

A close-up photograph of a double cheeseburger. The top bun is a golden brown with visible sesame seeds. The burger features two beef patties, melted cheese, lettuce, and pickles. A strip of bacon is visible on the left side. The bottom bun is also a golden brown. The background is dark, making the burger stand out.

Twitter Social Listening for McDonald's

Desmond
Mandy
Rainfield
Sophia

Agenda

01

Business Values

02

Data Collection

Tweepy API

03

Text Preprocessing

Regex, Stemming, Contractions

04

Model Training

LSTM Building & Evaluation

05

Analysis

Sentiment, Frequency Distribution,
Time-series

06

Codes

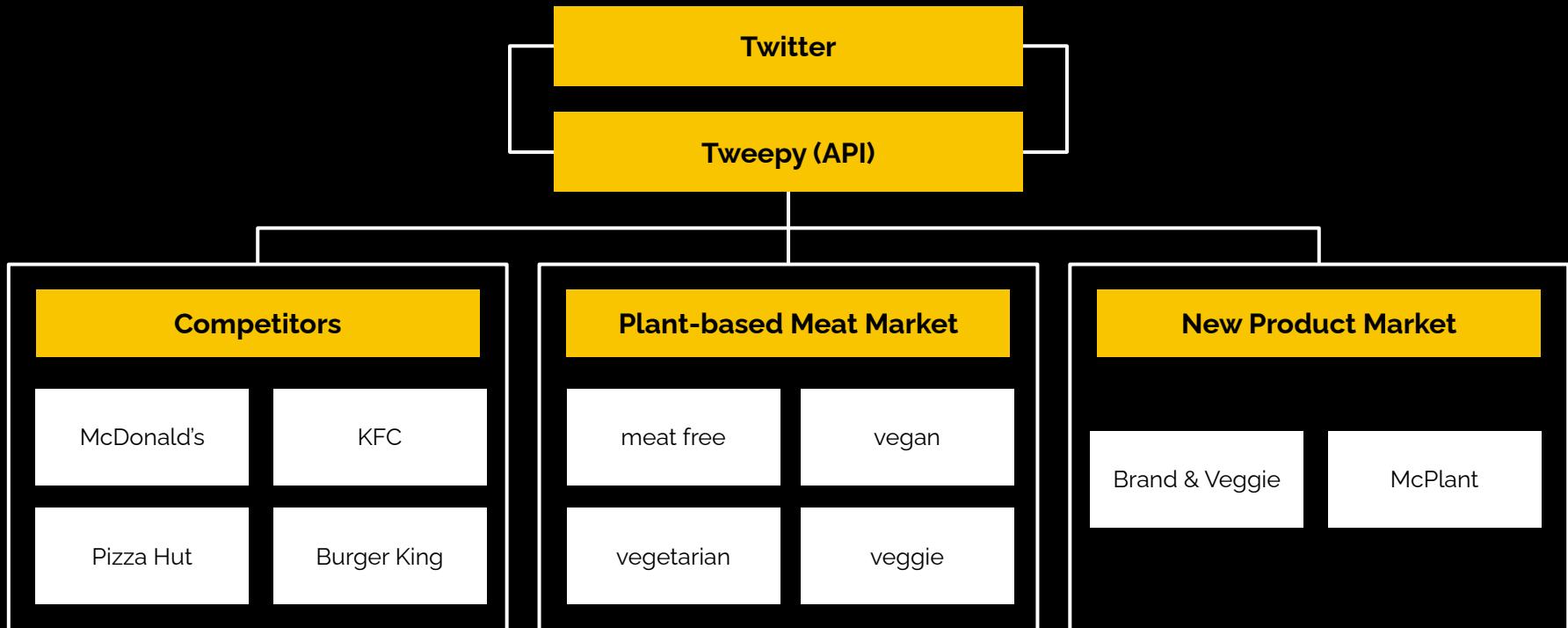
Business Values

To discover all real-time mentions and sentiments on Twitter related to McDonald's for instant Brand Reputation and Competitive Analysis.

It also offers valuable insights into Marketing Strategies, Product Development & Crisis Detection.



Data Collection



Text Preprocessing

Convert to Lowercase

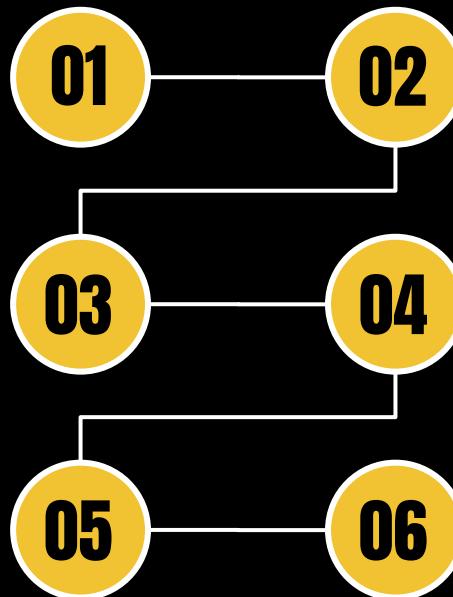
Handle consistency of different variations in input capitalization

Contractions

Convert abbreviations & shortened words into analyzable formats

Remove Punctuations, Numbers

Clear unnecessary context from being passed onto model



Remove Domain Specific Elements

E.g. Mentions "@", Hashtags "#", Retweets "RT", URLs, e.t.c

Stemming

Snowball Stemmer is used for Faster Computation Time & Higher Stemming Aggressiveness

Demojize

Remove emojis in tweets 🌱🍔🍟🍕

LSTM Model Training

Sentiment Dataset from Kaggle

1.6 million tweets with 50% Positive, 50% Negative



Data Preprocessing

Data Cleaning, Tokenization, Pad Sequences



Apply Pre-trained Word Embedding

- Unsupervised learning to obtain vector representations of words
- Performed training on aggregated global word-word co-occurrence statistics from a corpus of 6B tokens & 400K vocabularies

Text classification with Bi-LSTM

Model Summary

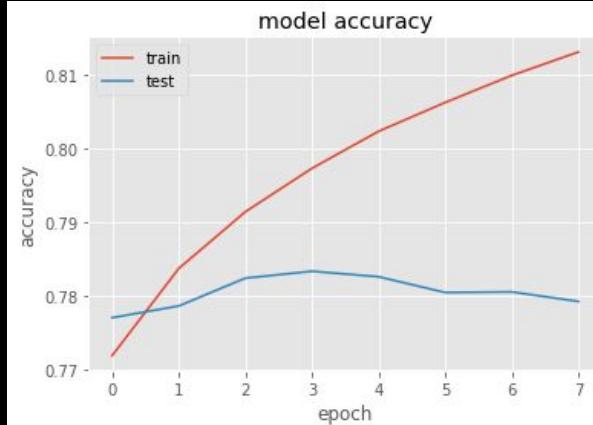
Layer (type)	Output Shape	Param #
<hr/>		
input_1 (InputLayer)	[None, 200]	0
embedding (Embedding)	(None, 200, 200)	64411600
conv1d (Conv1D)	(None, 196, 128)	128128
max_pooling1d (MaxPooling1D)	(None, 39, 128)	0
conv1d_1 (Conv1D)	(None, 35, 128)	82048
bidirectional (Bidirectional)	(None, 128)	98816
dense (Dense)	(None, 128)	16512
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 1)	129
<hr/>		
Total params:	64,737,233	
Trainable params:	325,633	
Non-trainable params:	64,411,600	

