Fake News Detection

Team Member :

Name	Reg No
ABDUR RAHIM S	821021104003
SiVA PRAKASH S	821021104043
VARUN RAJ M	821021104055
VIJAY RAJ B J	821021104057
SIVA G	821021104304

Table of Content

Fake News Detection Using NLP

Phase-3

Task :

In this part you will begin building your project by loading and preprocessing the dataset. Begin building the fake news detection model by loading and preprocessing the dataset.Load the fake news dataset and preprocess the textual data.

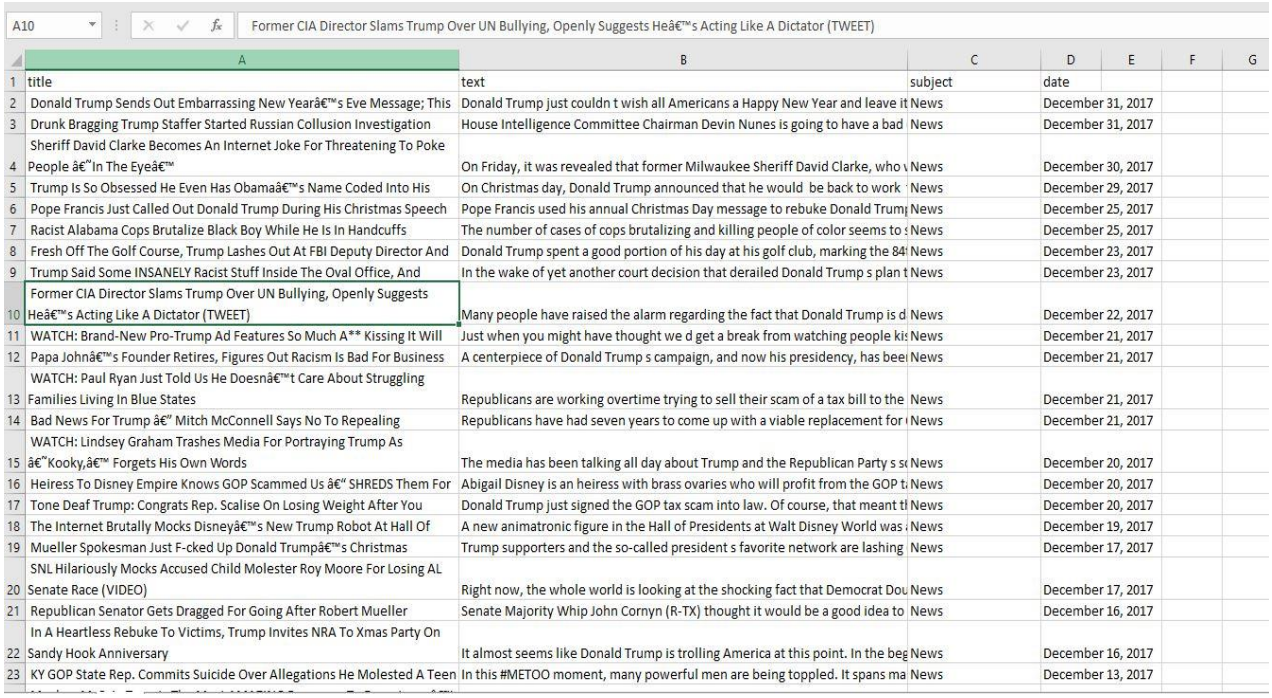
DataSet :

https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset

This Fake and Real News Dataset have two files :

