FAKE NEWS DETECTION USING NLP

PHASE-3



Introduction:

In an era where information is disseminated at an unprecedented pace through digital channels, the propagation of fake news has become a significant concern. Fake news, characterized by the spread of deceptive, false, or misleading information, poses a substantial threat to society, democracy, and public trust. Detecting and combating fake news is an essential task, and Natural Language Processing (NLP) has emerged as a powerful tool in this endeavor.

1. **The Proliferation of Fake News:** Fake news, often driven by malicious intent, disinformation campaigns, or personal gain, has the potential to mislead, polarize, and manipulate individuals and communities. It can impact elections, public health, and public opinion, making its detection and prevention of utmost importance.
2. **The Role of Natural Language Processing (NLP):** NLP is a subfield of artificial intelligence (AI) that focuses on the interaction between computers and human language. NLP leverages techniques from machine learning, linguistics, and computer science to understand, process, and generate human language. This technology plays a crucial role in fake news detection for several reasons:
   * **Text Analysis:** NLP allows us to analyze and understand the content of news articles, social media posts, and other textual sources. By applying techniques such as sentiment analysis, topic modeling, and named entity recognition, NLP can uncover patterns and anomalies in text data that may indicate the presence of fake news.
   * **Language Modeling:** NLP models, like transformers, have the capacity to generate text that resembles human language. Detecting fake news can involve comparing the linguistic patterns and structures used in a piece of content to those typically found in legitimate news sources.
   * **Fact-Checking:** NLP systems can be trained to fact-check claims and statements made in news articles. By comparing the information with reputable sources and databases, NLP can identify inconsistencies and falsehoods.
   * **Social Media Analysis:** Much fake news spreads through social media platforms. NLP can be used to analyze social media data to detect the propagation of fake news stories, identifying patterns of engagement and the dissemination of misinformation.
3. **Challenges in Fake News Detection:** Detecting fake news using NLP is not without its challenges. These challenges include:
   * **Semantic Ambiguity:** Fake news can be subtly deceptive, and it may not rely on blatant falsehoods. Identifying nuanced forms of misinformation is a complex task for NLP models.
   * **Data Quality:** Ensuring that the data used for training and testing NLP models is of high quality and unbiased is essential for accurate detection.
   * **Adversarial Techniques:** Those who spread fake news are often aware of detection techniques and may employ adversarial strategies to evade detection.
4. **The Objectives of This Research:** The primary objective of this research is to explore and develop NLP-based techniques for the automated detection of fake news. We aim to create and refine models that can assist in distinguishing genuine news from fake news with a high degree of accuracy. By addressing the challenges and harnessing the potential of NLP, we contribute to the ongoing effort to safeguard the integrity of information in the digital age.

In conclusion, fake news detection using NLP is a critical area of research and application in the field of natural language processing. This technology is pivotal in the fight against the dissemination of deceptive information and the preservation of the public's trust in the media and information sources. In the following sections, we will delve deeper into the techniques, methodologies, and models used in fake news detection through NLP.

**DATASET LINK:**

<https://kaggle.com/datasets/2260158d80a4cc41ce77ce3e795e64cef5459490fd452ca266a474422d1faf69>

**Task type:** Classification

**Models used:** LinearSVC, MultinomialNB, XGBoost, PyCaret, CatBoost

**Tools used:** NLP preprocessing tools, semi-supervised learning technique, new feature engineering, Word Cloud

*# This Python 3 environment comes with many helpful analytics libraries installed*

*# It is defined by the kaggle/python Docker image: https://github.com/kaggle/docker-python*

*# For example, here's several helpful packages to load*

import numpy as np *# linear algebra*

import pandas as pd *# data processing, CSV file I/O (e.g. pd.read\_csv)*

*# Input data files are available in the read-only "../input/" directory*

*# For example, running this (by clicking run or pressing Shift+Enter) will list all files under the input directory*

import os

for dirname, \_, filenames **in** os.walk('/kaggle/input'):

for filename **in** filenames:

print(os.path.join(dirname, filename))

*# You can write up to 20GB to the current directory (/kaggle/working/) that gets preserved as output when you create a version using "Save & Run All"*

*# You can also write temporary files to /kaggle/temp/, but they won't be saved outside of the current session*

/kaggle/input/fake-and-real-news-dataset/True.csv

/kaggle/input/fake-and-real-news-dataset/Fake.csv

1. Loading data

In [2]:

fake = pd.read\_csv('../input/fake-and-real-news-dataset/Fake.csv')

fake['flag'] = 0

fake

Out[2]:

|  | title | text | subject | date | flag |
| --- | --- | --- | --- | --- | --- |
| 0 | Donald Trump Sends Out Embarrassing New Year’... | Donald Trump just couldn t wish all Americans ... | News | December 31, 2017 | 0 |
| 1 | Drunk Bragging Trump Staffer Started Russian ... | House Intelligence Committee Chairman Devin Nu... | News | December 31, 2017 | 0 |
| 2 | Sheriff David Clarke Becomes An Internet Joke... | On Friday, it was revealed that former Milwauk... | News | December 30, 2017 | 0 |
| 3 | Trump Is So Obsessed He Even Has Obama’s Name... | On Christmas day, Donald Trump announced that ... | News | December 29, 2017 | 0 |
| 4 | Pope Francis Just Called Out Donald Trump Dur... | Pope Francis used his annual Christmas Day mes... | News | December 25, 2017 | 0 |
| ... | ... | ... | ... | ... | ... |
| 23476 | McPain: John McCain Furious That Iran Treated ... | 21st Century Wire says As 21WIRE reported earl... | Middle-east | January 16, 2016 | 0 |
| 23477 | JUSTICE? Yahoo Settles E-mail Privacy Class-ac... | 21st Century Wire says It s a familiar theme. ... | Middle-east | January 16, 2016 | 0 |
| 23478 | Sunnistan: US and Allied ‘Safe Zone’ Plan to T... | Patrick Henningsen 21st Century WireRemember ... | Middle-east | January 15, 2016 | 0 |
| 23479 | How to Blow $700 Million: Al Jazeera America F... | 21st Century Wire says Al Jazeera America will... | Middle-east | January 14, 2016 | 0 |
| 23480 | 10 U.S. Navy Sailors Held by Iranian Military ... | 21st Century Wire says As 21WIRE predicted in ... | Middle-east | January 12, 2016 | 0 |

23481 rows × 5 columns

In [3]:

true = pd.read\_csv('../input/fake-and-real-news-dataset/True.csv')

true['flag'] = 1

true

Out[3]:

|  | title | text | subject | date | flag |
| --- | --- | --- | --- | --- | --- |
| 0 | As U.S. budget fight looms, Republicans flip t... | WASHINGTON (Reuters) - The head of a conservat... | politicsNews | December 31, 2017 | 1 |
| 1 | U.S. military to accept transgender recruits o... | WASHINGTON (Reuters) - Transgender people will... | politicsNews | December 29, 2017 | 1 |
| 2 | Senior U.S. Republican senator: 'Let Mr. Muell... | WASHINGTON (Reuters) - The special counsel inv... | politicsNews | December 31, 2017 | 1 |
| 3 | FBI Russia probe helped by Australian diplomat... | WASHINGTON (Reuters) - Trump campaign adviser ... | politicsNews | December 30, 2017 | 1 |
| 4 | Trump wants Postal Service to charge 'much mor... | SEATTLE/WASHINGTON (Reuters) - President Donal... | politicsNews | December 29, 2017 | 1 |
| ... | ... | ... | ... | ... | ... |
| 21412 | 'Fully committed' NATO backs new U.S. approach... | BRUSSELS (Reuters) - NATO allies on Tuesday we... | worldnews | August 22, 2017 | 1 |
| 21413 | LexisNexis withdrew two products from Chinese ... | LONDON (Reuters) - LexisNexis, a provider of l... | worldnews | August 22, 2017 | 1 |
| 21414 | Minsk cultural hub becomes haven from authorities | MINSK (Reuters) - In the shadow of disused Sov... | worldnews | August 22, 2017 | 1 |
| 21415 | Vatican upbeat on possibility of Pope Francis ... | MOSCOW (Reuters) - Vatican Secretary of State ... | worldnews | August 22, 2017 | 1 |
| 21416 | Indonesia to buy $1.14 billion worth of Russia... | JAKARTA (Reuters) - Indonesia will buy 11 Sukh... | worldnews | August 22, 2017 | 1 |

21417 rows × 5 columns

In [4]:

df = pd.DataFrame()

df = true.append(fake)

2. EDA + Data cleaning

**Let's check the datatypes.**

In [5]:

df.info()

<class 'pandas.core.frame.DataFrame'>

Int64Index: 44898 entries, 0 to 23480

Data columns (total 5 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 title 44898 non-null object

1 text 44898 non-null object

2 subject 44898 non-null object

3 date 44898 non-null object

4 flag 44898 non-null int64

dtypes: int64(1), object(4)

memory usage: 2.1+ MB

**Removing the duplicates and preventing problems with indexing.**

In [6]:

df = df.drop\_duplicates()

df = df.reset\_index(drop=True)

**We can see that the date format is not the one we need. I will apply the appropriate date format for future purposes.**

In [7]:

*# Correcting some data*

df['date'] = df['date'].replace(['19-Feb-18'],'February 19, 2018')

df['date'] = df['date'].replace(['18-Feb-18'],'February 18, 2018')

df['date'] = df['date'].replace(['17-Feb-18'],'February 17, 2018')

df['date'] = df['date'].replace(['16-Feb-18'],'February 16, 2018')

df['date'] = df['date'].replace(['15-Feb-18'],'February 15, 2018')

df['date'] = df['date'].replace(['14-Feb-18'],'February 14, 2018')

df['date'] = df['date'].replace(['13-Feb-18'],'February 13, 2018')

df['date'] = df['date'].str.replace('Dec ', 'December ')

df['date'] = df['date'].str.replace('Nov ', 'November ')

df['date'] = df['date'].str.replace('Oct ', 'October ')

df['date'] = df['date'].str.replace('Sep ', 'September ')

df['date'] = df['date'].str.replace('Aug ', 'August ')

df['date'] = df['date'].str.replace('Jul ', 'July ')

df['date'] = df['date'].str.replace('Jun ', 'June ')

df['date'] = df['date'].str.replace('Apr ', 'April ')

df['date'] = df['date'].str.replace('Mar ', 'March ')

df['date'] = df['date'].str.replace('Feb ', 'February ')

df['date'] = df['date'].str.replace('Jan ', 'January ')

In [8]:

df['date'] = df['date'].str.replace(' ', '')

In [9]:

for i, val **in** enumerate(df['date']):

df['date'].iloc[i] = pd.to\_datetime(df['date'].iloc[i], format='%B**%d**,%Y', errors='coerce') *# by setting the parameter to "coerce", we will set unappropriate values to NaT (null)*

/opt/conda/lib/python3.7/site-packages/pandas/core/indexing.py:1636: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user\_guide/indexing.html#returning-a-view-versus-a-copy

self.\_setitem\_single\_block(indexer, value, name)

In [10]:

df['date'] = df['date'].astype('datetime64[ns]')

In [11]:

df.info()

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 44689 entries, 0 to 44688

Data columns (total 5 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 title 44689 non-null object

1 text 44689 non-null object

2 subject 44689 non-null object

3 date 44679 non-null datetime64[ns]

4 flag 44689 non-null int64

dtypes: datetime64[ns](1), int64(1), object(3)

memory usage: 1.7+ MB

In [12]:

import datetime as dt

df['year'] = pd.to\_datetime(df['date']).dt.to\_period('Y')

df['month'] = pd.to\_datetime(df['date']).dt.to\_period('M')

df['month'] = df['month'].astype(str)

**Next we will try to elicit insights from non-text features to get to know if they will help us boost the Text Classifier.**

Fake news dynamics:

In [13]:

sub = df[['month', 'flag']]

sub = sub.dropna()

sub = sub.groupby(['month'])['flag'].sum()

In [14]:

sub = sub.drop('NaT')

In [15]:

import matplotlib.pyplot as plt

plt.suptitle('Dynamics of fake news')

plt.xticks(rotation=90)

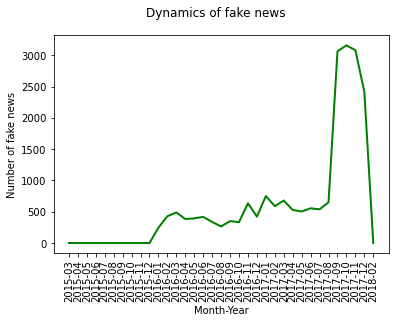
plt.ylabel('Number of fake news')

plt.xlabel('Month-Year')

plt.plot(sub.index, sub.values, linewidth=2, color='green')

Out[15]:

[<matplotlib.lines.Line2D at 0x7f16df024f10>]



**What a spike in the dynamics of fake news in late 2017!**

Subject distribution:

In [16]:

sub2 = df[['subject', 'flag']]

sub2 = sub2.dropna()

sub2 = sub2.groupby(['subject'])['flag'].sum()

In [17]:

plt.suptitle('Fake news among different categories')

plt.xticks(rotation=90)

plt.ylabel('Number of fake news')

plt.xlabel('Category')

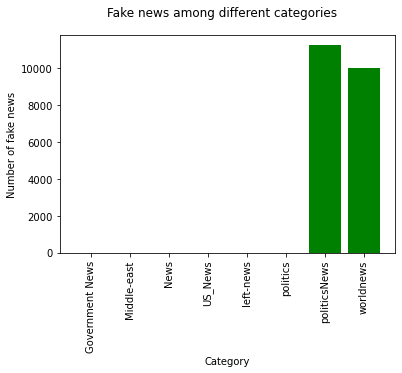
plt.bar(sub2.index, height=sub2.values, color='green')

*#ax1.plot(x, y)*

*#ax2.plot(x, -y)*

Out[17]:

<BarContainer object of 8 artists>



l

**As we have discovered, such features as**

* subject
* date

**might be also crucial for the algorithm to decide whether the piece of news is fake or real. We will try to include them in the model.**

3. Text preparation

In [18]:

nlp = df

**I will add the 'subject' feature to the title field as it might have an influence on the outcome of classification.**

In [19]:

*#nlp['title'] = nlp['title'] + ' ' + nlp['subject']*

3.1 Word Cloud visualization

**Here I am going to take one example and try visualize tfidf as a wordcloud.**

In [20]:

from sklearn.feature\_extraction.text import TfidfVectorizer

corpus = nlp[nlp['flag'] == 1]['title'].iloc[0:500] *# We will take a slice of fake news, to see what vocabulary there looks like*

tfidf1 = TfidfVectorizer()

vecs = tfidf1.fit\_transform(corpus)

feature\_names = tfidf1.get\_feature\_names()

dense = vecs.todense()

list\_words = dense.tolist()

df\_words = pd.DataFrame(list\_words, columns=feature\_names)

In [21]:

from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator

df\_words.T.sum(axis=1)

Cloud = WordCloud(background\_color="white", max\_words=100).generate\_from\_frequencies(df\_words.T.sum(axis=1))

In [22]:

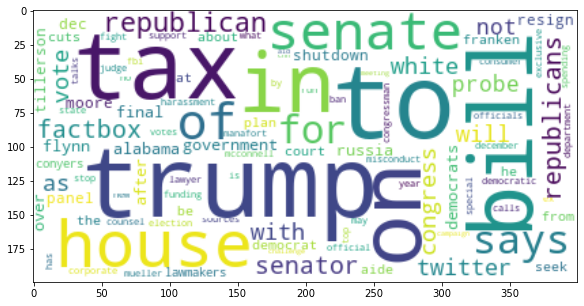
import matplotlib.pyplot as plt

plt.figure(figsize=(12,5))

plt.imshow(Cloud, interpolation='bilinear')

Out[22]:

<matplotlib.image.AxesImage at 0x7f16b19d6490>



**Indeed, looks definitely like fake news :)**

**And we can also see out 'subject' feature in the foreground as it has been added manually in every title. Therefore, out vectorizer considers it as an important & frequent word.**

3.2 Tfidf-vectorizing

**First, I will tokenize words to pass it on to the SnowballStemmer method, which will take out lemmas from words.**

In [23]:

import nltk

nltk.download('punkt')

from nltk import word\_tokenize

nlp['title'] = nlp['title'].apply(lambda x: word\_tokenize(str(x)))

[nltk\_data] Downloading package punkt to /usr/share/nltk\_data...

[nltk\_data] Package punkt is already up-to-date!

**An important step in every NLP-task is to get the roots of words in order not to distract the model by 'different' words.**

In [24]:

from nltk.stem import SnowballStemmer

snowball = SnowballStemmer(language='english')

nlp['title'] = nlp['title'].apply(lambda x: [snowball.stem(y) for y **in** x])

In [25]:

nlp['title'] = nlp['title'].apply(lambda x: ' '.join(x))

**Take the standard english bag of stopwords from nltk.**

In [26]:

from nltk.corpus import stopwords

nltk.download('words')

nltk.download('stopwords')

stopwords = stopwords.words('english')

[nltk\_data] Downloading package words to /usr/share/nltk\_data...

[nltk\_data] Package words is already up-to-date!

[nltk\_data] Downloading package stopwords to /usr/share/nltk\_data...

[nltk\_data] Package stopwords is already up-to-date!

**And finally TfidfVectorizing. You can also take CountVectorizer, but I prefer Tfidf as it has masses of advantages.**

In [27]:

from sklearn.feature\_extraction.text import TfidfVectorizer

tfidf = TfidfVectorizer()

X\_text = tfidf.fit\_transform(nlp['title'])

In [28]:

linkcode

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X\_text, nlp['flag'], test\_size=0.33, random\_state=1)

4. Model building:

**I will use several approaches to solve the classification task, such as:**

1) Traditional (which are known as efficient for text classification):

1.1) SVM

1.2) Naive Bayes

1.3) XGBoost

2) Not-very-traditional (Experimental): PyCaret NLP toolkit (I will apply unsupervised model to generate features which I will in turn pass on to the supervised model)

4.1 Linear SVC

In [29]:

scores = {}

In [30]:

from sklearn.svm import LinearSVC

from sklearn.model\_selection import cross\_val\_score

from sklearn.metrics import accuracy\_score

clf = LinearSVC(max\_iter=100, C=1.0)

clf.fit(X\_train, y\_train)

y\_pred\_SVM = clf.predict(X\_test)

print(cross\_val\_score(clf, X\_text, nlp['flag'], cv=3))

print(accuracy\_score(y\_pred\_SVM, y\_test))

scores['LinearSVC'] = accuracy\_score(y\_pred\_SVM, y\_test)

[0.91105592 0.93031686 0.92696026]

0.958706265256306

**This looks suspiciously good, but lets try another algorithm.**

4.2 Naive Bayes

In [31]:

from sklearn.naive\_bayes import MultinomialNB

clf2 = MultinomialNB()

clf2.fit(X\_train, y\_train)

y\_pred\_MNB = clf2.predict(X\_test)

print(cross\_val\_score(clf2, X\_text, nlp['flag'], cv=3))

print(accuracy\_score(y\_pred\_MNB, y\_test))

scores['MultinomialNB'] = accuracy\_score(y\_pred\_MNB, y\_test)

[0.88957508 0.89406552 0.92883996]

0.939924057499322

**Okay, this model performs a little worse, but still very good.**

4.3 XGBoost

In [32]:

from xgboost import XGBClassifier

clf3 = XGBClassifier(eval\_metric='rmse', use\_label\_encoder=False)

clf3.fit(X\_train, y\_train)

y\_pred\_XGB = clf3.predict(X\_test)

print(cross\_val\_score(clf3, X\_text, nlp['flag'], cv=3))

print(accuracy\_score(y\_pred\_XGB, y\_test))

scores['XGB'] = accuracy\_score(y\_pred\_XGB, y\_test)

[0.88615157 0.92353652 0.90695489]

0.9374830485489558

4.4 PyCaret + CatBoost

**PyCaret’s Natural Language Processing module is an unsupervised machine learning module that can be used for analyzing text data by creating topic models that can find hidden semantic structures within documents. PyCaret’s NLP module comes with a wide range of text pre-processing techniques. It has over 5 ready-to-use algorithms and several plots to analyze the performance of trained models and text corpus.**

In [33]:

linkcode

!pip install pycaret

**Setting up the model which will implement all traditional NLP-preprocessing operation (tokenizing, lemmatizing etc.**

**The PyCaret is almost fully automatic!**

In [34]:

from pycaret.nlp import \*

caret\_nlp = setup(data=nlp, target='title', session\_id=1)

| Description | Value |
| --- | --- |
| session\_id | 1 |
| Documents | 44689 |
| Vocab Size | 7568 |
| Custom Stopwords | False |

**LDA stands for Latent Dirichlet Allocation and is widely used in unsupervised learning tasks.**

In [35]:

lda = create\_model('lda')

In [36]:

lda\_data = assign\_model(lda)

**Here's the outcome dataset:**

In [37]:

lda\_data

Out[37]:

|  | title | text | subject | date | flag | year | month | Topic\_0 | Topic\_1 | Topic\_2 | Topic\_3 | Dominant\_Topic | Perc\_Dominant\_Topic |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | budget fight loom flip fiscal script | WASHINGTON (Reuters) - The head of a conservat... | politicsNews | 2017-12-31 | 1 | 2017 | 2017-12 | 0.190063 | 0.271491 | 0.203485 | 0.334961 | Topic 3 | 0.33 |
| 1 | transgend | WASHINGTON (Reuters) - Transgender people will... | politicsNews | 2017-12-29 | 1 | 2017 | 2017-12 | 0.197559 | 0.273950 | 0.204855 | 0.323636 | Topic 3 | 0.32 |
| 2 | senior let job | WASHINGTON (Reuters) - The special counsel inv... | politicsNews | 2017-12-31 | 1 | 2017 | 2017-12 | 0.187440 | 0.291736 | 0.201883 | 0.318941 | Topic 3 | 0.32 |
| 3 | diplomat | WASHINGTON (Reuters) - Trump campaign adviser ... | politicsNews | 2017-12-30 | 1 | 2017 | 2017-12 | 0.190208 | 0.273961 | 0.204864 | 0.330968 | Topic 3 | 0.33 |
| 4 | trump want postal charg much shipment | SEATTLE/WASHINGTON (Reuters) - President Donal... | politicsNews | 2017-12-29 | 1 | 2017 | 2017-12 | 0.185147 | 0.286900 | 0.199272 | 0.328681 | Topic 3 | 0.33 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 44684 | furious treat sailor well | 21st Century Wire says As 21WIRE reported earl... | Middle-east | 2016-01-16 | 0 | 2016 | 2016-01 | 0.187538 | 0.292114 | 0.201669 | 0.318679 | Topic 3 | 0.32 |
| 44685 | settl lawyer user | 21st Century Wire says It s a familiar theme. ... | Middle-east | 2016-01-16 | 0 | 2016 | 2016-01 | 0.187469 | 0.291635 | 0.201911 | 0.318984 | Topic 3 | 0.32 |
| 44686 | plan take territori | Patrick Henningsen 21st Century WireRemember ... | Middle-east | 2016-01-15 | 0 | 2016 | 2016-01 | 0.196395 | 0.271734 | 0.203346 | 0.328525 | Topic 3 | 0.33 |
| 44687 | final call quit | 21st Century Wire says Al Jazeera America will... | Middle-east | 2016-01-14 | 0 | 2016 | 2016-01 | 0.194696 | 0.269975 | 0.209132 | 0.326198 | Topic 3 | 0.33 |
| 44688 | hold iranian | 21st Century Wire says As 21WIRE predicted in ... | Middle-east | 2016-01-12 | 0 | 2016 | 2016-01 | 0.196119 | 0.279245 | 0.203361 | 0.321275 | Topic 3 | 0.32 |

44689 rows × 13 columns

**We'll utilize the 'Topic' features generated by PyCaret.**

In [38]:

from catboost import CatBoostClassifier

In [39]:

input\_cat = lda\_data.drop(['text','date','Perc\_Dominant\_Topic','flag','year'], axis=1)

input\_cat['month'] = input\_cat['month'].astype(str)

target\_cat = lda\_data['flag']

In [40]:

from sklearn.model\_selection import train\_test\_split

X\_train\_cat, X\_test\_cat, y\_train\_cat, y\_test\_cat = train\_test\_split(input\_cat, target\_cat, test\_size=0.33, random\_state=1)

In [41]:

clf4 = CatBoostClassifier(iterations=1000,

cat\_features=['title','subject','Dominant\_Topic','month']

)

In [42]:

clf4.fit(X\_train\_cat, y\_train\_cat, early\_stopping\_rounds=10)

Learning rate set to 0.043985

0: learn: 0.5145341 total: 83.5ms remaining: 1m 23s

1: learn: 0.3851221 total: 109ms remaining: 54.3s

2: learn: 0.2865127 total: 144ms remaining: 47.7s

3: learn: 0.2123637 total: 171ms remaining: 42.5s

4: learn: 0.1531220 total: 186ms remaining: 37s

5: learn: 0.1122692 total: 211ms remaining: 35s

6: learn: 0.0840460 total: 241ms remaining: 34.3s

7: learn: 0.0647696 total: 294ms remaining: 36.5s

8: learn: 0.0502976 total: 323ms remaining: 35.6s

9: learn: 0.0397706 total: 363ms remaining: 35.9s

10: learn: 0.0311634 total: 394ms remaining: 35.4s

11: learn: 0.0242480 total: 414ms remaining: 34.1s

12: learn: 0.0189828 total: 433ms remaining: 32.9s

13: learn: 0.0150631 total: 459ms remaining: 32.3s

14: learn: 0.0121618 total: 486ms remaining: 31.9s

15: learn: 0.0097319 total: 505ms remaining: 31.1s

16: learn: 0.0078641 total: 525ms remaining: 30.4s

17: learn: 0.0065187 total: 544ms remaining: 29.7s

18: learn: 0.0053943 total: 565ms remaining: 29.1s

19: learn: 0.0045027 total: 584ms remaining: 28.6s

20: learn: 0.0037708 total: 605ms remaining: 28.2s

21: learn: 0.0032655 total: 628ms remaining: 27.9s

22: learn: 0.0027977 total: 647ms remaining: 27.5s

23: learn: 0.0023977 total: 668ms remaining: 27.1s

24: learn: 0.0020752 total: 685ms remaining: 26.7s

25: learn: 0.0017890 total: 702ms remaining: 26.3s

26: learn: 0.0015641 total: 721ms remaining: 26s

27: learn: 0.0013903 total: 739ms remaining: 25.6s

28: learn: 0.0012290 total: 755ms remaining: 25.3s

29: learn: 0.0011056 total: 774ms remaining: 25s

30: learn: 0.0009950 total: 795ms remaining: 24.8s

31: learn: 0.0009029 total: 816ms remaining: 24.7s

32: learn: 0.0008133 total: 834ms remaining: 24.4s

33: learn: 0.0007523 total: 852ms remaining: 24.2s

34: learn: 0.0006801 total: 870ms remaining: 24s

35: learn: 0.0006323 total: 888ms remaining: 23.8s

36: learn: 0.0005810 total: 906ms remaining: 23.6s

37: learn: 0.0005368 total: 923ms remaining: 23.4s

38: learn: 0.0005118 total: 940ms remaining: 23.2s

39: learn: 0.0004745 total: 958ms remaining: 23s

40: learn: 0.0004406 total: 977ms remaining: 22.8s

41: learn: 0.0004097 total: 997ms remaining: 22.7s

42: learn: 0.0003892 total: 1.02s remaining: 22.7s

43: learn: 0.0003648 total: 1.04s remaining: 22.7s

44: learn: 0.0003366 total: 1.06s remaining: 22.5s

45: learn: 0.0003158 total: 1.09s remaining: 22.5s

46: learn: 0.0003086 total: 1.11s remaining: 22.5s

47: learn: 0.0002960 total: 1.13s remaining: 22.3s

48: learn: 0.0002827 total: 1.14s remaining: 22.2s

49: learn: 0.0002694 total: 1.16s remaining: 21.9s

50: learn: 0.0002602 total: 1.17s remaining: 21.8s

51: learn: 0.0002512 total: 1.19s remaining: 21.8s

52: learn: 0.0002424 total: 1.22s remaining: 21.7s

53: learn: 0.0002394 total: 1.24s remaining: 21.7s

54: learn: 0.0002319 total: 1.25s remaining: 21.6s

55: learn: 0.0002232 total: 1.28s remaining: 21.5s

56: learn: 0.0002176 total: 1.29s remaining: 21.4s

57: learn: 0.0002070 total: 1.31s remaining: 21.4s

58: learn: 0.0001998 total: 1.33s remaining: 21.3s

59: learn: 0.0001943 total: 1.35s remaining: 21.2s

60: learn: 0.0001890 total: 1.37s remaining: 21.1s

61: learn: 0.0001844 total: 1.39s remaining: 21s

62: learn: 0.0001810 total: 1.4s remaining: 20.9s

63: learn: 0.0001763 total: 1.42s remaining: 20.8s

64: learn: 0.0001719 total: 1.44s remaining: 20.7s

65: learn: 0.0001677 total: 1.46s remaining: 20.6s

66: learn: 0.0001648 total: 1.48s remaining: 20.6s

67: learn: 0.0001648 total: 1.49s remaining: 20.5s

68: learn: 0.0001619 total: 1.51s remaining: 20.4s

69: learn: 0.0001594 total: 1.53s remaining: 20.4s

70: learn: 0.0001571 total: 1.55s remaining: 20.3s

71: learn: 0.0001571 total: 1.57s remaining: 20.3s

72: learn: 0.0001546 total: 1.59s remaining: 20.2s

73: learn: 0.0001546 total: 1.61s remaining: 20.1s

74: learn: 0.0001546 total: 1.62s remaining: 20s

75: learn: 0.0001546 total: 1.64s remaining: 19.9s

76: learn: 0.0001521 total: 1.66s remaining: 19.9s

77: learn: 0.0001521 total: 1.68s remaining: 19.9s

78: learn: 0.0001503 total: 1.7s remaining: 19.8s

79: learn: 0.0001490 total: 1.72s remaining: 19.8s

80: learn: 0.0001490 total: 1.74s remaining: 19.7s

81: learn: 0.0001466 total: 1.76s remaining: 19.7s

82: learn: 0.0001400 total: 1.77s remaining: 19.6s

83: learn: 0.0001400 total: 1.79s remaining: 19.5s

84: learn: 0.0001390 total: 1.81s remaining: 19.5s

85: learn: 0.0001390 total: 1.82s remaining: 19.4s

86: learn: 0.0001377 total: 1.84s remaining: 19.3s

87: learn: 0.0001364 total: 1.86s remaining: 19.3s

88: learn: 0.0001353 total: 1.88s remaining: 19.2s

89: learn: 0.0001342 total: 1.89s remaining: 19.1s

90: learn: 0.0001328 total: 1.91s remaining: 19s

91: learn: 0.0001318 total: 1.92s remaining: 19s

92: learn: 0.0001318 total: 1.93s remaining: 18.9s

93: learn: 0.0001318 total: 1.95s remaining: 18.8s

94: learn: 0.0001309 total: 1.97s remaining: 18.7s

95: learn: 0.0001309 total: 1.98s remaining: 18.7s

96: learn: 0.0001299 total: 2s remaining: 18.6s

97: learn: 0.0001299 total: 2.01s remaining: 18.5s

98: learn: 0.0001290 total: 2.03s remaining: 18.5s

99: learn: 0.0001279 total: 2.05s remaining: 18.4s

100: learn: 0.0001267 total: 2.06s remaining: 18.3s

101: learn: 0.0001257 total: 2.08s remaining: 18.3s

102: learn: 0.0001257 total: 2.1s remaining: 18.3s

103: learn: 0.0001257 total: 2.11s remaining: 18.2s

104: learn: 0.0001257 total: 2.13s remaining: 18.1s

105: learn: 0.0001257 total: 2.15s remaining: 18.1s

106: learn: 0.0001257 total: 2.16s remaining: 18s

107: learn: 0.0001257 total: 2.17s remaining: 17.9s

108: learn: 0.0001257 total: 2.19s remaining: 17.9s

109: learn: 0.0001257 total: 2.2s remaining: 17.8s

110: learn: 0.0001249 total: 2.22s remaining: 17.8s

111: learn: 0.0001239 total: 2.23s remaining: 17.7s

112: learn: 0.0001239 total: 2.24s remaining: 17.6s

113: learn: 0.0001239 total: 2.26s remaining: 17.6s

114: learn: 0.0001239 total: 2.27s remaining: 17.5s

115: learn: 0.0001239 total: 2.29s remaining: 17.5s

116: learn: 0.0001239 total: 2.3s remaining: 17.4s

117: learn: 0.0001239 total: 2.32s remaining: 17.3s

118: learn: 0.0001239 total: 2.33s remaining: 17.3s

119: learn: 0.0001239 total: 2.35s remaining: 17.2s

120: learn: 0.0001228 total: 2.36s remaining: 17.1s

121: learn: 0.0001217 total: 2.38s remaining: 17.1s

122: learn: 0.0001217 total: 2.39s remaining: 17.1s

123: learn: 0.0001217 total: 2.41s remaining: 17s

124: learn: 0.0001217 total: 2.42s remaining: 17s

125: learn: 0.0001217 total: 2.44s remaining: 16.9s

126: learn: 0.0001217 total: 2.46s remaining: 16.9s

127: learn: 0.0001217 total: 2.48s remaining: 16.9s

128: learn: 0.0001217 total: 2.49s remaining: 16.8s

129: learn: 0.0001217 total: 2.51s remaining: 16.8s

130: learn: 0.0001217 total: 2.52s remaining: 16.7s

131: learn: 0.0001205 total: 2.54s remaining: 16.7s

132: learn: 0.0001205 total: 2.55s remaining: 16.6s

133: learn: 0.0001205 total: 2.57s remaining: 16.6s

134: learn: 0.0001205 total: 2.59s remaining: 16.6s

135: learn: 0.0001205 total: 2.6s remaining: 16.5s

136: learn: 0.0001205 total: 2.62s remaining: 16.5s

137: learn: 0.0001205 total: 2.63s remaining: 16.4s

138: learn: 0.0001205 total: 2.65s remaining: 16.4s

139: learn: 0.0001205 total: 2.66s remaining: 16.4s

140: learn: 0.0001205 total: 2.67s remaining: 16.3s

141: learn: 0.0001205 total: 2.69s remaining: 16.3s

142: learn: 0.0001205 total: 2.71s remaining: 16.2s

143: learn: 0.0001205 total: 2.72s remaining: 16.2s

144: learn: 0.0001205 total: 2.73s remaining: 16.1s

145: learn: 0.0001195 total: 2.75s remaining: 16.1s

146: learn: 0.0001195 total: 2.77s remaining: 16.1s

147: learn: 0.0001195 total: 2.79s remaining: 16s

148: learn: 0.0001195 total: 2.81s remaining: 16s

149: learn: 0.0001195 total: 2.83s remaining: 16s

150: learn: 0.0001195 total: 2.84s remaining: 16s

151: learn: 0.0001195 total: 2.85s remaining: 15.9s

152: learn: 0.0001195 total: 2.87s remaining: 15.9s

153: learn: 0.0001195 total: 2.88s remaining: 15.9s

154: learn: 0.0001195 total: 2.9s remaining: 15.8s

155: learn: 0.0001195 total: 2.92s remaining: 15.8s

156: learn: 0.0001195 total: 2.93s remaining: 15.8s

157: learn: 0.0001195 total: 2.95s remaining: 15.7s

158: learn: 0.0001195 total: 2.96s remaining: 15.7s

159: learn: 0.0001195 total: 2.99s remaining: 15.7s

160: learn: 0.0001195 total: 3s remaining: 15.7s

161: learn: 0.0001195 total: 3.02s remaining: 15.6s

162: learn: 0.0001184 total: 3.03s remaining: 15.6s

163: learn: 0.0001184 total: 3.05s remaining: 15.5s

164: learn: 0.0001184 total: 3.06s remaining: 15.5s

165: learn: 0.0001184 total: 3.08s remaining: 15.5s

166: learn: 0.0001184 total: 3.1s remaining: 15.5s

167: learn: 0.0001184 total: 3.12s remaining: 15.4s

168: learn: 0.0001184 total: 3.14s remaining: 15.4s

169: learn: 0.0001173 total: 3.15s remaining: 15.4s

170: learn: 0.0001173 total: 3.17s remaining: 15.4s

171: learn: 0.0001163 total: 3.19s remaining: 15.3s

172: learn: 0.0001163 total: 3.21s remaining: 15.3s

173: learn: 0.0001163 total: 3.22s remaining: 15.3s

174: learn: 0.0001163 total: 3.24s remaining: 15.3s

175: learn: 0.0001163 total: 3.25s remaining: 15.2s

176: learn: 0.0001163 total: 3.27s remaining: 15.2s

177: learn: 0.0001163 total: 3.29s remaining: 15.2s

178: learn: 0.0001163 total: 3.31s remaining: 15.2s

179: learn: 0.0001163 total: 3.33s remaining: 15.1s

180: learn: 0.0001163 total: 3.34s remaining: 15.1s

181: learn: 0.0001163 total: 3.36s remaining: 15.1s

182: learn: 0.0001163 total: 3.38s remaining: 15.1s

183: learn: 0.0001163 total: 3.4s remaining: 15.1s

184: learn: 0.0001163 total: 3.42s remaining: 15.1s

185: learn: 0.0001163 total: 3.44s remaining: 15s

186: learn: 0.0001163 total: 3.45s remaining: 15s

187: learn: 0.0001163 total: 3.47s remaining: 15s

188: learn: 0.0001163 total: 3.48s remaining: 14.9s

189: learn: 0.0001163 total: 3.5s remaining: 14.9s

190: learn: 0.0001163 total: 3.51s remaining: 14.9s

191: learn: 0.0001163 total: 3.53s remaining: 14.9s

192: learn: 0.0001163 total: 3.55s remaining: 14.8s

193: learn: 0.0001163 total: 3.56s remaining: 14.8s

194: learn: 0.0001163 total: 3.58s remaining: 14.8s

195: learn: 0.0001163 total: 3.59s remaining: 14.7s

196: learn: 0.0001163 total: 3.61s remaining: 14.7s

197: learn: 0.0001163 total: 3.63s remaining: 14.7s

198: learn: 0.0001163 total: 3.64s remaining: 14.7s

199: learn: 0.0001163 total: 3.66s remaining: 14.6s

200: learn: 0.0001163 total: 3.67s remaining: 14.6s

201: learn: 0.0001154 total: 3.69s remaining: 14.6s

202: learn: 0.0001154 total: 3.7s remaining: 14.5s

203: learn: 0.0001154 total: 3.71s remaining: 14.5s

204: learn: 0.0001154 total: 3.73s remaining: 14.5s

205: learn: 0.0001154 total: 3.75s remaining: 14.5s

206: learn: 0.0001154 total: 3.77s remaining: 14.5s

207: learn: 0.0001154 total: 3.79s remaining: 14.4s

208: learn: 0.0001154 total: 3.81s remaining: 14.4s

209: learn: 0.0001154 total: 3.82s remaining: 14.4s

210: learn: 0.0001154 total: 3.84s remaining: 14.3s

211: learn: 0.0001154 total: 3.85s remaining: 14.3s

212: learn: 0.0001154 total: 3.87s remaining: 14.3s

213: learn: 0.0001154 total: 3.88s remaining: 14.3s

214: learn: 0.0001154 total: 3.89s remaining: 14.2s

215: learn: 0.0001154 total: 3.91s remaining: 14.2s

216: learn: 0.0001154 total: 3.92s remaining: 14.2s

217: learn: 0.0001154 total: 3.94s remaining: 14.1s

218: learn: 0.0001154 total: 3.96s remaining: 14.1s

219: learn: 0.0001154 total: 3.97s remaining: 14.1s

220: learn: 0.0001154 total: 3.99s remaining: 14.1s

221: learn: 0.0001154 total: 4.01s remaining: 14s

222: learn: 0.0001154 total: 4.02s remaining: 14s

223: learn: 0.0001154 total: 4.04s remaining: 14s

224: learn: 0.0001154 total: 4.06s remaining: 14s

225: learn: 0.0001154 total: 4.08s remaining: 14s

226: learn: 0.0001154 total: 4.09s remaining: 13.9s

227: learn: 0.0001154 total: 4.11s remaining: 13.9s

228: learn: 0.0001143 total: 4.12s remaining: 13.9s

229: learn: 0.0001143 total: 4.14s remaining: 13.9s

230: learn: 0.0001143 total: 4.15s remaining: 13.8s

231: learn: 0.0001143 total: 4.17s remaining: 13.8s

232: learn: 0.0001143 total: 4.18s remaining: 13.8s

233: learn: 0.0001143 total: 4.19s remaining: 13.7s

234: learn: 0.0001134 total: 4.21s remaining: 13.7s

235: learn: 0.0001125 total: 4.22s remaining: 13.7s

236: learn: 0.0001125 total: 4.24s remaining: 13.6s

237: learn: 0.0001125 total: 4.26s remaining: 13.6s

238: learn: 0.0001125 total: 4.27s remaining: 13.6s

239: learn: 0.0001125 total: 4.29s remaining: 13.6s

240: learn: 0.0001125 total: 4.3s remaining: 13.5s

241: learn: 0.0001125 total: 4.32s remaining: 13.5s

242: learn: 0.0001125 total: 4.33s remaining: 13.5s

243: learn: 0.0001125 total: 4.34s remaining: 13.5s

244: learn: 0.0001125 total: 4.36s remaining: 13.4s

245: learn: 0.0001125 total: 4.38s remaining: 13.4s

246: learn: 0.0001125 total: 4.41s remaining: 13.4s

247: learn: 0.0001125 total: 4.43s remaining: 13.4s

248: learn: 0.0001125 total: 4.46s remaining: 13.4s

249: learn: 0.0001125 total: 4.48s remaining: 13.4s

250: learn: 0.0001125 total: 4.51s remaining: 13.4s

251: learn: 0.0001125 total: 4.53s remaining: 13.4s

252: learn: 0.0001125 total: 4.55s remaining: 13.4s

253: learn: 0.0001125 total: 4.57s remaining: 13.4s

254: learn: 0.0001125 total: 4.58s remaining: 13.4s

255: learn: 0.0001125 total: 4.6s remaining: 13.4s

256: learn: 0.0001125 total: 4.61s remaining: 13.3s

257: learn: 0.0001125 total: 4.63s remaining: 13.3s

258: learn: 0.0001103 total: 4.64s remaining: 13.3s

259: learn: 0.0001103 total: 4.66s remaining: 13.3s

260: learn: 0.0001103 total: 4.68s remaining: 13.2s

261: learn: 0.0001103 total: 4.69s remaining: 13.2s

262: learn: 0.0001103 total: 4.71s remaining: 13.2s

263: learn: 0.0001103 total: 4.72s remaining: 13.2s

264: learn: 0.0001103 total: 4.74s remaining: 13.1s

265: learn: 0.0001103 total: 4.75s remaining: 13.1s

266: learn: 0.0001103 total: 4.77s remaining: 13.1s

267: learn: 0.0001103 total: 4.79s remaining: 13.1s

268: learn: 0.0001103 total: 4.81s remaining: 13.1s

269: learn: 0.0001103 total: 4.82s remaining: 13s

270: learn: 0.0001103 total: 4.84s remaining: 13s

271: learn: 0.0001103 total: 4.85s remaining: 13s

272: learn: 0.0001103 total: 4.87s remaining: 13s

273: learn: 0.0001103 total: 4.88s remaining: 12.9s

274: learn: 0.0001103 total: 4.9s remaining: 12.9s

275: learn: 0.0001103 total: 4.91s remaining: 12.9s

276: learn: 0.0001103 total: 4.93s remaining: 12.9s

277: learn: 0.0001103 total: 4.94s remaining: 12.8s

278: learn: 0.0001103 total: 4.96s remaining: 12.8s

279: learn: 0.0001103 total: 4.97s remaining: 12.8s

280: learn: 0.0001103 total: 4.98s remaining: 12.8s

281: learn: 0.0001103 total: 5s remaining: 12.7s

282: learn: 0.0001103 total: 5.01s remaining: 12.7s

283: learn: 0.0001103 total: 5.03s remaining: 12.7s

284: learn: 0.0001103 total: 5.04s remaining: 12.7s

285: learn: 0.0001103 total: 5.06s remaining: 12.6s

286: learn: 0.0001103 total: 5.07s remaining: 12.6s

287: learn: 0.0001103 total: 5.09s remaining: 12.6s

288: learn: 0.0001103 total: 5.1s remaining: 12.5s

289: learn: 0.0001103 total: 5.11s remaining: 12.5s

290: learn: 0.0001103 total: 5.13s remaining: 12.5s

291: learn: 0.0001103 total: 5.14s remaining: 12.5s

292: learn: 0.0001103 total: 5.16s remaining: 12.4s

293: learn: 0.0001103 total: 5.17s remaining: 12.4s

294: learn: 0.0001103 total: 5.19s remaining: 12.4s

295: learn: 0.0001103 total: 5.21s remaining: 12.4s

296: learn: 0.0001103 total: 5.22s remaining: 12.4s

297: learn: 0.0001103 total: 5.24s remaining: 12.3s

298: learn: 0.0001103 total: 5.25s remaining: 12.3s

299: learn: 0.0001103 total: 5.27s remaining: 12.3s

300: learn: 0.0001103 total: 5.28s remaining: 12.3s

301: learn: 0.0001103 total: 5.3s remaining: 12.2s

302: learn: 0.0001103 total: 5.31s remaining: 12.2s

303: learn: 0.0001103 total: 5.32s remaining: 12.2s

304: learn: 0.0001103 total: 5.34s remaining: 12.2s

305: learn: 0.0001103 total: 5.36s remaining: 12.2s

306: learn: 0.0001103 total: 5.38s remaining: 12.1s

307: learn: 0.0001103 total: 5.39s remaining: 12.1s

308: learn: 0.0001103 total: 5.41s remaining: 12.1s

309: learn: 0.0001103 total: 5.42s remaining: 12.1s

310: learn: 0.0001103 total: 5.44s remaining: 12.1s

311: learn: 0.0001103 total: 5.45s remaining: 12s

312: learn: 0.0001103 total: 5.47s remaining: 12s

313: learn: 0.0001103 total: 5.49s remaining: 12s

314: learn: 0.0001103 total: 5.5s remaining: 12s

315: learn: 0.0001103 total: 5.52s remaining: 11.9s

316: learn: 0.0001103 total: 5.54s remaining: 11.9s

317: learn: 0.0001103 total: 5.55s remaining: 11.9s

318: learn: 0.0001103 total: 5.57s remaining: 11.9s

319: learn: 0.0001103 total: 5.58s remaining: 11.9s

320: learn: 0.0001103 total: 5.59s remaining: 11.8s

321: learn: 0.0001103 total: 5.61s remaining: 11.8s

322: learn: 0.0001103 total: 5.62s remaining: 11.8s

323: learn: 0.0001103 total: 5.64s remaining: 11.8s

324: learn: 0.0001103 total: 5.65s remaining: 11.7s

325: learn: 0.0001103 total: 5.67s remaining: 11.7s

326: learn: 0.0001103 total: 5.68s remaining: 11.7s

327: learn: 0.0001103 total: 5.7s remaining: 11.7s

328: learn: 0.0001103 total: 5.71s remaining: 11.7s

329: learn: 0.0001103 total: 5.73s remaining: 11.6s

330: learn: 0.0001103 total: 5.75s remaining: 11.6s

331: learn: 0.0001103 total: 5.76s remaining: 11.6s

332: learn: 0.0001103 total: 5.78s remaining: 11.6s

333: learn: 0.0001102 total: 5.79s remaining: 11.6s

334: learn: 0.0001102 total: 5.81s remaining: 11.5s

335: learn: 0.0001102 total: 5.82s remaining: 11.5s

336: learn: 0.0001102 total: 5.84s remaining: 11.5s

337: learn: 0.0001102 total: 5.86s remaining: 11.5s

338: learn: 0.0001102 total: 5.87s remaining: 11.5s

339: learn: 0.0001102 total: 5.89s remaining: 11.4s

340: learn: 0.0001102 total: 5.9s remaining: 11.4s

341: learn: 0.0001102 total: 5.92s remaining: 11.4s

342: learn: 0.0001102 total: 5.93s remaining: 11.4s

343: learn: 0.0001102 total: 5.95s remaining: 11.3s

344: learn: 0.0001102 total: 5.96s remaining: 11.3s

345: learn: 0.0001102 total: 5.98s remaining: 11.3s

346: learn: 0.0001102 total: 5.99s remaining: 11.3s

347: learn: 0.0001102 total: 6.01s remaining: 11.3s

348: learn: 0.0001102 total: 6.02s remaining: 11.2s

349: learn: 0.0001102 total: 6.04s remaining: 11.2s

350: learn: 0.0001102 total: 6.05s remaining: 11.2s

351: learn: 0.0001102 total: 6.07s remaining: 11.2s

352: learn: 0.0001102 total: 6.09s remaining: 11.2s

353: learn: 0.0001102 total: 6.1s remaining: 11.1s

354: learn: 0.0001102 total: 6.12s remaining: 11.1s

355: learn: 0.0001102 total: 6.13s remaining: 11.1s

356: learn: 0.0001102 total: 6.16s remaining: 11.1s

357: learn: 0.0001102 total: 6.18s remaining: 11.1s

358: learn: 0.0001102 total: 6.19s remaining: 11.1s

359: learn: 0.0001102 total: 6.21s remaining: 11s

360: learn: 0.0001102 total: 6.22s remaining: 11s

361: learn: 0.0001102 total: 6.24s remaining: 11s

362: learn: 0.0001102 total: 6.25s remaining: 11s

363: learn: 0.0001102 total: 6.27s remaining: 11s

364: learn: 0.0001102 total: 6.28s remaining: 10.9s

365: learn: 0.0001102 total: 6.3s remaining: 10.9s

366: learn: 0.0001102 total: 6.31s remaining: 10.9s

367: learn: 0.0001102 total: 6.33s remaining: 10.9s

368: learn: 0.0001102 total: 6.34s remaining: 10.8s

369: learn: 0.0001102 total: 6.36s remaining: 10.8s

370: learn: 0.0001102 total: 6.37s remaining: 10.8s

371: learn: 0.0001102 total: 6.39s remaining: 10.8s

372: learn: 0.0001102 total: 6.4s remaining: 10.8s

373: learn: 0.0001102 total: 6.42s remaining: 10.7s

374: learn: 0.0001102 total: 6.43s remaining: 10.7s

375: learn: 0.0001102 total: 6.45s remaining: 10.7s

376: learn: 0.0001102 total: 6.46s remaining: 10.7s

377: learn: 0.0001102 total: 6.48s remaining: 10.7s

378: learn: 0.0001102 total: 6.49s remaining: 10.6s

379: learn: 0.0001102 total: 6.51s remaining: 10.6s

380: learn: 0.0001102 total: 6.52s remaining: 10.6s

381: learn: 0.0001102 total: 6.54s remaining: 10.6s

382: learn: 0.0001102 total: 6.55s remaining: 10.6s

383: learn: 0.0001102 total: 6.57s remaining: 10.5s

384: learn: 0.0001102 total: 6.58s remaining: 10.5s

385: learn: 0.0001102 total: 6.59s remaining: 10.5s

386: learn: 0.0001102 total: 6.61s remaining: 10.5s

387: learn: 0.0001102 total: 6.62s remaining: 10.4s

388: learn: 0.0001102 total: 6.64s remaining: 10.4s

389: learn: 0.0001102 total: 6.65s remaining: 10.4s

390: learn: 0.0001102 total: 6.67s remaining: 10.4s

391: learn: 0.0001102 total: 6.68s remaining: 10.4s

392: learn: 0.0001102 total: 6.7s remaining: 10.3s

393: learn: 0.0001102 total: 6.71s remaining: 10.3s

394: learn: 0.0001102 total: 6.73s remaining: 10.3s

395: learn: 0.0001102 total: 6.75s remaining: 10.3s

396: learn: 0.0001102 total: 6.76s remaining: 10.3s

397: learn: 0.0001102 total: 6.77s remaining: 10.2s

398: learn: 0.0001102 total: 6.79s remaining: 10.2s

399: learn: 0.0001102 total: 6.8s remaining: 10.2s

400: learn: 0.0001102 total: 6.81s remaining: 10.2s

401: learn: 0.0001102 total: 6.83s remaining: 10.2s

402: learn: 0.0001102 total: 6.84s remaining: 10.1s

403: learn: 0.0001102 total: 6.86s remaining: 10.1s

404: learn: 0.0001102 total: 6.87s remaining: 10.1s

405: learn: 0.0001102 total: 6.89s remaining: 10.1s

406: learn: 0.0001102 total: 6.9s remaining: 10.1s

407: learn: 0.0001102 total: 6.92s remaining: 10s

408: learn: 0.0001102 total: 6.94s remaining: 10s

409: learn: 0.0001102 total: 6.96s remaining: 10s

410: learn: 0.0001102 total: 6.97s remaining: 9.99s

411: learn: 0.0001102 total: 6.99s remaining: 9.97s

412: learn: 0.0001102 total: 7.01s remaining: 9.96s

413: learn: 0.0001101 total: 7.03s remaining: 9.95s

414: learn: 0.0001101 total: 7.04s remaining: 9.93s

415: learn: 0.0001101 total: 7.06s remaining: 9.91s

416: learn: 0.0001101 total: 7.08s remaining: 9.89s

417: learn: 0.0001101 total: 7.09s remaining: 9.87s

418: learn: 0.0001101 total: 7.11s remaining: 9.85s

419: learn: 0.0001101 total: 7.12s remaining: 9.83s

420: learn: 0.0001101 total: 7.13s remaining: 9.81s

421: learn: 0.0001101 total: 7.15s remaining: 9.79s

422: learn: 0.0001101 total: 7.17s remaining: 9.77s

423: learn: 0.0001101 total: 7.18s remaining: 9.75s

424: learn: 0.0001101 total: 7.2s remaining: 9.74s

425: learn: 0.0001101 total: 7.21s remaining: 9.72s

426: learn: 0.0001101 total: 7.23s remaining: 9.7s

427: learn: 0.0001101 total: 7.24s remaining: 9.68s

428: learn: 0.0001101 total: 7.26s remaining: 9.66s

429: learn: 0.0001101 total: 7.27s remaining: 9.64s

430: learn: 0.0001101 total: 7.28s remaining: 9.62s

431: learn: 0.0001101 total: 7.3s remaining: 9.6s

432: learn: 0.0001101 total: 7.32s remaining: 9.58s

433: learn: 0.0001101 total: 7.33s remaining: 9.56s

434: learn: 0.0001101 total: 7.34s remaining: 9.54s

435: learn: 0.0001101 total: 7.36s remaining: 9.52s

436: learn: 0.0001101 total: 7.38s remaining: 9.5s

437: learn: 0.0001101 total: 7.39s remaining: 9.48s

438: learn: 0.0001101 total: 7.41s remaining: 9.46s

439: learn: 0.0001101 total: 7.42s remaining: 9.44s

440: learn: 0.0001101 total: 7.44s remaining: 9.43s

441: learn: 0.0001101 total: 7.45s remaining: 9.41s

442: learn: 0.0001101 total: 7.47s remaining: 9.39s

443: learn: 0.0001101 total: 7.48s remaining: 9.37s

444: learn: 0.0001101 total: 7.49s remaining: 9.35s

445: learn: 0.0001101 total: 7.51s remaining: 9.33s

446: learn: 0.0001101 total: 7.52s remaining: 9.31s

447: learn: 0.0001101 total: 7.54s remaining: 9.29s

448: learn: 0.0001101 total: 7.55s remaining: 9.27s

449: learn: 0.0001101 total: 7.57s remaining: 9.25s

450: learn: 0.0001101 total: 7.58s remaining: 9.23s

451: learn: 0.0001101 total: 7.59s remaining: 9.21s

452: learn: 0.0001101 total: 7.61s remaining: 9.19s

453: learn: 0.0001101 total: 7.62s remaining: 9.17s

454: learn: 0.0001101 total: 7.64s remaining: 9.15s

455: learn: 0.0001101 total: 7.65s remaining: 9.13s

456: learn: 0.0001101 total: 7.66s remaining: 9.11s

457: learn: 0.0001101 total: 7.68s remaining: 9.09s

458: learn: 0.0001101 total: 7.69s remaining: 9.06s

459: learn: 0.0001101 total: 7.7s remaining: 9.04s

460: learn: 0.0001101 total: 7.72s remaining: 9.03s

461: learn: 0.0001101 total: 7.74s remaining: 9.01s

462: learn: 0.0001101 total: 7.75s remaining: 8.99s

463: learn: 0.0001101 total: 7.77s remaining: 8.97s

464: learn: 0.0001101 total: 7.78s remaining: 8.95s

465: learn: 0.0001101 total: 7.8s remaining: 8.94s

466: learn: 0.0001101 total: 7.81s remaining: 8.92s

467: learn: 0.0001101 total: 7.83s remaining: 8.9s

468: learn: 0.0001101 total: 7.84s remaining: 8.88s

469: learn: 0.0001101 total: 7.85s remaining: 8.85s

470: learn: 0.0001101 total: 7.87s remaining: 8.84s

471: learn: 0.0001101 total: 7.88s remaining: 8.81s

472: learn: 0.0001101 total: 7.89s remaining: 8.8s

473: learn: 0.0001101 total: 7.91s remaining: 8.78s

474: learn: 0.0001101 total: 7.92s remaining: 8.76s

475: learn: 0.0001101 total: 7.94s remaining: 8.74s

476: learn: 0.0001101 total: 7.96s remaining: 8.72s

477: learn: 0.0001101 total: 7.97s remaining: 8.7s

478: learn: 0.0001101 total: 7.98s remaining: 8.68s

479: learn: 0.0001101 total: 8s remaining: 8.66s

480: learn: 0.0001101 total: 8.01s remaining: 8.65s

481: learn: 0.0001101 total: 8.03s remaining: 8.63s

482: learn: 0.0001101 total: 8.04s remaining: 8.61s

483: learn: 0.0001101 total: 8.05s remaining: 8.59s

484: learn: 0.0001101 total: 8.07s remaining: 8.57s

485: learn: 0.0001101 total: 8.09s remaining: 8.55s

486: learn: 0.0001101 total: 8.1s remaining: 8.54s

487: learn: 0.0001101 total: 8.12s remaining: 8.52s

488: learn: 0.0001101 total: 8.13s remaining: 8.5s

489: learn: 0.0001101 total: 8.15s remaining: 8.48s

490: learn: 0.0001101 total: 8.16s remaining: 8.46s

491: learn: 0.0001101 total: 8.18s remaining: 8.45s

492: learn: 0.0001101 total: 8.19s remaining: 8.43s

493: learn: 0.0001101 total: 8.21s remaining: 8.41s

494: learn: 0.0001101 total: 8.22s remaining: 8.39s

495: learn: 0.0001101 total: 8.24s remaining: 8.37s

496: learn: 0.0001101 total: 8.25s remaining: 8.35s

497: learn: 0.0001101 total: 8.27s remaining: 8.33s

498: learn: 0.0001101 total: 8.28s remaining: 8.31s

499: learn: 0.0001101 total: 8.29s remaining: 8.29s

500: learn: 0.0001101 total: 8.31s remaining: 8.28s

501: learn: 0.0001101 total: 8.32s remaining: 8.26s

502: learn: 0.0001101 total: 8.34s remaining: 8.24s

503: learn: 0.0001101 total: 8.36s remaining: 8.22s

504: learn: 0.0001101 total: 8.37s remaining: 8.2s

505: learn: 0.0001101 total: 8.38s remaining: 8.19s

506: learn: 0.0001101 total: 8.4s remaining: 8.17s

507: learn: 0.0001101 total: 8.41s remaining: 8.15s

508: learn: 0.0001101 total: 8.43s remaining: 8.13s

509: learn: 0.0001101 total: 8.44s remaining: 8.11s

510: learn: 0.0001101 total: 8.45s remaining: 8.09s

511: learn: 0.0001101 total: 8.47s remaining: 8.07s

512: learn: 0.0001101 total: 8.48s remaining: 8.05s

513: learn: 0.0001101 total: 8.5s remaining: 8.04s

514: learn: 0.0001101 total: 8.51s remaining: 8.02s

515: learn: 0.0001101 total: 8.53s remaining: 8s

516: learn: 0.0001101 total: 8.55s remaining: 7.99s

517: learn: 0.0001101 total: 8.56s remaining: 7.97s

518: learn: 0.0001101 total: 8.58s remaining: 7.95s

519: learn: 0.0001101 total: 8.59s remaining: 7.93s

520: learn: 0.0001101 total: 8.61s remaining: 7.91s

521: learn: 0.0001101 total: 8.62s remaining: 7.89s

522: learn: 0.0001101 total: 8.64s remaining: 7.88s

523: learn: 0.0001101 total: 8.65s remaining: 7.86s

524: learn: 0.0001101 total: 8.67s remaining: 7.84s

525: learn: 0.0001100 total: 8.68s remaining: 7.82s

526: learn: 0.0001100 total: 8.7s remaining: 7.81s

527: learn: 0.0001100 total: 8.71s remaining: 7.79s

528: learn: 0.0001100 total: 8.73s remaining: 7.77s

529: learn: 0.0001100 total: 8.74s remaining: 7.75s

530: learn: 0.0001100 total: 8.76s remaining: 7.74s

531: learn: 0.0001100 total: 8.77s remaining: 7.72s

532: learn: 0.0001100 total: 8.79s remaining: 7.7s

533: learn: 0.0001100 total: 8.8s remaining: 7.68s

534: learn: 0.0001100 total: 8.82s remaining: 7.66s

535: learn: 0.0001100 total: 8.83s remaining: 7.65s

536: learn: 0.0001100 total: 8.85s remaining: 7.63s

537: learn: 0.0001100 total: 8.86s remaining: 7.61s

538: learn: 0.0001100 total: 8.88s remaining: 7.59s

539: learn: 0.0001100 total: 8.89s remaining: 7.58s

540: learn: 0.0001100 total: 8.91s remaining: 7.56s

541: learn: 0.0001100 total: 8.92s remaining: 7.54s

542: learn: 0.0001100 total: 8.94s remaining: 7.52s

543: learn: 0.0001100 total: 8.95s remaining: 7.5s

544: learn: 0.0001100 total: 8.96s remaining: 7.48s

545: learn: 0.0001100 total: 8.98s remaining: 7.47s

546: learn: 0.0001100 total: 8.99s remaining: 7.45s

547: learn: 0.0001100 total: 9.01s remaining: 7.43s

548: learn: 0.0001100 total: 9.02s remaining: 7.41s

549: learn: 0.0001100 total: 9.04s remaining: 7.39s

550: learn: 0.0001100 total: 9.05s remaining: 7.38s

551: learn: 0.0001100 total: 9.07s remaining: 7.36s

552: learn: 0.0001100 total: 9.08s remaining: 7.34s

553: learn: 0.0001100 total: 9.1s remaining: 7.32s

554: learn: 0.0001100 total: 9.11s remaining: 7.3s

555: learn: 0.0001100 total: 9.13s remaining: 7.29s

556: learn: 0.0001100 total: 9.14s remaining: 7.27s

557: learn: 0.0001100 total: 9.15s remaining: 7.25s

558: learn: 0.0001100 total: 9.18s remaining: 7.24s

559: learn: 0.0001100 total: 9.19s remaining: 7.22s

560: learn: 0.0001100 total: 9.2s remaining: 7.2s

561: learn: 0.0001100 total: 9.22s remaining: 7.18s

562: learn: 0.0001100 total: 9.23s remaining: 7.17s

563: learn: 0.0001100 total: 9.25s remaining: 7.15s

564: learn: 0.0001100 total: 9.27s remaining: 7.13s

565: learn: 0.0001100 total: 9.28s remaining: 7.12s

566: learn: 0.0001100 total: 9.29s remaining: 7.1s

567: learn: 0.0001100 total: 9.31s remaining: 7.08s

568: learn: 0.0001100 total: 9.32s remaining: 7.06s

569: learn: 0.0001100 total: 9.34s remaining: 7.04s

570: learn: 0.0001100 total: 9.35s remaining: 7.03s

571: learn: 0.0001100 total: 9.37s remaining: 7.01s

572: learn: 0.0001100 total: 9.38s remaining: 6.99s

573: learn: 0.0001100 total: 9.4s remaining: 6.98s

574: learn: 0.0001100 total: 9.42s remaining: 6.96s

575: learn: 0.0001100 total: 9.43s remaining: 6.94s

576: learn: 0.0001100 total: 9.45s remaining: 6.92s

577: learn: 0.0001100 total: 9.46s remaining: 6.91s

578: learn: 0.0001100 total: 9.48s remaining: 6.89s

579: learn: 0.0001100 total: 9.5s remaining: 6.88s

580: learn: 0.0001100 total: 9.52s remaining: 6.86s

581: learn: 0.0001100 total: 9.53s remaining: 6.84s

582: learn: 0.0001100 total: 9.54s remaining: 6.83s

583: learn: 0.0001100 total: 9.56s remaining: 6.81s

584: learn: 0.0001100 total: 9.57s remaining: 6.79s

585: learn: 0.0001100 total: 9.59s remaining: 6.78s

586: learn: 0.0001100 total: 9.61s remaining: 6.76s

587: learn: 0.0001100 total: 9.62s remaining: 6.74s

588: learn: 0.0001100 total: 9.63s remaining: 6.72s

589: learn: 0.0001100 total: 9.65s remaining: 6.71s

590: learn: 0.0001100 total: 9.66s remaining: 6.69s

591: learn: 0.0001100 total: 9.68s remaining: 6.67s

592: learn: 0.0001100 total: 9.69s remaining: 6.65s

593: learn: 0.0001100 total: 9.71s remaining: 6.63s

594: learn: 0.0001100 total: 9.72s remaining: 6.62s

595: learn: 0.0001100 total: 9.74s remaining: 6.6s

596: learn: 0.0001100 total: 9.75s remaining: 6.58s

597: learn: 0.0001100 total: 9.76s remaining: 6.56s

598: learn: 0.0001100 total: 9.78s remaining: 6.55s

599: learn: 0.0001100 total: 9.81s remaining: 6.54s

600: learn: 0.0001100 total: 9.83s remaining: 6.52s

601: learn: 0.0001100 total: 9.85s remaining: 6.51s

602: learn: 0.0001100 total: 9.88s remaining: 6.5s

603: learn: 0.0001100 total: 9.9s remaining: 6.49s

604: learn: 0.0001100 total: 9.92s remaining: 6.48s

605: learn: 0.0001100 total: 9.94s remaining: 6.46s

606: learn: 0.0001100 total: 9.97s remaining: 6.46s

607: learn: 0.0001100 total: 10s remaining: 6.45s

608: learn: 0.0001100 total: 10s remaining: 6.43s

609: learn: 0.0001100 total: 10s remaining: 6.42s

610: learn: 0.0001100 total: 10.1s remaining: 6.41s

611: learn: 0.0001100 total: 10.1s remaining: 6.39s

612: learn: 0.0001100 total: 10.1s remaining: 6.38s

613: learn: 0.0001100 total: 10.1s remaining: 6.37s

614: learn: 0.0001100 total: 10.2s remaining: 6.36s

615: learn: 0.0001100 total: 10.2s remaining: 6.34s

616: learn: 0.0001100 total: 10.2s remaining: 6.33s

617: learn: 0.0001100 total: 10.2s remaining: 6.32s

618: learn: 0.0001100 total: 10.3s remaining: 6.31s

619: learn: 0.0001100 total: 10.3s remaining: 6.29s

620: learn: 0.0001100 total: 10.3s remaining: 6.28s

621: learn: 0.0001100 total: 10.3s remaining: 6.26s

622: learn: 0.0001100 total: 10.3s remaining: 6.25s

623: learn: 0.0001100 total: 10.3s remaining: 6.23s

624: learn: 0.0001100 total: 10.4s remaining: 6.22s

625: learn: 0.0001100 total: 10.4s remaining: 6.2s

626: learn: 0.0001100 total: 10.4s remaining: 6.19s

627: learn: 0.0001100 total: 10.4s remaining: 6.17s

628: learn: 0.0001100 total: 10.4s remaining: 6.16s

629: learn: 0.0001100 total: 10.5s remaining: 6.14s

630: learn: 0.0001100 total: 10.5s remaining: 6.13s

631: learn: 0.0001100 total: 10.5s remaining: 6.12s

632: learn: 0.0001100 total: 10.5s remaining: 6.1s

633: learn: 0.0001100 total: 10.5s remaining: 6.08s

634: learn: 0.0001099 total: 10.6s remaining: 6.07s

635: learn: 0.0001099 total: 10.6s remaining: 6.06s

636: learn: 0.0001099 total: 10.6s remaining: 6.04s

637: learn: 0.0001099 total: 10.6s remaining: 6.03s

638: learn: 0.0001099 total: 10.6s remaining: 6.01s

639: learn: 0.0001099 total: 10.7s remaining: 6s

640: learn: 0.0001099 total: 10.7s remaining: 5.98s

641: learn: 0.0001099 total: 10.7s remaining: 5.97s

642: learn: 0.0001099 total: 10.7s remaining: 5.95s

643: learn: 0.0001099 total: 10.7s remaining: 5.94s

644: learn: 0.0001099 total: 10.8s remaining: 5.92s

645: learn: 0.0001099 total: 10.8s remaining: 5.91s

646: learn: 0.0001099 total: 10.8s remaining: 5.89s

647: learn: 0.0001099 total: 10.8s remaining: 5.88s

648: learn: 0.0001099 total: 10.8s remaining: 5.87s

649: learn: 0.0001099 total: 10.9s remaining: 5.85s

650: learn: 0.0001099 total: 10.9s remaining: 5.84s

651: learn: 0.0001099 total: 10.9s remaining: 5.83s

652: learn: 0.0001099 total: 10.9s remaining: 5.81s

653: learn: 0.0001099 total: 11s remaining: 5.79s

654: learn: 0.0001099 total: 11s remaining: 5.78s

655: learn: 0.0001099 total: 11s remaining: 5.76s

656: learn: 0.0001099 total: 11s remaining: 5.75s

657: learn: 0.0001099 total: 11s remaining: 5.73s

658: learn: 0.0001099 total: 11s remaining: 5.71s

659: learn: 0.0001099 total: 11s remaining: 5.69s

660: learn: 0.0001099 total: 11.1s remaining: 5.67s

661: learn: 0.0001099 total: 11.1s remaining: 5.66s

662: learn: 0.0001099 total: 11.1s remaining: 5.64s

663: learn: 0.0001099 total: 11.1s remaining: 5.62s

664: learn: 0.0001099 total: 11.1s remaining: 5.6s

665: learn: 0.0001099 total: 11.1s remaining: 5.58s

666: learn: 0.0001099 total: 11.2s remaining: 5.57s

667: learn: 0.0001099 total: 11.2s remaining: 5.55s

668: learn: 0.0001099 total: 11.2s remaining: 5.53s

669: learn: 0.0001099 total: 11.2s remaining: 5.51s

670: learn: 0.0001099 total: 11.2s remaining: 5.5s

671: learn: 0.0001099 total: 11.2s remaining: 5.48s

672: learn: 0.0001099 total: 11.2s remaining: 5.46s

673: learn: 0.0001099 total: 11.3s remaining: 5.44s

674: learn: 0.0001099 total: 11.3s remaining: 5.42s

675: learn: 0.0001099 total: 11.3s remaining: 5.41s

676: learn: 0.0001099 total: 11.3s remaining: 5.39s

677: learn: 0.0001099 total: 11.3s remaining: 5.38s

678: learn: 0.0001099 total: 11.3s remaining: 5.36s

679: learn: 0.0001099 total: 11.3s remaining: 5.34s

680: learn: 0.0001099 total: 11.4s remaining: 5.32s

681: learn: 0.0001099 total: 11.4s remaining: 5.3s

682: learn: 0.0001099 total: 11.4s remaining: 5.29s

683: learn: 0.0001099 total: 11.4s remaining: 5.27s

684: learn: 0.0001099 total: 11.4s remaining: 5.25s

685: learn: 0.0001099 total: 11.4s remaining: 5.23s

686: learn: 0.0001099 total: 11.4s remaining: 5.21s

687: learn: 0.0001099 total: 11.5s remaining: 5.2s

688: learn: 0.0001099 total: 11.5s remaining: 5.18s

689: learn: 0.0001099 total: 11.5s remaining: 5.16s

690: learn: 0.0001099 total: 11.5s remaining: 5.14s

691: learn: 0.0001099 total: 11.5s remaining: 5.13s

692: learn: 0.0001099 total: 11.5s remaining: 5.11s

693: learn: 0.0001099 total: 11.5s remaining: 5.09s

694: learn: 0.0001099 total: 11.6s remaining: 5.07s

695: learn: 0.0001099 total: 11.6s remaining: 5.05s

696: learn: 0.0001099 total: 11.6s remaining: 5.04s

697: learn: 0.0001099 total: 11.6s remaining: 5.02s

698: learn: 0.0001099 total: 11.6s remaining: 5s

699: learn: 0.0001099 total: 11.6s remaining: 4.98s

700: learn: 0.0001099 total: 11.6s remaining: 4.97s

701: learn: 0.0001099 total: 11.7s remaining: 4.95s

702: learn: 0.0001099 total: 11.7s remaining: 4.93s

703: learn: 0.0001099 total: 11.7s remaining: 4.91s

704: learn: 0.0001099 total: 11.7s remaining: 4.9s

705: learn: 0.0001099 total: 11.7s remaining: 4.88s

706: learn: 0.0001099 total: 11.7s remaining: 4.86s

707: learn: 0.0001099 total: 11.7s remaining: 4.84s

708: learn: 0.0001099 total: 11.8s remaining: 4.83s

709: learn: 0.0001099 total: 11.8s remaining: 4.81s

710: learn: 0.0001099 total: 11.8s remaining: 4.79s

711: learn: 0.0001099 total: 11.8s remaining: 4.77s

712: learn: 0.0001099 total: 11.8s remaining: 4.76s

713: learn: 0.0001099 total: 11.8s remaining: 4.74s

714: learn: 0.0001099 total: 11.9s remaining: 4.72s

715: learn: 0.0001099 total: 11.9s remaining: 4.71s

716: learn: 0.0001099 total: 11.9s remaining: 4.69s

717: learn: 0.0001099 total: 11.9s remaining: 4.67s

718: learn: 0.0001099 total: 11.9s remaining: 4.65s

719: learn: 0.0001099 total: 11.9s remaining: 4.64s

720: learn: 0.0001099 total: 11.9s remaining: 4.62s

721: learn: 0.0001099 total: 12s remaining: 4.6s

722: learn: 0.0001099 total: 12s remaining: 4.58s

723: learn: 0.0001099 total: 12s remaining: 4.57s

724: learn: 0.0001099 total: 12s remaining: 4.55s

725: learn: 0.0001099 total: 12s remaining: 4.54s

726: learn: 0.0001099 total: 12s remaining: 4.52s

727: learn: 0.0001099 total: 12s remaining: 4.5s

728: learn: 0.0001099 total: 12.1s remaining: 4.48s

729: learn: 0.0001099 total: 12.1s remaining: 4.47s

730: learn: 0.0001099 total: 12.1s remaining: 4.45s

731: learn: 0.0001099 total: 12.1s remaining: 4.43s

732: learn: 0.0001099 total: 12.1s remaining: 4.41s

733: learn: 0.0001099 total: 12.1s remaining: 4.4s

734: learn: 0.0001099 total: 12.2s remaining: 4.38s

735: learn: 0.0001099 total: 12.2s remaining: 4.36s

736: learn: 0.0001099 total: 12.2s remaining: 4.35s

737: learn: 0.0001099 total: 12.2s remaining: 4.33s

738: learn: 0.0001099 total: 12.2s remaining: 4.31s

739: learn: 0.0001099 total: 12.2s remaining: 4.29s

740: learn: 0.0001099 total: 12.2s remaining: 4.28s

741: learn: 0.0001099 total: 12.3s remaining: 4.26s

742: learn: 0.0001099 total: 12.3s remaining: 4.24s

743: learn: 0.0001099 total: 12.3s remaining: 4.23s

744: learn: 0.0001099 total: 12.3s remaining: 4.21s

745: learn: 0.0001099 total: 12.3s remaining: 4.19s

746: learn: 0.0001099 total: 12.3s remaining: 4.17s

747: learn: 0.0001099 total: 12.3s remaining: 4.16s

748: learn: 0.0001099 total: 12.4s remaining: 4.14s

749: learn: 0.0001099 total: 12.4s remaining: 4.12s

750: learn: 0.0001099 total: 12.4s remaining: 4.11s

751: learn: 0.0001099 total: 12.4s remaining: 4.09s

752: learn: 0.0001099 total: 12.4s remaining: 4.07s

753: learn: 0.0001099 total: 12.4s remaining: 4.05s

754: learn: 0.0001099 total: 12.4s remaining: 4.04s

755: learn: 0.0001099 total: 12.5s remaining: 4.02s

756: learn: 0.0001099 total: 12.5s remaining: 4s

757: learn: 0.0001099 total: 12.5s remaining: 3.99s

758: learn: 0.0001099 total: 12.5s remaining: 3.97s

759: learn: 0.0001099 total: 12.5s remaining: 3.96s

760: learn: 0.0001099 total: 12.5s remaining: 3.94s

761: learn: 0.0001099 total: 12.6s remaining: 3.92s

762: learn: 0.0001099 total: 12.6s remaining: 3.9s

763: learn: 0.0001098 total: 12.6s remaining: 3.89s

764: learn: 0.0001098 total: 12.6s remaining: 3.87s

765: learn: 0.0001098 total: 12.6s remaining: 3.85s

766: learn: 0.0001098 total: 12.6s remaining: 3.84s

767: learn: 0.0001098 total: 12.6s remaining: 3.82s

768: learn: 0.0001098 total: 12.7s remaining: 3.8s

769: learn: 0.0001098 total: 12.7s remaining: 3.79s

770: learn: 0.0001098 total: 12.7s remaining: 3.77s

771: learn: 0.0001098 total: 12.7s remaining: 3.75s

772: learn: 0.0001098 total: 12.7s remaining: 3.73s

773: learn: 0.0001098 total: 12.7s remaining: 3.72s

774: learn: 0.0001098 total: 12.7s remaining: 3.7s

775: learn: 0.0001098 total: 12.8s remaining: 3.68s

776: learn: 0.0001098 total: 12.8s remaining: 3.67s

777: learn: 0.0001098 total: 12.8s remaining: 3.65s

778: learn: 0.0001098 total: 12.8s remaining: 3.63s

779: learn: 0.0001098 total: 12.8s remaining: 3.62s

780: learn: 0.0001098 total: 12.8s remaining: 3.6s

781: learn: 0.0001098 total: 12.9s remaining: 3.58s

782: learn: 0.0001098 total: 12.9s remaining: 3.56s

783: learn: 0.0001098 total: 12.9s remaining: 3.55s

784: learn: 0.0001098 total: 12.9s remaining: 3.53s

785: learn: 0.0001098 total: 12.9s remaining: 3.51s

786: learn: 0.0001098 total: 12.9s remaining: 3.5s

787: learn: 0.0001098 total: 12.9s remaining: 3.48s

788: learn: 0.0001098 total: 13s remaining: 3.46s

789: learn: 0.0001098 total: 13s remaining: 3.45s

790: learn: 0.0001098 total: 13s remaining: 3.43s

791: learn: 0.0001098 total: 13s remaining: 3.41s

792: learn: 0.0001098 total: 13s remaining: 3.4s

793: learn: 0.0001098 total: 13s remaining: 3.38s

794: learn: 0.0001098 total: 13s remaining: 3.36s

795: learn: 0.0001098 total: 13.1s remaining: 3.35s

796: learn: 0.0001098 total: 13.1s remaining: 3.33s

797: learn: 0.0001098 total: 13.1s remaining: 3.31s

798: learn: 0.0001098 total: 13.1s remaining: 3.3s

799: learn: 0.0001098 total: 13.1s remaining: 3.28s

800: learn: 0.0001098 total: 13.1s remaining: 3.26s

801: learn: 0.0001098 total: 13.1s remaining: 3.25s

802: learn: 0.0001098 total: 13.2s remaining: 3.23s

803: learn: 0.0001098 total: 13.2s remaining: 3.21s

804: learn: 0.0001098 total: 13.2s remaining: 3.2s

805: learn: 0.0001098 total: 13.2s remaining: 3.18s

806: learn: 0.0001098 total: 13.2s remaining: 3.16s

807: learn: 0.0001098 total: 13.2s remaining: 3.15s

808: learn: 0.0001098 total: 13.3s remaining: 3.13s

809: learn: 0.0001098 total: 13.3s remaining: 3.11s

810: learn: 0.0001098 total: 13.3s remaining: 3.1s

811: learn: 0.0001098 total: 13.3s remaining: 3.08s

812: learn: 0.0001098 total: 13.3s remaining: 3.06s

813: learn: 0.0001098 total: 13.3s remaining: 3.04s

814: learn: 0.0001098 total: 13.3s remaining: 3.03s

815: learn: 0.0001098 total: 13.4s remaining: 3.01s

816: learn: 0.0001098 total: 13.4s remaining: 2.99s

817: learn: 0.0001098 total: 13.4s remaining: 2.98s

818: learn: 0.0001098 total: 13.4s remaining: 2.96s

819: learn: 0.0001098 total: 13.4s remaining: 2.94s

820: learn: 0.0001098 total: 13.4s remaining: 2.93s

821: learn: 0.0001098 total: 13.4s remaining: 2.91s

822: learn: 0.0001098 total: 13.5s remaining: 2.89s

823: learn: 0.0001098 total: 13.5s remaining: 2.88s

824: learn: 0.0001098 total: 13.5s remaining: 2.86s

825: learn: 0.0001098 total: 13.5s remaining: 2.84s

826: learn: 0.0001098 total: 13.5s remaining: 2.83s

827: learn: 0.0001098 total: 13.5s remaining: 2.81s

828: learn: 0.0001098 total: 13.5s remaining: 2.79s

829: learn: 0.0001098 total: 13.5s remaining: 2.77s

830: learn: 0.0001098 total: 13.6s remaining: 2.76s

831: learn: 0.0001098 total: 13.6s remaining: 2.74s

832: learn: 0.0001098 total: 13.6s remaining: 2.73s

833: learn: 0.0001098 total: 13.6s remaining: 2.71s

834: learn: 0.0001098 total: 13.6s remaining: 2.69s

835: learn: 0.0001098 total: 13.6s remaining: 2.68s

836: learn: 0.0001098 total: 13.7s remaining: 2.66s

837: learn: 0.0001098 total: 13.7s remaining: 2.64s

838: learn: 0.0001098 total: 13.7s remaining: 2.63s

839: learn: 0.0001098 total: 13.7s remaining: 2.61s

840: learn: 0.0001097 total: 13.7s remaining: 2.59s

841: learn: 0.0001097 total: 13.7s remaining: 2.58s

842: learn: 0.0001097 total: 13.7s remaining: 2.56s

843: learn: 0.0001097 total: 13.8s remaining: 2.54s

844: learn: 0.0001097 total: 13.8s remaining: 2.52s

845: learn: 0.0001097 total: 13.8s remaining: 2.51s

846: learn: 0.0001097 total: 13.8s remaining: 2.49s

847: learn: 0.0001097 total: 13.8s remaining: 2.48s

848: learn: 0.0001097 total: 13.8s remaining: 2.46s

849: learn: 0.0001097 total: 13.8s remaining: 2.44s

850: learn: 0.0001097 total: 13.9s remaining: 2.43s

851: learn: 0.0001097 total: 13.9s remaining: 2.41s

852: learn: 0.0001097 total: 13.9s remaining: 2.39s

853: learn: 0.0001097 total: 13.9s remaining: 2.38s

854: learn: 0.0001097 total: 13.9s remaining: 2.36s

855: learn: 0.0001097 total: 13.9s remaining: 2.35s

856: learn: 0.0001097 total: 14s remaining: 2.33s

857: learn: 0.0001097 total: 14s remaining: 2.31s

858: learn: 0.0001097 total: 14s remaining: 2.3s

859: learn: 0.0001097 total: 14s remaining: 2.28s

860: learn: 0.0001097 total: 14s remaining: 2.26s

861: learn: 0.0001097 total: 14s remaining: 2.25s

862: learn: 0.0001097 total: 14.1s remaining: 2.23s

863: learn: 0.0001097 total: 14.1s remaining: 2.21s

864: learn: 0.0001097 total: 14.1s remaining: 2.2s

865: learn: 0.0001097 total: 14.1s remaining: 2.18s

866: learn: 0.0001097 total: 14.1s remaining: 2.17s

867: learn: 0.0001097 total: 14.1s remaining: 2.15s

868: learn: 0.0001097 total: 14.1s remaining: 2.13s

869: learn: 0.0001097 total: 14.2s remaining: 2.12s

870: learn: 0.0001097 total: 14.2s remaining: 2.1s

871: learn: 0.0001097 total: 14.2s remaining: 2.08s

872: learn: 0.0001097 total: 14.2s remaining: 2.07s

873: learn: 0.0001097 total: 14.2s remaining: 2.05s

874: learn: 0.0001097 total: 14.2s remaining: 2.03s

875: learn: 0.0001097 total: 14.3s remaining: 2.02s

876: learn: 0.0001097 total: 14.3s remaining: 2s

877: learn: 0.0001097 total: 14.3s remaining: 1.99s

878: learn: 0.0001097 total: 14.3s remaining: 1.97s

879: learn: 0.0001097 total: 14.3s remaining: 1.95s

880: learn: 0.0001097 total: 14.3s remaining: 1.94s

881: learn: 0.0001097 total: 14.3s remaining: 1.92s

882: learn: 0.0001097 total: 14.4s remaining: 1.9s

883: learn: 0.0001097 total: 14.4s remaining: 1.89s

884: learn: 0.0001097 total: 14.4s remaining: 1.87s

885: learn: 0.0001097 total: 14.4s remaining: 1.85s

886: learn: 0.0001097 total: 14.4s remaining: 1.84s

887: learn: 0.0001097 total: 14.4s remaining: 1.82s

888: learn: 0.0001097 total: 14.4s remaining: 1.8s

889: learn: 0.0001097 total: 14.5s remaining: 1.79s

890: learn: 0.0001097 total: 14.5s remaining: 1.77s

891: learn: 0.0001097 total: 14.5s remaining: 1.75s

892: learn: 0.0001097 total: 14.5s remaining: 1.74s

893: learn: 0.0001097 total: 14.5s remaining: 1.72s

894: learn: 0.0001097 total: 14.5s remaining: 1.71s

895: learn: 0.0001097 total: 14.6s remaining: 1.69s

896: learn: 0.0001097 total: 14.6s remaining: 1.67s

897: learn: 0.0001097 total: 14.6s remaining: 1.66s

898: learn: 0.0001097 total: 14.6s remaining: 1.64s

899: learn: 0.0001097 total: 14.6s remaining: 1.62s

900: learn: 0.0001097 total: 14.6s remaining: 1.61s

901: learn: 0.0001097 total: 14.6s remaining: 1.59s

902: learn: 0.0001097 total: 14.7s remaining: 1.57s

903: learn: 0.0001097 total: 14.7s remaining: 1.56s

904: learn: 0.0001097 total: 14.7s remaining: 1.54s

905: learn: 0.0001097 total: 14.7s remaining: 1.52s

906: learn: 0.0001097 total: 14.7s remaining: 1.51s

907: learn: 0.0001097 total: 14.7s remaining: 1.49s

908: learn: 0.0001097 total: 14.7s remaining: 1.48s

909: learn: 0.0001097 total: 14.8s remaining: 1.46s

910: learn: 0.0001097 total: 14.8s remaining: 1.44s

911: learn: 0.0001097 total: 14.8s remaining: 1.43s

912: learn: 0.0001097 total: 14.8s remaining: 1.41s

913: learn: 0.0001097 total: 14.8s remaining: 1.39s

914: learn: 0.0001097 total: 14.8s remaining: 1.38s

915: learn: 0.0001097 total: 14.9s remaining: 1.36s

916: learn: 0.0001097 total: 14.9s remaining: 1.34s

917: learn: 0.0001097 total: 14.9s remaining: 1.33s

918: learn: 0.0001097 total: 14.9s remaining: 1.31s

919: learn: 0.0001097 total: 14.9s remaining: 1.3s

920: learn: 0.0001097 total: 14.9s remaining: 1.28s

921: learn: 0.0001097 total: 14.9s remaining: 1.26s

922: learn: 0.0001097 total: 15s remaining: 1.25s

923: learn: 0.0001097 total: 15s remaining: 1.23s

924: learn: 0.0001097 total: 15s remaining: 1.21s

925: learn: 0.0001097 total: 15s remaining: 1.2s

926: learn: 0.0001097 total: 15s remaining: 1.18s

927: learn: 0.0001096 total: 15s remaining: 1.17s

928: learn: 0.0001096 total: 15s remaining: 1.15s

929: learn: 0.0001096 total: 15.1s remaining: 1.13s

930: learn: 0.0001096 total: 15.1s remaining: 1.12s

931: learn: 0.0001096 total: 15.1s remaining: 1.1s

932: learn: 0.0001096 total: 15.1s remaining: 1.08s

933: learn: 0.0001096 total: 15.1s remaining: 1.07s

934: learn: 0.0001096 total: 15.1s remaining: 1.05s

935: learn: 0.0001096 total: 15.1s remaining: 1.03s

936: learn: 0.0001096 total: 15.2s remaining: 1.02s

937: learn: 0.0001096 total: 15.2s remaining: 1s

938: learn: 0.0001096 total: 15.2s remaining: 987ms

939: learn: 0.0001096 total: 15.2s remaining: 972ms

940: learn: 0.0001096 total: 15.2s remaining: 956ms

941: learn: 0.0001096 total: 15.3s remaining: 940ms

942: learn: 0.0001096 total: 15.3s remaining: 925ms

943: learn: 0.0001096 total: 15.3s remaining: 909ms

944: learn: 0.0001096 total: 15.3s remaining: 893ms

945: learn: 0.0001096 total: 15.4s remaining: 877ms

946: learn: 0.0001096 total: 15.4s remaining: 861ms

947: learn: 0.0001096 total: 15.4s remaining: 845ms

948: learn: 0.0001096 total: 15.4s remaining: 828ms

949: learn: 0.0001096 total: 15.4s remaining: 812ms

950: learn: 0.0001096 total: 15.4s remaining: 796ms

951: learn: 0.0001096 total: 15.5s remaining: 779ms

952: learn: 0.0001096 total: 15.5s remaining: 763ms

953: learn: 0.0001096 total: 15.5s remaining: 747ms

954: learn: 0.0001096 total: 15.5s remaining: 731ms

955: learn: 0.0001096 total: 15.5s remaining: 714ms

956: learn: 0.0001096 total: 15.5s remaining: 698ms

957: learn: 0.0001096 total: 15.5s remaining: 682ms

958: learn: 0.0001096 total: 15.6s remaining: 665ms

959: learn: 0.0001096 total: 15.6s remaining: 649ms

960: learn: 0.0001096 total: 15.6s remaining: 633ms

961: learn: 0.0001096 total: 15.6s remaining: 617ms

962: learn: 0.0001096 total: 15.6s remaining: 600ms

963: learn: 0.0001096 total: 15.6s remaining: 584ms

964: learn: 0.0001096 total: 15.7s remaining: 568ms

965: learn: 0.0001096 total: 15.7s remaining: 551ms

966: learn: 0.0001096 total: 15.7s remaining: 535ms

967: learn: 0.0001096 total: 15.7s remaining: 519ms

968: learn: 0.0001096 total: 15.7s remaining: 503ms

969: learn: 0.0001096 total: 15.7s remaining: 486ms

970: learn: 0.0001096 total: 15.7s remaining: 470ms

971: learn: 0.0001096 total: 15.8s remaining: 454ms

972: learn: 0.0001096 total: 15.8s remaining: 438ms

973: learn: 0.0001096 total: 15.8s remaining: 421ms

974: learn: 0.0001096 total: 15.8s remaining: 405ms

975: learn: 0.0001096 total: 15.8s remaining: 389ms

976: learn: 0.0001096 total: 15.8s remaining: 373ms

977: learn: 0.0001096 total: 15.8s remaining: 356ms

978: learn: 0.0001096 total: 15.9s remaining: 340ms

979: learn: 0.0001096 total: 15.9s remaining: 324ms

980: learn: 0.0001096 total: 15.9s remaining: 308ms

981: learn: 0.0001096 total: 15.9s remaining: 292ms

982: learn: 0.0001096 total: 15.9s remaining: 275ms

983: learn: 0.0001096 total: 15.9s remaining: 259ms

984: learn: 0.0001096 total: 16s remaining: 243ms

985: learn: 0.0001096 total: 16s remaining: 227ms

986: learn: 0.0001096 total: 16s remaining: 211ms

987: learn: 0.0001096 total: 16s remaining: 194ms

988: learn: 0.0001096 total: 16s remaining: 178ms

989: learn: 0.0001096 total: 16s remaining: 162ms

990: learn: 0.0001096 total: 16s remaining: 146ms

991: learn: 0.0001096 total: 16.1s remaining: 129ms

992: learn: 0.0001096 total: 16.1s remaining: 113ms

993: learn: 0.0001096 total: 16.1s remaining: 97.1ms

994: learn: 0.0001096 total: 16.1s remaining: 80.9ms

995: learn: 0.0001096 total: 16.1s remaining: 64.7ms

996: learn: 0.0001096 total: 16.1s remaining: 48.5ms

997: learn: 0.0001096 total: 16.1s remaining: 32.4ms

998: learn: 0.0001096 total: 16.2s remaining: 16.2ms

999: learn: 0.0001096 total: 16.2s remaining: 0us

<catboost.core.CatBoostClassifier at 0x7f167ddb8a50>

In [43]:

scores['CatBoost'] = clf4.score(X\_test\_cat, y\_test\_cat)

In [44]:

scores

Out[44]:

{'LinearSVC': 0.958706265256306,

'MultinomialNB': 0.939924057499322,

'XGB': 0.9374830485489558,

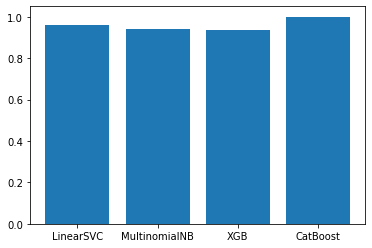
'CatBoost': 1.0}

In [45]:

plt.bar(scores.keys(), scores.values())

Out[45]:

<BarContainer object of 4 artists>



5. Conclusion

**We have trained & tested 4 models for NLP task (implementing the traditional NLP preprocessing strategies). They all perform very good, however this is most likely due to the high correlation of the target other categorical features (such as 'subject'). If we did not add it to analysis, the result could have been totally different.**