Sentiment Analysis of Tweets about Remote Work

Sherly Hartono

Motivation

- Given the changing landscape of the workplace from office to home, people have different opinions regarding these large scale transitions. Many say this trend would likely continue after the pandemic.
- There will be many challenges that companies will need to overcome in order for remote work to stay. But the question still remains,
- **How do remote employees actually feel about remote work? and what are the challenges that they are facing daily?**
- This insight will help businesses to plan and prioritize their next step in this paradigm shift.

Approach

<table>
<tr><td>Collect Data</td><td>Using Twint</td></tr>
<tr><td>Data Wrangling</td><td>Remove irrelevant tweets
Clean each tweets for Rule-Based and Transformers modelling</td></tr>
<tr><td>Rule Based NLP</td><td>User VADER Package</td></tr>
<tr><td>Transformer: BERT</td><td>Create two types of word clusters
Get Cosine differences between tweets and the two clusters</td></tr>
<tr><td>Topic Modelling</td><td>Choose results from the three models
Clustering on negative tweets
Generate Topics using TF-IDF</td></tr>
</table>

Collecting Data

- Twint: An advanced Twitter scraping & OSINT tool written in Python
- Doesn't use Twitter's API
- Scrape a user's followers, following, Tweets and more while evading most API limitations.
- Data can be saved in json file or converted to a pandas dataframe to store selected information

```python
# Configure
config = twint.Config()

config.Search = "remote work"
config.Lang = "en"
config.Since = "2020-08-01"
config.Until = "2021-05-13"
config.Limit = 10000
config.Pandas = True
config.Filter_retweets = True


# Run
twint.run.Search(config)
```

```
1392630666471297029 2021-05-13 06:59:59 +0700 <RecruiterDotCom> Recruiter .com'
s April 2021 recruiter index® has found that the demand for in-person jobs is o
utpacing that of remote work.   Hit the link below for the full Recruiter Index
®.   https://t.co/vh2wufdTp0 #recruiterindex #recruiters #recruitment #jobmarke
t #labormarket #hiringtrends
1392630316494295042 2021-05-13 06:58:35 +0700 <rucsb> @TimSackett @lruettimann
@FrankZupan @Lars +1. Real Estate Market would crash if there is no demand for
commercial space.   Hybrid work / Remote work works . If we design for it. For
decades, Office space worked as space to socialize with fellow human beings.
1392630178359042054 2021-05-13 06:58:02 +0700 <ArneEkstrom1> Congratulations to
Dr. Michael Starrett on successfully defending his dissertation!  Mike worked o
n everything from immersive VR with a treadmill to remote VR testing!  Great wo
```

Data Wrangling

Remove tweets:

- Remove non english tweets using langdetect()
- Remove duplicate tweets
- Remove job opening and advertisement tweets
- Remove username with ‘remote’
-

Clean tweets:

- Emojis are converted to words using emoji.demojized()
  😎 → *:smiling_face_with_sunglasses:*
- Remove special characters

my_functions.py 4

```python
from __future__ import division
import pandas as pd
import spacy
import string
import re, nltk
import nltk.corpus
from nltk.corpus import stopwords

from collections import Counter

def print_tweet(the_df, end_index):⋯

def print_empty_tweet(the_df):⋯

def delete_empty_tweet(the_df):⋯

def print_sentiment_tweet(the_df, sentiment = 'NEGATIVE'):⋯

# For VADER and BERT: Read learn more
# Remove all characters after the phrase read more, learn more, find ou
def remove_regex(the_regex, the_df):⋯

def remove_read_more(the_df):⋯

# 1. Remove End Hashtag
def remove_end_hashtag(the_df):   ⋯


#2. Lemmatizing
def get_lemmatized_text(the_text):⋯

def lemmatized_df(the_df):⋯

#3. Remove Special Char
def remove_char_from_text(the_text):⋯

def remove_char_vader(the_df):⋯

# OPTIONAL
# 1. Split Hashtags
WORDS = nltk.corpus.brown.words()
COUNTS = Counter(WORDS)

def pdist(counter):⋯
```

```python
def pdist(counter):⋯

P = pdist(COUNTS)

def Pwords(words):⋯

def product(nums):⋯

def splits(text, start=0, L=20):⋯

def segment(text):⋯


def get_capital_letter_index(the_word):⋯

def isAllCapital(the_word):⋯


def get_split_word(the_word):⋯


def split_hashtag(the_df):⋯

# 2. Remove Stopwords
def remove_stopwords(the_df):⋯

# 3. Final cleaning
def final_cleaning(the_df):⋯


# 4. Remove Short tweets
def remove_short_tweets(the_df):⋯



# PIPELINE
def custom_pipeline(the_df,⋯
```

Rule Based Training : VADER

1. CLEANING

- Attuned to sentiments expressed in social media
- Create functions for this to create different pipeline (split and non split hashtags)
- Hashtags are removed but only the ones at the end of the tweets.
- Lemmatizing
- Remove phrases like read more, learn more, find out more, as they usually come with links where to read them after.