- Fake.csv
- True.csv

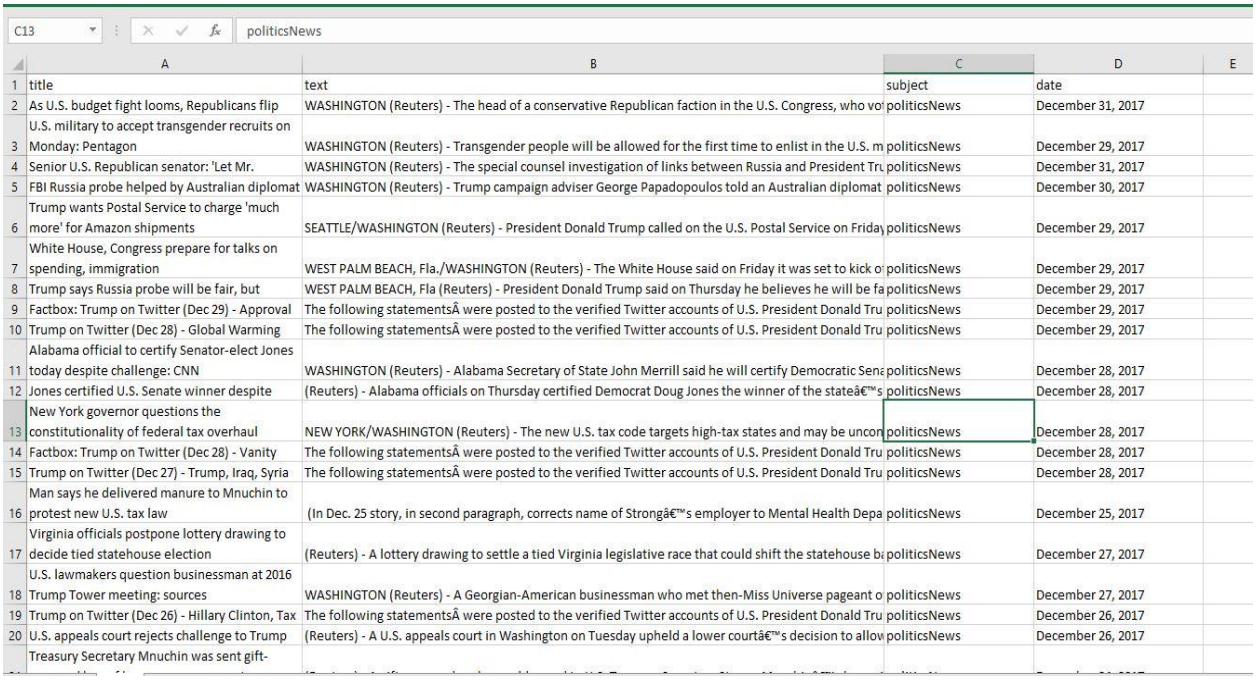
Fake.csv:



A10 — Former CIA Director Slams Trump Over UN Bullying, Openly Suggests Heâ€™s Acting Like A Dictator (TWEET)

	A	B	C	D
1	title	text	subject	date
2	Donald Trump Sends Out Embarrassing New Yearâ€™s Eve Message; This	Donald Trump just couldn t wish all Americans a Happy New Year and leave it	News	December 31, 2017
3	Drunk Bragging Trump Staffer Started Russian Collusion Investigation	House Intelligence Committee Chairman Devin Nunes is going to have a bad	News	December 31, 2017
4	Sheriff David Clarke Becomes An Internet Joke For Threatening To Poke People â€™In The Eyeâ€™	On Friday, it was revealed that former Milwaukee Sheriff David Clarke, who	News	December 30, 2017
5	Trump Is So Obsessed He Even Has Obamaâ€™s Name Coded Into His	On Christmas day, Donald Trump announced that he would be back to work	News	December 29, 2017
6	Pope Francis Just Called Out Donald Trump During His Christmas Speech	Pope Francis used his annual Christmas Day message to rebuke Donald Trump	News	December 25, 2017
7	Racist Alabama Cops Brutalize Black Boy While He Is In Handcuffs	The number of cases of cops brutalizing and killing people of color seems to	News	December 25, 2017
8	Fresh Off The Golf Course, Trump Lashes Out At FBI Deputy Director And	Donald Trump spent a good portion of his day at his golf club, marking the 84	News	December 23, 2017
9	Trump Said Some INSANELY Racist Stuff Inside The Oval Office, And	In the wake of yet another court decision that derailed Donald Trump s plan t	News	December 23, 2017
10	Former CIA Director Slams Trump Over UN Bullying, Openly Suggests Heâ€™s Acting Like A Dictator (TWEET)	Many people have raised the alarm regarding the fact that Donald Trump is d	News	December 22, 2017
11	WATCH: Brand-New Pro-Trump Ad Features So Much A** Kissing It Will	Just when you might have thought we d get a break from watching people kis	News	December 21, 2017
12	Papa Johnâ€™s Founder Retires, Figures Out Racism Is Bad For Business	A centerpiece of Donald Trump s campaign, and now his presidency, has bee	News	December 21, 2017
13	WATCH: Paul Ryan Just Told Us He Doesnâ€™t Care About Struggling Families Living In Blue States	Republicans are working overtime trying to sell their scam of a tax bill to the	News	December 21, 2017
14	Bad News For Trump â€" Mitch McConnell Says No To Repealing	Republicans have had seven years to come up with a viable replacement for	News	December 21, 2017
15	WATCH: Lindsey Graham Trashes Media For Portraying Trump As â€™Kooky,â€™ Forgets His Own Words	The media has been talking all day about Trump and the Republican Party s s	News	December 20, 2017
16	Heiress To Disney Empire Knows GOP Scammed Us â€" SHREDS Them For	Abigail Disney is an heiress with brass ovaries who will profit from the GOP t	News	December 20, 2017
17	Tone Deaf Trump: Congrats Rep. Scalise On Losing Weight After You	Donald Trump just signed the GOP tax scam into law. Of course, that meant t	News	December 20, 2017
18	The Internet Brutally Mocks Disneyâ€™s New Trump Robot At Hall Of	A new animatronic figure in the Hall of Presidents at Walt Disney World was	News	December 19, 2017
19	Mueller Spokesman Just F-cked Up Donald Trumpâ€™s Christmas	Trump supporters and the so-called president s favorite network are lashing	News	December 17, 2017
20	SNL Hilariously Mocks Accused Child Molester Roy Moore For Losing AL Senate Race (VIDEO)	Right now, the whole world is looking at the shocking fact that Democrat Dou	News	December 17, 2017
21	Republican Senator Gets Dragged For Going After Robert Mueller	Senate Majority Whip John Cornyn (R-TX) thought it would be a good idea to	News	December 16, 2017
22	In A Heartless Rebuke To Victims, Trump Invites NRA To Xmas Party On Sandy Hook Anniversary	It almost seems like Donald Trump is trolling America at this point. In the beg	News	December 16, 2017
23	KY GOP State Rep. Commits Suicide Over Allegations He Molested A Teen	In this #METOO moment, many powerful men are being toppled. It spans ma	News	December 13, 2017

