SPAM DETECTION

A Project Report

submitted in partial fulfillment of the requirements

of

Edunet Certificate

by

Mitali Saini,

Student of

Aryabhatta College

ACKNOWLEDGEMENT

The success and final outcome of this project required a lot of guidance and assistance and I'm extremely fortunate to have got this all along the completion of my project work. Whatever I've done is only due to such guidance and assistance and I would not forget to thank them.

I would like to pay thanks to my teacher of the Techsaksham, for giving me an opportunity to do the capstone project work in TSP course on "Industrial Artificial Intelligence with cloud computing and providing me all guidance and support which made me complete the project on time. I'm extremely grateful to him for providing such a great support and guidance.

I'm thankful to all those who have helped me completing this project including my teachers, parents and so on. I've done this project on my own with the help of Internet.

I'm thankful to and fortunate enough to get constant encouragement, support and guidance from all the teaching staffs which helped me in successfully completing my project.

Mitali Saini

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Introduction of the title

In today's digital age, managing email inboxes has become increasingly challenging due to proliferation of spam messages. The project, "Spam Detection", aims to address the issue by implementing advanced algorithms and machine learning techniques to identify and filter out spam emails effectively. By doing so, we not only streamline the email experience but also enhance cybersecurity measures, ensuring that users' data remains safe and their inboxes remain clutter free. Let's join us in this journey towards a cleaner and safer email environment.

1.2 Problem Statement

The influx of spam emails continues to be a significant challenge for individuals and organizations, leading to cluttered inboxes, wasted time, and potential security risks. Current spam filters often struggle to differentiate between legitimate and malicious emails, resulting in missed important messages or allowing harmful content to reach users' inboxes.

The project, "Spam Detection," addresses this critical issue by developing and implementing an advanced spam detection system. By leveraging machine learning algorithms, natural language processing techniques, and data analysis, we aim to create a robust solution that accurately identifies and filters out spam emails while minimizing false positives.

CHAPTER 3

PROPOSED METHODOLOGY

2.1 System Design

2.1.1 Data Collection

The dataset selected is taken from Kaggle repository (spam.csv) having 2 attributes one includes categorical values (has spam or not) and other one includes email text.

```
df.head(5)
```

	target	text
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...

2.1.2 Preprocessing

Data has been preprocessed by adding new attributes using natural language processing and wordcloud. The new attributes created are: num_characters (contains the number of words in a particular row), num_words (contains the number of words in a particular row) and num_sentences (contains the number of sentences in a particular row) and transformed_text by using NLTK tools.

2.1.3 Model Selection

Model is being divided into 2 datasets: training data and testing data by using train_test_split. By using classification techniques, model is going to predict the results that the email has spam or not, such as Naïve Bayes, Support Vector Machine, Decision Tree, K nearest neighbors, Logistic Regression. Various ensemble techniques

are also being used such as Random Forest, XGB Classifier, AdaBoost Classifier, Gradient Boosting Classifier and so on.

2.1.4 Model Evaluation

The model is being evaluated using the accuracy and precision of each machine learning Techniques.

2.1.5 Deployment

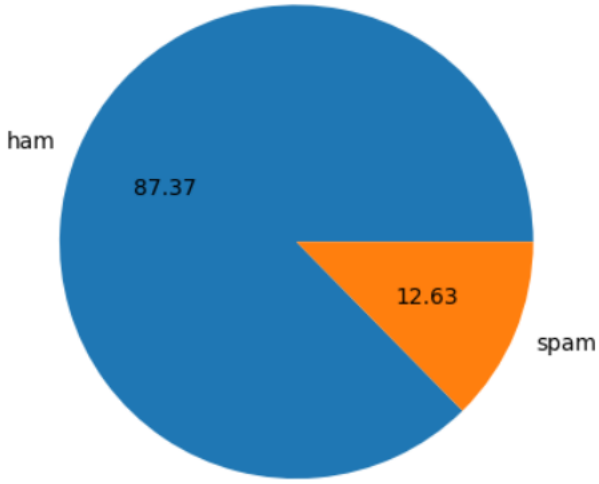
For deployment, Python's streamlit module is used that gives a web interface to predict the output.

