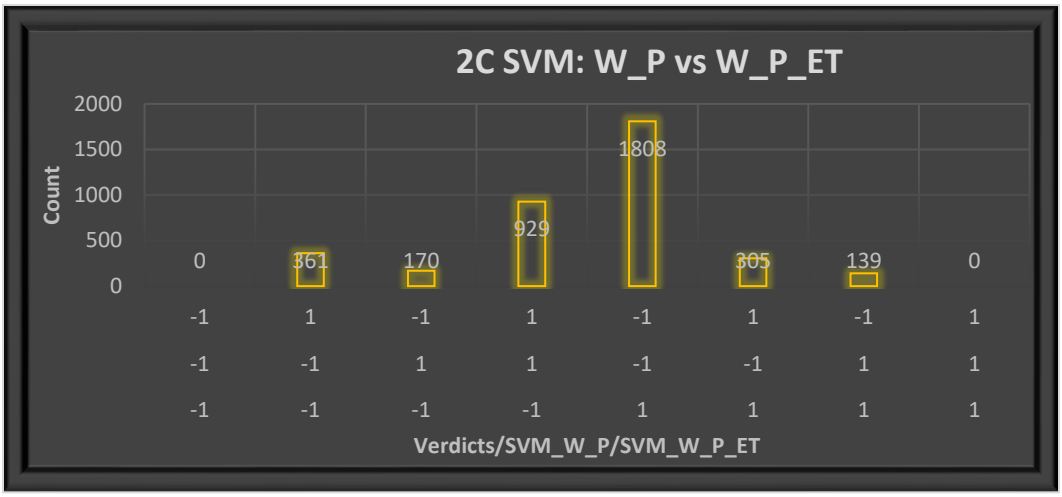

Figure 6.2.3: Incorrect predictions in 2C SVM: W_P vs. W_P_ET

The SVM using W_P_ET has achieved the best result in two-class classification. It yields weighted recall of 84% and maximum recall for CFS of 60%. Use of POS tags, entities and both do not influence SVM to improve the classification any further.

6.3 RFC ANALYSIS FOR TWO CLASS

Case 1: W vs. W_ET

	Verdicts	RFC_W	RFC_W_ET
#CFS (1)	4843	1616	1764
#N_UFS (-1)	15774	19001	18853
Total	20617	20617	20617

Table 6.3.1 (a): Classifier prediction count for CFS and N_UFS in two-class RFC: W vs. W_ET

Verdicts_NFS	310
Verdicts_CFS	3561
Total	3871

Table 6.3.1 (b): Mistakes made by classifiers in two-class RFC: W vs. W_ET

In this experiment, we have compared two-class RFC using W and W_ET, i.e. Words as features and words combined with entities as features. Table 6.3.1 (a) shows the counts of CFS, N_UFS in verdicts (labels), predictions of RFC_W and RFC_W_ET.

From Table 6.1, precision(p_wavg), recall (r_wavg) and f-measure (f_wavg) remains the same in both the cases. Even though the recall of CFS is the very low, the r_wavg remains high due to the weight of N_UFS. Table 6.3.1 (b) shows the mistakes done by either of the classifiers concerning the verdicts. Figure 6.3.1 illustrates the graph with all the combinations of predictions with the count of incorrectly predicted sentences in each category.