Model Training

```
Epoch 1/20
8750/8750 [=====] - 1128s 129ms/step - loss: 0.4757 - accuracy: 0.7720 - val_loss: 0.4665 - val_accuracy: 0.7771
Epoch 2/20
8750/8750 [=====] - 1119s 128ms/step - loss: 0.4566 - accuracy: 0.7838 - val_loss: 0.4630 - val_accuracy: 0.7787
Epoch 3/20
8750/8750 [=====] - 1115s 127ms/step - loss: 0.4441 - accuracy: 0.7915 - val_loss: 0.4601 - val_accuracy: 0.7825
Epoch 4/20
8750/8750 [=====] - 1109s 127ms/step - loss: 0.4345 - accuracy: 0.7974 - val_loss: 0.4596 - val_accuracy: 0.7834
Epoch 5/20
8750/8750 [=====] - 1106s 126ms/step - loss: 0.4262 - accuracy: 0.8024 - val_loss: 0.4592 - val_accuracy: 0.7827
Epoch 6/20
8750/8750 [=====] - 1109s 127ms/step - loss: 0.4187 - accuracy: 0.8063 - val_loss: 0.4640 - val_accuracy: 0.7805
Epoch 7/20
8750/8750 [=====] - 1108s 127ms/step - loss: 0.4127 - accuracy: 0.8100 - val_loss: 0.4646 - val_accuracy: 0.7806
Epoch 8/20
8750/8750 [=====] - 1103s 126ms/step - loss: 0.4068 - accuracy: 0.8131 - val_loss: 0.4683 - val_accuracy: 0.7793
```

Text classification with Bi-LSTM

Model Evaluation



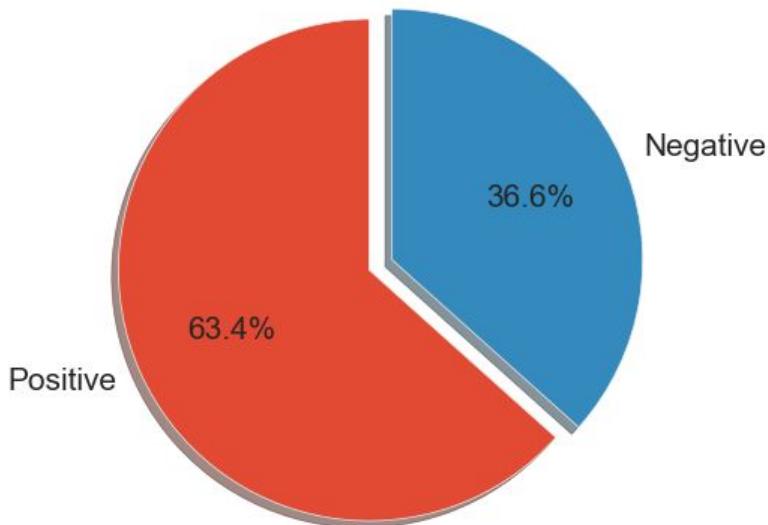
	precision	recall	f1-score	support
0	0.76	0.81	0.79	239925
1	0.80	0.74	0.77	240075
accuracy			0.78	480000
macro avg	0.78	0.78	0.78	480000
weighted avg	0.78	0.78	0.78	480000

01

McDonald's Sentiment Analysis



McDonald's - Sentiment Breakdown

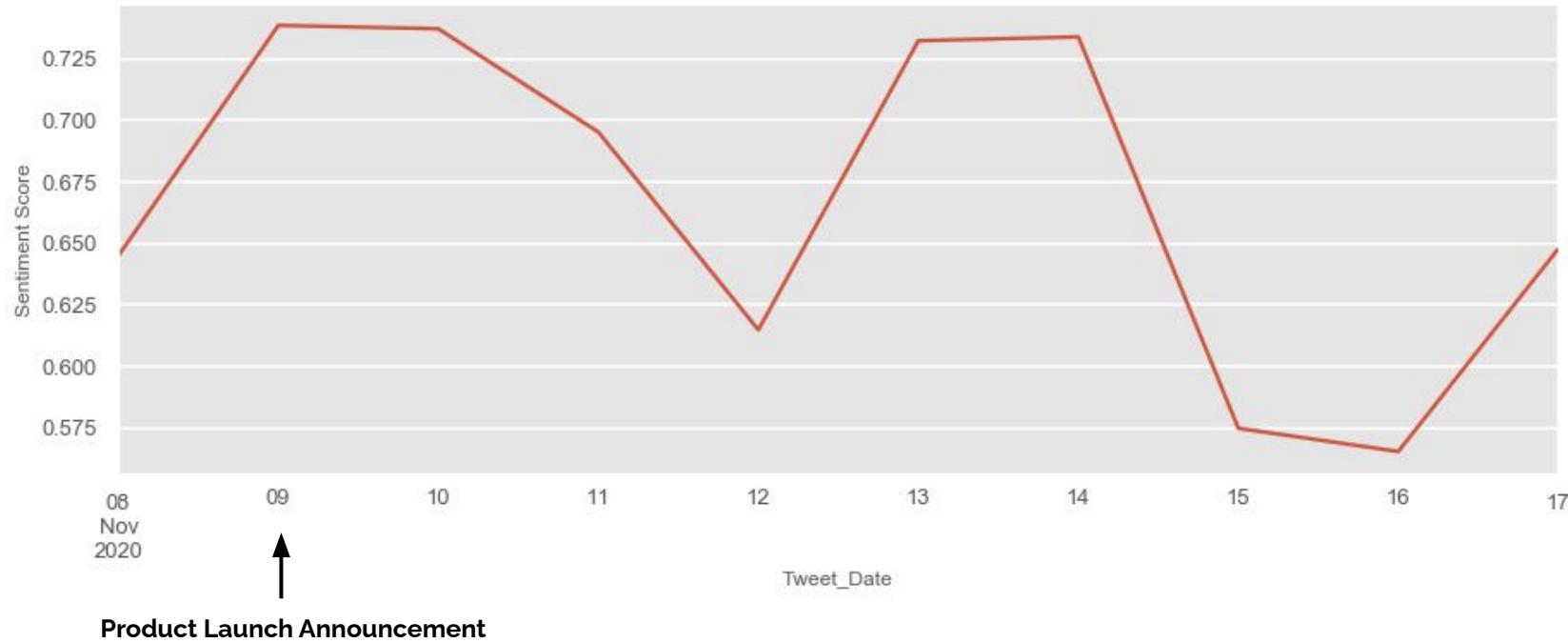


63%

Positive Mentions towards McDonald's

- Positive tweets might predominately wash through after the launch of 'McPlant' on Nov 9th.

