Natural Language Processing 02 (DSMM Group 2)

FINAL PROJECT

Yogesh Kumar (c0852435)

FAKE/REAL Job Postings Classification

Creating dataframe

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv('C:/Users/yogi/documents/fake_job_postings.csv')
print(df.head(10))
print(df.columns)
```

```
   job_id                                      title               location  \
0       1                           Marketing Intern       US, NY, New York
1       2  Customer Service - Cloud Video Production         NZ, , Auckland
2       3    Commissioning Machinery Assistant (CMA)          US, IA, Wever
3       4          Account Executive - Washington DC     US, DC, Washington
4       5                        Bill Review Manager     US, FL, Fort Worth
5       6                           Accounting Clerk               US, MD,
6       7                      Head of Content (m/f)         DE, BE, Berlin
7       8              Lead Guest Service Specialist  US, CA, San Francisco
8       9                                 HP BSM SME      US, FL, Pensacola
9      10     Customer Service Associate - Part Time        US, AZ, Phoenix

   department salary_range                                    company_profile  \
0   Marketing          NaN  We're Food52, and we've created a groundbreaki...
1     Success          NaN  90 Seconds, the worlds Cloud Video Production ...
2         NaN          NaN  Valor Services provides Workforce Solutions th...
3       Sales          NaN  Our passion for improving quality of life thro...
4         NaN          NaN  SpotSource Solutions LLC is a Global Human Cap...
5         NaN          NaN                                                NaN
6  ANDROIDPIT  20000-28000  Founded in 2009, the Fonpit AG rose with its i...
7         NaN          NaN  Airenvy's mission is to provide lucrative yet ...
8         NaN          NaN  Solutions3 is a woman-owned small business who...
9         NaN          NaN  Novitex Enterprise Solutions, formerly Pitney ...

                                         description  \
0  Food52, a fast-growing, James Beard Award-winn...
1  Organised - Focused - Vibrant - Awesome!Do you...
2  Our client, located in Houston, is actively se...
3  THE COMPANY: ESRI – Environmental Systems Rese...
4  JOB TITLE: Itemization Review ManagerLOCATION:...
5  Job OverviewApex is an environmental consultin...
6  Your Responsibilities: Manage the English-spea...
7  Who is Airenvy?Hey there! We are seasoned entr...
8  Implementation/Configuration/Testing/Training ...
9  The Customer Service Associate will be based i...

                                        requirements  \
0  Experience with content management systems a m...
1  What we expect from you:Your key responsibilit...
2  Implement pre-commissioning and commissioning ...
3  EDUCATION: Bachelor's or Master's in GIS, busi...
4  QUALIFICATIONS:RN license in the State of Texa...
5                                                NaN
6  Your Know-How:                                ...
7  Experience with CRM software, live chat, and p...
8  MUST BE A US CITIZEN.An active TS/SCI clearanc...
9  Minimum Requirements:Minimum of 6 months custo...

                                            benefits  telecommuting  \
0                                                NaN              0
1  What you will get from usThrough being part of...              0
2                                                NaN              0
3  Our culture is anything but corporate—we have ...              0
4                              Full Benefits Offered              0
5                                                NaN              0
6  Your Benefits: Being part of a fast-growing co...              0
7  Competitive Pay. You'll be able to eat steak e...              0
8                                                NaN              0
9                                                NaN              0

   has_company_logo  has_questions employment_type required_experience  \
0                 1              0           Other          Internship
1                 1              0       Full-time      Not Applicable
2                 1              0             NaN                 NaN
3                 1              0       Full-time    Mid-Senior level
4                 1              1       Full-time    Mid-Senior level
5                 0              0             NaN                 NaN
6                 1              1       Full-time    Mid-Senior level
7                 1              1             NaN                 NaN
8                 1              1       Full-time           Associate
9                 1              0       Part-time         Entry level

          required_education                             industry  \
0                        NaN                                  NaN
1                        NaN            Marketing and Advertising
2                        NaN                                  NaN
3          Bachelor's Degree                    Computer Software
4          Bachelor's Degree               Hospital & Health Care
5                        NaN                                  NaN
6            Master's Degree                         Online Media
7                        NaN                                  NaN
8                        NaN  Information Technology and Services
9  High School or equivalent                   Financial Services

               function  fraudulent
0             Marketing           0
1      Customer Service           0
2                   NaN           0
3                 Sales           0
4  Health Care Provider           0
5                   NaN           0
6            Management           0
7                   NaN           0
8                   NaN           0
9      Customer Service           0
Index(['job_id', 'title', 'location', 'department', 'salary_range',
       'company_profile', 'description', 'requirements', 'benefits',
       'telecommuting', 'has_company_logo', 'has_questions', 'employment_type',
       'required_experience', 'required_education', 'industry', 'function',
       'fraudulent'],
      dtype='object')
```

Data Pre-Processing

In [2]:

```
#checking null values
print(df.info())
print(df.isnull().sum().sort_values())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17880 entries, 0 to 17879
Data columns (total 18 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   job_id               17880 non-null  int64
 1   title                17880 non-null  object
 2   location             17534 non-null  object
 3   department           6333 non-null   object
 4   salary_range         2868 non-null   object
 5   company_profile      14572 non-null  object
 6   description          17879 non-null  object
 7   requirements         15185 non-null  object
 8   benefits             10670 non-null  object
 9   telecommuting        17880 non-null  int64
 10  has_company_logo     17880 non-null  int64
 11  has_questions        17880 non-null  int64
 12  employment_type      14409 non-null  object
 13  required_experience  10830 non-null  object
 14  required_education   9775 non-null   object
 15  industry             12977 non-null  object
 16  function             11425 non-null  object
 17  fraudulent           17880 non-null  int64
dtypes: int64(5), object(13)
memory usage: 2.5+ MB
None
job_id                     0
has_questions              0
has_company_logo           0
telecommuting              0
fraudulent                 0
title                      0
description                1
location                 346
requirements            2695
company_profile         3308
employment_type         3471
industry                4903
function                6455
required_experience     7050
benefits                7210
required_education      8105
department             11547
salary_range           15012
dtype: int64
```

In [3]:

```
# filling null values with
df.fillna(' ',inplace=True)
df.isnull().sum().sort_values()
```

Out[3]:

```
job_id                 0
industry               0
required_education     0
required_experience    0
employment_type        0
has_questions          0
has_company_logo       0
telecommuting          0
benefits               0
requirements           0
description            0
company_profile        0
salary_range           0
department             0
location               0
title                  0
function               0
fraudulent             0
dtype: int64
```

In [4]:

```
df.describe()
print(df.head(10))
```

```
   job_id                                      title               location  \
0       1                           Marketing Intern       US, NY, New York
1       2  Customer Service - Cloud Video Production         NZ, , Auckland
2       3    Commissioning Machinery Assistant (CMA)          US, IA, Wever
3       4          Account Executive - Washington DC     US, DC, Washington
4       5                        Bill Review Manager     US, FL, Fort Worth
5       6                           Accounting Clerk               US, MD,
6       7                      Head of Content (m/f)         DE, BE, Berlin
7       8              Lead Guest Service Specialist  US, CA, San Francisco
8       9                                 HP BSM SME      US, FL, Pensacola
9      10     Customer Service Associate - Part Time        US, AZ, Phoenix

   department salary_range                                    company_profile  \
0   Marketing               We're Food52, and we've created a groundbreaki...
1     Success               90 Seconds, the worlds Cloud Video Production ...
2                           Valor Services provides Workforce Solutions th...
3       Sales               Our passion for improving quality of life thro...
4                           SpotSource Solutions LLC is a Global Human Cap...
5
6  ANDROIDPIT  20000-28000  Founded in 2009, the Fonpit AG rose with its i...
7                           Airenvy's mission is to provide lucrative yet ...
8                           Solutions3 is a woman-owned small business who...
9                           Novitex Enterprise Solutions, formerly Pitney ...

