

SENTIMENT ANALYSIS FOR MARKETING

Phase 3

Development part 1

Introduction:

Sentiment Analysis is a use case of [Natural Language Processing \(NLP\)](#) and comes under the category of [text classification](#). To put it simply, Sentiment Analysis involves classifying a text into various sentiments, such as positive or negative, Happy, Sad or Neutral, etc. Thus, the ultimate goal of sentiment analysis is to decipher the underlying mood, emotion, or sentiment of a text. This is also known as **Opinion Mining**.

Loading data:

IN:

```
Import pandas as pd
```

```
Df = pd.read_csv("Tweeter.csv")
```

```
Load the dataset..,
```

```
Print(df.head())
```

OP:

tweet_id	airline_sen...	# airline_sen...	negativere...	# negativere...	airline
570301031407624196	negative	1.0	Bad Flight	0.7033	Virgin America
570300817074462722	negative	1.0	Can't Tell	1.0	Virgin America
570300767074181121	negative	1.0	Can't Tell	0.6842	Virgin America

tweet_id	airline_sen...	# airline_sen...	negativere...	# negativere...	airline
570306133677760513	neutral	1.0			Virgin America
570301130888122368	positive	0.3486		0.0	Virgin America
570301083672813571	neutral	0.6837			Virgin America

tweet_id	airline_sen...	# airline_sen...	negativere...	# negativere...	airline
570300767074181121	negative	1.0	Can't Tell	0.6842	Virgin America
570300616901320704	positive	0.6745		0.0	Virgin America
570300248553349120	neutral	0.634			Virgin America

As the dataset about tweeter in airline sentiment and have 16,424 datas,let check the few data in 6 columns.

Preprocessing dataset:

1.Text Cleaning (or)preprocessing:

Remove special characters, URLs, and other unwanted elements from the text.

IN[]:

Import re

```
Def clean_text(text):
```

```
    Text = re.sub(r'http\S+', '', text) # Remove URLs
```

```
    Text = re.sub(r'[^\A-Za-z0-9]+', '', text) # Remove special characters
```

```
    Text = text.lower() # Convert to lowercase
```

```
    Return text
```

```
Clean the dataset..,
```

```
Cleaned_text = clean_text(text)
```

```
OP[]:
```

tweet_id	airline_sen...	# airline_sen...	negativere...	# negativere...	airline
570306133677760513	neutral	1.0			Virgin America
570301130888122368	positive	0.3486		0.0	Virgin America
570301083672813571	neutral	0.6837			Virgin America

2.Tokenization:

Now we will tokenize all the cleaned tweets in our dataset. Tokens are individual terms or words, and tokenization is the process of splitting a string of text into tokens

```
IN[]:
```

```
From nltk.tokenize import word_tokenize
```

```
Def tokenize_text(text):
```

```
    Tokens = word_tokenize(text)
```

```
    Return tokens
```

Splits the datasets,..

```
tokenized_tweet = combi['tidy_tweet'].apply(lambda x:  
x.split())  
tokenized_tweet.head()
```

Tokens = tokenize_text(cleaned_text)

OP[]:

	airline_sentiment	Text
0	neutral	@VirginAmerica What @dhepburn said.
1	positive	VirginAmerica plus youve added commercials t...
2	neutral	VirginAmerica I didnt today... Must mean I n...
3	negative	VirginAmerica its really aggressive to blast...
4	negative	VirginAmerica and its a really big bad thing...

3.Stop Word Removal:

Remove common stop words like "and", "the", "is", that do not provide significant information.

IN[]:

```
From nltk.corpus import stopwords
```

```
Def remove_stopwords(tokens):
```

```
Stop_words = set(stopwords.words('english'))
```

```
Filtered_tokens = [word for word in tokens if word.lower() not in  
stop_words]
```

```
Return filtered_tokens
```

```
Filtered_tokens = remove_stopwords(tokens)
```

```
OP[]:
```

tweet_id	airline_sen...	# airline_sen...	A negativere...	# negativere...	A airline
570306133677760513	neutral	1.0			Virgin America
570301130888122368	positive	0.3486		0.0	Virgin America
570301083672813571	neutral	0.6837			Virgin America

4.Lemmatization:

Reduce words to their base form (lemmas).Remove prefixes and suffixs.

```
IN[]:
```

```
Import spacy
```

```
Nlp = spacy.load("en_core_web_sm")
```

```
Def lemmatize_text(text):
```

Doc = nlp(text)

Lemmatized_text = ' '.join([token.lemma_ for token in doc])

Return lemmatized_text

Lemmatized_text = lemmatize_text(" ".join(filtered_tokens))

OP[]:

	tweet_id	airline_sentiment	airline_sentiment_confidence	negativereason	negativereason_confidence	airline	airline_sentiment_gold	name	negativereason_gold	retweet_count	text	tweet_coord	tweet_created	tweet_location	user_timezone
0	570306133677760513	neutral	1.0000	NaN	NaN	Virgin America	NaN	cairdin	NaN	0	@VirginAmerica What @dhepburn said.	NaN	2015-02-24 11:35:52 -0800	NaN	Eastern Time (US & Canada)
1	570301130888122368	positive	0.3486	NaN	0.0000	Virgin America	NaN	jnardino	NaN	0	@VirginAmerica plus you've added commercials t...	NaN	2015-02-24 11:15:59 -0800	NaN	Pacific Time (US & Canada)

	tweet_id	airline_sentiment	airline_sentiment_confidence	negative_reason	negative_reason_confidence	airline	airline_sentiment_gold	name	negative_reason_gold	retweet_count	text	tweet_coord	tweet_created	tweet_location	user_timezone
2	570301083672813571	neutral	0.6837	NaN	NaN	Virgin America	NaN	ynonalyon	NaN	0	@Virgin America I didn't today... Must mean I n...	NaN	2015-02-24 11:15:48 -0800	Lets Play	Central Time (US & Canada)
3	570301031407624196	negative	1.0000	Bad Flight	0.7033	Virgin America	NaN	jnardino	NaN	0	@Virgin America it's really aggressive to blast ...	NaN	2015-02-24 11:15:36 -0800	NaN	Pacific Time (US & Canada)
4	570300817074462722	negative	1.0000	Can't Tell	1.0000										

In [2]:

```
train = pd.read_csv('train_E6oV3IV.csv')
test = pd.read_csv('test_tweets_airlines.csv')
```



```
In [18]:  
sent_data.head()
```

```
Out[18]:
```

	airline_sentiment	text
0	neutral	@VirginAmerica What @dhepburn said.
1	positive	@VirginAmerica plus you've added commercials t...
2	neutral	@VirginAmerica I didn't today... Must mean I n...
3	negative	@VirginAmerica it's really aggressive to blast...
4	negative	@VirginAmerica and it's a really big bad thing...

Data preprocessing is a critical step in sentiment analysis as it lays the foundation for building effective sentiment classification models. By transforming raw text data into a clean, structured format, preprocessing helps in extracting meaningful features that reflect the sentiment expressed in the text.