McDonald's - Sentiment Over Time



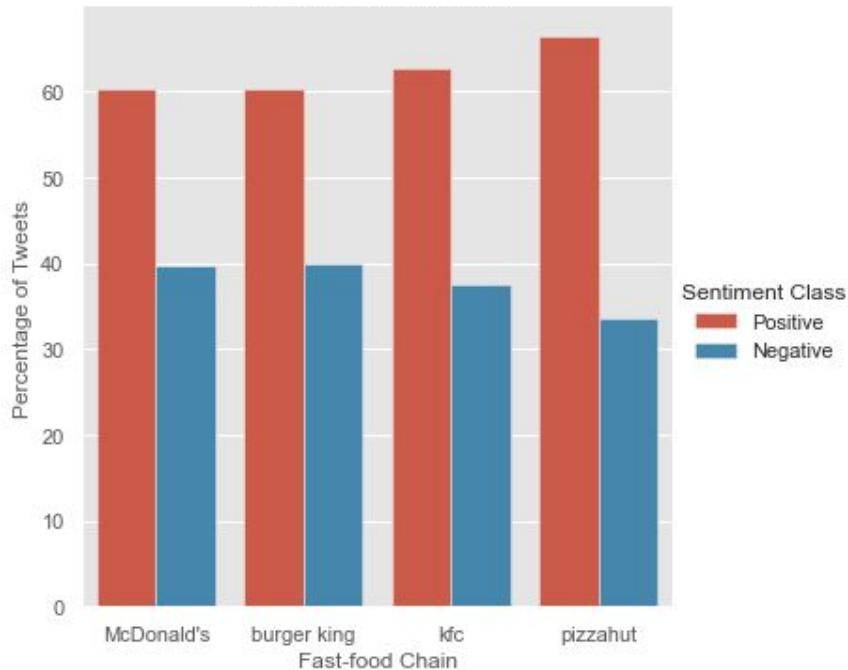


02

Competitive Analysis

McDonald's, KFC, Pizza Hut, Burger King

Competitive Analysis - Sentiment Class Ratio

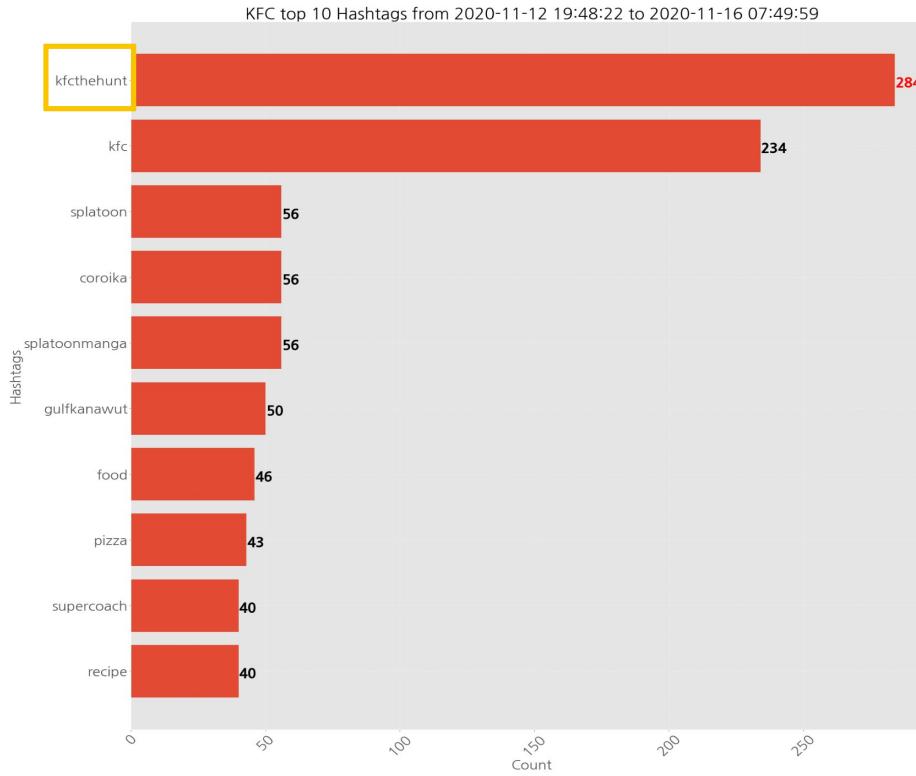


Pizza Hut > KFC > Burger King = McDonald's

Pizza Hut portrays a better brand image with most positive tweets and least negative tweets.

- Purely measuring tweet content about the brand itself without skewing towards their use of veg meat.
- However, the positive perception might be built up with actions/campaigns they launched throughout this window.

Top 10 Hashtag Count - KFC



#KFCTheHunt



KFC Gaming @kfcgaming · Nov 11
Step Eight is done!

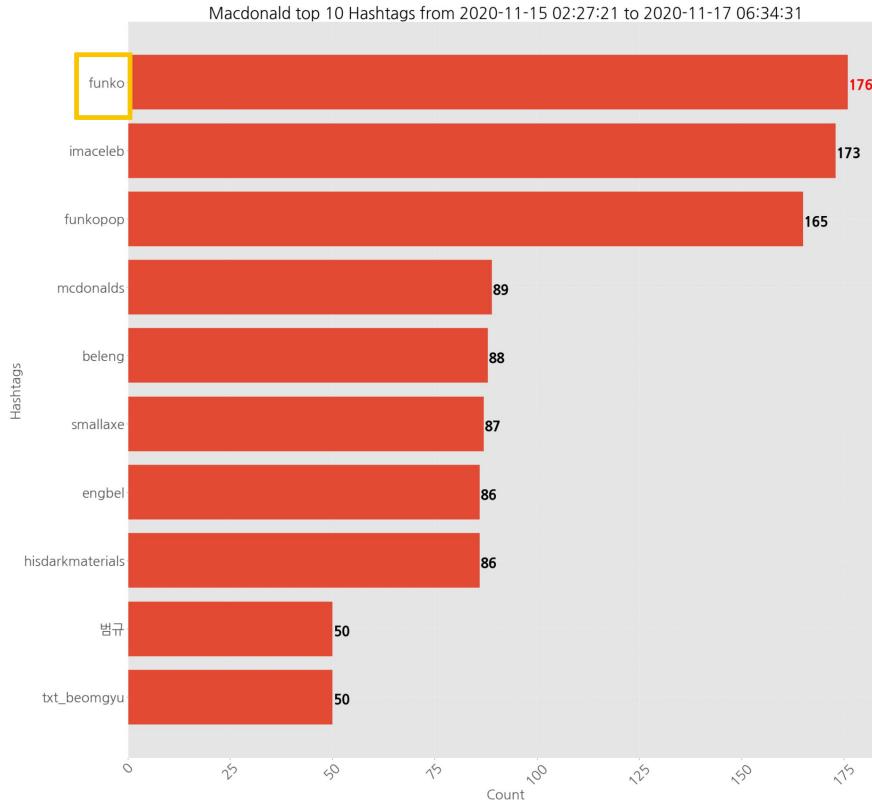
You know the drill. Retweet and like to be in with a chance of winning an Xbox Series X console and a month's supply of KFC!