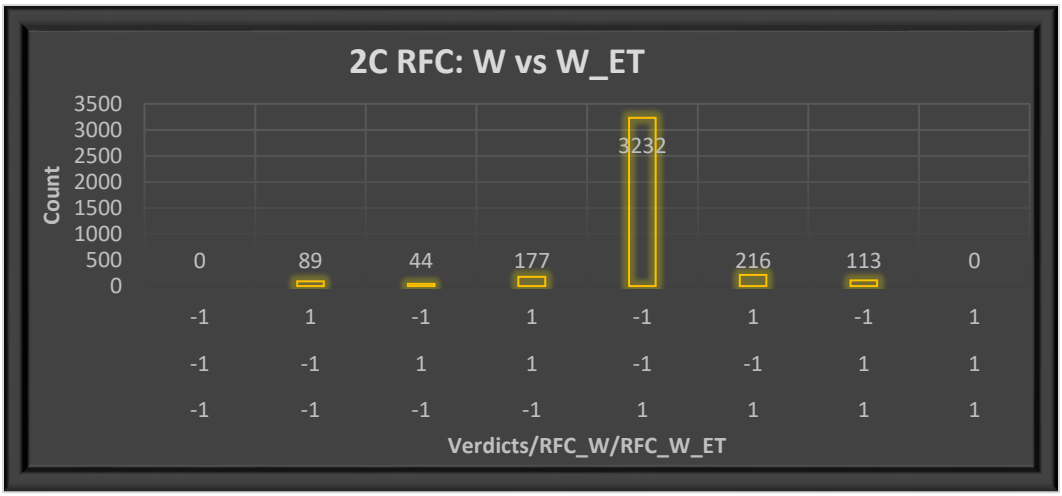

Figure 6.3.1: Incorrect predictions in 2C RFC: W vs. W_ET

From the above graph, we see that RFC predicts more sentences towards NFS regardless of whether it has entities as features. 92% of errors are in this category.

Finally, we see here that entities have not contributed in this case. The overall performance (f_wavg) has not improved.

Case 2: W_P vs. W_ET

	Verdicts	RFC_W_P	RFC_W_ET
#CFS (1)	4843	1797	1764
#N_UFS (-1)	15774	18820	18853
Total	20617	20617	20617

Table 6.3.2 (a): Classifier prediction count for CFS and N_UFS in two-class RFC: W_P vs. W_ET

Verdicts_NFS	360
Verdicts_CFS	3558
Total	3918

Table 6.3.2 (b): Mistakes made by classifiers in two-class RFC: W_P vs. W_ET

In this experiment, we have compared two-class RFC using W_P and W_ET, i.e. Words combined with POS tags as features and words coupled with entities as features. Table 6.3.2 (a) shows the counts of CFS, N_UFS in verdicts (labels), predictions of RFC_W_P and RFC_W_ET.

From Table 6.1, both POS tags and entities do not contribute to the classifier performance. Precision(p_wavg), recall (r_wavg) and f-measure (f_wavg) remains the same in both the cases. Table 6.3.2 (b) shows the mistakes done by either of the classifiers concerning the verdicts. Figure 6.3.2 illustrates the graph with all the combinations of predictions with the count of sentences in each category.

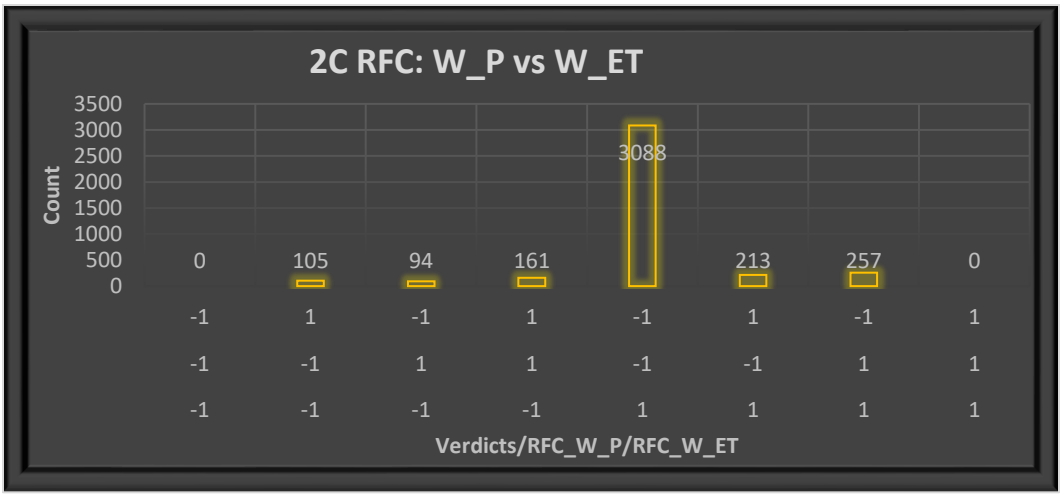

Figure 6.3.2: Incorrect predictions in 2C RFC: W_P vs. W_ET

Finally, POS tags and entities contribute very less to the classification. The overall performance (f_wavg) was not improved.