True.csv :

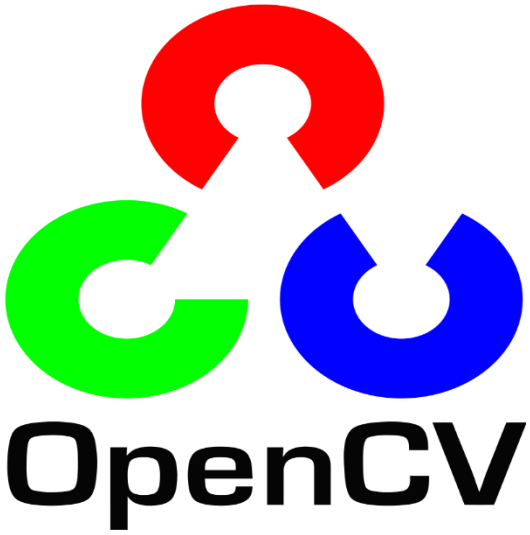


C13 — politicsNews

	A	B	C	D
1	title	text	subject	date
2	As U.S. budget fight looms, Republicans flip	WASHINGTON (Reuters) - The head of a conservative Republican faction in the U.S. Congress, who vo	politicsNews	December 31, 2017
3	U.S. military to accept transgender recruits on Monday: Pentagon	WASHINGTON (Reuters) - Transgender people will be allowed for the first time to enlist in the U.S. m	politicsNews	December 29, 2017
4	Senior U.S. Republican senator: 'Let Mr.	WASHINGTON (Reuters) - The special counsel investigation of links between Russia and President Tru	politicsNews	December 31, 2017
5	FBI Russia probe helped by Australian diplomat	WASHINGTON (Reuters) - Trump campaign adviser George Papadopoulos told an Australian diplomat	politicsNews	December 30, 2017
6	Trump wants Postal Service to charge 'much more' for Amazon shipments	SEATTLE/WASHINGTON (Reuters) - President Donald Trump called on the U.S. Postal Service on Friday	politicsNews	December 29, 2017
7	White House, Congress prepare for talks on spending, immigration	WEST PALM BEACH, Fla./WASHINGTON (Reuters) - The White House said on Friday it was set to kick o	politicsNews	December 29, 2017
8	Trump says Russia probe will be fair, but	WEST PALM BEACH, Fla (Reuters) - President Donald Trump said on Thursday he believes he will be fa	politicsNews	December 29, 2017
9	Factbox: Trump on Twitter (Dec 29) - Approval	The following statementsÂ were posted to the verified Twitter accounts of U.S. President Donald Tru	politicsNews	December 29, 2017
10	Trump on Twitter (Dec 28) - Global Warming	The following statementsÂ were posted to the verified Twitter accounts of U.S. President Donald Tru	politicsNews	December 29, 2017
11	Alabama official to certify Senator-elect Jones today despite challenge: CNN	WASHINGTON (Reuters) - Alabama Secretary of State John Merrill said he will certify Democratic Sena	politicsNews	December 28, 2017
12	Jones certified U.S. Senate winner despite	(Reuters) - Alabama officials on Thursday certified Democrat Doug Jones the winner of the stateâ€™s	politicsNews	December 28, 2017
13	New York governor questions the constitutionality of federal tax overhaul	NEW YORK/WASHINGTON (Reuters) - The new U.S. tax code targets high-tax states and may be uncon	politicsNews	December 28, 2017
14	Factbox: Trump on Twitter (Dec 28) - Vanity	The following statementsÂ were posted to the verified Twitter accounts of U.S. President Donald Tru	politicsNews	December 28, 2017
15	Trump on Twitter (Dec 27) - Trump, Iraq, Syria	The following statementsÂ were posted to the verified Twitter accounts of U.S. President Donald Tru	politicsNews	December 28, 2017
16	Man says he delivered manure to Mnuchin to protest new U.S. tax law	(In Dec. 25 story, in second paragraph, corrects name of Strongâ€™s employer to Mental Health Depa	politicsNews	December 25, 2017
17	Virginia officials postpone lottery drawing to decide tied statehouse election	(Reuters) - A lottery drawing to settle a tied Virginia legislative race that could shift the statehouse b	politicsNews	December 27, 2017
18	U.S. lawmakers question businessman at 2016 Trump Tower meeting: sources	WASHINGTON (Reuters) - A Georgian-American businessman who met then-Miss Universe pageant o	politicsNews	December 27, 2017
19	Trump on Twitter (Dec 26) - Hillary Clinton, Tax	The following statementsÂ were posted to the verified Twitter accounts of U.S. President Donald Tru	politicsNews	December 26, 2017
20	U.S. appeals court rejects challenge to Trump	(Reuters) - A U.S. appeals court in Washington on Tuesday upheld a lower courtâ€™s decision to allow	politicsNews	December 26, 2017
	Treasury Secretary Mnuchin was sent gift-			