                                         description  \
0  Food52, a fast-growing, James Beard Award-winn...
1  Organised - Focused - Vibrant - Awesome!Do you...
2  Our client, located in Houston, is actively se...
3  THE COMPANY: ESRI – Environmental Systems Rese...
4  JOB TITLE: Itemization Review ManagerLOCATION:...
5  Job OverviewApex is an environmental consultin...
6  Your Responsibilities: Manage the English-spea...
7  Who is Airenvy?Hey there! We are seasoned entr...
8  Implementation/Configuration/Testing/Training ...
9  The Customer Service Associate will be based i...

                                        requirements  \
0  Experience with content management systems a m...
1  What we expect from you:Your key responsibilit...
2  Implement pre-commissioning and commissioning ...
3  EDUCATION: Bachelor's or Master's in GIS, busi...
4  QUALIFICATIONS:RN license in the State of Texa...
5
6  Your Know-How:                                ...
7  Experience with CRM software, live chat, and p...
8  MUST BE A US CITIZEN.An active TS/SCI clearanc...
9  Minimum Requirements:Minimum of 6 months custo...

                                            benefits  telecommuting  \
0                                                                 0
1  What you will get from usThrough being part of...              0
2                                                                 0
3  Our culture is anything but corporate—we have ...              0
4                              Full Benefits Offered              0
5                                                                 0
6  Your Benefits: Being part of a fast-growing co...              0
7  Competitive Pay. You'll be able to eat steak e...              0
8                                                                 0
9                                                                 0

   has_company_logo  has_questions employment_type required_experience  \
0                 1              0           Other          Internship
1                 1              0       Full-time      Not Applicable
2                 1              0
3                 1              0       Full-time    Mid-Senior level
4                 1              1       Full-time    Mid-Senior level
5                 0              0
6                 1              1       Full-time    Mid-Senior level
7                 1              1
8                 1              1       Full-time           Associate
9                 1              0       Part-time         Entry level

          required_education                             industry  \
0
1                                       Marketing and Advertising
2
3          Bachelor's Degree                    Computer Software
4          Bachelor's Degree               Hospital & Health Care
5
6            Master's Degree                         Online Media
7
8                                 Information Technology and Services
9  High School or equivalent                   Financial Services

               function  fraudulent
0             Marketing           0
1      Customer Service           0
2                                 0
3                 Sales           0
4  Health Care Provider           0
5                                 0
6            Management           0
7                                 0
8                                 0
9      Customer Service           0
```

In [5]:

```
#taking only required field and merging them in single field
df['text'] = df['description'] + " " + df['requirements']
df['text'].head()
df['text'][0]
```

Out[5]:

```
'Food52, a fast-growing, James Beard Award-winning online food community and crowd-sourced and curated recipe hub, is currently interviewing full- and part-time unpaid interns to work in a small team of editors, executives, and developers in its New York City headquarters.Reproducing and/or repackaging existing Food52 content for a number of partner sites, such as Huffington Post, Yahoo, Buzzfeed, and more in their various content management systemsResearching blogs and websites for the Provisions by Food52 Affiliate ProgramAssisting in day-to-day affiliate program support, such as screening affiliates and assisting in any affiliate inquiriesSupporting with PR &amp; Events when neededHelping with office administrative work, such as filing, mailing, and preparing for meetingsWorking with developers to document bugs and suggest improvements to the siteSupporting the marketing and executive staff Experience with content management systems a major plus (any blogging counts!)Familiar with the Food52 editorial voice and aestheticLoves food, appreciates the importance of home cooking and cooking with the seasonsMeticulous editor, perfectionist, obsessive attention to detail, maddened by typos and broken links, delighted by finding and fixing themCheerful under pressureExcellent communication skillsA+ multi-tasker and juggler of responsibilities big and smallInterested in and engaged with social media like Twitter, Facebook, and PinterestLoves problem-solving and collaborating to drive Food52 forwardThinks big picture but pitches in on the nitty gritty of running a small company (dishes, shopping, administrative support)Comfortable with the realities of working for a startup: being on call on evenings and weekends, and working long hours'
```

In [6]:

```
# Dropping unnecessary columns from dataframe
print(df.columns)
df.drop(columns=df.columns[0:len(df.columns)-2],inplace=True)
print(df.head())
```

```
Index(['job_id', 'title', 'location', 'department', 'salary_range',
       'company_profile', 'description', 'requirements', 'benefits',
       'telecommuting', 'has_company_logo', 'has_questions', 'employment_type',
       'required_experience', 'required_education', 'industry', 'function',
       'fraudulent', 'text'],
      dtype='object')
   fraudulent                                               text
0           0  Food52, a fast-growing, James Beard Award-winn...
1           0  Organised - Focused - Vibrant - Awesome!Do you...
2           0  Our client, located in Houston, is actively se...
3           0  THE COMPANY: ESRI – Environmental Systems Rese...
4           0  JOB TITLE: Itemization Review ManagerLOCATION:...
```

In [7]:

```
#function to remove punctuation and digits from document

import string
import re

print(string.punctuation)

def remove_punction(text):
    text_nonpunct=[]
    for i in text:
        text_nonpunct.append([re.sub('\d','',char) for char in i if char not in string.punctuation])
    return text_nonpunct

df['clean_text']=remove_punction(df['text'])