Case 3: W_P vs. W_P_ET

	Verdicts	RFC_W_P	RFC_W_P_ET
#CFS (1)	4843	1797	1925
#N_UFS (-1)	15774	18820	18692
Total	20617	20617	20617

Table 6.3.3 (a): Classifier prediction count for CFS and N_UFS in Two-Class RFC: W_P vs. W_P_ET

Verdicts_NFS	329
Verdicts_CFS	3400
Total	3729

Table 6.3.3 (b): Mistakes made by classifiers in Two-Class RFC: W_P vs. W_P_ET

In this experiment, we have compared two-class RFC using W_P and W_P_ET, i.e. Words combined with POS tags as features and words coupled with entities and POS tags as features. Table 6.3.3 (a) shows the counts of CFS, N_UFS in verdicts (labels), predictions of RFC_W_P and RFC_W_P_ET.

From Table 6.1, using of entities did not show any improvement in the classifier performance. Precision(p_wavg), recall (r_wavg) and f-measure (f_wavg) remains the same in both the cases. Table 6.3.3 (b) shows the mistakes made by both the classifiers concerning the verdicts. Figure 6.3.3 illustrates the graph with all the combinations of predictions with the count of incorrectly predicted sentences in each category.

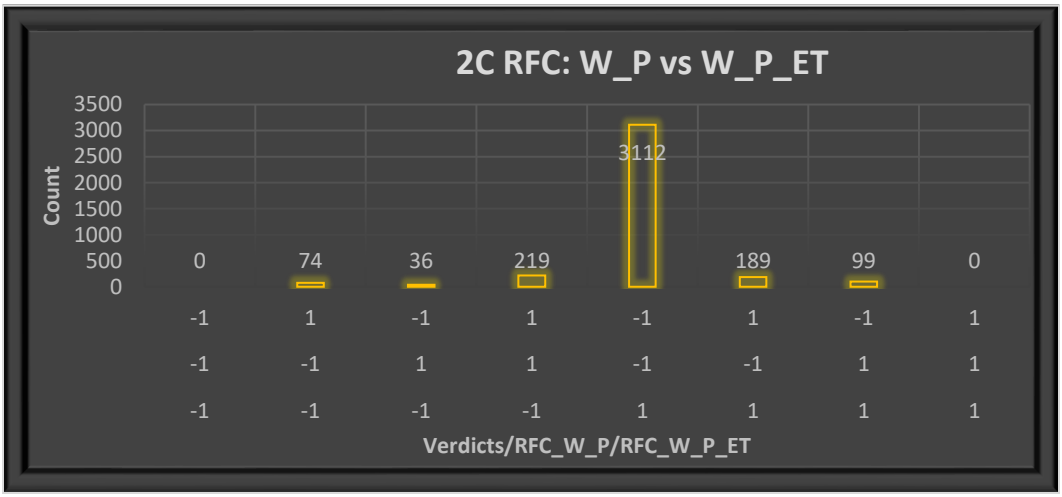

Figure 6.3.3: Incorrect predictions in 2C RFC: W_P vs. W_P_ET

Finally, RFC performed the worst in all the two-class classification. It had the worst recall for CFS. We see from the graphs that the algorithm is erroneous with CFS in any case of the feature set. Entities do not improve the performance in RFC.

Chapter 7 REASONING

In Chapter 5 and 6 we see the performance of classifiers with various feature sets. We saw particular cases where entities have helped and also cases where they do not contribute to improving the performance of classifiers. The reasons for such behavior can vary by problem, domain, and classifications. In our project, we use sentences from presidential debates as our dataset, extracted features from the sentences into vectors and used standard classifiers. Below are a few reasons that can be explained by this work.