OpenCV:

OpenCV is a great tool for image processing and performing computer vision tasks. It is an open-source library that can be used to perform tasks like face detection, objection tracking, landmark detection, and much more. It supports multiple languages including python, java C++. Although, For this article, we will be limiting to python only.



Current Version :

opencv-python 4.8.1.78

Package Link :

https://pypi.org/project/opencv-python/

TensorFlow :

TensorFlow is an open-source library developed by Google primarily for deep learning applications. It also supports traditional machine learning. TensorFlow was originally developed for large numerical computations without keeping deep learning in mind. TensorFlow is an end-to-end open-source machine learning platform with a focus on deep neural networks. Deep learning is a subtype of machine learning that analyses massive amounts of unstructured data. Since it works with structured data, deep learning is different from normal machine learning.

Current Version :

tensorflow 2.14.0

Package Link :

https://pypi.org/project/tensorflow/

Convolutional Neural Network(CNN):

A CNN is a kind of network architecture for deep learning algorithms and is specifically used for image recognition and tasks that involve the processing of pixel data. There are other types of neural networks in deep learning, but for identifying and recognizing objects, CNNs are the network architecture of choice.

There are four types of layers for a convolutional neural network:

- convolutional layer,
- pooling layer,
- ReLU correction layer and
- fully-connected layer.

Download The Dataset:

- Go to the Kaggle dataset page: https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset.
- Download the dataset files, which typically come in the form of CSV or other common formats.

Import Necessary Libraries:

- In Python, you'll want to import libraries that you'll use for data manipulation and machine learning.
- Common libraries include pandas, numpy, scikit-learn, and NLTK (Natural Language Toolkit).

Load the Data:

- Read the downloaded dataset files into pandas dataframes.

Data Preprocessing:

- Combine the fake and real news data.
- Label the data as 'fake' and 'real' for classification
- Remove any unnecessary column
- Handle missing values if necessary
- Clean and preprocess the textual data (text of the news articles)

Preprocess the Data:

Text Cleaning	*Remove any unnecessary characters, symbols, and HTML tags from the text*
Lowercasing	*IConvert all text to lowercase to ensure uniformity.*
Tokenization	*Split the text into individual words or tokens.*
Stopword Removal	*Remove common stopwords like "the," "and," "in," etc.*
Vectorization	*Convert text data into numerical form using techniques like TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings.*
Label Encoding	*If your dataset contains labels (real or fake news), you may need to encode them into numerical values (e.g., 0 for fake, 1 for real).*

Combine and Shuffle Data:

- Combine the fake and real news data.
- Shuffle the data to ensure it's not ordered.

Python

IN [0]

```
import pandas as pd
import numpy as np
import nltk
from sklearn.model_selection import train_test_split
```

IN [1]

```
# Assuming you have downloaded 'fake.csv' and 'true.csv' files
fake_news_df = pd.read_csv('fake.csv')
real_news_df = pd.read_csv('true.csv')
```

IN[2]

```
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer

nltk.download('stopwords')
nltk.download('punkt')

stop_words = set(stopwords.words('english'))
lemmatizer = WordNetLemmatizer()

def preprocess_text(text):
    # Tokenize the text
    words = word_tokenize(text)
```

Python

IN[3]

```
# Join the words back into a single string
        return ' '.join(words)


fake_news_df['text'] = fake_news_df['text'].apply(preprocess_text)
real_news_df['text'] =
real_news_df['text'].apply(preprocess_text)
```

IN[4]

```
# Combine the data
data = pd.concat([fake_news_df, real_news_df],
ignore_index=True)


# Shuffle the data
data = data.sample(frac=1).reset_index(drop=TrueP
```

Introduction:

Fake news has become increasingly prevalent in recent years. With more and more of our news being consumed from online sources, being able to discern whether or not news is real is vital. This article is presented as a fun look at how deep learning approaches could be used to solve this problem in Python.**It is broken into 5 sections:**

- Description of Data
- Pre-processing
- Embedding Layers
- Model Architecture
- Results

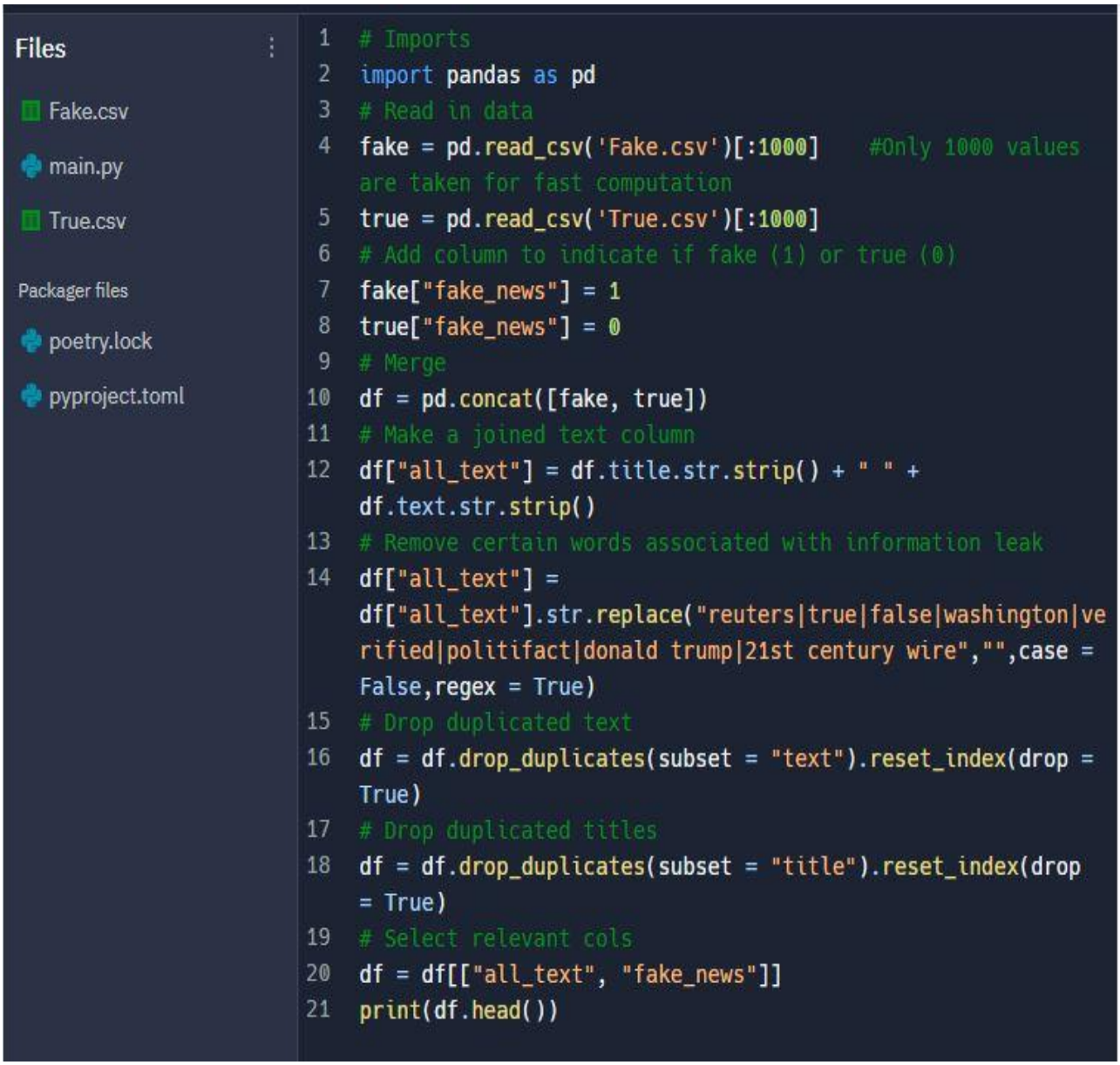
Description of Data:

The data for this task is taken from a Kaggle dataset of around 45K news articles. Around half of the data are fake articles pulled from various US-based websites that have been deemed unreliable by fact-checking organisations, with the other half being true articles pulled from Reuters, a US news website.

all_text	fake_news
Senate Democrats push for new gun control measures () - Leading U.S. Senate Democrats on Monday urged quick passage of legislation defeated last year to impose additional gun controls in the wake of the weekend mass shooting in Florida. Four Democratic senators, led by Chuck Schumer of New York, the No. 2 Senate Democrat, called for immediate passage of a bill preventing people on "terror watch lists" and other "suspected terrorists" from buying firearms or explosives. Last December, Democrats attempted to pass this legislation but were blocked by Republicans, who said the government could mistakenly place innocent people on watch lists. The new push for legislation came after a man with an assault rifle entered a gay nightclub in Orlando and killed 49 people and wounded 53 others. The gunman subsequently was killed by law enforcement officers. Federal agents had interviewed the gunman twice in recent years. Joining Schumer in calling for passage of the legislation are Senators Dianne Feinstein of California, Bill Nelson of Florida and Richard Blumenthal of Connecticut. Meanwhile, Democratic President Barack Obama on Monday blamed weak gun laws for allowing disturbed individuals to gain access to powerful weapons. Hillary Clinton, the leading Democratic candidate to replace Obama in the White House, said people on watch lists should be barred from buying guns and said there should be a debate about possible restrictions on assault weapons. , the presumptive Republican presidential nominee, took a different approach, repeating his call for temporarily banning Muslims from entering the country. The Orlando gunman was born in the United States. His parents were Afghan immigrants.	0
Racist Cowards Nearly Beat Black Ex-Marine To Death In Wyoming Clayton Denny is a former U.S. Marine who moved to Sheridan, Wyoming in August to work as a personal trainer and exercise coach. After a bike ride through town on Saturday evening, Clayton decided to cap off his night with a visit to a local bar known as the Beaver Creek Saloon.While there, he overheard Jacob Olson, 26, and Dylan Dygert, 21, say a racial slur. Denny, who is of African-American and Chippewa Cree descent, approached the two men to confront them about it.All seemed to be going well as Denny shook hands with one of the men. But then something changed as the encounter became more heated. The bartender ordered them to take the argument outside. The trio resumed their argument in an alley. Denny turned and walked away and that s when Olson and Dygert attacked. A push, white flash, bits and pieces of pain, Denny recalled about what happened. Dygert and Olson pushed Denny to the ground and began kicking and punching him while yelling racial epithets at him. They beat him so badly that Denny thought he might die.Even Olson believed he may have left Denny for dead as a witness told police that Olson told someone on the phone that he may have killed someone.Aside from scratches, bruises, and cuts, Denny suffered a broken orbital bone, which he would need surgery for in order to drain fluid and clean out an infection. He also suffered two swollen eyes, multiple cuts on his head that had to be stapled, and a concussion.Olson and Dygert have both been arrested by Sheridan police and are being held on $20,000 bond.Of course, they claimed they only attacked and viciously beat Denny in self defense. However, they didn t know that the incident had been caught on video, which clearly showed that Olson and Dygert went after Denny after he had already turned and walked away from them. In other words, Denny wasn t a threat and Olson and Dygert lied to police in an effort to save their racist asses.Both men face 10 years in prison for the racist attack and hate crime charges are being considered by the County Attorney s Office. And if they don t charge them with a hate crime, the Department Of Justice should do it for them.But due to this disgusting attack, Denny says he and his girlfriend have received tremendous support from the Sheridan community. I do want people to know that the community of Sheridan has been awesome. We ve had people stopping by all weekend and all week, just saying, you know giving their condolences and helping us out. It sounds like Denny has all the support he needs to get through this ordeal. Let s just hope Olson and Dygert use their time in prison wisely to think about what they did and learn from it.Featured Image: Billings Gazette	1

Pre-processing

The first step in pre-processing is to load the data in and create a field to indicate whether the article was fake (1) or true (0). Both title and text fields (of each article) are used to de-duplicate the data and are then merged into one long “all_text” field. To address previously identified leakages the date and subject columns are dropped, as are a number of informative keywords (“reuters”, “politifact” etc).




```python
# Imports
import pandas as pd
# Read in data
fake = pd.read_csv('Fake.csv')[:1000]    #Only 1000 values are taken for fast computation
true = pd.read_csv('True.csv')[:1000]
# Add column to indicate if fake (1) or true (0)
fake["fake_news"] = 1
true["fake_news"] = 0
# Merge
df = pd.concat([fake, true])
# Make a joined text column
df["all_text"] = df.title.str.strip() + " " + df.text.str.strip()
# Remove certain words associated with information leak
df["all_text"] = df["all_text"].str.replace("reuters|true|false|washington|verified|politifact|donald trump|21st century wire","",case = False,regex = True)
# Drop duplicated text
df = df.drop_duplicates(subset = "text").reset_index(drop = True)
# Drop duplicated titles
df = df.drop_duplicates(subset = "title").reset_index(drop = True)
# Select relevant cols
df = df[["all_text", "fake_news"]]
print(df.head())
```

```
#
Imports
        from keras.preprocessing.text import Tokenizer
        from keras.preprocessing.sequence import pad_sequences
        from sklearn.model_selection import train_test_split
        import numpy as np


        def tokenize_padder(train_text, test_text,
                            chars_to_filter = '!"#$%&()*+,-./:;<=>?@[\\]^_`{|}~\t\n',
                             oov_token = "OOV",
                             maxlen = 500,
                             padding = "pre",
                             truncating = "post"
                            ):
            # Create tokenizer
            tokenizer = Tokenizer(filters = chars_to_filter,
                                  oov_token = oov_token)


            # Fit tokenizer on training data only
            tokenizer.fit_on_texts(train_text)


            # Generate sequences
            train_sequences = tokenizer.texts_to_sequences(train_text)
            test_sequences = tokenizer.texts_to_sequences(test_text)


            # Pad and trim sequences
            # Pre-padding is empirically better for sequence modelling
            # Post-truncating ensures the titles are included in observations
            train_padded = pad_sequences(train_sequences, maxlen = maxlen, padding =
        padding, truncating = truncating)
            test_padded = pad_sequences(test_sequences, maxlen = maxlen, padding = padding,
        truncating = truncating)


            return tokenizer, train_padded, test_padded


        # Split into test and train data
        X = df.all_text.values
        y = np.array(df["fake_news"], dtype = "float32")
        text_train, text_test, y_train, y_test = train_test_split(X, y,
                                                                  test_size = 0.2, shuffle = True,
```

```
                                                    # reproducible split
                                                   random_state = 1)

# Process, tokenize, pad/trim
tokenizer, X_train, X_test = tokenize_padder(text_train, text_test)
```

```
# Tokenized text in number form
X_train[10][:100]
```

```
array([  261,  2792,  1394,    26,   849,    58,     4,     2,   461,
          297,    76,  1282,     3,   420,    20, 18675,   297,   822,
           34,    14,  1550,    43,   107,   420,     8,  1622,    11,
          564,    30,  2482,    15,    15,   438,    10,     6,    10,
           15,    18,   495,     4, 17902, 15696,    40,  1018,   205,
          825,    72,    61,     9,    60,   132,    76,     9,    46,
          565,     3,  1487,    20,   170,   226,    20,   859,   719,
           72,   241,    24,     7,    20,    96, 18675,   297,    37,
          168,    28,   749,     4,  1394,    21,  9395, 27300,     4,
         1385,   103,    12,  1018,  1123,    10,   192,    37,   446,
          509,    59,    28,   108,  2470,    80,  6193,    21,   748,
          213], dtype=int32)
```

```
# Convert tokenized text in number form back into text again
tokenizer.sequences_to_texts([X_train[10]])[0][:100]
```

Embedding Layers:

Embedding is a technique often used in Natural Language Processing to represent words in a reduced number of dimensions. For example, if our fake news dataset contained a vocabulary of only 1000 words (e.g. only 1000 unique words) then each word could be represented by a vector of 999 0s and a single 1. 999 instances of it not being one of our 1000 words and 1 instance of it being 1 of our 1000 words. This is known as **one-hot encoding.**

spaCy

spaCy is an open-source library for NLP in Python. We will be using the en_core_web_sm package from it, which is a small English-language pipeline that has been trained on blogs, news and comments (so is appropriate to our task). We'll use the package's pre-trained embedding with our data, e.g. to represent our text data in spaCy's embedding. We'll be using it statically, by setting trainable=False as seen in the code below. Alternatively, we could just use the spaCy embedding to seed our model with and then have it be updated throughout training. However, it perhaps makes for a more interesting comparison with the custom, Keras embedding if the spaCy embedding remains fixed.

Keras / Custom:

Program :

```
#Imports
        from keras.initializers import Constant
        import spacy
        from keras.layers import Embedding



        def spacy_embedding(tokenizer, maxlen = 500, show_progress = False):



            # Load the spacy pipeline
            nlp = spacy.load("en_core_web_sm")
            # Get vocab size of tokenizer
            vocab_size = len(tokenizer.word_index) + 1
```

```
# Get the number of embedding dimensions SpaCy uses
embedding_dim = nlp("any_word").vector.shape[0]
# Create a matrix to use in embedding layer
embedding_matrix = np.zeros((vocab_size, embedding_dim))



# Iterate through our vocabulary, mapping words to spacy embedding
# this will take a while to run
for i, word in enumerate(tokenizer.word_index):
    embedding_matrix[i] = nlp(word).vector
    # Show progress if desired
    if show_progress:
        if i % 10000 == 0 and i > 0:
            print(round(i*100/vocab_size, 3), "% complete")




# Load the embedding matrix as the weights matrix for the embedding layer
# Set trainable to False as the layer is already "learned"
Embedding_layer = Embedding(
    vocab_size,
    embedding_dim,
    input_length = maxlen,
    embeddings_initializer=Constant(embedding_matrix),
    trainable=False,
    name = "spacy_embedding")
```

```
    return Embedding_layer




def keras_embedding(tokenizer, embedding_dim = 256, maxlen = 500):



    # Get vocab size of tokenizer

    vocab_size = len(tokenizer.word_index) + 1



    # Load the embedding matrix as the weights matrix for the embedding layer

    # Set trainable to False as the layer is already "learned"

    Embedding_layer = Embedding(

        vocab_size,

        embedding_dim,

        input_length = maxlen,

        name = "keras_embedding")



    return Embedding_layer







# Generate the embeddings

embed_dict = dict()
```

```
embed_dict["spacy"] = spacy_embedding(tokenizer, show_progress = True, maxlen =
500)

embed_dict["keras"] = keras_embedding(tokenizer, maxlen = 500)
```

Model Architecture

Since our data is sequential (e.g. words in a sentence) a Gated Recurrent Unit could be used in the neural network we're building (see here or here for more details). ***GRUs have been found to perform similarly*** to other recurrent units such as the long short-term memory (LSTM) unit (Chung et al., 2014) but with the added advantage that they can be ***faster to train***.

Additionally, we will be using the Keras `Bidirectional` layer. Whereas a standard GRU trains only once on the input sequence, a bidirectional GRU will train twice- once on the input sequence and once again on a reversed copy of the input sequence. The hope is that this will provide extra context to the network that yields faster and better learning. Indeed, bidirectional models have been shown to outperform unidirectional models in other fake news detection research (Bahad et al., 2019).

Fake News Classification with Keras

```
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 keras_embedding (Embedding)  (None, 500, 256)          33604608

 Bidirectional_GRU (Bidirect  (None, 64)               55680
 ional)

 Linear_Dense (Dense)         (None, 256)              16640

 Batch_Norm1 (BatchNormaliza  (None, 256)              1024
 tion)

 ReLU_Activation (Activation  (None, 256)              0
 )

 Output (Dense)               (None, 1)                257

=================================================================
Total params: 33,678,209
Trainable params: 33,677,697
Non-trainable params: 512
```

```
from keras.callbacks import EarlyStopping
from tensorflow.keras.layers import Dense, BatchNormalization,
Reshape, Activation
from tensorflow.keras.layers import Embedding, GRU, Bidirectional
from tensorflow.keras import Sequential
# Model compilation params
compile_hp = dict()
compile_hp["loss"] = "binary_crossentropy"
compile_hp["optimizer"] = optimizers.Adam(learning_rate = 0.001)
compile_hp["metrics"] = ["accuracy"]
compile_hp["maxlen"] = 500


# Model fitting params
fit_hp = dict()
fit_hp["batch_size"] = 64
fit_hp["epochs"] = 100
```

```
fit_hp["validation_split"] = 0.3
# Create callback to select the best model
fit_hp["callbacks"] = EarlyStopping(monitor = "val_loss",
                                    mode = "min",
                                    restore_best_weights =
True,
                                    patience = 10)


def bi_gru(loss = "binary_crossentropy",
               optimizer = "adam",
              metrics = ["accuracy"],
              batch_normalize = False,
              embedding = None,
              maxlen = 500,
              hidden_dense_units = 256,
              dense_kernel_initializer = "glorot_uniform",
              rnn_units = 32,
              rnn_kernel_initializer = "glorot_uniform"):


    # Build model
    model = Sequential(name = "GRU")


    # Add embedding if desired
    if embedding:
        # Embedding contains input shape
        model.add(embedding)
    else:
        # Otherwise reshape data to work with GRU
        model.add(Reshape((maxlen, 1), input_shape = (maxlen, ),
name = "Reshaping"))


    # Add GRU
    model.add(Bidirectional(GRU(rnn_units,
                                kernel_initializer =
rnn_kernel_initializer),
                                name = "Bidirectional_GRU"))


    # Baseline model
    model.add(Dense(hidden_dense_units, name = "Linear_Dense",
                    kernel_initializer = dense_kernel_initializer))
```

```
    # Batch normalised model
    if batch_normalize:
        model.add(BatchNormalization(name = "Batch_Norm1"))


    # Apply non-linear activation, specified in this way to be
consistent
    # with the original paper
    model.add(Activation("relu", name = "ReLU_Activation"))


    # Output layer
    model.add(Dense(1, activation = "sigmoid", name = "Output",
                    kernel_initializer = dense_kernel_initializer))
    # Compile model
    model.compile(loss = loss, optimizer = optimizer,
                  metrics = metrics)


    return model


# Set embedding
embedding_layer = "keras"
# Toggle batch normalization
batch_normalize = True


# Build and fit model with embedding
model = bi_gru(**compile_hp, batch_normalize=batch_normalize,
                   embedding = embed_dict[embedding_layer])
model.summary()
history = model.fit(X_train, y_train, **fit_hp)
```

Results

Both the Keras- and spaCy-embedded models will take a good amount of time to train, but ultimately we'll end up with something that we can evaluate on our test data with.

Overall, the **Keras-embedded model performed better**– achieving a test accuracy of **99.1%** vs the spaCy model's **94.8%**.

	embedding	accuracy
0	keras	0.991116
0	spacy	0.948393

Conclusion:

- Pre-process your text data with Keras Tokenizer and the pad_sequences function
- Decide on an embedding to use; either pre-trained/transfer learning or a custom one learned from your dataset
- Add in a bi-directional GRU to your network (after the embedding), along with a batch normalized Dense layer and a sigmoid-activated output layer
- Train and fit your model – it may take a while to train, but will eventually achieve high test accuracies on the fake news dataset