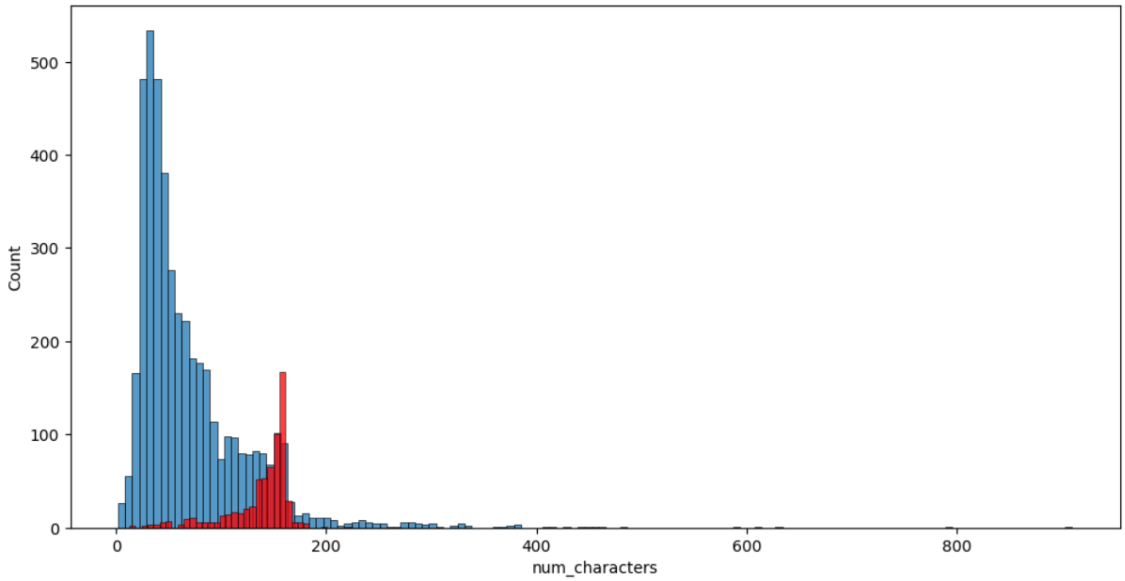
CHAPTER 3

IMPLEMENTATION and RESULT

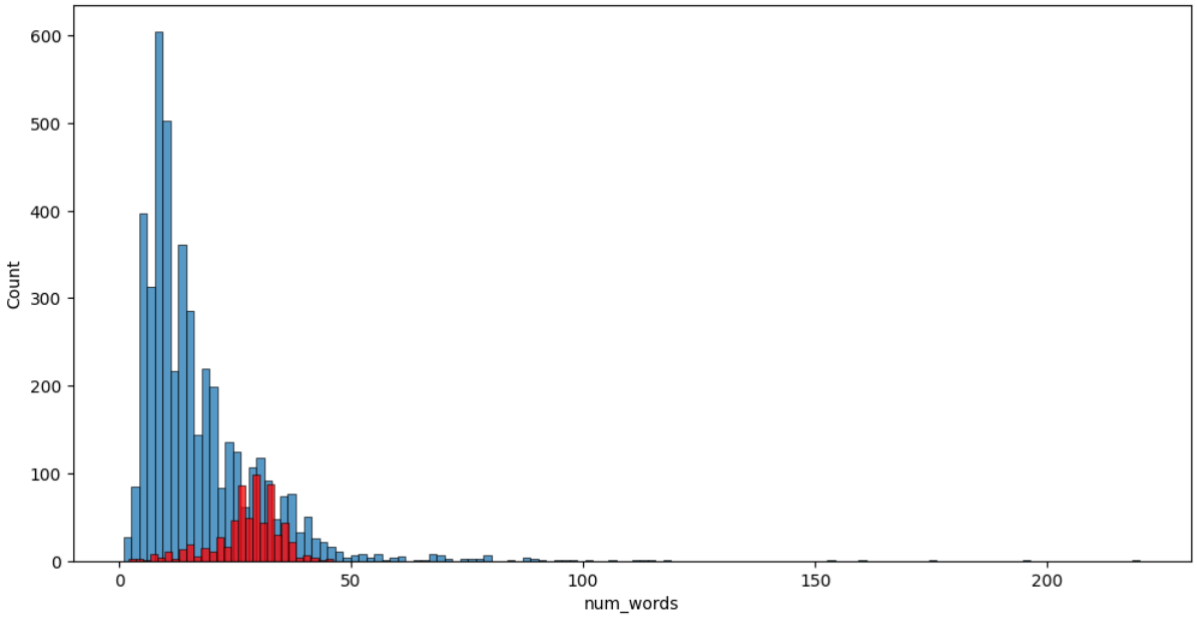
3.1 EDA results

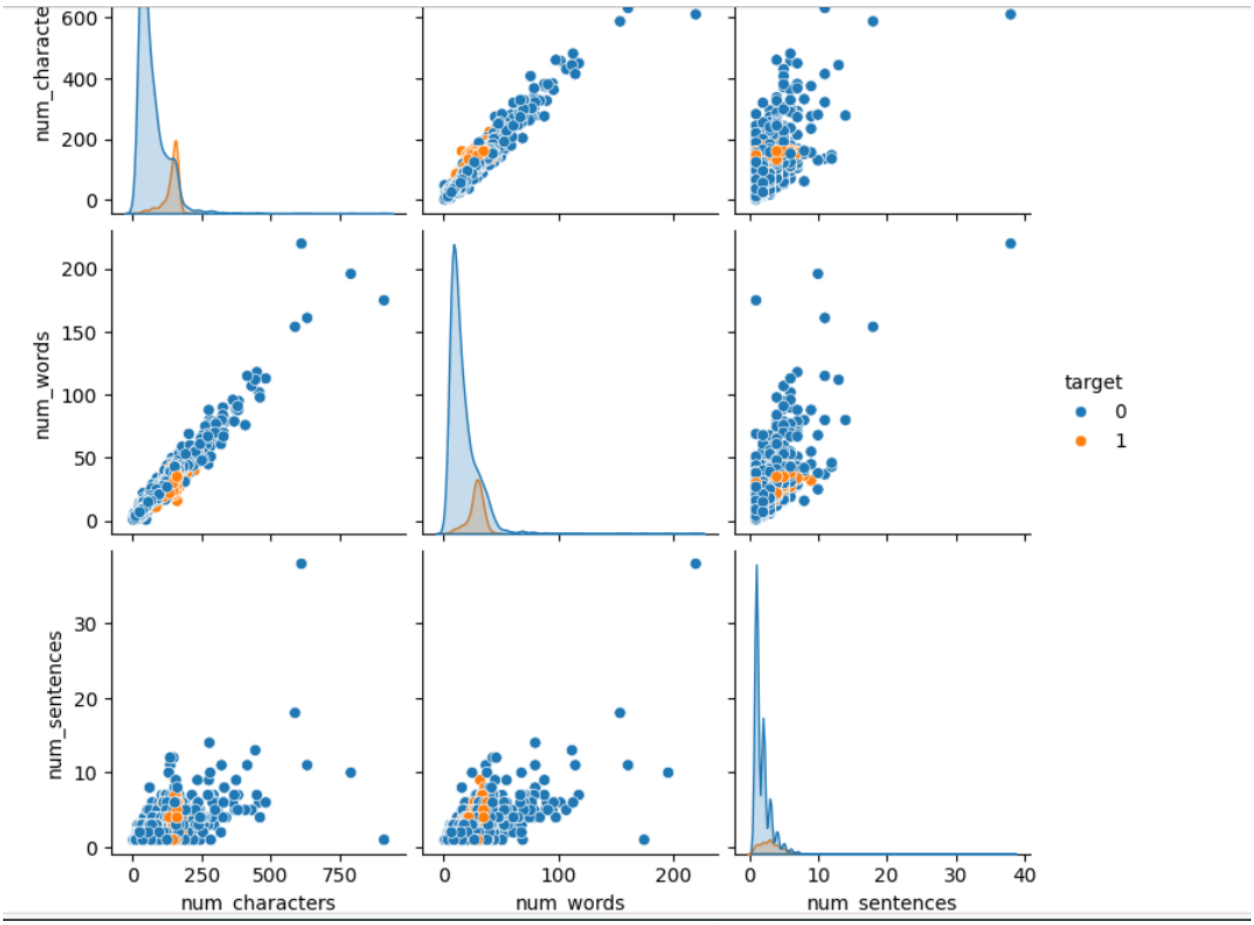
	target	text	num_characters	num_words	num_sentences
0	0	Go until jurong point, crazy.. Available only ...	111	24	2
1	0	Ok lar... Joking wif u oni...	29	8	2
2	1	Free entry in 2 a wkly comp to win FA Cup fina...	155	37	2
3	0	U dun say so early hor... U c already then say...	49	13	1
4	0	Nah I don't think he goes to usf, he lives aro...	61	15	1