Step Nine is live... #KFCTheHunt



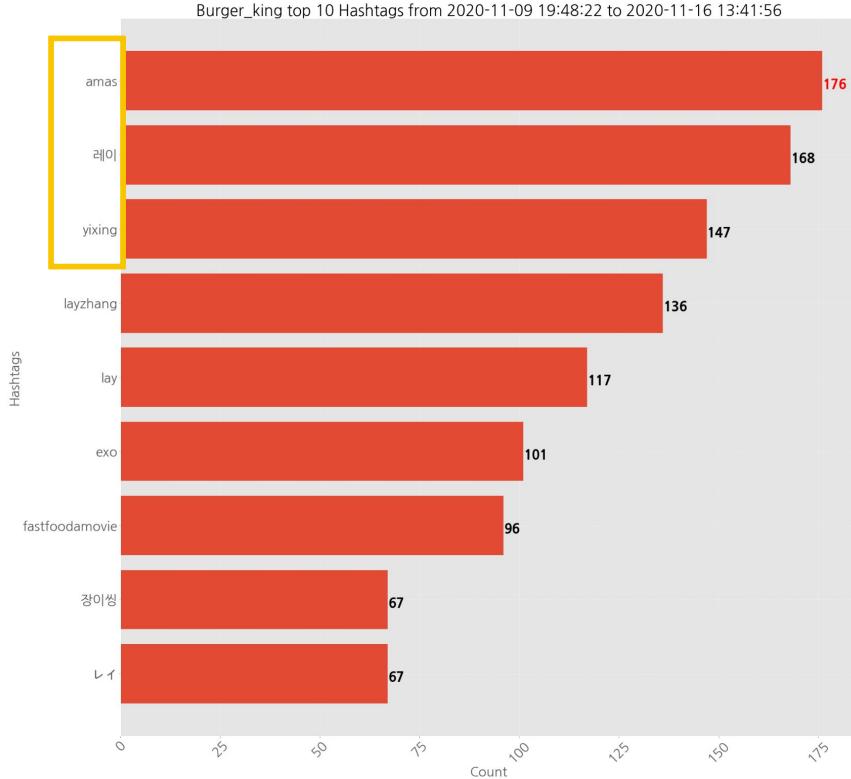
- Giveaway: KFC has teamed up with Xbox to give away its latest Gaming Setup for winners of KFC-themed gaming contest

Top 10 Hashtag Count - McDonald's



- Souvenirs: Release of Funko Pop! figures commemorating McDonald's collection of mascots

Top 10 Hashtag Count - Burger King



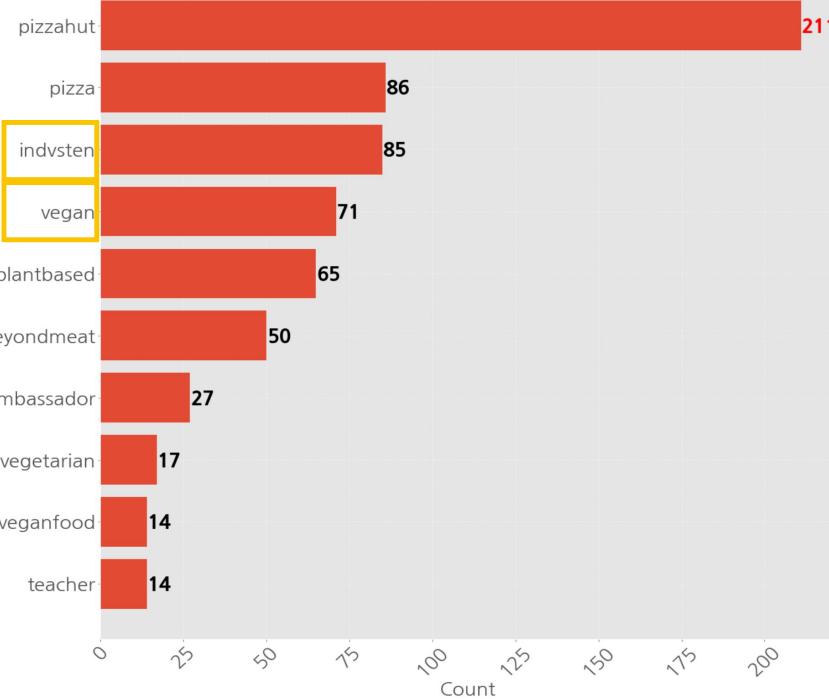
#layzhang



- Celebrities: Announcement of K-Pop idol, Lay Zhang becoming brand ambassador of Burger King

Top 10 Hashtag Count - Pizza Hut

Pizzahut top 10 Hashtags from 2020-11-10 09:33:35 to 2020-11-17 04:00:24



#indvsten #vegan



- Sponsorship: mentioning Pizza Hut's sponsorship on NFL, a professional American Football League



03

Vegan Market Analysis

Beyond Meat, Impossible Foods and more

Vegan Market Trending Tweets (Most-liked)

Negative

1683

@Veganella_ Ohhh wow! But why do you vegans insist on fake meat if you hate it so much!!???

1334

McDonald's and Pizza Hut to offer vegetarian meat substitutes. I bet the cows are partying.

897

@SimonJadis @AITA_reddit It's pretty weird how some meat eaters get personally offended that someone else chooses not to eat meat. I'm not a vegetarian but it in no way hurts my feelings.

891

White veganism is honestly going to be more harmful to the environment than anyone just not going vegan. Cause y'all are offering "solutions" that don't really solve the problem and you're **distracting away from the core of the issue**, which is colonialism, not meat consumption.

866

Plant-based meat is **not a thing**. I will die on this hill.