print(df['clean_text'][0])
```

```
!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~
['F', 'o', 'o', 'd', '', '', ' ', 'a', ' ', 'f', 'a', 's', 't', 'g', 'r', 'o', 'w', 'i', 'n', 'g', ' ', 'J', 'a', 'm', 'e', 's', ' ', 'B', 'e', 'a', 'r', 'd', ' ', 'A', 'w', 'a', 'r', 'd', 'w', 'i', 'n', 'n', 'i', 'n', 'g', ' ', 'o', 'n', 'l', 'i', 'n', 'e', ' ', 'f', 'o', 'o', 'd', ' ', 'c', 'o', 'm', 'm', 'u', 'n', 'i', 't', 'y', ' ', 'a', 'n', 'd', ' ', 'c', 'r', 'o', 'w', 'd', 's', 'o', 'u', 'r', 'c', 'e', 'd', ' ', 'a', 'n', 'd', ' ', 'c', 'u', 'r', 'a', 't', 'e', 'd', ' ', 'r', 'e', 'c', 'i', 'p', 'e', ' ', 'h', 'u', 'b', ' ', 'i', 's', ' ', 'c', 'u', 'r', 'r', 'e', 'n', 't', 'l', 'y', ' ', 'i', 'n', 't', 'e', 'r', 'v', 'i', 'e', 'w', 'i', 'n', 'g', ' ', 'f', 'u', 'l', 'l', ' ', 'a', 'n', 'd', ' ', 'p', 'a', 'r', 't', 't', 'i', 'm', 'e', ' ', 'u', 'n', 'p', 'a', 'i', 'd', ' ', 'i', 'n', 't', 'e', 'r', 'n', 's', ' ', 't', 'o', ' ', 'w', 'o', 'r', 'k', ' ', 'i', 'n', ' ', 'a', ' ', 's', 'm', 'a', 'l', 'l', ' ', 't', 'e', 'a', 'm', ' ', 'o', 'f', ' ', 'e', 'd', 'i', 't', 'o', 'r', 's', ' ', 'e', 'x', 'e', 'c', 'u', 't', 'i', 'v', 'e', 's', ' ', 'a', 'n', 'd', ' ', 'd', 'e', 'v', 'e', 'l', 'o', 'p', 'e', 'r', 's', ' ', 'i', 'n', ' ', 'i', 't', 's', ' ', 'N', 'e', 'w', ' ', 'Y', 'o', 'r', 'k', ' ', 'C', 'i', 't', 'y', ' ', 'h', 'e', 'a', 'd', 'q', 'u', 'a', 'r', 't', 'e', 'r', 's', 'R', 'e', 'p', 'r', 'o', 'd', 'u', 'c', 'i', 'n', 'g', ' ', 'a', 'n', 'd', 'o', 'r', ' ', 'r', 'e', 'p', 'a', 'c', 'k', 'a', 'g', 'i', 'n', 'g', ' ', 'e', 'x', 'i', 's', 't', 'i', 'n', 'g', ' ', 'F', 'o', 'o', 'd', '', '', ' ', 'c', 'o', 'n', 't', 'e', 'n', 't', ' ', 'f', 'o', 'r', ' ', 'a', ' ', 'n', 'u', 'm', 'b', 'e', 'r', ' ', 'o', 'f', ' ', 'p', 'a', 'r', 't', 'n', 'e', 'r', ' ', 's', 'i', 't', 'e', 's', ' ', 's', 'u', 'c', 'h', ' ', 'a', 's', ' ', 'H', 'u', 'f', 'f', 'i', 'n', 'g', 't', 'o', 'n', ' ', 'P', 'o', 's', 't', ' ', 'Y', 'a', 'h', 'o', 'o', ' ', 'B', 'u', 'z', 'z', 'f', 'e', 'e', 'd', ' ', 'a', 'n', 'd', ' ', 'm', 'o', 'r', 'e', ' ', 'i', 'n', ' ', 't', 'h', 'e', 'i', 'r', ' ', 'v', 'a', 'r', 'i', 'o', 'u', 's', ' ', 'c', 'o', 'n', 't', 'e', 'n', 't', ' ', 'm', 'a', 'n', 'a', 'g', 'e', 'm', 'e', 'n', 't', ' ', 's', 'y', 's', 't', 'e', 'm', 's', 'R', 'e', 's', 'e', 'a', 'r', 'c', 'h', 'i', 'n', 'g', ' ', 'b', 'l', 'o', 'g', 's', ' ', 'a', 'n', 'd', ' ', 'w', 'e', 'b', 's', 'i', 't', 'e', 's', ' ', 'f', 'o', 'r', ' ', 't', 'h', 'e', ' ', 'P', 'r', 'o', 'v', 'i', 's', 'i', 'o', 'n', 's', ' ', 'b', 'y', ' ', 'F', 'o', 'o', 'd', '', '', ' ', 'A', 'f', 'f', 'i', 'l', 'i', 'a', 't', 'e', ' ', 'P', 'r', 'o', 'g', 'r', 'a', 'm', 'A', 's', 's', 'i', 's', 't', 'i', 'n', 'g', ' ', 'i', 'n', ' ', 'd', 'a', 'y', 't', 'o', 'd', 'a', 'y', ' ', 'a', 'f', 'f', 'i', 'l', 'i', 'a', 't', 'e', ' ', 'p', 'r', 'o', 'g', 'r', 'a', 'm', ' ', 's', 'u', 'p', 'p', 'o', 'r', 't', ' ', 's', 'u', 'c', 'h', ' ', 'a', 's', ' ', 's', 'c', 'r', 'e', 'e', 'n', 'i', 'n', 'g', ' ', 'a', 'f', 'f', 'i', 'l', 'i', 'a', 't', 'e', 's', ' ', 'a', 'n', 'd', ' ', 'a', 's', 's', 'i', 's', 't', 'i', 'n', 'g', ' ', 'i', 'n', ' ', 'a', 'n', 'y', ' ', 'a', 'f', 'f', 'i', 'l', 'i', 'a', 't', 'e', ' ', 'i', 'n', 'q', 'u', 'i', 'r', 'i', 'e', 's', 'S', 'u', 'p', 'p', 'o', 'r', 't', 'i', 'n', 'g', ' ', 'w', 'i', 't', 'h', ' ', 'P', 'R', ' ', 'a', 'm', 'p', ' ', 'E', 'v', 'e', 'n', 't', 's', ' ', 'w', 'h', 'e', 'n', ' ', 'n', 'e', 'e', 'd', 'e', 'd', 'H', 'e', 'l', 'p', 'i', 'n', 'g', ' ', 'w', 'i', 't', 'h', ' ', 'o', 'f', 'f', 'i', 'c', 'e', ' ', 'a', 'd', 'm', 'i', 'n', 'i', 's', 't', 'r', 'a', 't', 'i', 'v', 'e', ' ', 'w', 'o', 'r', 'k', ' ', 's', 'u', 'c', 'h', ' ', 'a', 's', ' ', 'f', 'i', 'l', 'i', 'n', 'g', ' ', 'm', 'a', 'i', 'l', 'i', 'n', 'g', ' ', 'a', 'n', 'd', ' ', 'p', 'r', 'e', 'p', 'a', 'r', 'i', 'n', 'g', ' ', 'f', 'o', 'r', ' ', 'm', 'e', 'e', 't', 'i', 'n', 'g', 's', 'W', 'o', 'r', 'k', 'i', 'n', 'g', ' ', 'w', 'i', 't', 'h', ' ', 'd', 'e', 'v', 'e', 'l', 'o', 'p', 'e', 'r', 's', ' ', 't', 'o', ' ', 'd', 'o', 'c', 'u', 'm', 'e', 'n', 't', ' ', 'b', 'u', 'g', 's', ' ', 'a', 'n', 'd', ' ', 's', 'u', 'g', 'g', 'e', 's', 't', ' ', 'i', 'm', 'p', 'r', 'o', 'v', 'e', 'm', 'e', 'n', 't', 's', ' ', 't', 'o', ' ', 't', 'h', 'e', ' ', 's', 'i', 't', 'e', 'S', 'u', 'p', 'p', 'o', 'r', 't', 'i', 'n', 'g', ' ', 't', 'h', 'e', ' ', 'm', 'a', 'r', 'k', 'e', 't', 'i', 'n', 'g', ' ', 'a', 'n', 'd', ' ', 'e', 'x', 'e', 'c', 'u', 't', 'i', 'v', 'e', ' ', 's', 't', 'a', 'f', 'f', ' ', 'E', 'x', 'p', 'e', 'r', 'i', 'e', 'n', 'c', 'e', ' ', 'w', 'i', 't', 'h', ' ', 'c', 'o', 'n', 't', 'e', 'n', 't', ' ', 'm', 'a', 'n', 'a', 'g', 'e', 'm', 'e', 'n', 't', ' ', 's', 'y', 's', 't', 'e', 'm', 's', ' ', 'a', ' ', 'm', 'a', 'j', 'o', 'r', ' ', 'p', 'l', 'u', 's', ' ', 'a', 'n', 'y', ' ', 'b', 'l', 'o', 'g', 'g', 'i', 'n', 'g', ' ', 'c', 'o', 'u', 'n', 't', 's', 'F', 'a', 'm', 'i', 'l', 'i', 'a', 'r', ' ', 'w', 'i', 't', 'h', ' ', 't', 'h', 'e', ' ', 'F', 'o', 'o', 'd', '', '', ' ', 'e', 'd', 'i', 't', 'o', 'r', 'i', 'a', 'l', ' ', 'v', 'o', 'i', 'c', 'e', ' ', 'a', 'n', 'd', ' ', 'a', 'e', 's', 't', 'h', 'e', 't', 'i', 'c', 'L', 'o', 'v', 'e', 's', ' ', 'f', 'o', 'o', 'd', ' ', 'a', 'p', 'p', 'r', 'e', 'c', 'i', 'a', 't', 'e', 's', ' ', 't', 'h', 'e', ' ', 'i', 'm', 'p', 'o', 'r', 't', 'a', 'n', 'c', 'e', ' ', 'o', 'f', ' ', 'h', 'o', 'm', 'e', ' ', 'c', 'o', 'o', 'k', 'i', 'n', 'g', ' ', 'a', 'n', 'd', ' ', 'c', 'o', 'o', 'k', 'i', 'n', 'g', ' ', 'w', 'i', 't', 'h', ' ', 't', 'h', 'e', ' ', 's', 'e', 'a', 's', 'o', 'n', 's', 'M', 'e', 't', 'i', 'c', 'u', 'l', 'o', 'u', 's', ' ', 'e', 'd', 'i', 't', 'o', 'r', ' ', 'p', 'e', 'r', 'f', 'e', 'c', 't', 'i', 'o', 'n', 'i', 's', 't', ' ', 'o', 'b', 's', 'e', 's', 's', 'i', 'v', 'e', ' ', 'a', 't', 't', 'e', 'n', 't', 'i', 'o', 'n', ' ', 't', 'o', ' ', 'd', 'e', 't', 'a', 'i', 'l', ' ', 'm', 'a', 'd', 'd', 'e', 'n', 'e', 'd', ' ', 'b', 'y', ' ', 't', 'y', 'p', 'o', 's', ' ', 'a', 'n', 'd', ' ', 'b', 'r', 'o', 'k', 'e', 'n', ' ', 'l', 'i', 'n', 'k', 's', ' ', 'd', 'e', 'l', 'i', 'g', 'h', 't', 'e', 'd', ' ', 'b', 'y', ' ', 'f', 'i', 'n', 'd', 'i', 'n', 'g', ' ', 'a', 'n', 'd', ' ', 'f', 'i', 'x', 'i', 'n', 'g', ' ', 't', 'h', 'e', 'm', 'C', 'h', 'e', 'e', 'r', 'f', 'u', 'l', ' ', 'u', 'n', 'd', 'e', 'r', ' ', 'p', 'r', 'e', 's', 's', 'u', 'r', 'e', 'E', 'x', 'c', 'e', 'l', 'l', 'e', 'n', 't', ' ', 'c', 'o', 'm', 'm', 'u', 'n', 'i', 'c', 'a', 't', 'i', 'o', 'n', ' ', 's', 'k', 'i', 'l', 'l', 's', 'A', ' ', 'm', 'u', 'l', 't', 'i', 't', 'a', 's', 'k', 'e', 'r', ' ', 'a', 'n', 'd', ' ', 'j', 'u', 'g', 'g', 'l', 'e', 'r', ' ', 'o', 'f', ' ', 'r', 'e', 's', 'p', 'o', 'n', 's', 'i', 'b', 'i', 'l', 'i', 't', 'i', 'e', 's', ' ', 'b', 'i', 'g', ' ', 'a', 'n', 'd', ' ', 's', 'm', 'a', 'l', 'l', 'I', 'n', 't', 'e', 'r', 'e', 's', 't', 'e', 'd', ' ', 'i', 'n', ' ', 'a', 'n', 'd', ' ', 'e', 'n', 'g', 'a', 'g', 'e', 'd', ' ', 'w', 'i', 't', 'h', ' ', 's', 'o', 'c', 'i', 'a', 'l', ' ', 'm', 'e', 'd', 'i', 'a', ' ', 'l', 'i', 'k', 'e', ' ', 'T', 'w', 'i', 't', 't', 'e', 'r', ' ', 'F', 'a', 'c', 'e', 'b', 'o', 'o', 'k', ' ', 'a', 'n', 'd', ' ', 'P', 'i', 'n', 't', 'e', 'r', 'e', 's', 't', 'L', 'o', 'v', 'e', 's', ' ', 'p', 'r', 'o', 'b', 'l', 'e', 'm', 's', 'o', 'l', 'v', 'i', 'n', 'g', ' ', 'a', 'n', 'd', ' ', 'c', 'o', 'l', 'l', 'a', 'b', 'o', 'r', 'a', 't', 'i', 'n', 'g', ' ', 't', 'o', ' ', 'd', 'r', 'i', 'v', 'e', ' ', 'F', 'o', 'o', 'd', '', '', ' ', 'f', 'o', 'r', 'w', 'a', 'r', 'd', 'T', 'h', 'i', 'n', 'k', 's', ' ', 'b', 'i', 'g', ' ', 'p', 'i', 'c', 't', 'u', 'r', 'e', ' ', 'b', 'u', 't', ' ', 'p', 'i', 't', 'c', 'h', 'e', 's', ' ', 'i', 'n', ' ', 'o', 'n', ' ', 't', 'h', 'e', ' ', 'n', 'i', 't', 't', 'y', ' ', 'g', 'r', 'i', 't', 't', 'y', ' ', 'o', 'f', ' ', 'r', 'u', 'n', 'n', 'i', 'n', 'g', ' ', 'a', ' ', 's', 'm', 'a', 'l', 'l', ' ', 'c', 'o', 'm', 'p', 'a', 'n', 'y', ' ', 'd', 'i', 's', 'h', 'e', 's', ' ', 's', 'h', 'o', 'p', 'p', 'i', 'n', 'g', ' ', 'a', 'd', 'm', 'i', 'n', 'i', 's', 't', 'r', 'a', 't', 'i', 'v', 'e', ' ', 's', 'u', 'p', 'p', 'o', 'r', 't', 'C', 'o', 'm', 'f', 'o', 'r', 't', 'a', 'b', 'l', 'e', ' ', 'w', 'i', 't', 'h', ' ', 't', 'h', 'e', ' ', 'r', 'e', 'a', 'l', 'i', 't', 'i', 'e', 's', ' ', 'o', 'f', ' ', 'w', 'o', 'r', 'k', 'i', 'n', 'g', ' ', 'f', 'o', 'r', ' ', 'a', ' ', 's', 't', 'a', 'r', 't', 'u', 'p', ' ', 'b', 'e', 'i', 'n', 'g', ' ', 'o', 'n', ' ', 'c', 'a', 'l', 'l', ' ', 'o', 'n', ' ', 'e', 'v', 'e', 'n', 'i', 'n', 'g', 's', ' ', 'a', 'n', 'd', ' ', 'w', 'e', 'e', 'k', 'e', 'n', 'd', 's', ' ', 'a', 'n', 'd', ' ', 'w', 'o', 'r', 'k', 'i', 'n', 'g', ' ', 'l', 'o', 'n', 'g', ' ', 'h', 'o', 'u', 'r', 's']
```

In [8]:

```
# a function to combine all elements
def remove_punct_combine(text):
    comb_text=[]
    for x in text:
        comb_text.append("".join([char for char in x if char not in string.punctuation]))
    return comb_text

df['clean_text']=remove_punct_combine(df['clean_text'])
print(df['clean_text'][0])
```

```
Food a fastgrowing James Beard Awardwinning online food community and crowdsourced and curated recipe hub is currently interviewing full and parttime unpaid interns to work in a small team of editors executives and developers in its New York City headquartersReproducing andor repackaging existing Food content for a number of partner sites such as Huffington Post Yahoo Buzzfeed and more in their various content management systemsResearching blogs and websites for the Provisions by Food Affiliate ProgramAssisting in daytoday affiliate program support such as screening affiliates and assisting in any affiliate inquiriesSupporting with PR amp Events when neededHelping with office administrative work such as filing mailing and preparing for meetingsWorking with developers to document bugs and suggest improvements to the siteSupporting the marketing and executive staff Experience with content management systems a major plus any blogging countsFamiliar with the Food editorial voice and aestheticLoves food appreciates the importance of home cooking and cooking with the seasonsMeticulous editor perfectionist obsessive attention to detail maddened by typos and broken links delighted by finding and fixing themCheerful under pressureExcellent communication skillsA multitasker and juggler of responsibilities big and smallInterested in and engaged with social media like Twitter Facebook and PinterestLoves problemsolving and collaborating to drive Food forwardThinks big picture but pitches in on the nitty gritty of running a small company dishes shopping administrative supportComfortable with the realities of working for a startup being on call on evenings and weekends and working long hours
```

In [9]:

```
# fucntion to tokenize Sentence
! pip install nltk
import nltk
from nltk.tokenize import word_tokenize
def token_word(text):
    sents=[]
    for x in text:
        sents.append(word_tokenize(x))
    return sents
df["token_word"]=token_word(df["clean_text"])
print(df["token_word"][0])
```