1. The possibility of wrong verdicts for the sentences: When we study the incorrectly predicted sentences of classifiers, we see that verdicts for some of the sentences are wrong. This errors may be due to the majority vote between the top labelers. Ambiguous sentences are usually labeled as UFS.
2. Curse of learning: This is mainly due to the imbalance in the classes of the datasets. In our data, we find that NFS and UFS constitute to over 75% and CFS is only about 25%. This imbalance biases the classifiers towards the NFS class.
3. Limited training data: The training sentences are limited as the domain is restricted to presidential debates. It is hard to find or construct more labeled sentences of presidential debates. Particularly in the case of CFS, the sentences are very limited.
4. Entities are sparse: Some sentences do not have a single entity. Such absence makes the entity features sparse whereas the POS tags are available in every sentence.

These are some of the reasons the performance of the classifiers is influenced and for entities to not contribute to the classifiers.

Chapter 8 CONCLUSION

In this project, we have extracted named entities from sentences of presidential debates transcripts using Illinois NE tagger. We have experimented the factual claim classifier using three classifiers with four different feature sets by two classification types. These results assist us in understanding scenarios where the entities have helped in the classification process. We saw that entities when used individually helped in better classifying CFS. We saw this in results of NBC. We also saw that POS tags perform better as features than entities in various cases. However, the performance of classifiers was not improved significantly by using either POS tags or entities or both as features. We saw that the two-class performances were better than three-class performances and entities showed the same behavior in both classification types. We have also analyzed the errors made by the classifiers by comparing two feature sets at a time.

References

[1] N. Hassan, C. Li, and M. Tremayne. Detecting check-worthy factual claims in presidential debates. In CIKM, pages 1835–1838, 2015.

[2] N. Hassan, B. Adair, J. T. Hamilton, C. Li, M. Tremayne, J. Yang, and C. Yu. The quest to automate fact-checking. Proceedings of the 2015 Computation+Journalism Symposium, 2015.

[3] N. Hassan, M. Tremayne, F. Arslan, and C. Li. Comparing automated factual claim detection against judgments of journalism organizations. In Computation+Journalism Symposium, 2016.

[4] Samet Atdağ and Vincent Labatut. A Comparison of Named Entity Recognition Tools Applied to Biographical Texts. 2nd International Conference on Systems and Computer Science, Villeneuve d'Ascq (FR), 2013.

[5] S. Sulaiman, R. Abdul Wahid, S. Sarkawi, and N. Omar. Using Stanford NER and Illinois NER to Detect Malay Named Entity Recognition. International Journal of Computer Theory and Engineering, Vol. 9, No. 2, April 2017.

[6] L. Ratinov and D. Roth, "Design challenges and misconceptions in named entity recognition," in *13th Conference on Computational Natural Language Learning*, 2009, pp. 147-155.

[7] Mark Sammons, Tom Redman, and Dan Roth. Illinois Named Entity Recognizer: Addendum to Ratinov and Roth ’09 reporting improved results. 2017

[8] Cognitive computation group. [Online]. Available: https://cogcomp.cs.illinois.edu/page/software_view/Curator

[9] David Nadeau, Satoshi Sekine. A survey of named entity recognition and classification. National Research Council Canada / New York University.

[10] Joachim Giard, Jérôme Ambroise, Jean-Luc Gala, and Benoît Macq. Regression applied to protein binding site prediction and comparison with classification. BMC Bioinformatics, 2009.

[11] McCallum, A., & Nigam, K. (1998). A comparison of event models for naive Bayes text classification. In AAAI-98 Workshop on Learning for Text Categorization. Tech. rep. WS-98-05, AAAI Press.

[12] Stanford NLP Named Entity Recognition Results. [Online]. Available: https://nlp.stanford.edu/projects/project-ner.shtml

Biographical Information

Abu Ayub Ansari Syed is a student of Computer Science who received his Bachelors in Engineering from M.S. Ramaiah Institute of Technology, Bangalore, India in 2014. He has completed Masters in Science in Department of Computer Science from The University of Texas at Arlington, the United States in 2017. His interests are in Data Mining, Machine Learning, and Data Analysis.