Positive

3653

McDonald's says it expects to launch a plant-based product line called "McPlant"; Beyond Meat shares plunge more than 6%

3462

McDonald's To Launch 'McPlant' Meat Alternative

2254

It genuinely baffles me how angry vegan meat alternatives make people. Alcohol-free beer, fat-free yogurt, sugar-free sweets, decaf coffee and faux fur don't generate the same level of mockery or dismissal. You don't see people saying: 'JUST DRINK BEER IF YOU WANT IT THAT BADLY.'

1851

I really don't care about pizza huts beyond meat option. Call me when they have **vegan cheese**.

1180

Happiness is Contagious :red_heart_selector: Friends Not Food #govegan #vegan #plantbased #turkey #poultry #farmers #meat #agriculture from life on earth

Vegan Market Trending Tweets (Most-retweeted)

Negative

50277

RT @spoiledsoymilk_: You don't have to be vegan to reduce your meat and **dairy intake**.

23414

RT @sadkxit: asking a gay couple who the man and woman are in their relationship is like asking a vegetarian which vegetable in their salad...

16025

RT @f1uctuate: not being funny but i've never met a vegan/vegetarian who's forced their views on me but i've met plenty of meat eaters who...

9931

RT @greenhouseenyt: Meatpacking companies said their plants needed to stay open, warning that the U.S. faced a severe meat shortage. At lea...

1570

RT @ggreenwald: New study confirms even further what has long been known: a key cause for future pandemics is the animal agriculture/factor...

Positive

26896

RT @__maryboy: Beyond meat implies the existence of bed meat and bath meat

17326

RT @sarah_ogun: Ethnic vegetarians / vegans will just be eating their regular cultural foods— just without the meat, and then white vegetar...

14254

RT @ArianaGrande: I've eaten organically since I was little and always kept meat minimal but today marks my first day as a 100% Vegan!!!! J...

6294

RT @CholesTroll: On #NationalCheeseburgerDay don't let dairy industry make u forget **where the cheese comes from & how**

1180

A vegan said to me “people who sell meat are disgusting”! I said “people who sell fruit & veg are grocer”.

Most-discussed Topics - Vegan Market

01

Plant-based meat
is a **Distraction**
from Core Issue

02

Option of
Vegan Cheese
is a concern

03

Plant-based meat
is **not a thing**

04

Celebration on
McPlant new
Meat Alternative

Vegan Market - Frequency Distribution



What's on trend in
vegan market ?

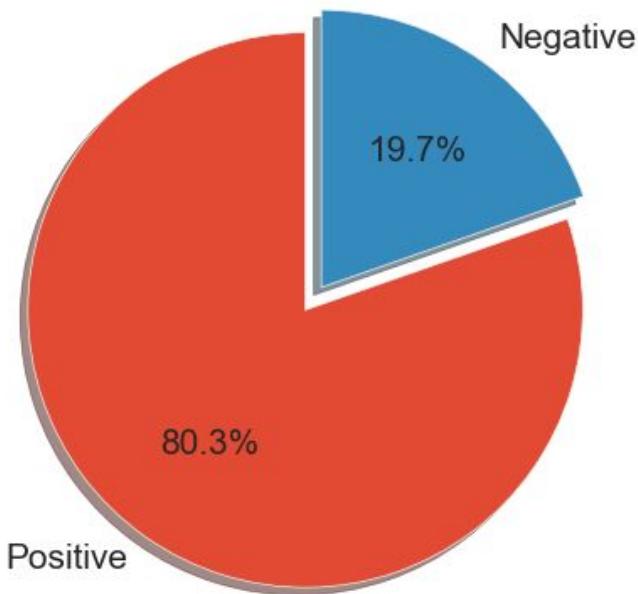
- Beyond Meat
- Beer
- Yogurt
- McPlant
- Sugar
- Fat
- Alcohol