```
Requirement already satisfied: nltk in c:\users\yogi\anaconda3\lib\site-packages (3.6.5)
Requirement already satisfied: click in c:\users\yogi\anaconda3\lib\site-packages (from nltk) (8.0.3)
Requirement already satisfied: joblib in c:\users\yogi\anaconda3\lib\site-packages (from nltk) (1.1.0)
Requirement already satisfied: regex>=2021.8.3 in c:\users\yogi\anaconda3\lib\site-packages (from nltk) (2021.8.3)
Requirement already satisfied: tqdm in c:\users\yogi\anaconda3\lib\site-packages (from nltk) (4.62.3)
Requirement already satisfied: colorama in c:\users\yogi\anaconda3\lib\site-packages (from click->nltk) (0.4.4)
['Food', 'a', 'fastgrowing', 'James', 'Beard', 'Awardwinning', 'online', 'food', 'community', 'and', 'crowdsourced', 'and', 'curated', 'recipe', 'hub', 'is', 'currently', 'interviewing', 'full', 'and', 'parttime', 'unpaid', 'interns', 'to', 'work', 'in', 'a', 'small', 'team', 'of', 'editors', 'executives', 'and', 'developers', 'in', 'its', 'New', 'York', 'City', 'headquartersReproducing', 'andor', 'repackaging', 'existing', 'Food', 'content', 'for', 'a', 'number', 'of', 'partner', 'sites', 'such', 'as', 'Huffington', 'Post', 'Yahoo', 'Buzzfeed', 'and', 'more', 'in', 'their', 'various', 'content', 'management', 'systemsResearching', 'blogs', 'and', 'websites', 'for', 'the', 'Provisions', 'by', 'Food', 'Affiliate', 'ProgramAssisting', 'in', 'daytoday', 'affiliate', 'program', 'support', 'such', 'as', 'screening', 'affiliates', 'and', 'assisting', 'in', 'any', 'affiliate', 'inquiriesSupporting', 'with', 'PR', 'amp', 'Events', 'when', 'neededHelping', 'with', 'office', 'administrative', 'work', 'such', 'as', 'filing', 'mailing', 'and', 'preparing', 'for', 'meetingsWorking', 'with', 'developers', 'to', 'document', 'bugs', 'and', 'suggest', 'improvements', 'to', 'the', 'siteSupporting', 'the', 'marketing', 'and', 'executive', 'staff', 'Experience', 'with', 'content', 'management', 'systems', 'a', 'major', 'plus', 'any', 'blogging', 'countsFamiliar', 'with', 'the', 'Food', 'editorial', 'voice', 'and', 'aestheticLoves', 'food', 'appreciates', 'the', 'importance', 'of', 'home', 'cooking', 'and', 'cooking', 'with', 'the', 'seasonsMeticulous', 'editor', 'perfectionist', 'obsessive', 'attention', 'to', 'detail', 'maddened', 'by', 'typos', 'and', 'broken', 'links', 'delighted', 'by', 'finding', 'and', 'fixing', 'themCheerful', 'under', 'pressureExcellent', 'communication', 'skillsA', 'multitasker', 'and', 'juggler', 'of', 'responsibilities', 'big', 'and', 'smallInterested', 'in', 'and', 'engaged', 'with', 'social', 'media', 'like', 'Twitter', 'Facebook', 'and', 'PinterestLoves', 'problemsolving', 'and', 'collaborating', 'to', 'drive', 'Food', 'forwardThinks', 'big', 'picture', 'but', 'pitches', 'in', 'on', 'the', 'nitty', 'gritty', 'of', 'running', 'a', 'small', 'company', 'dishes', 'shopping', 'administrative', 'supportComfortable', 'with', 'the', 'realities', 'of', 'working', 'for', 'a', 'startup', 'being', 'on', 'call', 'on', 'evenings', 'and', 'weekends', 'and', 'working', 'long', 'hours']
```

In [10]:

```
# function to remove stopword removed
from nltk.corpus import stopwords
nltk.download('stopwords')
from string import punctuation
customStopWords = set(stopwords.words('english')+list(punctuation))

def stop_word(text):
    stop_words=[]
    for x in text:
        stop_words.append([word for word in x if word not in customStopWords])
    return stop_words
df['stop_word']=stop_word(df['token_word'])

print(df['stop_word'][0])
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\Yogi\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
['Food', 'fastgrowing', 'James', 'Beard', 'Awardwinning', 'online', 'food', 'community', 'crowdsourced', 'curated', 'recipe', 'hub', 'currently', 'interviewing', 'full', 'parttime', 'unpaid', 'interns', 'work', 'small', 'team', 'editors', 'executives', 'developers', 'New', 'York', 'City', 'headquartersReproducing', 'andor', 'repackaging', 'existing', 'Food', 'content', 'number', 'partner', 'sites', 'Huffington', 'Post', 'Yahoo', 'Buzzfeed', 'various', 'content', 'management', 'systemsResearching', 'blogs', 'websites', 'Provisions', 'Food', 'Affiliate', 'ProgramAssisting', 'daytoday', 'affiliate', 'program', 'support', 'screening', 'affiliates', 'assisting', 'affiliate', 'inquiriesSupporting', 'PR', 'amp', 'Events', 'neededHelping', 'office', 'administrative', 'work', 'filing', 'mailing', 'preparing', 'meetingsWorking', 'developers', 'document', 'bugs', 'suggest', 'improvements', 'siteSupporting', 'marketing', 'executive', 'staff', 'Experience', 'content', 'management', 'systems', 'major', 'plus', 'blogging', 'countsFamiliar', 'Food', 'editorial', 'voice', 'aestheticLoves', 'food', 'appreciates', 'importance', 'home', 'cooking', 'cooking', 'seasonsMeticulous', 'editor', 'perfectionist', 'obsessive', 'attention', 'detail', 'maddened', 'typos', 'broken', 'links', 'delighted', 'finding', 'fixing', 'themCheerful', 'pressureExcellent', 'communication', 'skillsA', 'multitasker', 'juggler', 'responsibilities', 'big', 'smallInterested', 'engaged', 'social', 'media', 'like', 'Twitter', 'Facebook', 'PinterestLoves', 'problemsolving', 'collaborating', 'drive', 'Food', 'forwardThinks', 'big', 'picture', 'pitches', 'nitty', 'gritty', 'running', 'small', 'company', 'dishes', 'shopping', 'administrative', 'supportComfortable', 'realities', 'working', 'startup', 'call', 'evenings', 'weekends', 'working', 'long', 'hours']
```

In [11]:

```
# function for stemming document
ps = nltk.PorterStemmer()
def stemming(tokenized_text):
    st=[]
    for x in tokenized_text:
        st.append([ps.stem(word) for word in x])
    return st

df['stemming']=stemming(df['stop_word'])

print(df['stemming'][0])
```

```
['food', 'fastgrow', 'jame', 'beard', 'awardwin', 'onlin', 'food', 'commun', 'crowdsourc', 'curat', 'recip', 'hub', 'current', 'interview', 'full', 'parttim', 'unpaid', 'intern', 'work', 'small', 'team', 'editor', 'execut', 'develop', 'new', 'york', 'citi', 'headquartersreproduc', 'andor', 'repackag', 'exist', 'food', 'content', 'number', 'partner', 'site', 'huffington', 'post', 'yahoo', 'buzzfe', 'variou', 'content', 'manag', 'systemsresearch', 'blog', 'websit', 'provis', 'food', 'affili', 'programassist', 'daytoday', 'affili', 'program', 'support', 'screen', 'affili', 'assist', 'affili', 'inquiriessupport', 'pr', 'amp', 'event', 'neededhelp', 'offic', 'administr', 'work', 'file', 'mail', 'prepar', 'meetingswork', 'develop', 'document', 'bug', 'suggest', 'improv', 'sitesupport', 'market', 'execut', 'staff', 'experi', 'content', 'manag', 'system', 'major', 'plu', 'blog', 'countsfamiliar', 'food', 'editori', 'voic', 'aestheticlov', 'food', 'appreci', 'import', 'home', 'cook', 'cook', 'seasonsmetic', 'editor', 'perfectionist', 'obsess', 'attent', 'detail', 'madden', 'typo', 'broken', 'link', 'delight', 'find', 'fix', 'themcheer', 'pressureexcel', 'commun', 'skillsa', 'multitask', 'juggler', 'respons', 'big', 'smallinterest', 'engag', 'social', 'media', 'like', 'twitter', 'facebook', 'pinterestlov', 'problemsolv', 'collabor', 'drive', 'food', 'forwardthink', 'big', 'pictur', 'pitch', 'nitti', 'gritti', 'run', 'small', 'compani', 'dish', 'shop', 'administ', 'supportcomfort', 'realiti', 'work', 'startup', 'call', 'even', 'weekend', 'work', 'long', 'hour']
```

In [12]:

```
#function for lemmatization a document
nltk.download("wordnet")
wn = nltk.WordNetLemmatizer()
def lemmatizing(tokenized_text):
    lem=[]
    for x in tokenized_text:
        lem.append([wn.lemmatize(word) for word in x])
    return lem

df['lemmatize']=lemmatizing(df['stemming'])
```

```
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\Yogi\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

In [13]:

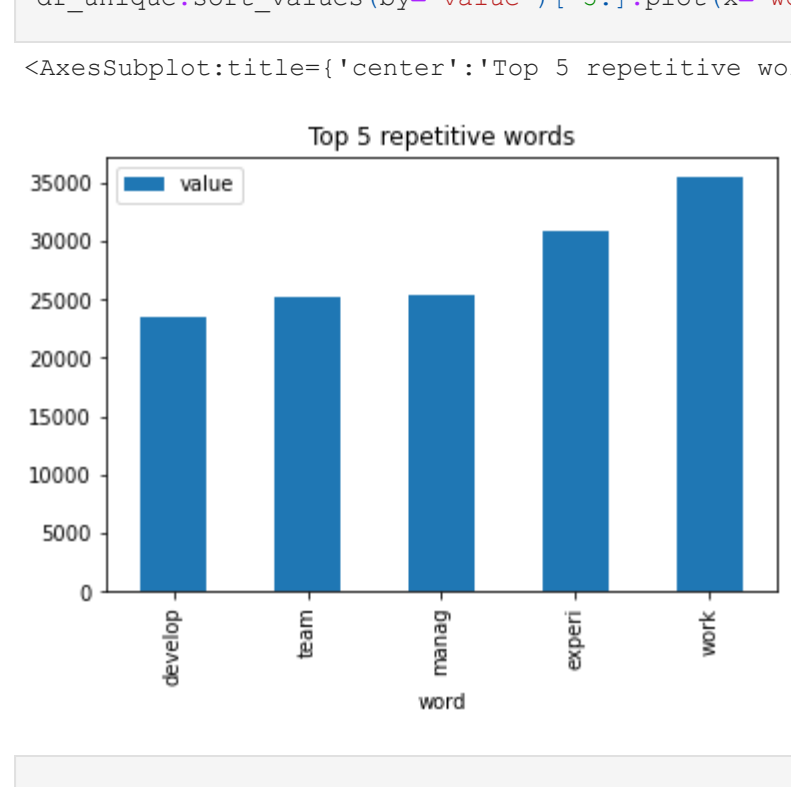
```
#function to create final document/text for vectorisation
def final_doc(text):
    str=""
    str=" ".join(text)
    return str

df["full_doc"]=df['lemmatize'].apply(final_doc)

#dropping previously created columns and retaining only final clean document/text for model
df.drop(columns=df.columns[1:-1],inplace=True)
print(df)
```

```
       fraudulent                                           full_doc
0               0  food fastgrow jame beard awardwin onlin food c...
1               0  organis focus vibrant awesomedo passion custom...
2               0  our client locat houston activ seek experienc ...
3               0  the compani esri – environ system research...
4               0  job titl item review managerloc fort worth te ...
...           ...                                                ...
17875           0  just case first time ' visit websit vend award...
17876           0  the payrol account focu primarili payrol funct...
17877           0  experienc project cost control staff engin req...
17878           0  nemsia studio look experienc visualgraph desig...
17879           0  who wevend award win web base point sale softw...

[17880 rows x 2 columns]
```

In [15]:

```
# frequency of each word in document
unq_val={}
for x in df["full_doc"]:
    for y in x.split(" "):
        if y in unq_val:
            unq_val[y]=unq_val[y]+1
        else:
            unq_val[y]=1

df_unique=pd.DataFrame(list(unq_val.items()),columns=['word','value'])

#print(df_unique)

df_unique.sort_values(by='value')[-5:].plot(x="word", y="value", kind="bar",title='Top 5 repetitive words')
```

Out[15]: `<AxesSubplot:title={'center':'Top 5 repetitive words'}, xlabel='word'>`



In [30]:

```
#creating target and features and splitting them into train and test datasets
from sklearn.model_selection import train_test_split

X = df['full_doc']
y = df['fraudulent']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 33)
```

In [36]:

```
#Vectorization using CountVectorizer
from sklearn.feature_extraction.text import CountVectorizer

cv=CountVectorizer(binary=False,ngram_range=(2,3))
cv_train= cv.fit_transform(X_train)
cv_test=cv.transform(X_test)
print(cv_train.shape)
print(cv_test.shape)
print(cv_train[0:10])
print(cv_test[0:10])
```

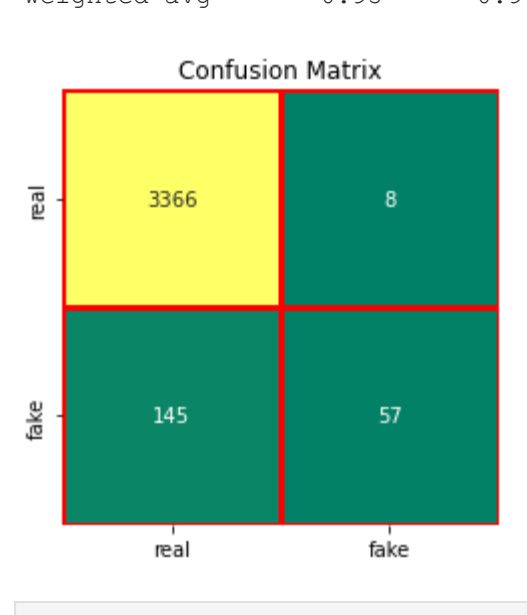
```
(14304, 1960816)
(3576, 1960816)
  (0, 1869951)	1
  (0, 977429)	1
  (0, 1695456)	1
  (0, 1208363)	1
  (0, 1320000)	2
  (0, 443901)	1
  (0, 898672)	1
  (0, 1327590)	1
  (0, 1720458)	1
  (0, 1927135)	1
  (0, 1905663)	2
  (0, 270874)	2
  (0, 551679)	1
  (0, 1323216)	1
  (0, 999012)	1
  (0, 447882)	1
  (0, 1842821)	1
  (0, 656692)	1
  (0, 1325412)	1
  (0, 1380784)	1
  (0, 1137132)	1
  (0, 1944330)	1
  (0, 1693475)	1
  (0, 1191417)	1
  (0, 1857521)	1
  :	:
  (9, 421162)	1
  (9, 1801238)	1
  (9, 894895)	1
  (9, 925919)	1
  (9, 1938727)	1
  (9, 933248)	1
  (9, 1494259)	1
  (9, 1880606)	1
  (9, 1195055)	1
  (9, 1765510)	1
  (9, 1142561)	1
  (9, 1768124)	1
  (9, 854954)	1
  (9, 480702)	1
  (9, 227843)	1
  (9, 480717)	1
  (9, 969257)	1
  (9, 854930)	1
  (9, 854926)	1
  (9, 1798886)	1
  (9, 169180)	1
  (9, 854961)	1
  (9, 1080477)	1
  (9, 1454610)	1
  (9, 895363)	1
  (0, 4191)	1
  (0, 48040)	1
  (0, 48041)	1
  (0, 50884)	1
  (0, 50921)	1
  (0, 53788)	1
  (0, 53789)	1
  (0, 67342)	1
  (0, 67344)	1
  (0, 69211)	1
  (0, 131243)	1
  (0, 131244)	1
  (0, 204634)	1
  (0, 226912)	1
  (0, 226913)	1
  (0, 261271)	1
  (0, 261282)	1
  (0, 278196)	1
  (0, 278212)	1
  (0, 278213)	1
  (0, 281029)	1
  (0, 281030)	1
  (0, 281311)	1
  (0, 281312)	1
  (0, 305348)	1
  :	:
  (9, 1889934)	1
  (9, 1889936)	1
  (9, 1894094)	1
  (9, 1894095)	1
  (9, 1895334)	1
  (9, 1895335)	1
  (9, 1908628)	1
  (9, 1908629)	1
  (9, 1910344)	1
  (9, 1910369)	1
  (9, 1911879)	1
  (9, 1911882)	1
  (9, 1912842)	1
  (9, 1912846)	1
  (9, 1926317)	1
  (9, 1926318)	1
  (9, 1927356)	1
  (9, 1927357)	1
  (9, 1928201)	1
  (9, 1928211)	1
  (9, 1930439)	1
  (9, 1930488)	1
  (9, 1934313)	1
  (9, 1934317)	1
  (9, 1949719)	1
```