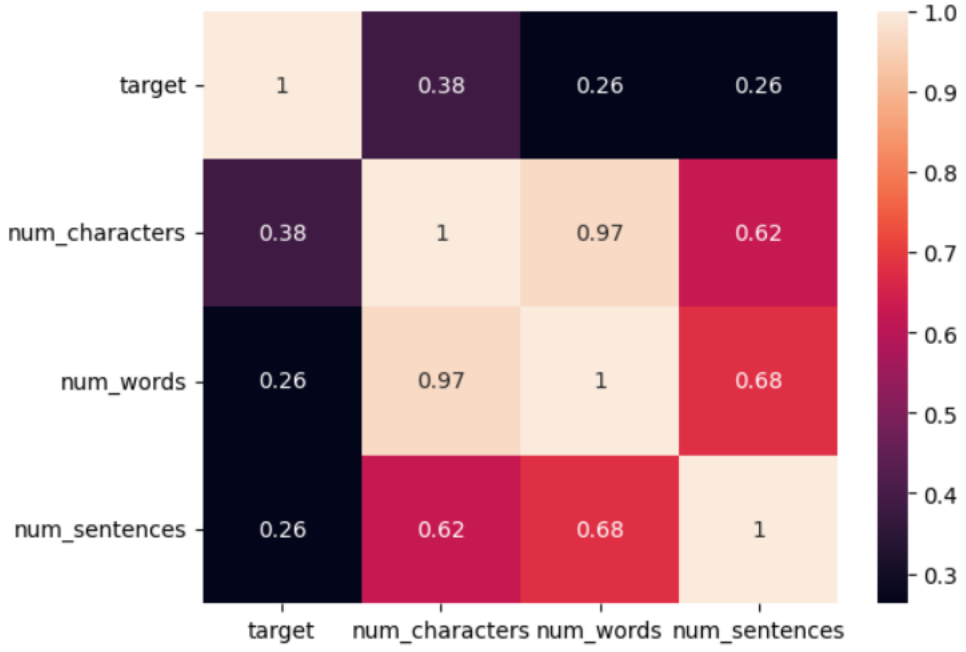


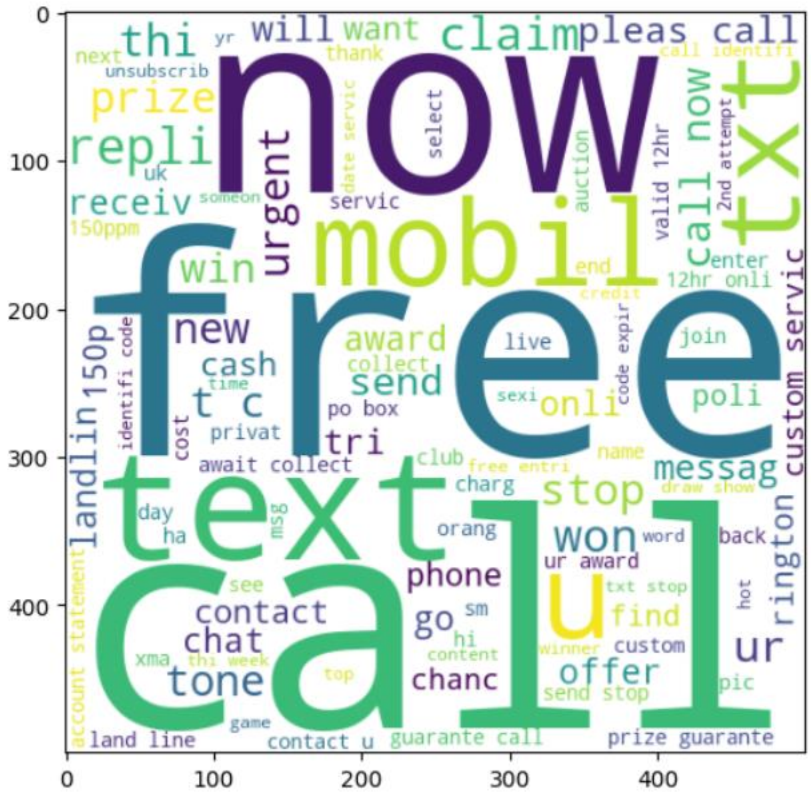



```
<AxesSubplot: xlabel='num_words', ylabel='Count'>