Study reveals growing ***#cybersecurity #risks*** *driven by* ***#remotework***

⟶

Study reveals growing ***cyber security risks*** *driven by* ***remote work***

2. SCORE AND PREDICTION

	tweet	original_tweet	VaderScore	VaderSentiment
0	real Estate Market would crash demand commerci...	Real Estate Market would crash if there is no ...	-0.4939	NEGATIVE
1	Concur . company must work office say go . imp...	Concur. My company was 100% "you MUST work in ...	-0.0325	NEGATIVE
2	not ask really need thing ? I think would fair...	Why not ask if we really need that thing? I th...	0.4939	POSITIVE
3	dear Line Managers , Appraisal subordinate bas...	Dear Line Managers, Appraisal your subordinate...	0.3818	POSITIVE
4	I opportunity work cross functionally engage c...	I have had more opportunities to work cross-fu...	0.8225	POSITIVE
...	...	...	...	...

Transformers : BERT

STEP 1: Create negative and word clusters

Method 1

Extract all unique words from our tweets and create their embeddings

	words	embeddings	sentence_idx
0	real	[0.84305966, 0.26398277, 0.32490465, 0.0248823...	0
1	estate	[0.5303595, -0.2668757, 0.75979805, 0.19747669...	0
2	market	[0.28153154, -0.28520963, 0.5332744, 0.2479845...	0
3	would	[-0.30469832, -0.46680853, -0.14788364, 0.0218...	0
4	crash	[-0.24616344, 0.039229684, 0.06317264, 0.44311...	0
...	...	...	...
64894	responsibility	[-0.008578587, 0.1220773, 0.24025224, -0.05287...	2994
64895	staff	[0.31579545, 0.04895662, 0.29348424, -0.167502...	2994
64896	welfare	[0.20462838, 0.31706288, 0.05414169, -0.091104...	2994
64897	cannot	[-0.12302854, -0.030138453, 0.32204738, -0.080...	2994
64898	avoided	[-0.40538037, -0.13855177, -0.17537802, 0.0789...	2994

64899 rows × 3 columns

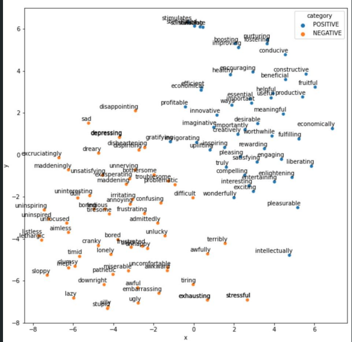
Method 2

Generate our own positive and negative words using gensim and create their embeddings

```
print(dict_clusters["POSITIVE"],'\n')
print(dict_clusters["NEGATIVE"],'\n')

['essential', 'depressing', 'constructive', 'helpful', 'truly', 'creatively', 'healthy', 'boosting', 'invigorating', 'intellectually', 'interesting', 'desirable', 'fostering', 'nurturing', 'exhausting', 'stimulated', 'gratifying', 'encouraging', 'profitable', 'emotionally', 'stimulating', 'efficient', 'fruitful', 'less', 'conducive', 'inspiring', 'uplifting', 'stressful', 'worthwhile', 'useful', 'surprisingly', 'pleasing', 'enormously', 'importantly', 'fulfilling', 'innovative', 'wonderfully', 'economically', 'exciting', 'meaningful', 'important', 'satisfying', 'entertaining', 'compelling', 'beneficial', 'liberating', 'ways', 'stimulate', 'quite', 'enlightening', 'improving', 'enjoyable', 'economical', 'incredibly', 'immensely', 'pleasurable', 'extremely', 'engaging', 'productive', 'imaginative', 'stimulates', 'rewarding']

['dull', 'confusing', 'bored', 'depressing', 'pathetic', 'lazy', 'listless', 'tedious', 'exhausting', 'frustrated', 'unsatisfying', 'bothersome', 'maddeningly', 'unhappy', 'disheartening', 'bit', 'exceedingly', 'lonely', 'stressful', 'exasperating', 'ugly', 'excruciatingly', 'terribly', 'miserable', 'frustrating', 'downright', 'irritating', 'annoying', 'disappointing', 'dreary', 'cranky', 'awful', 'timid', 'silly', 'stupid', 'troublesome', 'sloppy', 'difficult', 'dispiriting', 'unlucky', 'uninteresting', 'tiresome', 'uninspired', 'lethargic', 'boring', 'enjoyable', 'incredibly', 'sad', 'inept', 'extremely', 'problematic', 'tired', 'awkward', 'embarrassing', 'tiring', 'unfocused', 'unnerving', 'clumsy', 'uninspiring', 'uncomfortable', 'admittedly', 'pretty', 'maddening', 'awfully', 'aimless']
```