04

McPlant Sentiment Analysis



McPlant - Sentiment Breakdown

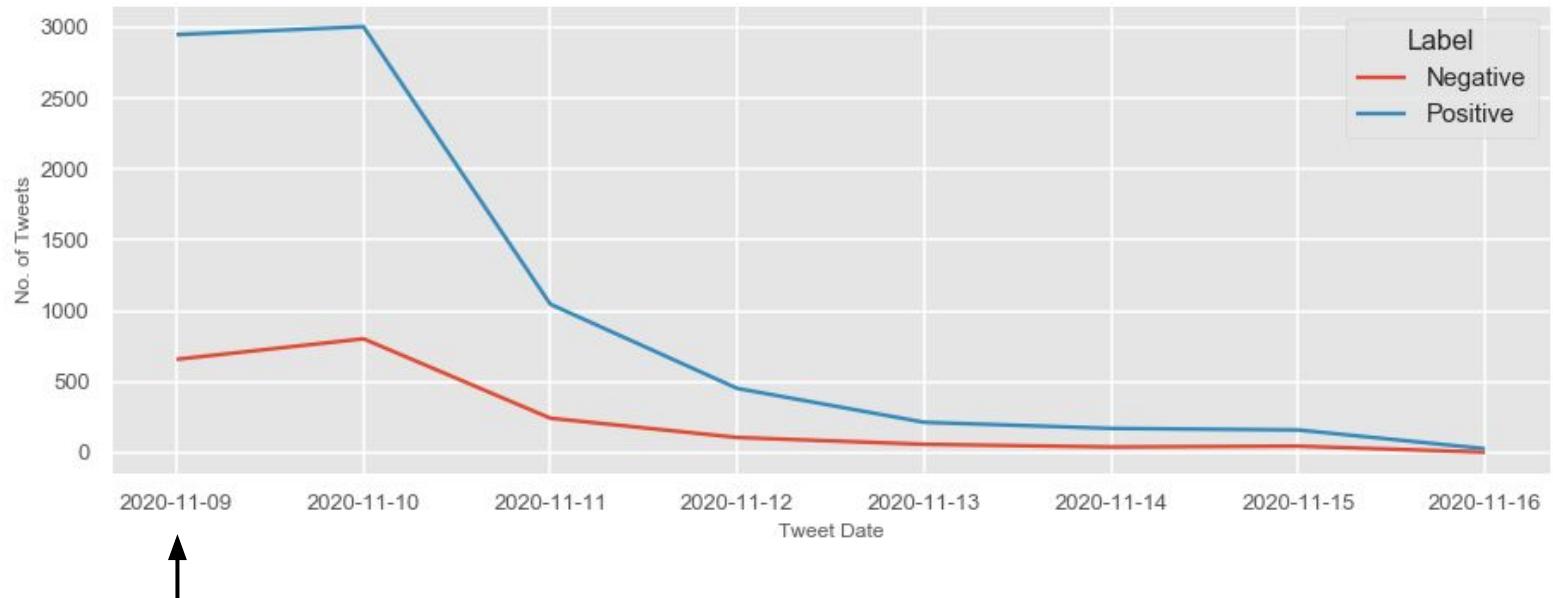


80%

Positive Mentions on McPlant

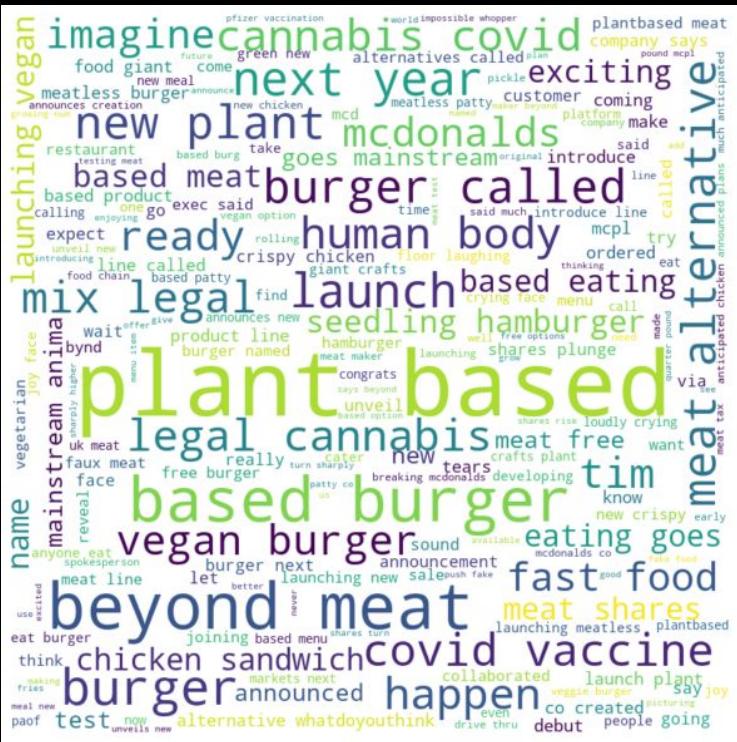
during the week after product
launch announcement on Nov 9,
2020

McPlant - Sentiment Over Time



Product Launch Announcement ↑

McPlant - Frequency Distribution



What's mostly discussed around McPlant ?

- (Legalize) Cannabis
 - COVID-19 vaccine
 - Product Name
 - Chicken Sandwich
 - Goes mainstream
 - Exciting
 - Beyond Meat

McPlant Trending Tweets (Most-liked)

Negative

- | | |
|------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2513 | Is McDonald's really gunna call their veggie burger a Mcplant?!
 |
| 634 | Introducing the 'P.L.T.'
 |
| 512 | thinking about the original mcplant |
| 354 | The McPlant could be with us very soon.
 |
| 233 | i'm surprised mcdonald's didn't already have the McPlant name set aside for the brain implant they'll probably release in a few years |
| 182 | But will there be a rib-less McRib? |
| 142 | Beyond Meat Collapses After Missing Revenue Forecast, Blames COVID |
| 121 | mcplant is a gross name for a food product sorry |

Positive

- | | |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 3461 | McDonald's To Launch 'McPlant' Meat Alternative
#WhatDoYouThink? |
| 1931 | McDonald's announces creation of McPlant plant-based burger, and unveil new Crispy Chicken Sandwich that will debut in 2021.  |
| 1790 | SO EXCITING: @McDonalds is launching a #vegan burger called the #McPlant.  When plant-based eating goes mainstream, animals win.  |
| 929 | Fast food giant McDonald's announced it was launching a new plant-based burger named the "McPlant" to cater to the growing number of people who do not eat meat. |
| 919 | Would anyone eat a burger called McPlant? McDonald's is about to find out. The company says it will test the new plant-based burger in key markets next year. |
| 842 | McDonald's launching meatless 'McPlant' burger |
| 666 | *BREAKING* #McDonalds has announced it will introduce a line of #plantbased meat alternatives called 'McPlant' in 2021. The range is said to include plant-based burgers, chicken substitutes and breakfast sandwiches. No further details yet - we'll let you know as soon as we hear! |
| 574 | Nice work on the McPlant  burger, @McDonalds. We have plants as well. |

McPlant Trending Tweets (Most-Retweeted)

Negative

- 247 RT @NewsyNick: Picturing me in 2021 enjoying my Pfizer vaccination with a quarter-pound McPlant
- 63 RT @MuscleSkoals: thinking about the original mcplant
- 59 RT @FoxBusiness: McDonald's to introduce McPlant and drive-thru-only restaurants
- 56 RT @gordovegan: You know damn well that McPlant won't be vegan. Can't even get vegan fries.
- 43 RT @oneunderscore__: Put the vaccine in the McPlant.
- 29 RT @spillseshYT: Is McDonald's really gunna call their veggie burger a Mcplant?! 💀💀💀
- 26 RT @Mikelsaac: mmmm gimme dat mcplant
- 26 RT @Sappykinz: it's really called mcplant omfg I'm dying

Positive

- 1728 RT @StockCats: imagine what happens to the human body when you mix legal cannabis, a Covid vaccine and a McPlant burger at the same time
- 714 RT @CNBCnow: McDonald's says it expects to launch a plant-based product line called "McPlant"; Beyond Meat shares plunge more than 6%
- 357 RT @CNBCnow: McDonald's says it expects to launch a plant-based product line called "McPlant"; Beyond Meat shares plunge more than 6%
- 355 RT @loganclarkhall: just ordered the mcplant
- 330 RT @TheOnion: McDonald's To Launch 'McPlant' Meat Alternative #WhatDoYouThink?
- 200 RT @businessinsider: McDonald's reveals the McPlant, as the fast-food giant crafts plant-based burgers and 'chicken'
- 163 RT @darth: mcdonalds congratulations on the mcplant burger hey speaking of new ideas mcdonalds
- 131 RT @CNBCnow: Beyond Meat shares turn sharply higher after spokesperson for the company says "Beyond Meat and McDonalds co-created the plant..."

Most-discussed Topics - McPlant

01

The **Naming** of
McPlant is
unsatisfying

02

Celebration on
plant-based
option going
mainstream

03

McPlant is **not**
vegan



05

Conclusion

Key Findings



Sports & Entertainment

Sports & Entertainment capture Mass Favourability and hatch conversations more easily.



100% Vegan

Public concerns towards how fast-food industry process, and produce 100% vegan food

Improvements



Limited API Window

Up to recent 7 days of tweets. Grow more data for analysis with streaming function.



Geographical Location

Not all Twitter users open locations to Twitter.



Models for Labelling

Binary classification limited by the dataset. More models specifically adapted to NLP like BERT can be tested.