```

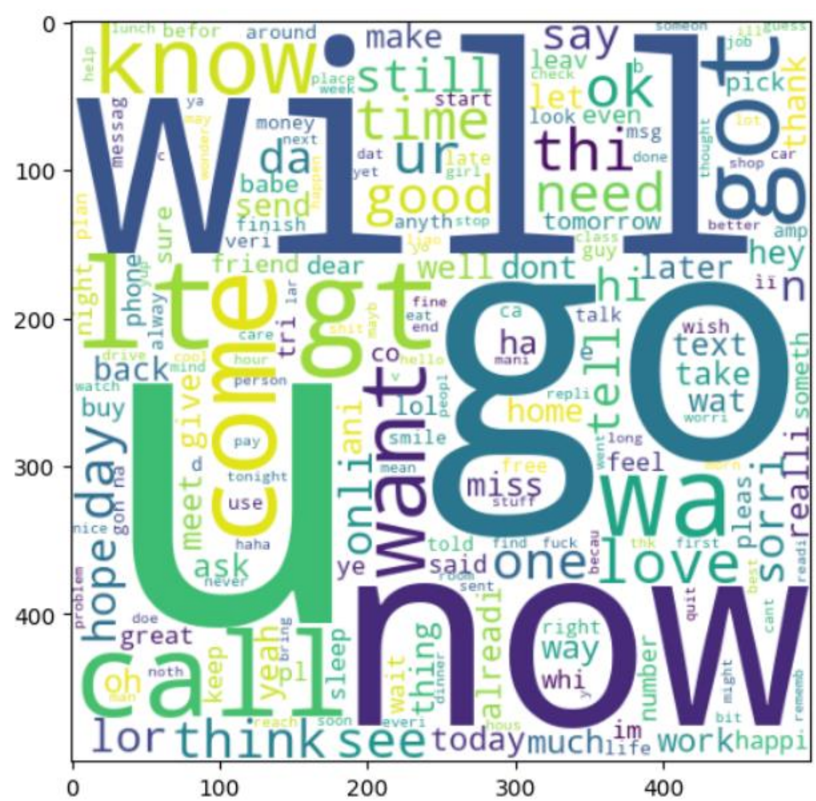



num_characters
num_words
num_sentences
target
0
1
0
250
500
750
0
100
200
0
10
20
30
40