Transformers : BERT

STEP 2: Create Sentence Embedding

STEP 3: Compute Cosine Distance between sentences and the two average clusters

Average Embedding of word clusters

```
df_own_dict
```

<class 'list'>

	sentiment	embedding
0	positive_own	[-0.23684903980000002, 0.11508918550000001, 0....
1	negative_own	[-0.45195508, 0.0403723456, 0.5418859124000001...

```
df_data_dict
```

	sentiment	embedding
0	positive_data	[-0.6036096215, 0.11336755750000001, 0.6376053...
1	negative_data	[-0.7820361257, 0.20127013330000001, 0.5900069...

Distance between

Sentence Embedding

```
]: df_sentence.head(10)
```

	tweet	tweet_embedding
0	Real Estate Market would crash if there is no ...	[0.2257416844, -0.0169709176, 0.6545215249, -1...
1	Concur. My company was 100% "you MUST work in ...	[-0.3543173969, -0.3487285674, 0.2804761231000...
2	Why not ask if we really need that thing ? I t...	[-0.733558774, -0.3607640862, 0.1692087054, -0...
3	Dear Line Managers Appraisal your subordinate ...	[-0.6378751993, 0.2945272923, 0.22799032930000...
4	I have had more opportunities to work cross-fu...	[-0.2694126964, -0.3058058321, 0.7276366353, -...
5	Study reveals growing cybersecurity risks driv...	[-1.0207954645, -0.6177105904, 0.7027013302, -...
6	As a remote employee you may be tempted to che...	[-0.6545557976, 0.033209234500000004, 0.194388...
7	I am lucky mine is moving to a hybrid model. S...	[-0.6591749787, -0.9909458756, 0.4506825209, -...
8	Shifting to a #remotework environment created ...	[-0.4876919985, -0.2119547725, 0.6156088710000...
9	professionals from a range of industries who n...	[-0.7931020856000001, -0.4555655718, 0.8284254...

Transformers : BERT

STEP 5: Get results of prediction by VADER and two clustering results

	tweet	tweet_embedding	data_predicted	own_predicted	VaderSentiment
0	Concur. My company was 100% "you MUST work in ...	[-0.3543173969, -0.3487285674, 0.2804761231000...	POSITIVE	NEGATIVE	NEGATIVE
1	So weird how we all decided to never replace t...	[0.3376812041, -0.1152411997, 0.530261457, 0.3...	POSITIVE	NEGATIVE	NEGATIVE
2	Glad you exposed me to this. What bizarre argu...	[-0.5353716612, 0.3261777461, 0.2838433087, -0...	POSITIVE	NEGATIVE	NEGATIVE
3	Any piece that tries to say why we should ditc...	[-0.2667962313, 0.4768332839, 0.48676985500000...	POSITIVE	NEGATIVE	NEGATIVE
4	So are you telling me I should leave my eight-...	[-0.8893477321000001, -0.361951381, -0.2480771...	NEGATIVE	NEGATIVE	NEGATIVE
...	...	...	...	...	...
128	Yeah. A lot of boundaries have dissolved with ...	[0.06808185580000001, -0.4628287554, 0.2636246...	NEGATIVE	NEGATIVE	NEGATIVE
129	Getting very tired of the corporate-sponsored ...	[-0.21660083530000002, -0.0097534209, 0.465162...	NEGATIVE	NEGATIVE	NEGATIVE
130	Counter #2 Most people I know loathe their cow...	[-0.2670706809, 0.0491347052, 0.39799004790000...	NEGATIVE	NEGATIVE	NEGATIVE
131	its funny bc the argument against remote work ...	[-0.0675944909, 0.10974104700000001, 0.4871161...	POSITIVE	NEGATIVE	NEGATIVE
132	The announcing and act of departing a zoom cal...	[-0.6881164908, -0.2324504852, 0.3338147998, -...	NEGATIVE	NEGATIVE	NEGATIVE