06

Code

Walkthrough

Getting Tweets

```
# Build Cursor function to get all the tweets mentioning kfc + vegan meat combination

def GetTweets(brands, number_of_tweets):
    meat_catalogue = ["vegan", "meat free", "vegetarian", "veg", "meatless", "plant based", "veggie"]

    query = []
    for brand in brands:
        query.append(brand)
        for combo in meat_catalogue:
            query.append(brand+" "+combo)

# Creation of query method using parameters with a for Loop
tweets_list = []
for keyword in query:
    tweets = tweepy.Cursor(api.search,q=query,lang='en',tweet_mode='extended').items(number of tweets)
    # Pulling information from tweets iterable object
    # Add or remove tweet information you want in the below List comprehension
    for tweet in tweets:
        tweets_list.append([tweet.full_text, tweet.created_at, tweet.id_str,
                           tweet.user.screen_name, tweet.user.id_str, tweet.user.location, tweet.retweet_count,
                           tweet.favorite_count, tweet.in_reply_to_status_id_str, tweet.in_reply_to_user_id_str,
                           tweet.user.followers_count, tweet.user.friends_count, tweet.coordinates, tweet.place,
                           tweet.entities['hashtags'], keyword])

# Creation of dataframe from tweets_list
# Add or remove columns as you remove tweet information
tweets_df = pd.DataFrame(tweets_list,columns=['Tweet Text', 'Tweet Datetime', 'Tweet Id',
                                                'Twitter @ Name', 'User Id', 'User Location',
                                                'Retweets', 'Favorites', 'Replied Tweet Id',
                                                'Replied Tweet User Id Str', 'User Follower Counts',
                                                'User Following Counts', 'Tweet Coordinates',
                                                'Place Info','Hashtags','Keyword'])

# Checks if there are coordinates attached to tweets, if so extracts them
tweets_df['Tweet Coordinates'] = tweets_df.apply(extract_coordinates,axis=1)

# Checks if there is place information available, if so extracts them
tweets_df['Place Info'] = tweets_df.apply(extract_place,axis=1)

return tweets_df
```

Passing
Keyword Parameters
to API

Calling API

Saving to Dataframe

Converting emojis to text

```
def demoji(text):  
  
    # import Library  
    import emoji  
  
    #Convert Emojis to plain text  
    process_text=emoji.demojize(text)  
    return process_text  
  
def find_emoji(text):  
  
    #find the 'emoji' but in text form  
  
    emoji_text=re.findall(':\w+:',text)  
    return emoji_text
```

Demojize

```
df_text['Tweet Text'][22]
```

```
'#Tuesday Special HOT Offers !!! 🍝\n#Pasta Lover !!!!😊\n\n#PASTA COMBO !!!\nRegular Pasta + 375ML Can for $12.95 PICK UP OR  
DELIVERY\n\nPizza near You at Orange\n\nCONTACTLESS Delivery and Pick Up now available.\n\nOrder Online NOW !\n\n📞 (02) 63 60  
09 06\n\n➡️ https://t.co/m6SPD7kk2j https://t.co/AW83SGbsyJ'
```

```
df_text['demoji_text'][22]
```

```
'#Tuesday Special HOT Offers !!! 🤙\n#Pasta Lover !!!!😋\n\n#PASTA COMBO !!!\nRegular Pasta + 3  
75ML Can for $12.95 PICK UP OR DELIVERY\n\nPizza near You at Orange\n\nCONTACTLESS Delivery and Pick Up now available.\n\nOrder  
Online NOW !\n\n📞 (02) 63 60 09 06\n\n➡️ right\_arrow\_selector: https://t.co/m6SPD7kk2j https://t.co/AW83SGbsyJ'
```

Text Cleaning

```
def cleantext(text):

    text = text.lower().strip() # Convert text to lower case
    text = re.sub(r'@[A-zA-Z0-9]+','',text) # remove @mentions
    text = re.sub(r'RT[\s]+','',text) #remove retweet
    text = re.sub(r'https://[S+]','',text) #remove hyperlink
    text = re.sub(r'#','',text) #remove #

    text = text.replace('\n',' ') #remove \n
    text = contractions.fix(text)

    stemmer = SnowballStemmer(language='english')
    text = stemmer.stem(text)
    text = re.sub("[\.,\!,\!?\!:;-\-=\*\`\`\(\)\`\$\+]","",text) # remove punctuation
    text = re.sub("[0123456789]","",text) # remove number

    text = text.replace(' ','')

    return text.strip()
```

Before

```
df_text['emoji_text'][22]

'#Tuesday Special HOT Offers !!! :raising_hands:\n#Pasta Lover !!!!:face_savoring_food:\n \n#PASTA COMBO !!!\nRegular Pasta + 3
75ML Can for $12.95 PICK UP OR DELIVERY\n\nPizza near You at Orange\n\nCONTACTLESS Delivery and Pick Up now available.\n\nOrder
Online NOW !\n\n:telephone_selector: (02) 63 60 09 06\n:right_arrow_selector: https://t.co/m6SPD7KK2j https://t.co/AW83SGbsyJ'
```

After

```
df['Clean_text'][22]

'tuesday special hot offers raising hands pasta lover face savoring food pasta comboregular pasta + ml can for $ pick up or del
iverypizza near you at orangecontactless delivery and pick up now availableorder online now telephone selector right arrow sele
ctor'
```

Sentiment analysis using LSTM model

```
# Get sentiment score using trained LSTM model
def getpredict(csvpath,tweetcolumn):
    # Load LSTM model
    model = keras.models.load_model("../twitter_sentiment_LSTM_model")

    #read csv
    df = pd.read_csv(csvpath)

    # convert data to numpy
    testdata = df[tweetcolumn].to_numpy()
    with open('tokenizer.pickle', 'rb') as handle:
        tokenizer = pickle.load(handle)
    newinput = pad_sequences(tokenizer.texts_to_sequences(testdata), maxlen = 200)

    #predict score
    scores = model.predict(newinput)
    scoredf = pd.DataFrame(scores,columns= ['Score'])
    finaldf = pd.concat([df,scoredf],axis=1)

    return finaldf

#Converting score to label
def sentiment_class(score):
    if score >= 0.5:
        return "Positive"
    else:
        return "Negative"
```

Load LSTM model

Predict Tweet Sentiment

```
| df_score[['Score','Sentiment Class']].iloc[22]
| Score          0.981893
| Sentiment Class Positive
| Name: 22, dtype: object
```

Example of Cleaned Text

```
df_text['Clean_text'][22]
```

```
'tuesday special hot offers raising hands pasta lover face savoring food pasta comboregular pasta + ml can for $ pick up or del
iverypizza near you at orangecontactless delivery and pick up now availableorder online now telephone selector right arrow sele
ctor'
```

LSTM Model - Data Acquisition

The screenshot shows a Kaggle dataset page for the "Sentiment140 dataset with 1.6 million tweets". The page features a background image of three large smiley faces (green, red, yellow) representing sentiment analysis. At the top left is a "Dataset" icon. On the right, there's a yellow circular profile picture, a small upward arrow icon, and a box containing the number "884". The title "Sentiment140 dataset with 1.6 million tweets" is prominently displayed in white text. Below it, the subtitle "Sentiment analysis with tweets" is visible. The author information shows a user icon and the name "Μαρίος Μιχαηλίδης Kazanova" with a note that it was updated 3 years ago (Version 2). The navigation bar includes tabs for "Data" (which is selected), "Tasks", "Notebooks (153)", "Discussion (10)", "Activity", and "Metadata". To the right of the navigation bar are buttons for "Download (81 MB)" and "New Notebook", along with a vertical ellipsis menu. Below the navigation bar, there are sections for "Usability" (rating 8.8), "License" (Other, specified in description), and "Tags" (internet, online communities, social networks, linguistics, languages).