target
num_characters
num_words
num_sentences
1
0.38
0.26
0.26
0.38
1
0.97
0.62
0.26
0.97
1
0.68
0.26
0.62
0.68
1
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3



free
call
now
text
txt
mobil
claim
pleas call
will
want
thi
prize
repli
urgent
receiv
win
new
award
cash
send
t c
poli
onli
tri
messag
stop
won
phone
contact
chat
tone
go
offer
chanc
find
ur
u
rington
custom servic
landlin
150p
call now
0
100
200
300
400



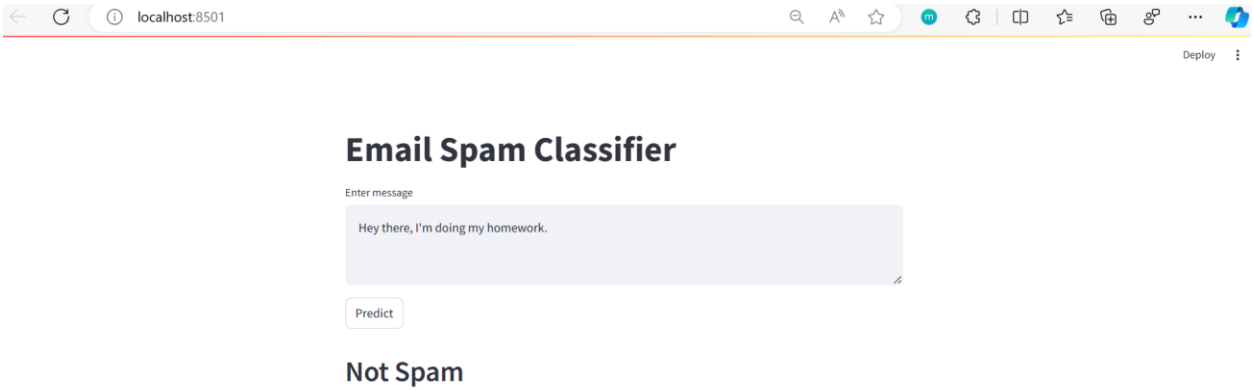
will
go
u
now
know
got
gt
lt
come
want
wa
good
time
still
ok
thi
need
ur
da
say
make
tell
hi
love
one
call
think
see
lor
sorri
realli
day
hope
text
take
wat
home
miss
later
dont
well
hey
today
much
work
happi
0
100
200
300
400

3.2 Accuracy and Precision

```
For  SVC
Accuracy -  0.9816247582205029
Precision -  0.983739837398374
For  KN
Accuracy -  0.90715667311412
Precision -  1.0
For  NB
Accuracy -  0.9680851063829787
Precision -  1.0
For  DT
Accuracy -  0.9429400386847195
Precision -  0.7883211678832117
For  LR
Accuracy -  0.9622823984526112
Precision -  0.9459459459459459
For  RF
Accuracy -  0.9709864603481625
Precision -  1.0
```

```
For  AdaBoost
Accuracy -  0.9700193423597679
Precision -  0.928
For  BgC
Accuracy -  0.965183752417795
Precision -  0.9180327868852459
For  ETC
Accuracy -  0.9787234042553191
Precision -  0.9833333333333333
For  GBDT
Accuracy -  0.960348162475822
Precision -  0.9532710280373832
For  xgb
Accuracy -  0.9806576402321083
Precision -  0.9682539682539683
```

3.3 Deployment Results



4. Conclusion:

By seeing the output, it is being concluded that the software has successfully predicted that the email has not spam.