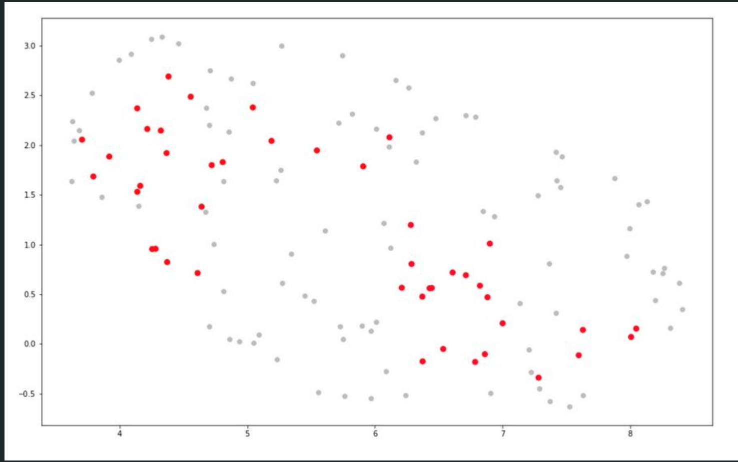
TOPIC MODELLING

STEP 1: Dimensionality reduction to our negative tweets

STEP2: Cluster our negative tweets into topics

The intuition behind the method is as follows. When you apply TF-IDF as usual on a set of documents, what you are basically doing is comparing the importance of words between documents.

What if, we instead treat all documents in a single category (e.g., a cluster) as a single document and then apply TF-IDF? The result would be a very long document per category and the resulting TF-IDF score would demonstrate the important words in a topic.



TOPIC MODELLING

STEP 3: TF-IDF SCORE

Now, we have a single **importance** value for each word in a cluster which can be used to create the topic. If we take the top 10 most important words in each cluster, then we would get a good representation of a cluster, and thereby a topic.

```
0: [('workers', 0.038508393381364706),
 ('communication', 0.03385479321030039),
 ('person', 0.031286203277695206),
 ('article', 0.031286203277695206),
 ('disagree', 0.031286203277695206),
 ('pandemic', 0.03080671470509176),
 ('companies', 0.02929385014096119),
 ('culture', 0.02929385014096119),
 ('company', 0.0262896337425575),
 ('arguing', 0.024983344926558383),
 ('documentation', 0.024983344926558383),
 ('forced', 0.024983344926558383),
 ('concept', 0.024983344926558383),
 ('propaganda', 0.024983344926558383),
 ('shows', 0.024983344926558383),
 ('ppl', 0.024983344926558383),
 ('corporate', 0.024983344926558383),
 ('employee', 0.02256986214020026),
 ('fear', 0.02256986214020026),
 ('colleagues', 0.02256986214020026)],
```

```
1: [('home', 0.04191234731448262),
 ('going', 0.034411755787602506),
 ('really', 0.03132381892730256),
 ('day', 0.028650261179633346),
 ('email', 0.027957883780068058),
 ('office', 0.027113120920628328),
 ('lot', 0.02651546654705727),
 ('life', 0.026177483104688724),
 ('year', 0.023492864195476915),
 ('time', 0.023215946033186635),
 ('energy', 0.022325542274796852),
 ('use', 0.022325542274796852),
 ('likely', 0.022325542274796852),
 ('spent', 0.022325542274796852),
 ('turning', 0.022325542274796852),
 ('hired', 0.022325542274796852),
 ('mother', 0.022325542274796852),
 ('chance', 0.022325542274796852),
 ('slack', 0.02016881297634917),
 ('sick', 0.02016881297634917)]}
```

STEP 4: Evaluation

The topic on the left seems to be about disagreement between workers/ employees and companies as predicted. The keywords on the right are not that obvious. But we know from manual evaluation that there are concerns about cybersecurity and isolation.

FUTURE WORK

Due to CPU constraints, I am not able to fully use the 100,000 tweets that I extracted which could be the reason why our K-Means clustering is not doing so well.

Another method that I would like to try is to train the DistilBERT model on labelled data, that is tweets that have already been labelled negative, positive, and neutral but on other topics, then use the trained model to predict my tweets about remote work.

It would also be more insightful to include texts from other sources such as news portals, Quora, and reddit which have longer text and thus will give us a better topic prediction.