Dataset

Sentiment140 dataset with 1.6 million tweets

Sentiment analysis with tweets

Μαρίος Μιχαηλίδης Kazanova • updated 3 years ago (Version 2)

Data Tasks Notebooks (153) Discussion (10) Activity Metadata

Download (81 MB) New Notebook :

Usability 8.8 License Other (specified in description)

Tags internet, online communities, social networks, linguistics, languages

^<https://www.kaggle.com/kazanova/sentiment140>

LSTM Model - Data Preprocessing

```
X = newdf['tweet_text'].apply(str)
y = newdf['sentiment']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=20)

print(len(X_train))
print(len(X_test))

1120000
480000

# tokenizer create tokens for every word in the data corpus and map them to a index using dictionary.
# word_index: contains the index for each word
# vocab_size: represents the total number of word in the data corpus

from keras.preprocessing.text import Tokenizer

tokenizer = Tokenizer()
tokenizer.fit_on_texts(X_train)
word_index = tokenizer.word_index

vocab_len = len(set(tokenizer.word_index))+1
vocab_len
```

LSTM Model - Embedding

```
path_to_glove_file = "/content/drive/My Drive/Colab Notebooks/glove.6B.200d.txt"

# load the entire GloVe word embedding file into memory as a dictionary of word to embedding array

embeddings_index = {}
with open(path_to_glove_file) as f:
    for line in f:
        word, coefs = line.split(maxsplit=1)
        coefs = np.fromstring(coefs, "f", sep=" ")
        embeddings_index[word] = coefs


# create a weight matrix for words in training docs
embedding_dim = 200

# Prepare embedding matrix
embedding_matrix = np.zeros((vocab_len, embedding_dim))
for word, i in word_index.items():
    embedding_vector = embeddings_index.get(word)
    if embedding_vector is not None:
        # Words not found in embedding index will be all-zeros
        embedding_matrix[i] = embedding_vector


#define our embedding layers
#we do not want to update the learned word weights in this model
from tensorflow.keras.layers import Embedding

embedding_layer = Embedding(
    vocab_len,
    embedding_dim,
    embeddings_initializer=keras.initializers.Constant(embedding_matrix),
    trainable=False,
)
```

LSTM Model - Training

```
# Embedding Layer - we have an embedding layer instead of loading random weight we will load the weights from our glove embeddings
# Conv1D Layer - Its using to convolve data into smaller feature vectors.
# Bidirectional RNN is to split the state neurons of a regular RNN in a part that is responsible for
# the positive time direction (forward states) and a part for the negative time direction (backward states)
# LSTM - Long Short Term Memory, its a variant of RNN which has memory state cell to learn the context of words
# Dense - Fully Connected Layers for classification

from tensorflow.keras import layers

int_sequences_input = layers.Input(shape=(200,), dtype='int32')
embedding_sequences = embedding_layer(int_sequences_input)
x = layers.Conv1D(128, 5, activation="relu")(embedding_sequences)
x = layers.MaxPooling1D(5)(x)
x = layers.Conv1D(128, 5, activation='relu')(x)
x = layers.Bidirectional(layers.LSTM(64, dropout=0.2, recurrent_dropout=0.2))(x)
x = layers.Dense(128, activation='relu')(x)
x = layers.Dropout(0.5)(x)
preds = layers.Dense(1, activation="sigmoid")(x)
model = tf.keras.Model(int_sequences_input, preds)
model.summary()
```

Layer (type)	Output Shape	Param #
=====		
input_1 (InputLayer)	[(None, 200)]	0
embedding (Embedding)	(None, 200, 200)	64411600
conv1d (Conv1D)	(None, 196, 128)	128128
max_pooling1d (MaxPooling1D)	(None, 39, 128)	0
conv1d_1 (Conv1D)	(None, 35, 128)	82048
bidirectional (Bidirectional)	(None, 128)	98816
dense (Dense)	(None, 128)	16512
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 1)	129
=====		
Total params:	64,737,233	
Trainable params:	325,633	
Non-trainable params:	64,411,600	

LSTM Model - Training

```
from tensorflow.keras.optimizers import Adam

model.compile(optimizer="adam", loss='binary_crossentropy',metrics=['accuracy'])

my_callbacks = tf.keras.callbacks.EarlyStopping(patience=3, monitor = 'val_loss')
history = model.fit(X_train, y_train, batch_size=128, epochs=20, validation_data=(X_test,y_test),callbacks=my_callbacks)
```

```
Epoch 1/20
8750/8750 [=====] - 1128s 129ms/step - loss: 0.4757 - accuracy: 0.7720 - val_loss: 0.4665 - val_accuracy: 0.7771
Epoch 2/20
8750/8750 [=====] - 1119s 128ms/step - loss: 0.4566 - accuracy: 0.7838 - val_loss: 0.4630 - val_accuracy: 0.7787
Epoch 3/20
8750/8750 [=====] - 1115s 127ms/step - loss: 0.4441 - accuracy: 0.7915 - val_loss: 0.4601 - val_accuracy: 0.7825
Epoch 4/20
8750/8750 [=====] - 1109s 127ms/step - loss: 0.4345 - accuracy: 0.7974 - val_loss: 0.4596 - val_accuracy: 0.7834
Epoch 5/20
8750/8750 [=====] - 1106s 126ms/step - loss: 0.4262 - accuracy: 0.8024 - val_loss: 0.4592 - val_accuracy: 0.7827
Epoch 6/20
8750/8750 [=====] - 1109s 127ms/step - loss: 0.4187 - accuracy: 0.8063 - val_loss: 0.4640 - val_accuracy: 0.7805
Epoch 7/20
8750/8750 [=====] - 1108s 127ms/step - loss: 0.4127 - accuracy: 0.8100 - val_loss: 0.4646 - val_accuracy: 0.7806
Epoch 8/20
8750/8750 [=====] - 1103s 126ms/step - loss: 0.4068 - accuracy: 0.8131 - val_loss: 0.4683 - val_accuracy: 0.7793
```