Machine learning models implementation

In [35]:

```
from sklearn.tree import DecisionTreeClassifier

dt_model= DecisionTreeClassifier(criterion='entropy',max_depth=5)

dt_model.fit(cv_train,y_train)

dt_pred=dt_model.predict(X_test_tf)

from sklearn.metrics import precision_score, recall_score, confusion_matrix, classification_report, accuracy_sc
from sklearn.model_selection import cross_val_score

acc1=accuracy_score(y_test,dt_pred)
pre1=precision_score(y_test,dt_pred)

print("Accuracy Score :",acc1)
print("Precision Score :",pre1)

import seaborn as sns
sns.heatmap(confusion_matrix(y_test,dt_pred),annot=True, cmap='summer', cbar=False, linewidths=3, linecolor='r'
plt.title('Confusion Matrix')
print(classification_report(y_test,dt_pred))
```

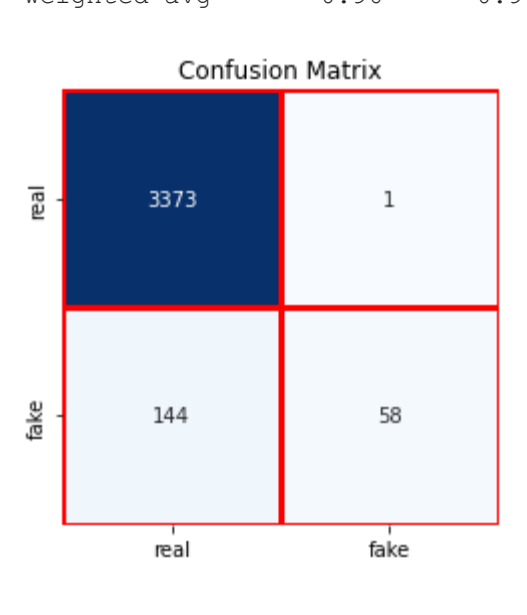
```
Accuracy Score : 0.9572147651006712
Precision Score : 0.8769230769230769
              precision    recall  f1-score   support

           0       0.96      1.00      0.98      3374
           1       0.88      0.28      0.43       202

    accuracy                           0.96      3576
   macro avg       0.92      0.64      0.70      3576
weighted avg       0.95      0.96      0.95      3576
```



In [18]:

```
from sklearn.svm import SVC

svc_model= SVC(kernel='sigmoid', gamma=1.0)
svc_model.fit(cv_train,y_train)

svc_pred=dt_model.predict(cv_test)
acc2=accuracy_score(y_test,svc_pred)
pre2=precision_score(y_test,svc_pred)
print("Accuracy Score :",accuracy_score(y_test,svc_pred))
print("Precision Score :",precision_score(y_test,svc_pred))

import seaborn as sns
plt.title('Confusion Matrix')
sns.heatmap(confusion_matrix(y_test,svc_pred),annot=True, cmap='Blues', cbar=False, linewidths=3, linecolor='r'
print(classification_report(y_test,svc_pred))
```

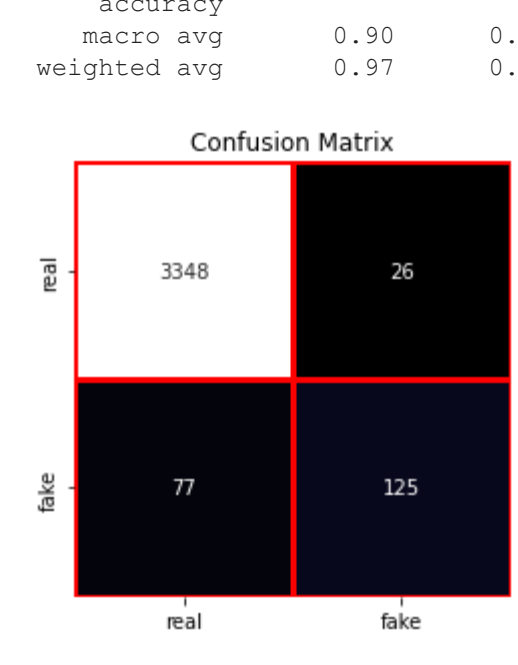
```
Accuracy Score : 0.9594519015659956
Precision Score : 0.9830508474576272
              precision    recall  f1-score   support

           0       0.96      1.00      0.98      3374
           1       0.98      0.29      0.44       202

    accuracy                           0.96      3576
   macro avg       0.97      0.64      0.71      3576
weighted avg       0.96      0.96      0.95      3576
```



In [19]:

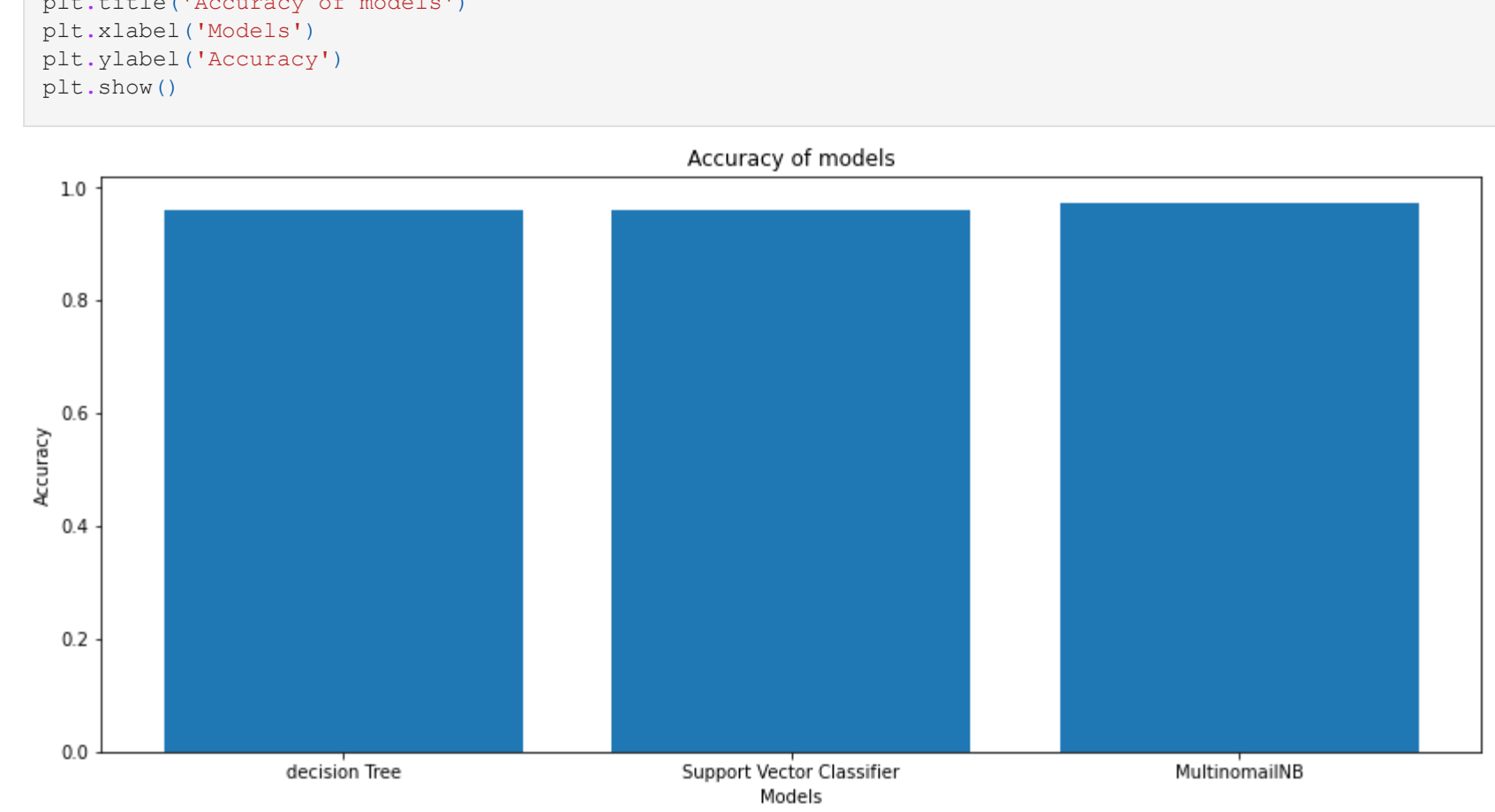
```
from sklearn.naive_bayes import MultinomialNB

mnb_model = MultinomialNB()

mnb_model.fit(cv_train,y_train)

mnb_pred=mnb_model.predict(cv_test)
acc3=accuracy_score(y_test,mnb_pred)
pre3=precision_score(y_test,mnb_pred)
print("Accuracy Score :",accuracy_score(y_test,mnb_pred))
print("Precision Score :",precision_score(y_test,mnb_pred))

sns.heatmap(confusion_matrix(y_test,mnb_pred),annot=True, cmap='CMRmap', cbar=False, linewidths=3, linecolor='r
plt.title('Confusion Matrix')
print(classification_report(y_test,mnb_pred))
```

```
Accuracy Score : 0.9711968680089486
Precision Score : 0.8278145695364238
              precision    recall  f1-score   support

           0       0.98      0.99      0.98      3374
           1       0.83      0.62      0.71       202

    accuracy                           0.97      3576
   macro avg       0.90      0.81      0.85      3576
weighted avg       0.97      0.97      0.97      3576
```



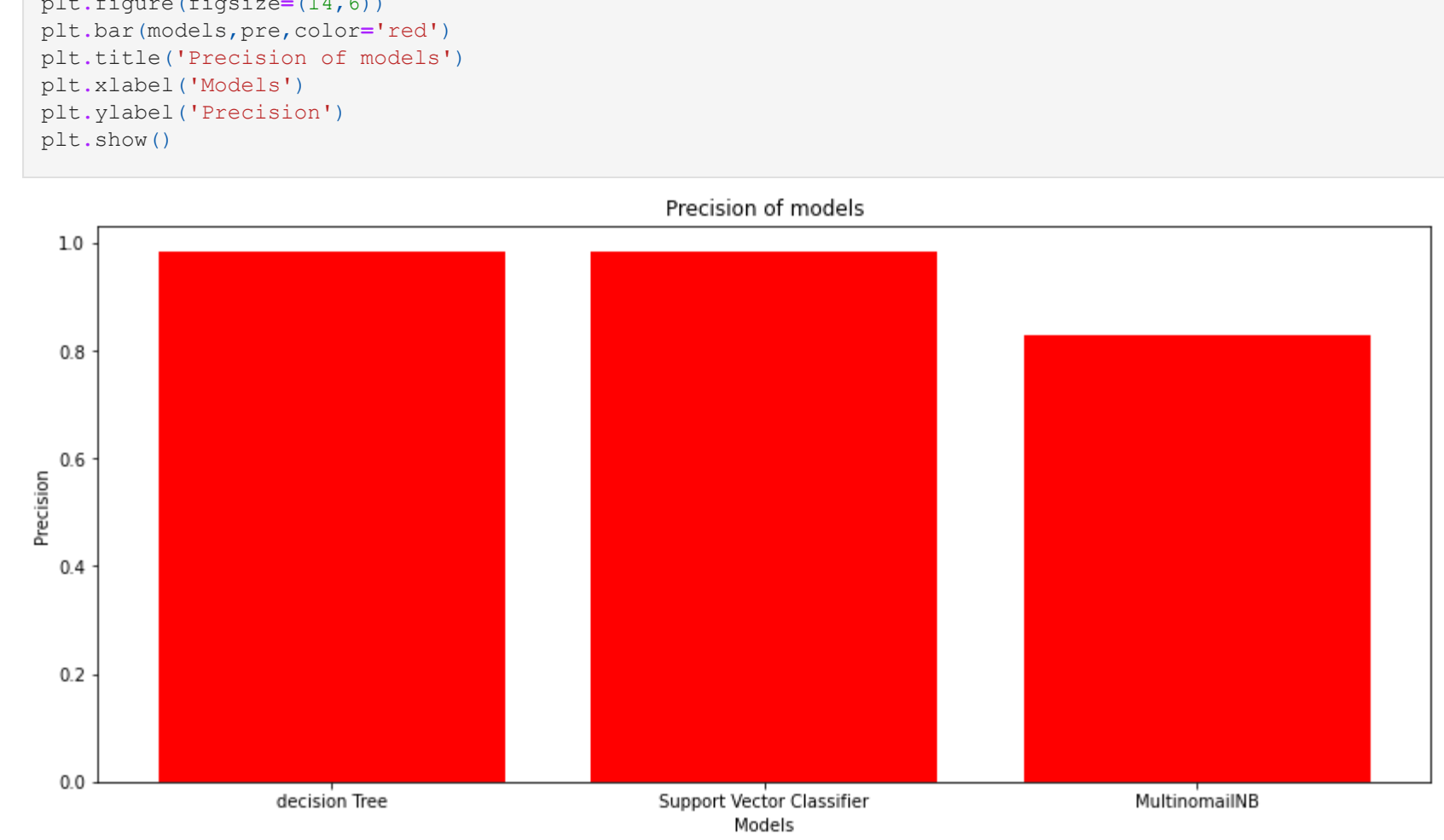
In [20]:

```
plt.figure(figsize=(14,6))
models = ["decision Tree","Support Vector Classifier","MultinomialNB"]
acc= [acc1,acc2,acc3]
pre= [pre1,pre2,pre3]
plt.bar(models,acc)
plt.title('Accuracy of models')
plt.xlabel('Models')
plt.ylabel('Accuracy')
plt.show()
```



In [21]:

```
plt.figure(figsize=(14,6))
plt.bar(models,pre,color='red')
plt.title('Precision of models')
plt.xlabel('Models')
plt.ylabel('Precision')
plt.show()
```



Conclusion : I will choose Descision tree classifier as final model

After Training and predicting with Multiple algorithms on test data set, We can conclude that multinomial model have highest accuracy but less precision and decision tree model and SVM(Support Vector Classifier) model also have high accuracy with good precision score but SVM(Support Vector Classifier) have more processing time than decision tree model. we can choose decision tree classifier as best algorithms to to sovle NLP problem, there are a couple of dvantages of using an descision tree classifier. Firstly, They are very fast and efficient compared to other classification algorithms. As decision trees are simple hence they require less effort for understanding an algorithm. The decision tree is one of the machine learning algorithms where we don't worry about its feature scaling.

In [ ]:

In [ ]:

In [ ]:

In [ ]: