

Fake News Identifier

Name: Benoush Valentine F

Reg no: 610821205006

Dept: B.Tech.,IT

Year: 3rd

Overview

A fake news identifier is a tool that uses artificial intelligence (AI) to automatically detect fake news articles. These tools typically work by analyzing a variety of factors, including the article's content, its source, and its language style.

Some of the most common features that fake news identifiers look for include:

- Biased or inflammatory language: Fake news articles often use language that is designed to evoke strong emotions, such as anger, fear, or disgust.
- False or exaggerated claims: Fake news articles often make claims that are not supported by evidence, or that are exaggerated beyond what is reasonable.
- Lack of credible sourcing: Fake news articles often do not cite any sources for their claims, or they cite sources that are not credible.
- Unrealistic or unbelievable events: Fake news articles often report events that are so unrealistic or unbelievable that they are clearly false.
- Poor grammar and spelling: Fake news articles often contain grammatical errors and spelling mistakes.

Fake news identifiers are not perfect, and they can sometimes incorrectly identify real news articles as fake. However, they can be a useful tool for helping people to identify and avoid fake news.

Here are some examples of fake news identifier tools:

- Fake News Detector: This tool from the Stanford University News Lab uses AI to analyze news articles for signs of fake news.
- NewsGuard: This tool gives news websites a rating based on their transparency, editorial standards, and fact-checking practices.
- BuzzFeed News's Fake News Detector: This tool uses AI to analyze news articles for signs of fake news, and it also includes a database of known fake news websites.

It is important to note that fake news identifiers are not a substitute for critical thinking. It is still important to be skeptical of news articles, even if they have been rated as credible by a fake news identifier tool.

Here are some tips for identifying fake news articles:

- Check the source: Is the article from a reputable news organization?
- Check the author: Is the author identified? Do they have any credentials?
- Check the date: Is the article recent?
- Check the headline: Is the headline sensational or misleading?
- Check the content: Does the article make any claims that seem unrealistic or unbelievable?
- Check the sources: Are the sources cited credible?

Developed Source Code:-

```
import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

from nltk.corpus import stopwords

from nltk.stem.porter import PorterStemmer

from sklearn.feature_extraction.text import TfidfVectorizer

from sklearn.model_selection import train_test_split

from sklearn.linear_model import LogisticRegression

from sklearn.metrics import accuracy_score , confusion_matrix

import nltk

nltk.download('stopwords')

print(stopwords.words('english'))

news_data = pd.read_csv('/kaggle/input/fake-news/train.csv')

news_data.head()

news_data.shape

news_data.info()

news_data.isnull().sum()

news_data = news_data.fillna("")

news_data['content'] = news_data['author']+' '+news_data['title']

print(news_data['content'])

X = news_data.drop(labels= 'label' , axis=1)

Y = news_data['label']

print(X)

print(Y)

port_stem = PorterStemmer()

def stemming(content):

stemmed_content = re.sub('[^a-zA-Z]', ' ',content)
```

```

stemmed_content = stemmed_content.lower()

stemmed_content = stemmed_content.split()

stemmed_content = [port_stem.stem(word) for word in stemmed_content if not word in stopwords.words('english')]

stemmed_content = ' '.join(stemmed_content)

return stemmed_content

import re

news_data['content'] = news_data['content'].apply(stemming)

print(news_data['content'])

X = news_data['content'].values

Y = news_data['label'].values

print(X)

print(Y)

Y.shape

vectorizer = TfidfVectorizer()

vectorizer.fit(X)

X = vectorizer.transform(X)

print(X)

X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=2)

model = LogisticRegression()

model.fit(X_train, Y_train)

X_train_prediction = model.predict(X_train)

training_data_accuracy = accuracy_score(X_train_prediction, Y_train)

print('Accuracy score of the training data : ', training_data_accuracy)

X_test_prediction = model.predict(X_test)

testing_data_accuracy = accuracy_score(X_test_prediction, Y_test)

print('Accuracy score of the testing data : ', testing_data_accuracy)

y_pred = model.predict(X_test)

confusion_matrix = confusion_matrix(Y_test, y_pred.round())

sns.heatmap(confusion_matrix, annot=True, fmt="d", cbar = False)

plt.title("Confusion Matrix")

plt.show()

X_new = X_test[1939]

prediction = model.predict(X_new)

print(prediction)

if (prediction[0]==0):

    print('The news is Real')

else:

    print('The news is Fake')

```