IS434 Social Analytics and Applications (SMU-X)
AY2019/2020 Term 1 – G2

Project Final Report

Date of Submission: 12 November 2019

Group 10

Group Members:

Tan Wei Long Ryan
Ong De Lin
Lee Jia Ern Janell
Lau Jun Rong
Benedict Then Ji Xiang

Contents

1.0 Background

The client we have chosen to work with is Daimler Fleet Management (DFM).

Daimler Fleet Management

DFM is a pioneer in the development of innovative mobility services and is a division of the Daimler Group. DFM offers independent multi-brand leasing and fleet management solutions while still benefiting from being part of the Daimler Group, which has close affiliation with Mercedes-Benz Passenger Cars, Vans and Trucks. Their current car leasing fleet ranges across a broad spectrum which encompasses a range of different car sizes and price levels.

Due to existing confidentiality clauses on their customer database, DFM was unable to provide us with any datasets to work with. However, DFM made sure to equip us with comprehensive knowledge on their business model as well as knowledge on the social media platforms that they focus most of their marketing efforts on i.e. LinkedIn and Facebook. We also identified that DFM was still using traditional print media such as flyers and magazines for their generalized advertising methods.

2.0 Problem Statement

Currently, DFM **spends a large proportion of their yearly budget on marketing efforts** for their car leasing/rental services. However, the rewards from their marketing campaigns and initiatives are minimal. This is largely attributed to the fact that their **marketing efforts are too generalized** which leads to a **mismatch to what they are advertising and the interest of prospective customers**.

3.0 Project Objectives

We aim to help DFM minimize marketing costs as well as increase their probability of successful customer acquisitions for their car renting/leasing services through:

1. Identifying prospective customers who are likely to be interested in DFM’s services from social media data
2. Improve DFM’s current offerings (increasing/refining their current fleet of cars to be more aligned to current market trends)

4.0 Proposed Approach

Based on the resources provided, our group devised a suitable framework that will help us meet our project objectives. Our approach is a top down approach comprising of 4 steps, namely, **Search, Heuristics, Identify and Target** *(Refer to Figure 1).*

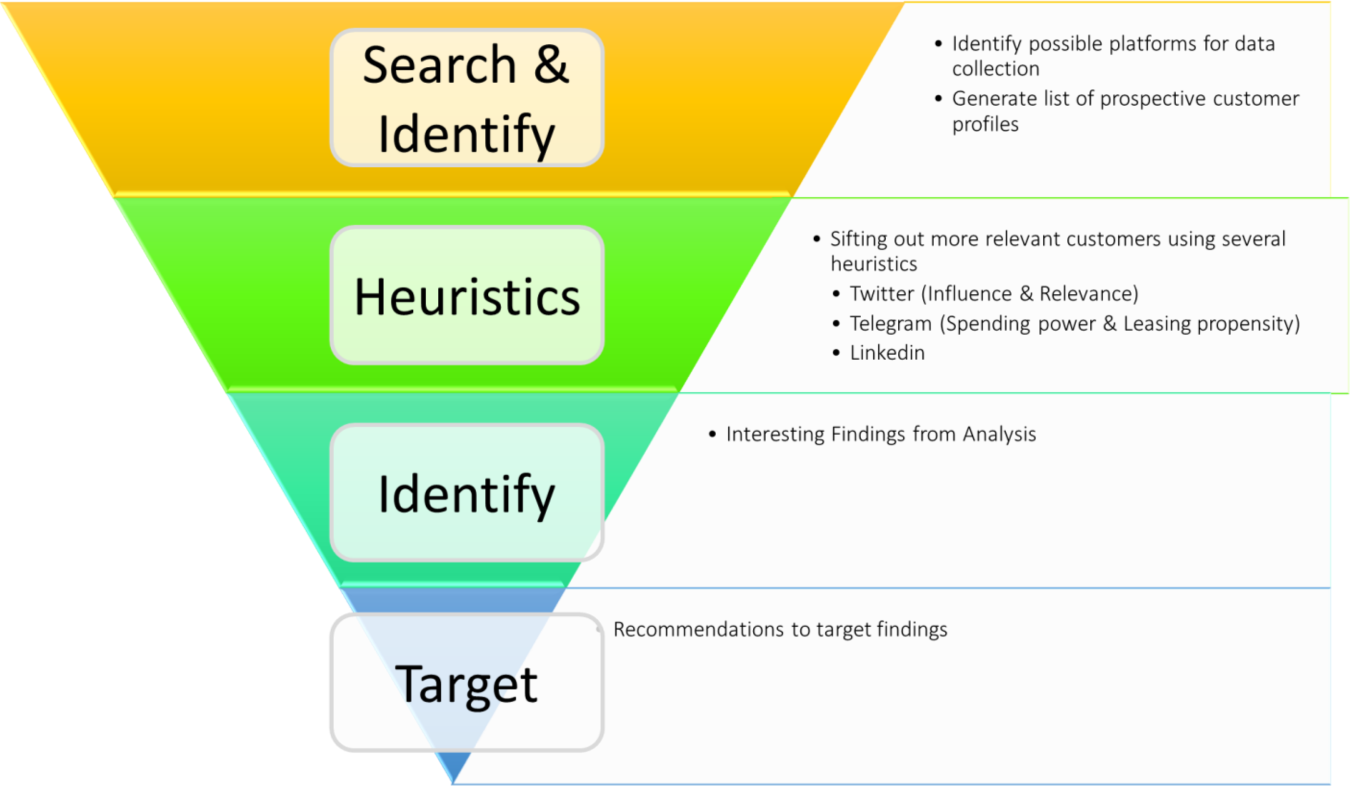

Search & Identify
- Identify possible platforms for data collection
- Generate list of prospective customer profiles
Heuristics
- Sifting out more relevant customers using several heuristics
- Twitter (Influence & Relevance)
- Telegram (Spending power & Leasing propensity)
- Linkedin
Identify
- Interesting Findings from Analysis
Target
Recommendations to target findings

Figure 1: S.H.I.T APPROACH

In order to kick start our analysis, our group must first search and identify social media platforms where data on prospective car leasing customers can be obtained from. We scoured an array of platforms including social networking sites, forums, blogs and review sites to see what kind of information can be obtained from these platforms. Next, we decided to narrow the scope of our data collection to three platforms, namely Twitter, LinkedIn and Telegram. [Search]

After collecting the data from our search, we were able to obtain a large list of prospective customers for DFM. From this list, we had to sift out the users that were most likely to be useful leads for DFM to reach out to. After which, these users were ranked based on our predefined heuristics that differed among the 3 platforms. [Heuristics]

From the top few users that were sifted out after the heuristics stage, we identified various interesting findings and trends. We looked into the individual profiles of the prospective customers to identify patterns which can give insights that DFM can leverage on to solve their business problem. [Identify]

Based on the insights that we obtained after our analysis, we will go further in depth by suggesting plausible recommendations to DFM which can help them to tap upon this market potential. [Target]

4.1. Twitter

Twitter is a social networking service that is based on user interactions that focus on microblogging using messages known as "tweets". Thus, it was one of the platforms identified by our group as it allows us to identify global trends as well as to help us map out the social network of users.

4.1.1. Topic Modelling

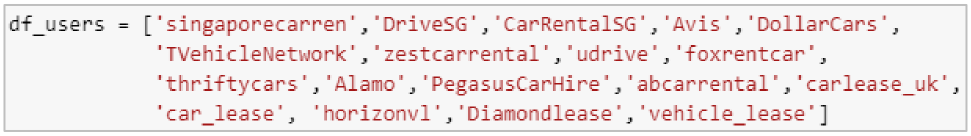
Topic Modelling is essentially a method for finding common topics from a collection of documents that best represents the information in the collection.

In order to gain an understanding of the car leasing industry and the different types of customer profiles, we decided to **leverage on topic modelling to generate key leads.** In addition, this will help us identify potential influencers for DFM to reach out to.

Due to the nature of tweets, traditional algorithms like Latent Dirichlet Allocation (LDA) will perform poorly. This is due to the short text limit, by the nature of the social media platform as well as the 140-character limit imposed on tweets, as well as the tendency for tweets to have 1 topic per tweet. Hence, to perform the topic modelling, we used a library that implemented the “Gibbs sampling algorithm for a Dirichlet Mixture Model”. This approach is optimized for short text topic modelling, and was first proposed by Yin and Wang in 2014 [1] of which we found the implementation of the algorithm on github [2].

4.1.1.1. Collecting

From Twitter, we crawled the recent tweets from accounts of various popular car leasing/rental company Twitter accounts using Tweepy (*Refer to Figure 2*).



```
df_users = ['singaporecarren','DriveSG','CarRentalSG','Avis','DollarCars',
            'TVehicleNetwork','zestcarrental','udrive','foxrentcar',
            'thriftycars','Alamo','PegasusCarHire','abcarrental','carlease_uk',
            'car_lease', 'horizonvl','Diamondlease','vehicle_lease']
```

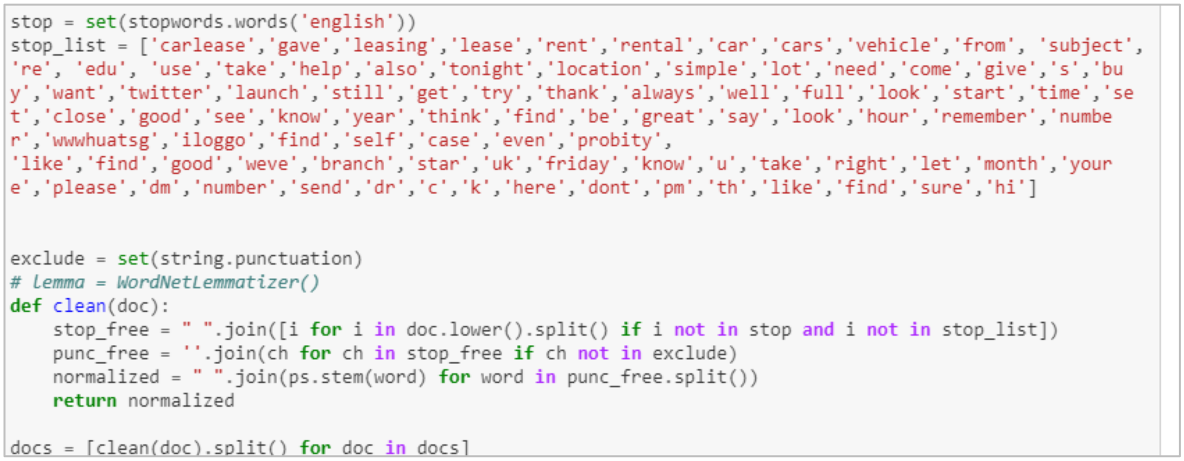
Figure 2: Twitter accounts of Car Leasing/Rental Companies

4.1.1.2. Cleaning

As we are only interested in the words used by the Tweets from various car leasing/rental companies for our topic modelling, we removed the users’ name, Twitter mentions i.e. @twitterhandle and ‘RT’ for this dataset from Twitter. Links were also removed, retaining only characters and spaces. After that, we removed blanks and dropped duplicates. We also had to remove stop words, punctuations and any other irrelevant words that we have defined ourselves. Lastly, the words were stemmed to reduce the words to a common base form.

In summary, these were the actions we took for each tweet that was crawled.

1. Removed the user's name
2. Removed Twitter mentions (I.e. @twitterhandle and "RT")
3. Removed links, retaining only characters and spaces in the text
4. Removed blanks and dropped duplicates
5. Removed stop words, punctuations and irrelevant words in our defined list (R*efer to Figure 3*)
6. Stemmed the words



```
stop = set(stopwords.words('english'))
stop_list = ['carlease','gave','leasing','lease','rent','rental','car','cars','vehicle','from', 'subject',
're', 'edu', 'use','take','help','also','tonight','location','simple','lot','need','come','give','s','bu
y','want','twitter','launch','still','get','try','thank','always','well','full','look','start','time','se
t','close','good','see','know','year','think','find','be','great','say','look','hour','remember','numbe
r','wwwhuatsg','iloggo','find','self','case','even','probity',
'like','find','good','weve','branch','star','uk','friday','know','u','take','right','let','month','your
e','please','dm','number','send','dr','c','k','here','dont','pm','th','like','find','sure','hi']


exclude = set(string.punctuation)
# Lemma = WordNetLemmatizer()
def clean(doc):
    stop_free = " ".join([i for i in doc.lower().split() if i not in stop and i not in stop_list])
    punc_free = ''.join(ch for ch in stop_free if ch not in exclude)
    normalized = " ".join(ps.stem(word) for word in punc_free.split())
    return normalized

docs = [clean(doc).split() for doc in docs]
```

Figure 3: List containing words that were deemed to be irrelevant to the analysis

4.1.1.3. Analysis

From these tweets, we generated a list of keywords which were then grouped into different meaningful topics under a common theme after multiple iterations of the topic modelling algorithm (*Refer to Figures 4, 5, 6 and 7*). **Each of these topics that were generated represent a prospective customer profile with defining characteristics**.

For example, the topic in the group shown in Figure 3 contains keywords like "money", "deal" and "save" which can indicate that the customers that fall within this profile are price sensitive.

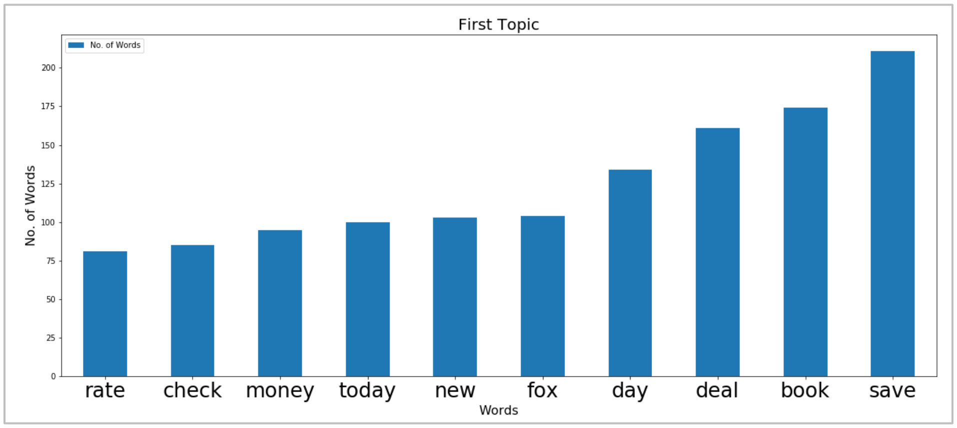


Figure 4: Example of a topic returned with keywords suggesting customers who are price sensitive

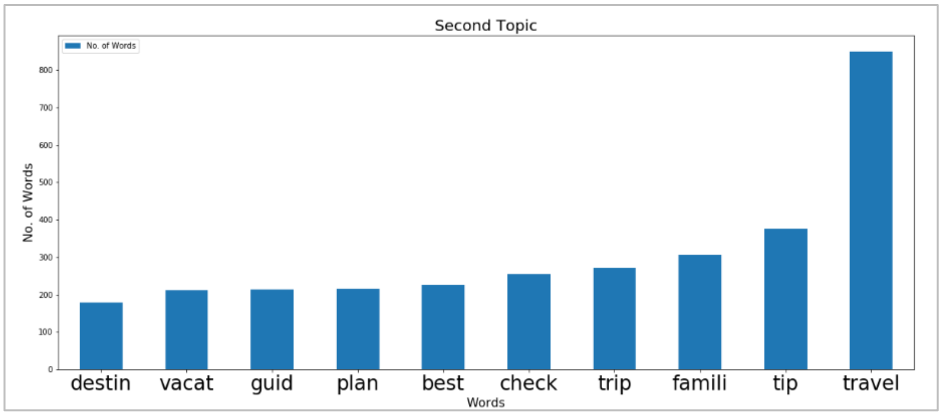


Figure 5: Example of a topic returned with keywords suggesting customers that likes to travel with their family

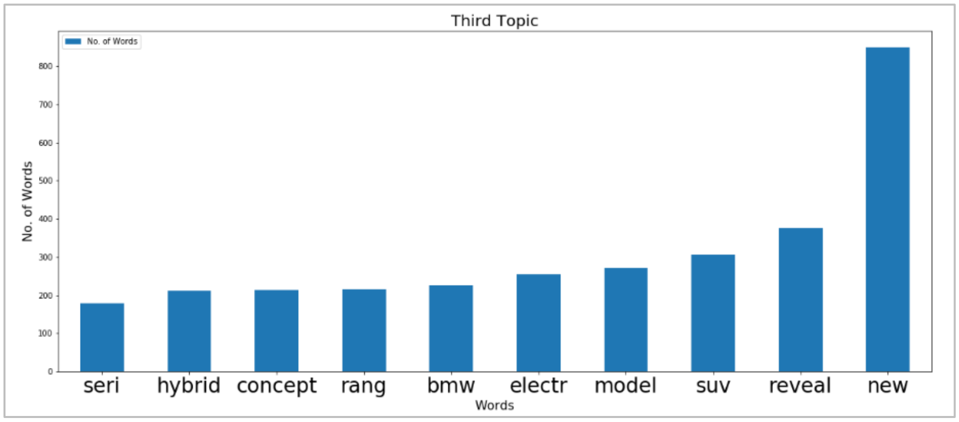


Figure 6: Example of a topic returned with keywords suggesting customers that prefer greener car models such as hybrid and electric cars.

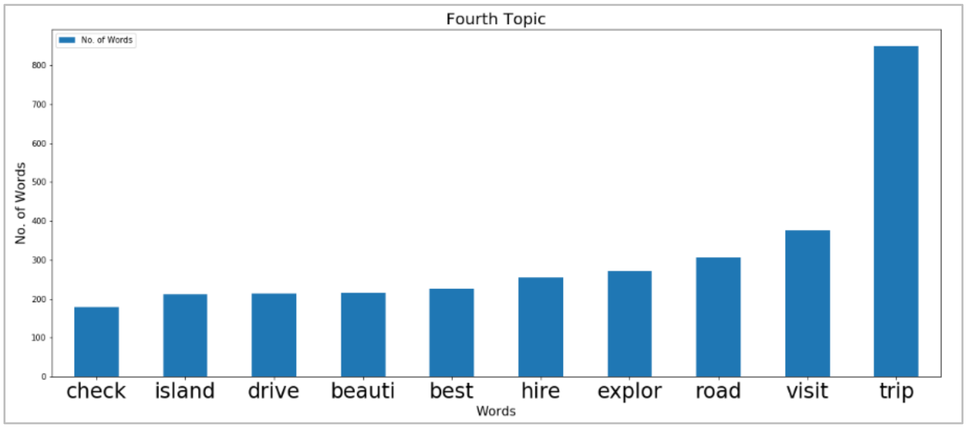


Figure 7: Example of a topic returned with keywords suggesting customers that like to go to road trips

4.1.1.4 Interesting Finding Insights

Increasing trend in Electric Vehicles

One of the most interesting findings we have obtained from topic modelling is the cluster that contains words such as “Electric”, “New”, “Hybrid”, “Model” etc *(Refer to Figure 8)*. This tells us that there is some **buzz and discussions going around on Twitter regarding Electric/Hybrid vehicles**.

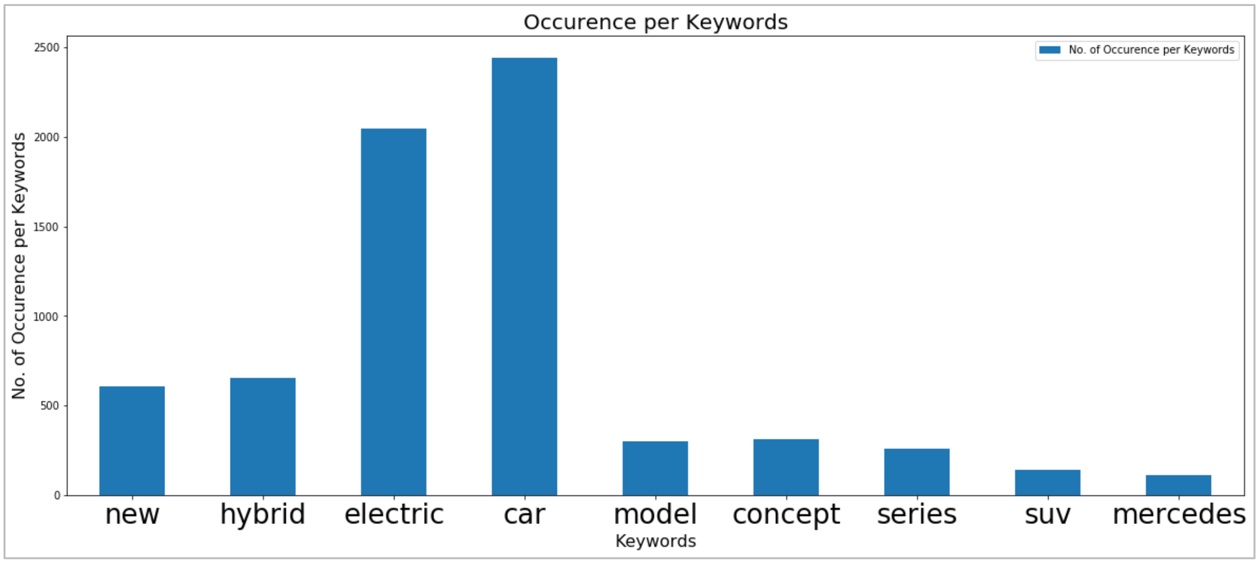
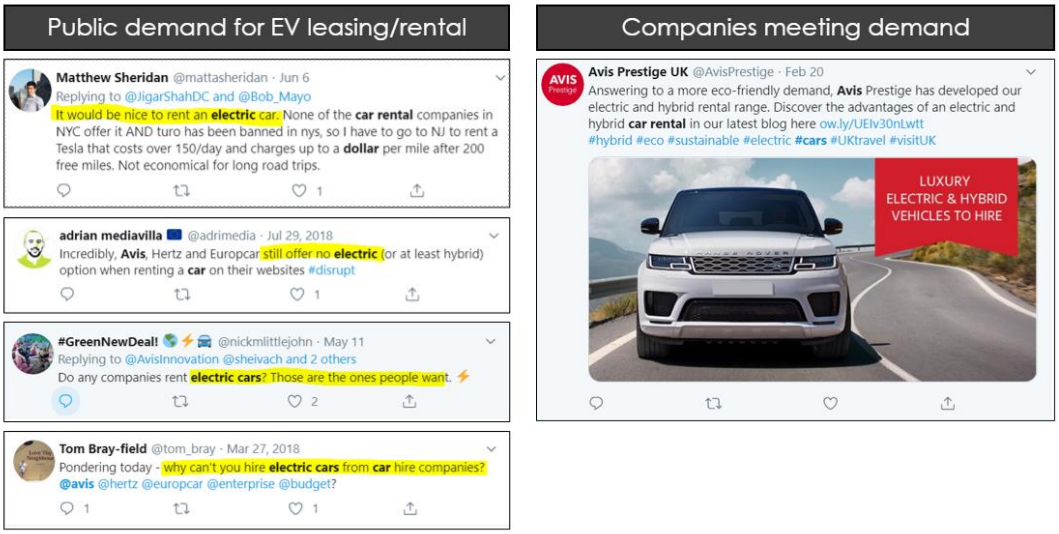


Figure 8: Example of a topic returned with keywords suggesting customers that like cars that run on clean energy

With that, we followed up by researching on the tweets of the various car leasing companies’ Twitter account that we used previously for our topic modeling and found some substantial evidence that shows that there is indeed a demand for electric and hybrid vehicles from the public.

For example, in 2018, a leading car leasing company, Avis was receiving feedbacks, demands and even critics/sarcasm from the public for not having electric and hybrid vehicle leasing options. *(Refer to Figure 9)*

Moving forward in 2019, Avis posted a tweet fulfilling the public demands of having electric vehicle options.



Public demand for EV leasing/rental

Matthew Sheridan @mattasheridan · Jun 6
Replying to @JigarShahDC and @Bob_Mayo
It would be nice to rent an **electric** car. None of the **car rental** companies in NYC offer it AND turo has been banned in nys, so I have to go to NJ to rent a Tesla that costs over 150/day and charges up to a **dollar** per mile after 200 free miles. Not economical for long road trips.
1

adrian mediavilla 🇪🇺 @adrimedia · Jul 29, 2018
Incredibly, **Avis**, Hertz and Europcar still offer no **electric** (or at least hybrid) option when renting a **car** on their websites #disrupt
1

#GreenNewDeal! 🌎⚡🚘 @nickmlittlejohn · May 11
Replying to @AvisInnovation @sheivach and 2 others
Do any companies rent **electric cars**? Those are the ones people want. ⚡
2

Tom Bray-field @tom_bray · Mar 27, 2018
Pondering today - why can't you hire **electric cars** from **car** hire companies? **@avis** @hertz @europcar @enterprise @budget?
1 1

Companies meeting demand

Avis Prestige UK @AvisPrestige · Feb 20
Answering to a more eco-friendly demand, **Avis** Prestige has developed our electric and hybrid rental range. Discover the advantages of an electric and hybrid **car rental** in our latest blog here ow.ly/UEIv30nLwtt
#hybrid #eco #sustainable #electric **#cars** #UKtravel #visitUK

Figure 9: Illustrations of consumers tweeting about wanting electric vehicles and some company's response

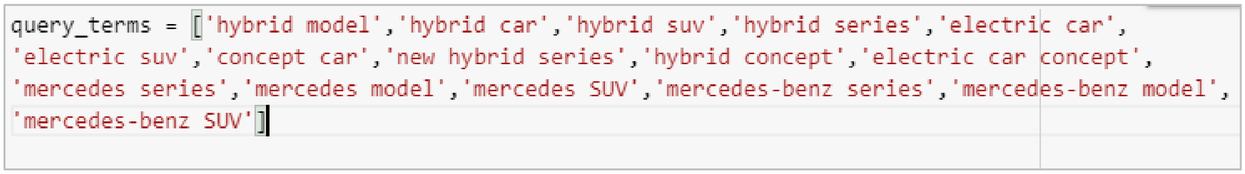
In addition, large automobile manufacturers such as Tesla, Volkswagen and BMW are already joining in the bandwagon of building a portfolio of electric vehicles. This tells us that there is an increasing trend in electric and hybrid vehicles and DFM can consider including electric and hybrid vehicles into their car leasing fleet to improve and expand on their current offerings so that it is more aligned with the current trends in the market.

4.1.2. Identification of Influencers

After gaining an understanding of the car leasing industry and the different types of customer profiles, we decided to leverage on the key leads generated from topic modelling to obtain the list of users who posted tweets containing the keywords in the different profiles.

4.1.2.1. Collecting

From Twitter, we crawled the recent tweets that contained the keywords in our selected profile using Tweepy (*Refer to Figure 10 for search terms used*).



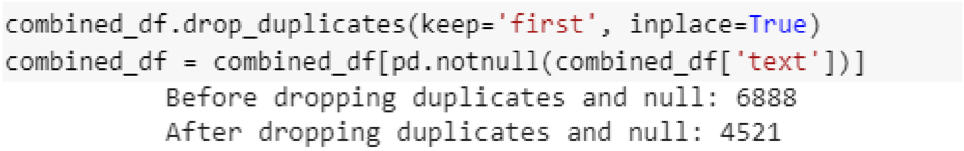
```
query_terms = ['hybrid model','hybrid car','hybrid suv','hybrid series','electric car',
'electric suv','concept car','new hybrid series','hybrid concept','electric car concept',
'mercedes series','mercedes model','mercedes SUV','mercedes-benz series','mercedes-benz model',
'mercedes-benz SUV']
```

Figure 10: Query terms for selected topic (environmentally friendly) used when using Tweepy

4.1.2.2. Cleaning

4.1.2.2.1. Dropping Duplicates & Records with Missing Tweets

Before we start conducting text cleaning, we dropped duplicates and records with missing tweets. This is to ensure that our dataset is filled with unique values before we conduct our analysis. We are left with 4,521 records from the initial 6,888. (*Refer to Figure 11)*



```
combined_df.drop_duplicates(keep='first', inplace=True)
combined_df = combined_df[pd.notnull(combined_df['text'])]
            Before dropping duplicates and null: 6888
            After dropping duplicates and null: 4521
```

Figure 11: Dropping of duplicated records and records with missing tweets

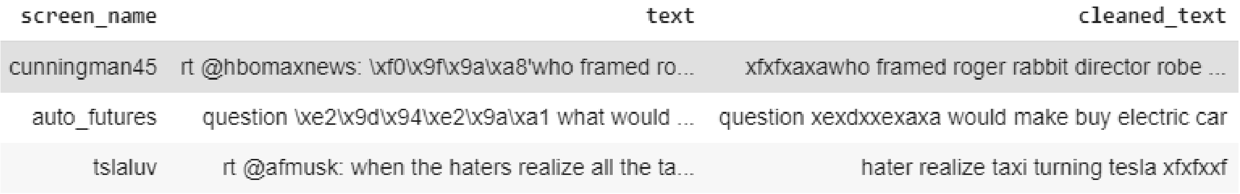
4.1.2.2.1. Text Cleaning for Tweets & Profiles

We used the same data cleaning steps for the data set used for topic modelling apart from the last step, where the words were lemmatized instead of stemmed so that the meaning of the words are still maintained in their own context.

In summary, these were the actions we took for each tweet and profile that was crawled.

1. Removed the user's name
2. Removed Twitter mentions (I.e. @twitterhandle and "RT")
3. Removed links, retaining only characters and spaces in the text
4. Removed blanks and dropped duplicates
5. Removed stopwords and punctuations
6. Lemmatizing the words

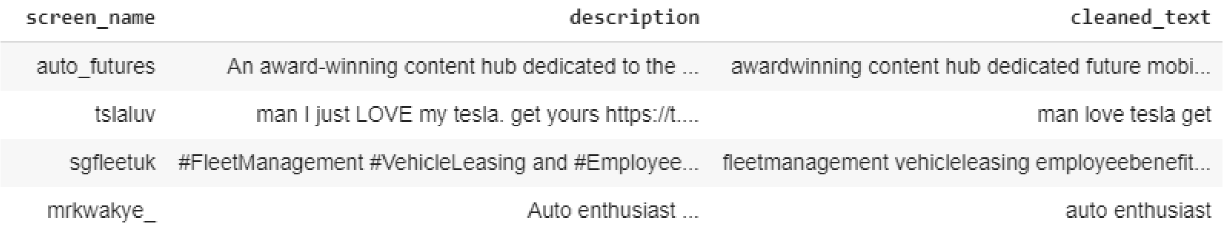
After the text cleaning for the users' tweets, we will store the cleaned text which will be used to find users whose tweets are relevant to our defined keywords *(Refer to Figure 12)*



screen_name	text	cleaned_text
cunningman45	rt @hbomaxnews: \xf0\x9f\x9a\xa8'who framed ro...	xfxfxaxawho framed roger rabbit director robe ...
auto_futures	question \xe2\x9d\x94\xe2\x9a\xa1 what would ...	question xexdxxexaxa would make buy electric car
tslaluv	rt @afmusk: when the haters realize all the ta...	hater realize taxi turning tesla xfxfxxf

Figure 12: Example of Cleaned Text for tweets

After the text cleaning for the users' profiles, we will store the cleaned text which will be used to find users whose profiles are relevant to our defined keywords. *(Refer to Figure 13)*



screen_name	description	cleaned_text
auto_futures	An award-winning content hub dedicated to the ...	awardwinning content hub dedicated future mobi...
tslaluv	man I just LOVE my tesla. get yours https://t....	man love tesla get
sgfleetuk	#FleetManagement #VehicleLeasing and #Employee...	fleetmanagement vehicleleasing employeebenefit...
mrkwakye_	Auto enthusiast ...	auto enthusiast

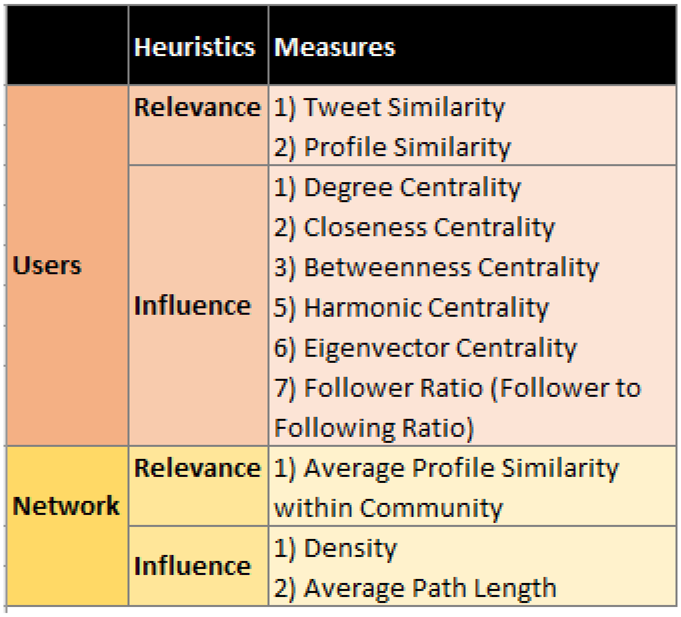
Figure 13: Example of cleaned text for profiles

4.1.2.3. Analysis

As the list of users contained every single user that had tweets containing the keywords we defined, we had to sift out the users that were more relevant to DFM from this large list.

We scored each individual user as well as the community (network) which they belonged in based on several measures which we grouped under two categories, relevance and influence (*Refer to Figure 14*). Using our scoring algorithm, we gave an overall score to each user, and sieved out the users that had the highest overall scores.

In this section, we will explain what relevance and influence means, how we calculated the weighted score for each user followed by explaining what each measure under relevance and influence represents.



	Heuristics	Measures
Users	Relevance	1) Tweet Similarity 2) Profile Similarity
	Influence	1) Degree Centrality 2) Closeness Centrality 3) Betweenness Centrality 5) Harmonic Centrality 6) Eigenvector Centrality 7) Follower Ratio (Follower to Following Ratio)
Network	Relevance	1) Average Profile Similarity within Community
	Influence	1) Density 2) Average Path Length

Figure 14: The measures that will be used to rank the users and network

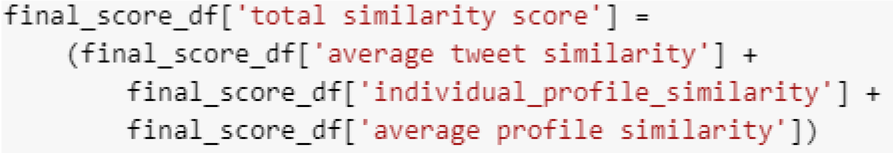
1) Relevance

Relevance is referred to as how closely associated an individual is in the context of possessing the expertise and subject-matter credibility with their followers. The relevance of an individual user is determined by the similarity of our identified keywords and their tweets or profiles. This allows us to sift out potential customers who are more relevant to DFM by selecting users that scored higher in their tweet and profile similarities

This method of matching the tweets of users to a list of seeded words was proposed in a research paper [3] that attempts to differentiate and identify high-value social entities on social media. We used a pre-trained word embedding model to calculate the similarity scores. This will help us obtain scores for the users' tweet and profile similarities.

We assigned the final relevance score to a user by aggregating the scores from their individual measures (tweet and profile similarities) as well as their network measures such as the average profile similarity within community.

The final relevance score for each user was standardized from 0 to 1.

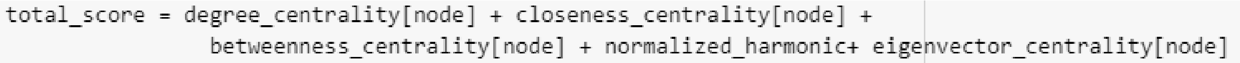


```
final_score_df['total similarity score'] =
    (final_score_df['average tweet similarity'] +
        final_score_df['individual_profile_similarity'] +
        final_score_df['average profile similarity'])
```

Figure 15: Formula for aggregating the relevance score for each individual user

2) Influence

Influence is a measure of how much power an individual has which affects the decisions of others because of their authority, knowledge, position or relationship within the audiences in their social network. Influence is calculated by taking the sum of a user's weighted centrality scores based on the entire network (using number of retweets as weights for each centrality calculated).

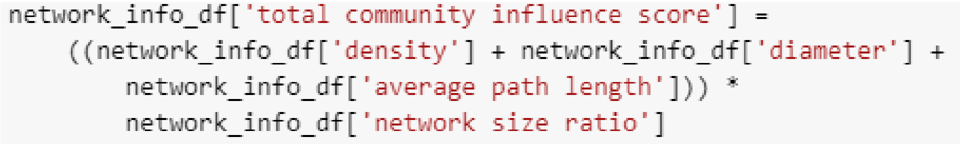


```
total_score = degree_centrality[node] + closeness_centrality[node] +
                  betweenness_centrality[node] + normalized_harmonic+ eigenvector_centrality[node]
```

Figure 16: Formula for aggregating the centrality measure scores for each individual user

We also made use of Follower to Following ratio to determine the influence of the users. The Follower to Following Ratio allows us to quickly gauge the quality of the user's account. For example, those with low Follower to Following ratio are typically low-quality accounts that depend only on the Follow/Unfollow method to gain followers, whereas accounts with high Follower to Following ratio are most likely influencers and celebrities [4]. Similar to the centrality measures, the Follower to Following ratio has also been standardized to a score between 0 and 1.

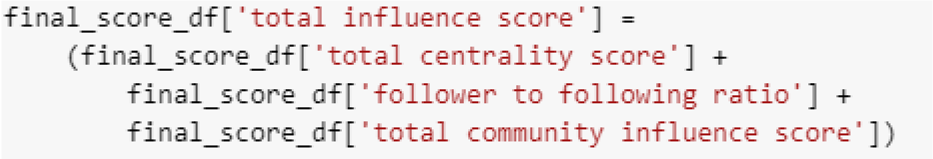
Network Influence is calculated by taking the sum of density, diameter and average path length and this is normalized by its relative community size to the network size.



```
network_info_df['total community influence score'] =
    ((network_info_df['density'] + network_info_df['diameter'] +
        network_info_df['average path length'])) *
        network_info_df['network size ratio']
```

Figure 17: Formula for calculating the network influence score for each individual user

The total influence score is simply the sum of the total centrality score, follower to following ratio and their community influence score. This has also been standardized to a score between 0 and 1 after summation.



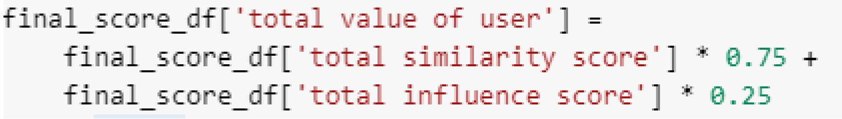
```
final_score_df['total influence score'] =
    (final_score_df['total centrality score'] +
        final_score_df['follower to following ratio'] +
        final_score_df['total community influence score'])
```

Figure 18: Formula for obtaining the final influence score for each individual user

3) Total Value of User

In calculating the overall score for the user based on their relevance and influencer score, we assigned a higher weight for relevance than influence.

Based on an article that compared influencer marketing and reach versus relevance [5], research has suggested that campaigns are more successful when it involves a well-respected, industry-relevant online influencer with smaller, highly engaged audiences as compared to campaigns that try to include big names that are not a good brand match. Hence, we will put a higher weight *(Refer to Figure 19)* on relevance score to obtain users that will be more effective for DFM to reach out to.



```
final_score_df['total value of user'] =
    final_score_df['total similarity score'] * 0.75 +
    final_score_df['total influence score'] * 0.25
```

Figure 19: Formula for calculating final overall score for user by adding weights to influence and relevance scores

After getting the final score of the users based on their relevance and influence, we filter out those users who have no degree centrality as these users are isolates in the network who are not connected to any other users in the network. After doing so, we are able to identify the group of valued prospective customers for DFM. This will be shared in Section 4.1.2.4.

4.1.2.3.1. Users

4.1.2.3.1.1. Relevance

1) Tweet Similarity

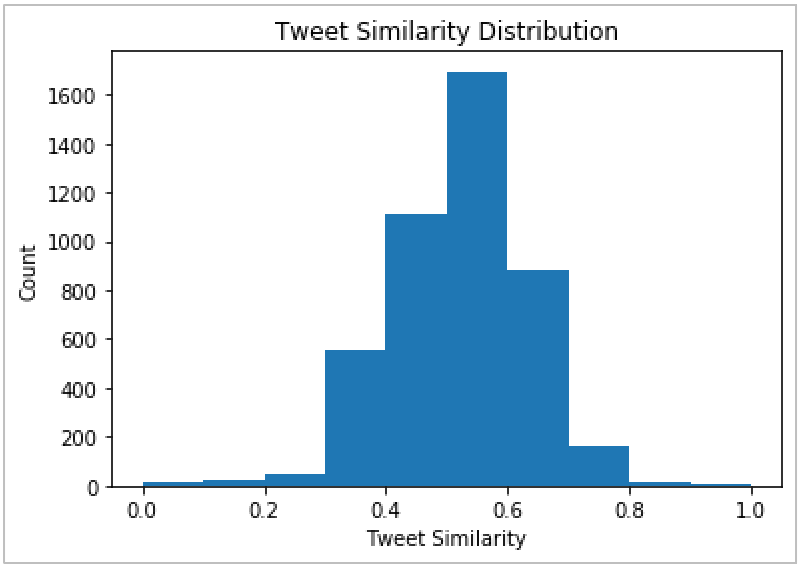
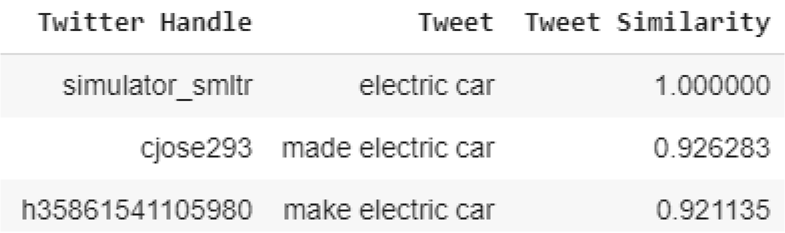


Figure 20: Distribution of Tweet Similarity Scores

The tweets of users are matched to a list of seeded words, to generate a tweet similarity score for each user *(Refer to Figure 21).* There is a normal distribution of how relevant our users are based on their tweets to our keywords. Having a normal distribution means that we have a limited pool of potential influencers or celebrities (Score of 0.8 or higher) to identify. Having a higher score for tweet similarity score means that their tweets have are more relevant to our keywords. As you can see from the below diagram, the users mentioned the words, “electric car” in their tweets. These words can be found in our keywords, therefore, giving them a higher score.

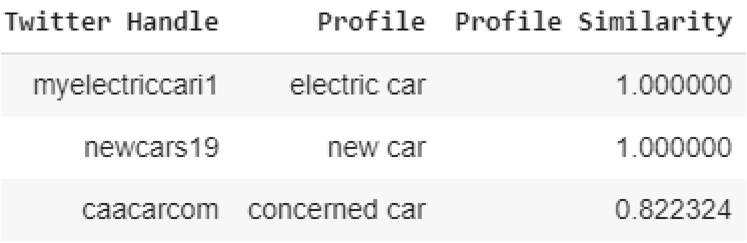


Twitter Handle	Tweet	Tweet Similarity
simulator_smltr	electric car	1.000000
cjose293	made electric car	0.926283
h35861541105980	make electric car	0.921135

Figure 21: Example to illustrate how tweet similarity scores are calculated against “electric car”

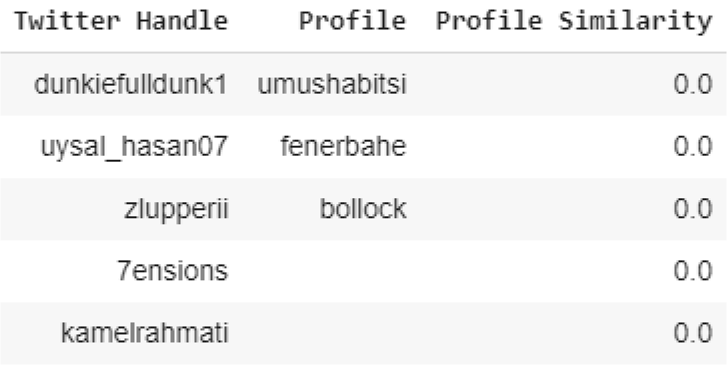
2) Profile Similarity

The text in the profiles of users are matched to a list of seeded words, to generate a profile similarity score for each user *(Refer to Figure 22).* There is a huge number of users whose profile similarity scores lie between 0 and 0.1. This could be due to their profiles being empty or not written in English *(Refer to Figure 23)*.



Twitter Handle	Profile	Profile Similarity
myelectriccari1	electric car	1.000000
newcars19	new car	1.000000
caacarcom	concerned car	0.822324

Figure 22: Example to illustrate how profile similarity scores are calculated against “electric car”



Twitter Handle	Profile	Profile Similarity
dunkiefulldunk1	umushabitsi	0.0
uysal_hasan07	fenerbahe	0.0
zlupperii	bollock	0.0
7ensions		0.0
kamelrahmati		0.0

Figure 23: Example to illustrate how profile similarity scores are calculated against “electric car”

There is a normal distribution of how relevant our users are based on their profiles to our keywords *(Refer to Figure 24)*. Having a normal distribution means that we have a limited pool of potential influencers or celebrities (Score of 0.8 or higher).

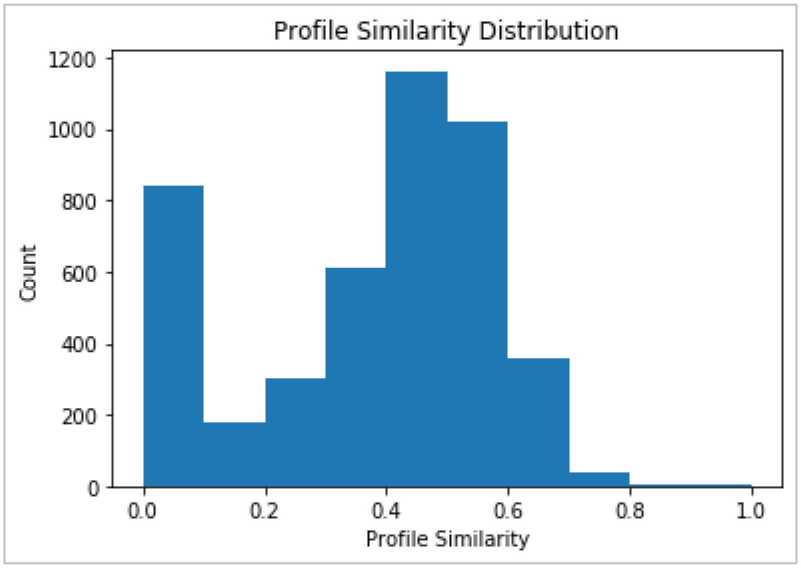


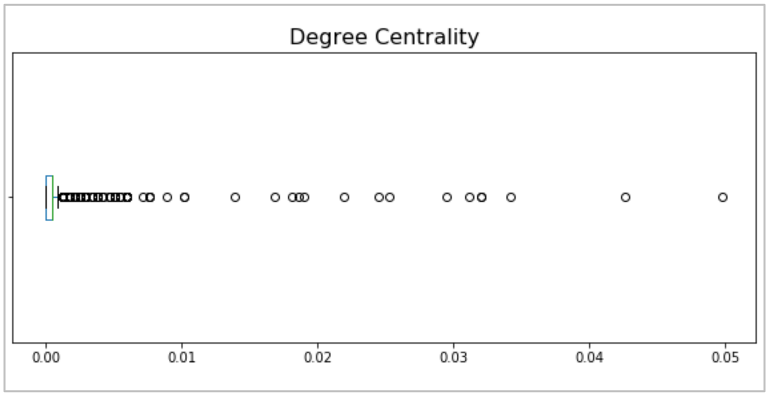
Figure 24: Distribution of profile similarity scores

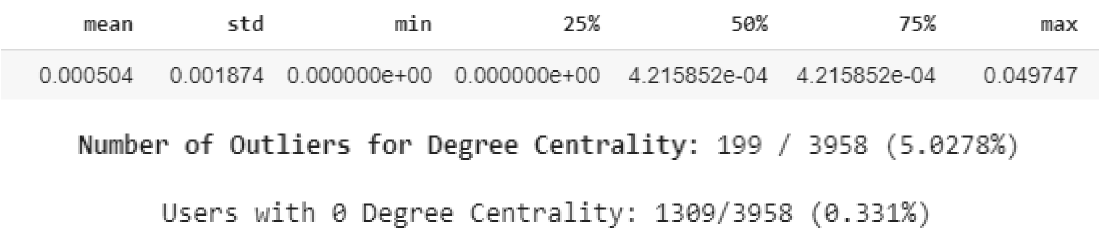
We wanted to include this as a measure as users tends to have interesting profiles which give insights into their characteristics and preferences. For example, they might have a profile that they are a car enthusiast. In our scenario, they would be relevant to our keywords as they have the word, 'car'. This measure serves as an additional score as it might help us identify users who are more relevant to DFM's brand from their profile information.

4.1.2.3.1.2. Influence

1) Degree Centrality

The degree centrality score is dependent on the number of links held by each node and essentially tells us which individuals are the most likely to hold the most information or can quickly connect with a large amount of people.





mean	std	min	25%	50%	75%	max
0.000504	0.001874	0.000000e+00	0.000000e+00	4.215852e-04	4.215852e-04	0.049747

```
Number of Outliers for Degree Centrality: 199 / 3958 (5.0278%)

Users with 0 Degree Centrality: 1309/3958 (0.331%)
```

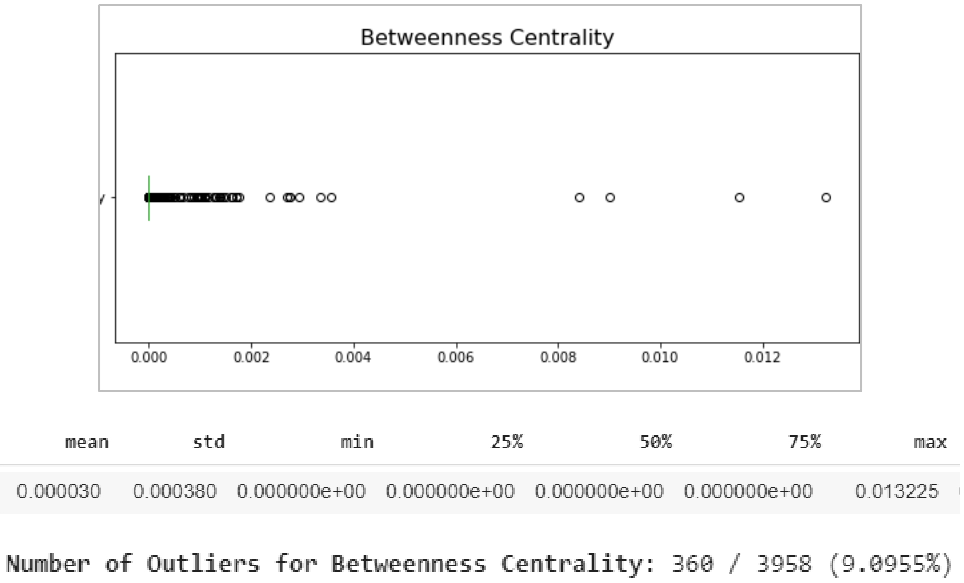
Figure 25: Descriptions of Degree Centrality Statistics

The minimum degree centrality is 0 which means that there are users who do not collaborate with anyone. The maximum degree centrality is 0.049747 which is held by the user that has the highest number of interactions in the network. There are 199 outliers *(Refer to Figure 25)* which means that these users interacted with other users much more frequently than the normal distribution of the entire network. These outliers are well-connected in the network.

However, it is still important to note that there are 1309 users who have 0 degree centrality which means that they are isolates. We should also account for these users when doing our final analysis to identify prospective customers.

2) Betweenness Centrality

The betweenness centrality score measures the number of times a node lies on the shortest path to other nodes. It highlights to us the individuals that are likely to influence the flow of information in a network.



mean	std	min	25%	50%	75%	max
0.000030	0.000380	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.013225

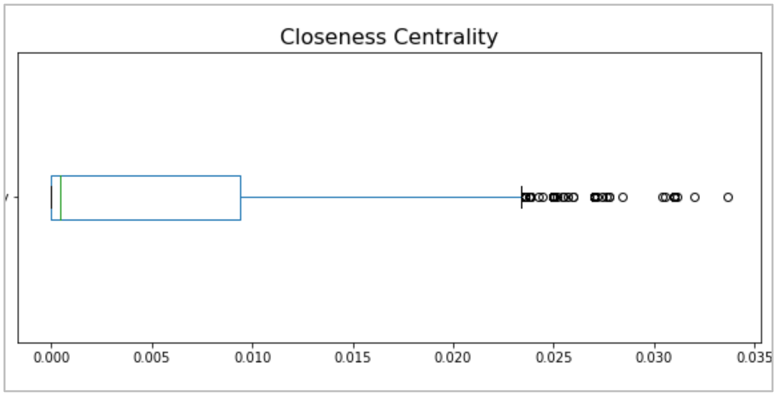
```
Number of Outliers for Betweenness Centrality: 360 / 3958 (9.0955%)
```

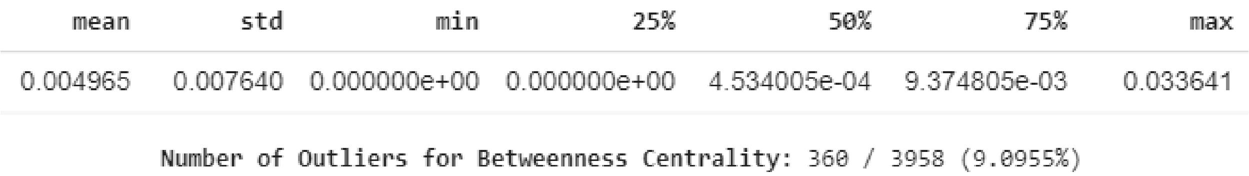
Figure 26: Descriptions of Betweenness Centrality Statistics

The minimum betweenness centrality is 0.000030 *(Refer to Figure 26)* which means that there are users who does not hold much importance as they are not located near any users in the network to influence. The maximum betweenness centrality is 0.013225 which means that the user is an important person in the information flow within the network. These 360 outliers are important as information tends to flow through them most of the time.

3) Closeness Centrality

The closeness centrality score refers to how “close” each node is to other nodes within the network and helps us find individuals who can influence the entire network the fastest.





mean	std	min	25%	50%	75%	max
0.004965	0.007640	0.000000e+00	0.000000e+00	4.534005e-04	9.374805e-03	0.033641

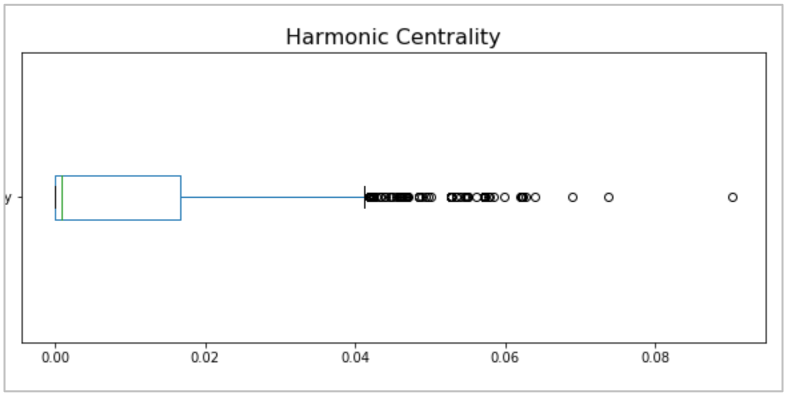
```
Number of Outliers for Betweenness Centrality: 360 / 3958 (9.0955%)
```

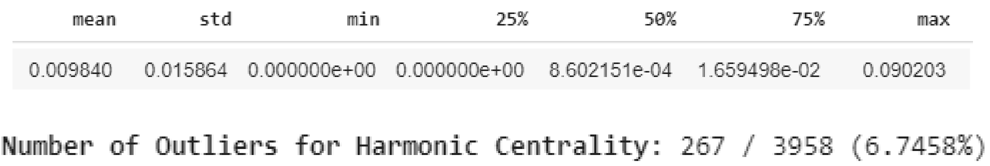
Figure 27: Descriptions of Closeness Centrality Statistics

The minimum closeness centrality is 0 *(Refer to Figure 27)* which means that there are users who are not located near any users in the network. The maximum closeness centrality is 0.033641 which means that the user is very well-positioned in the network. The distribution of closeness centrality is right-skewed. There are 360 outliers which means that these authors are most likely located at a dense location of the network. This means that they are important users who are able to spread information fast.

4) Harmonic Centrality

The harmonic centrality score is a variant of closeness centrality that focuses on solving the issue faced by using the closeness centrality on unconnected graphs.





mean	std	min	25%	50%	75%	max
0.009840	0.015864	0.000000e+00	0.000000e+00	8.602151e-04	1.659498e-02	0.090203

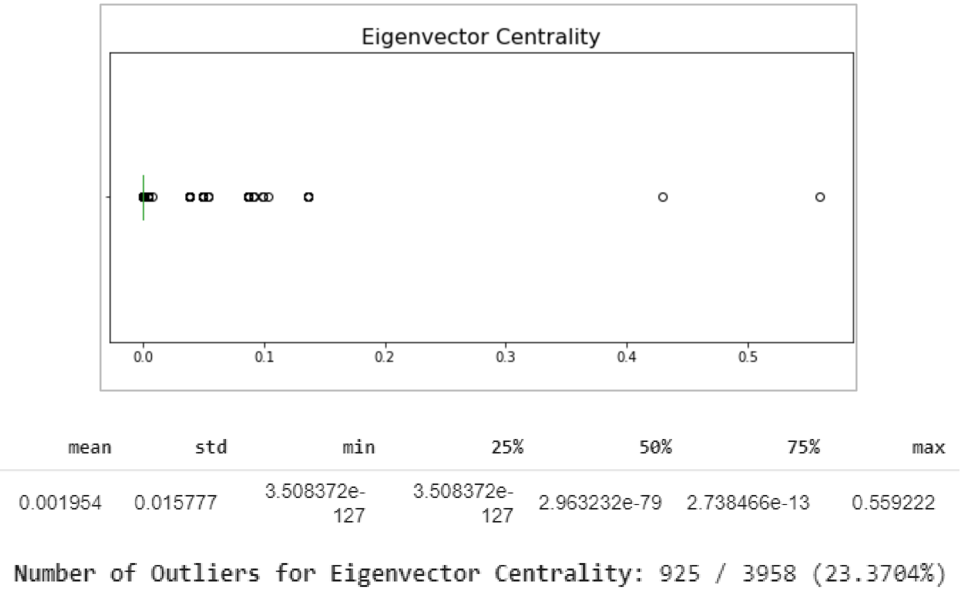
```
Number of Outliers for Harmonic Centrality: 267 / 3958 (6.7458%)
```

Figure 28: Descriptions of Harmonic Centrality Statistics

The minimum harmonic centrality is 0 *(Refer to Figure 28)* which means that there are users who are not located near any users in the network. The maximum harmonic centrality is 0.090203 which means that the user is very well-positioned in the network. The distribution of harmonic centrality is right-skewed. There are 267 outliers which means that these authors are most likely located at a dense location of the network. These outliers might be the same outliers as the ones found in closeness centrality.

5) Eigenvector Centrality

The eigenvector centrality score also measures the node's influence based on the number of links it has to other nodes. However, eigenvector goes more in depth by considering how well-connected a node is to other well-connected nodes and goes beyond direct connections.



mean	std	min	25%	50%	75%	max
0.001954	0.015777	3.508372e-127	3.508372e-127	2.963232e-79	2.738466e-13	0.559222

```
Number of Outliers for Eigenvector Centrality: 925 / 3958 (23.3704%)
```

Figure 29: Descriptions of Eigenvector Centrality Statistics

The maximum eigenvector centrality is 0.559222 which suggests that the user is a highly influential user in the network. There are 925 outliers which means that these users have more influence than the general distribution of the network. These outliers might be people of high standing and status, explaining their high eigenvector centrality score.

6) Follower Ratio (Follower to Following ratio)

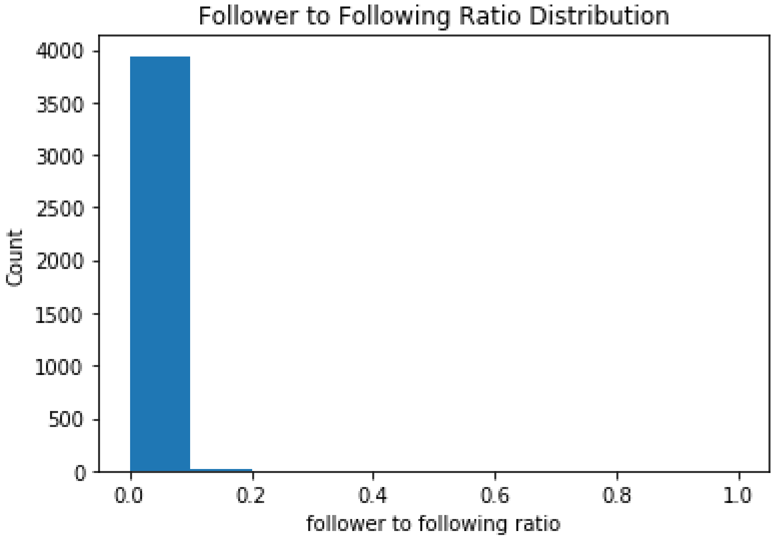
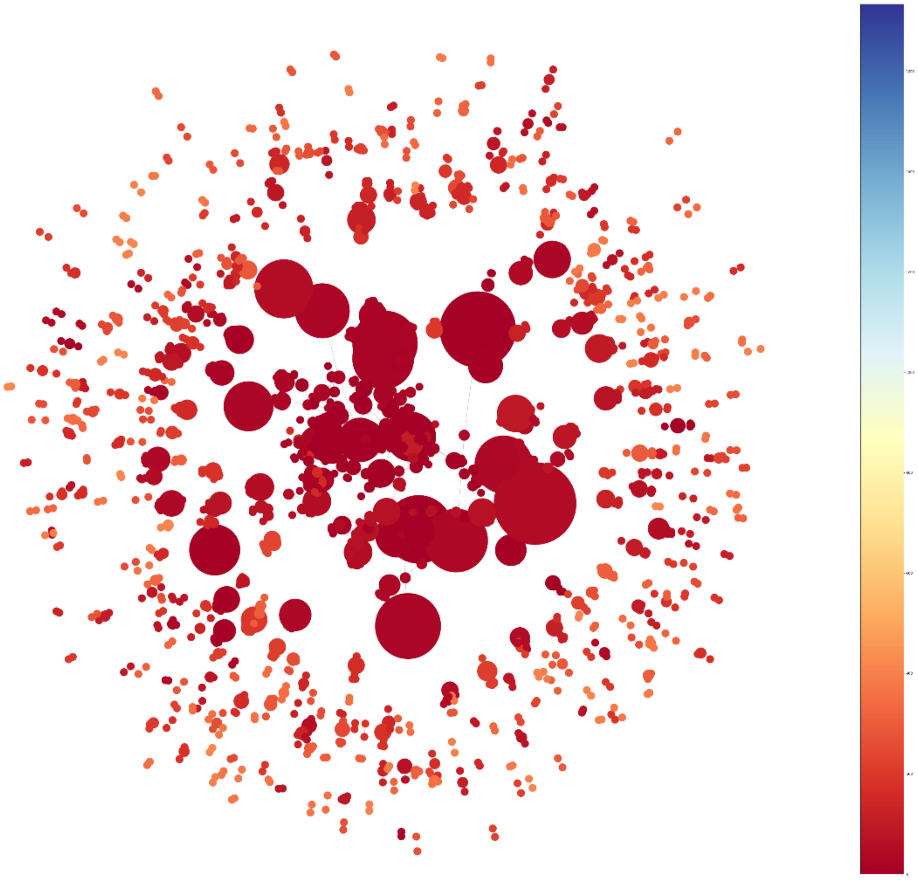


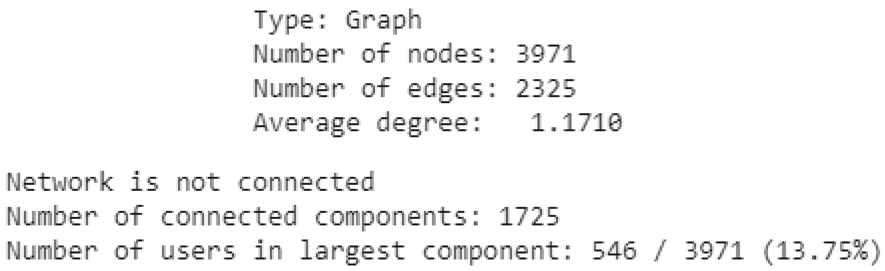
Figure 30: Distribution of Follower Ratio

The Follower Ratio is simply the number of followers someone has, divided by the number of people following them. As such, a value above 1 means an account has more followers than accounts they are following. To give a gauge, celebrities might have a ratio of 100,000+, while non-celebrities will often have a ratio below 1.

The distribution is right-skewed. We have lots of users who have low follower to following ratio. Based on Twitter's definition, this means that we have lots of users who are not influencers. This is a more realistic way of evaluating our potential influencers.

4.1.2.3.2. Network





```
Type: Graph
Number of nodes: 3971
Number of edges: 2325
Average degree:   1.1710

Network is not connected
Number of connected components: 1725
Number of users in largest component: 546 / 3971 (13.75%)
```

Figure 31: Overview of the Social Network and its Statistics

Based on our keywords, we managed to identify 3,971 prospective customers. This network has 2,325 edges which suggest that there are only 2,325 retweets among these 3,971 potential customers. The average degree is 1.1710 which means that each customer knows about 1 to 2 different users in the network.

From the network diagram (*Refer to Figure 31)*, we also found out that the network is not connected. The number of connected components is 1725. In the largest community, there are 546 potential customers which made up 13.75% of the entire network. As the network size of the largest component has also far exceeded the Dunbar's number of 150, cohesion between users might break down and hinder communication. Therefore, to find out the best-suited community for each potential customer, we used the Louvain community detection algorithm which attempts to optimize the "modularity" of a community of the network. This will result in achieving communities with high modularity having dense connections between potential customers in the same community but sparse connections between potential customers in different communities.

4.1.2.3.2.1. Relevance

1) Average Profile Similarity within Community

We used Louvain Community Detection Algorithm to generate communities in the network of which these users belonged to. This algorithm generated 1174 communities. For each community, we did an average profile score *(Refer to Figure 32)*, which is able to serve as a gauge of how relevant a user in each particular community is to DFM. This is important as we might have identified an influencer who is related to our keywords but the community around that influencer is not which would lead to less effective results as compared to identifying a relevant influencer who is in a relevant community.

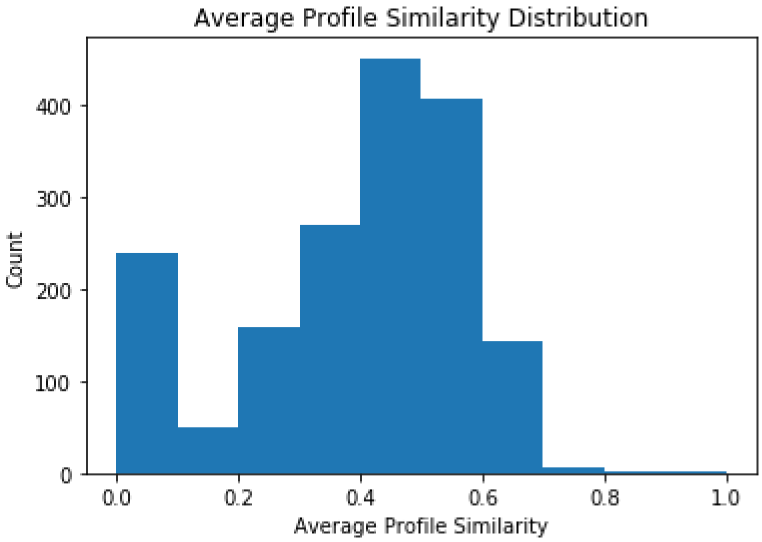


Figure 32: Distribution of the average profile similarity score within the communities generated

Other than the communities that have a score of 0 to 0.1, the average profile similarity scores for the communities follow a normal distribution. There is a huge number of communities which have a score close to 0 as they might be users who have missing profiles or isolates of the network.

4.1.2.3.2.2. Influence

1) Density

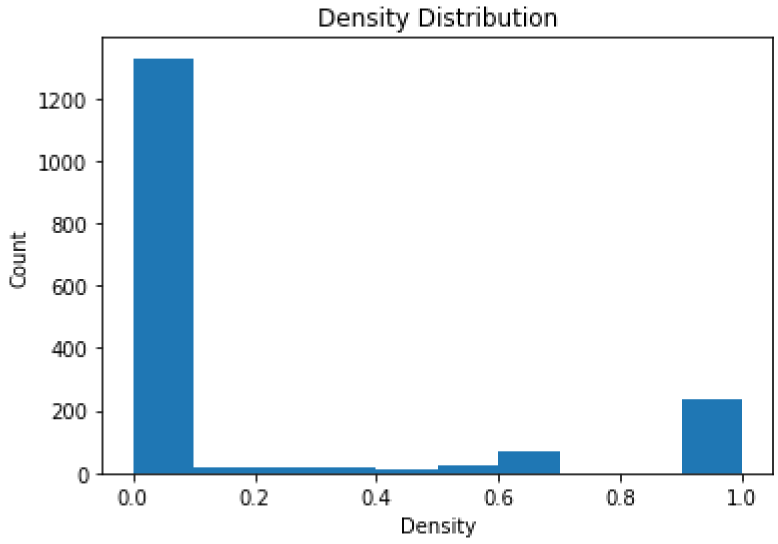


Figure 33: Distribution of the density scores for the communities generated

Density involves the proportion of all possible links that are present and describes the general level of cohesion in a graph. Density has been normalized to a score of 0 and 1. A higher density score will mean that information or resources are diffused among the users faster. The distribution we can observe is right tailed. However, we still have around 300 communities in this network which is considered dense.

2) Average Path Length

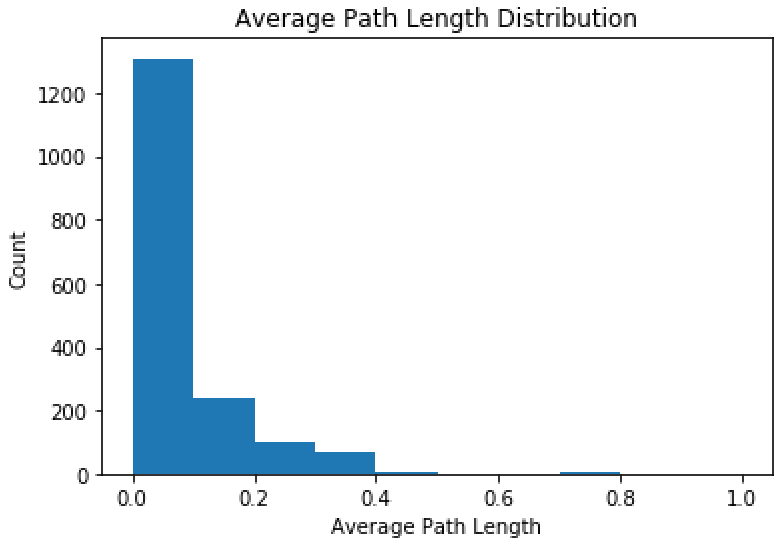


Figure 34: Distribution of the average path length scores for the communities generated

Average Path Length refers to the sum of all distances (geodesic or shortest paths) divided by the total number of node-to-node pairs. It is a measure of the efficiency of information or mass transport on a network. Average Path Length has been normalized to a score of 0 and 1. A higher score means that information can be passed within a higher degree of separation.

4.1.2.4 Interesting Finding Insights

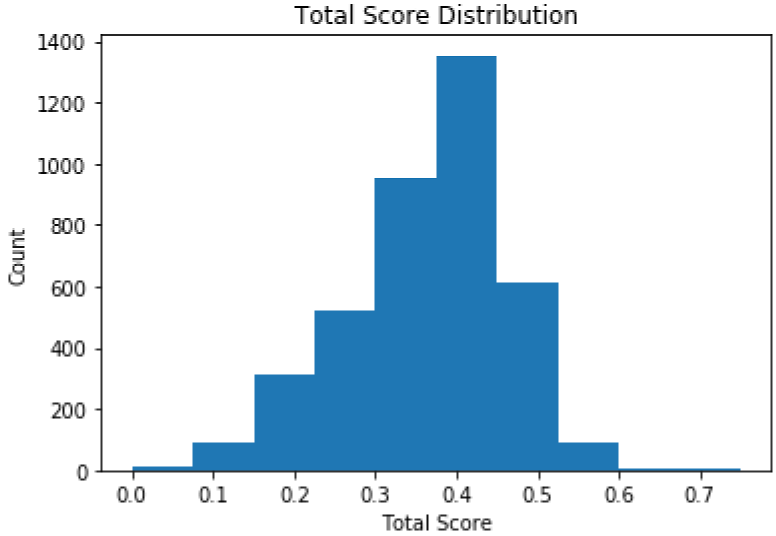


Figure 35: Distribution of the Total Scores for All Users

From the weighted total scores obtained for each user *(Refer to Figure 35)*, we observe that the scores follow a normal distribution. Majority of the users are centered around the 0.4 to 0.5 range. However, as mentioned above, we should remove those users who have no degree centrality as they are isolates and are not connected to anyone in the network which will make them to be of little value to DFM.

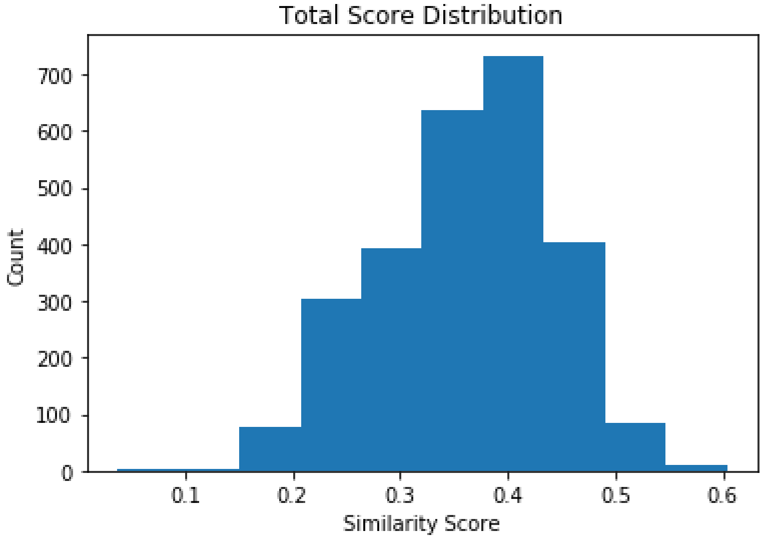
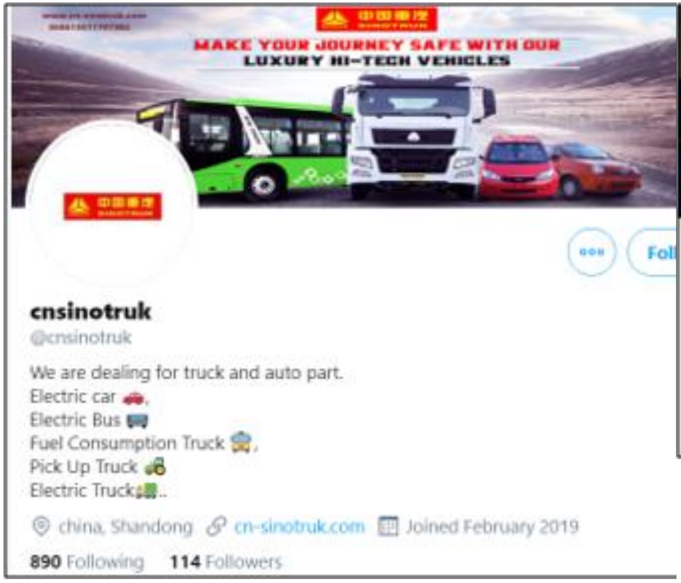


Figure 36: Distribution of the Total Scores for All Users after removing isolates

After filtering those users that are isolates of the network, we identified a few interesting profiles that stood out amongst the profile of users who had a relatively high scoring based on the users who scored between 0.5 to 0.6.

The top scorers were **largely made up of Automobile, Autoparts and car review Twitter accounts.** Besides that, there was an individual named Assand Razzouk with a high follower count and is **an advocate and public figure towards clean energy** *(Refer to Figure 35).*

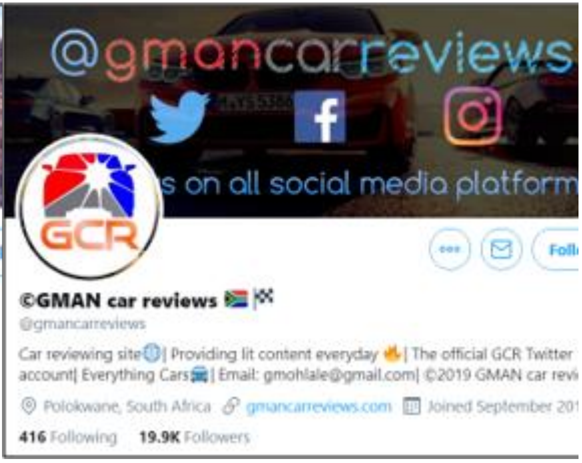


cnsinotruk
@cnsinotruk

We are dealing for truck and auto part.
Electric car 🚗,
Electric Bus 🚌
Fuel Consumption Truck 🚖,
Pick Up Truck 🚜
Electric Truck🚛..

china, Shandong cn-sinotruk.com Joined February 2019

890 Following 114 Followers

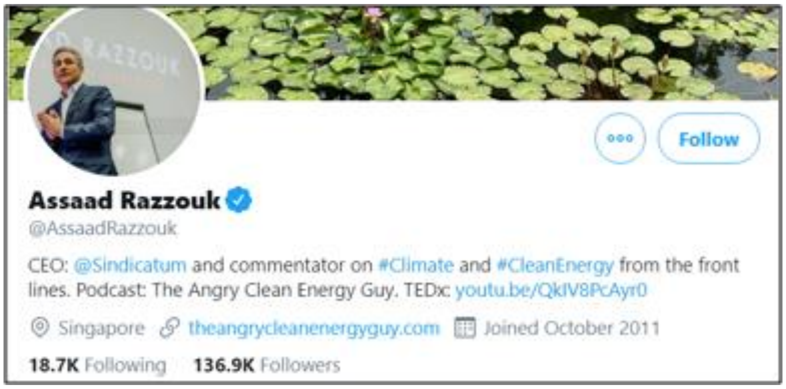


©GMAN car reviews 🇿🇦🏁
@gmancarreviews

Car reviewing site🌐| Providing lit content everyday 🔥| The official GCR Twitter account| Everything Cars🚘| Email: gmohlale@gmail.com| ©2019 GMAN car revi

Polokwane, South Africa gmancarreviews.com Joined September 201

416 Following 19.9K Followers



Assaad Razzouk ✓
@AssaadRazzouk

CEO: @Sindicatum and commentator on #Climate and #CleanEnergy from the front lines. Podcast: The Angry Clean Energy Guy. TEDx: youtu.be/QkIV8PcAyr0

Singapore theangrycleanenergyguy.com Joined October 2011

18.7K Following 136.9K Followers

User	Total Similarity Score	Total Influence Score	Total Value of User
cnsinotruk	0.488274	0.950583	0.603851
gmancarreviews	0.780686	0.002147	0.586051
assaadrazzouk	0.760481	0.003617	0.571265

Figure 37: Illustrations of the Identified Twitter Profiles and respective scores of these users

We suggest that DFM can strike a partnership through collaborating with these influencers to market their car leasing services on their twitter accounts as these are influential figures in the car leasing industry. In addition, DFM can analyze the followers of all these Twitter users to gather more interesting insights. For example, users who follow Assand Razzouk tells us that they might have an interest towards clean energy whereas users that follow these car review accounts have an interest in cars. By analyzing these followers, DFM is able build a persona or set of characteristics that they can target for their proposed car leasing services.

4.2. LinkedIn

From LinkedIn, we are able to obtain personal details about a person such as their education, employment history, interests and others. From their education and job positions, we are able to deduce their rough spending power which is able to help us narrow down which tier of car leasing options are viable for them. In addition, their interests and personal description can give insights as to which type of car within the viable leasing options would be more optimal for them.

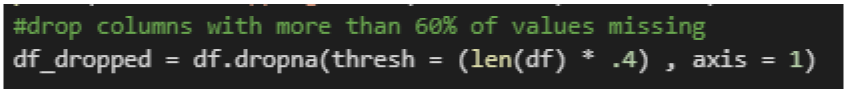
4.2.1. Collecting

We used a Google Chrome Extension, Linked Helper, to extract data from the profiles of our profiled users. As there is an advised limit of extracting a maximum of 150 users per day, each of us crawled around 100-150 users for 3 days. However, the limitation on this application is that the user profiles are sorted from how far your connection link is from them. If they are your direct connections, then they will appear first, and if they are connections of connections (2nd link connection), then they will have priority after the direct connections and so on. Hence, the profiles extracted will have a dependency on who we are connected to and hence, skewing the profiles to possibly contain more SMU students.

4.2.2. Cleaning

LinkedIn has many data fields that are optional. A quick check shows that we have 130 columns. To allow us to get a meaningful analysis, we need to keep the columns where data is populated significantly. Hence, we dropped any columns where more than 60% of the records are missing. The number of columns dropped is 65, leaving us with 65 columns to analyze. Organization and Education columns span from 1 to 7, which is LinkedIn's option of allowing users to enter up to 7 of their previous education or working history.

4.2.2.1 Removing unnecessary columns in dataset



```
#drop columns with more than 60% of values missing
df_dropped = df.dropna(thresh = (len(df) * .4) , axis = 1)
```

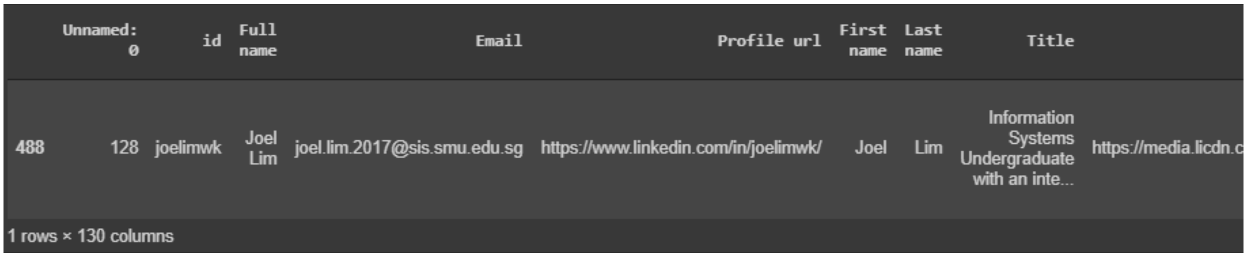
Figure 38: Code snippet of dropping columns

Due to the large number of columns, we have summarized and grouped relevant columns together to present it here. The Columns dropped are:

1. Address
2. Birthday

3. Education (Degree, description, start and end dates) 3 and above
4. Email Followers
5. Organization (Description, Start/end dates, domain) 5 and above
6. Phone
7. Twitter Website

We then observe the columns to see if there are any valuable columns that we need but may have been dropped. Firstly, we may possibly need their address to adopt location targeted marketing, if needed. We checked the Address column, which only had 1 record out of 1268 records. Hence, we will still proceed with dropping this column.



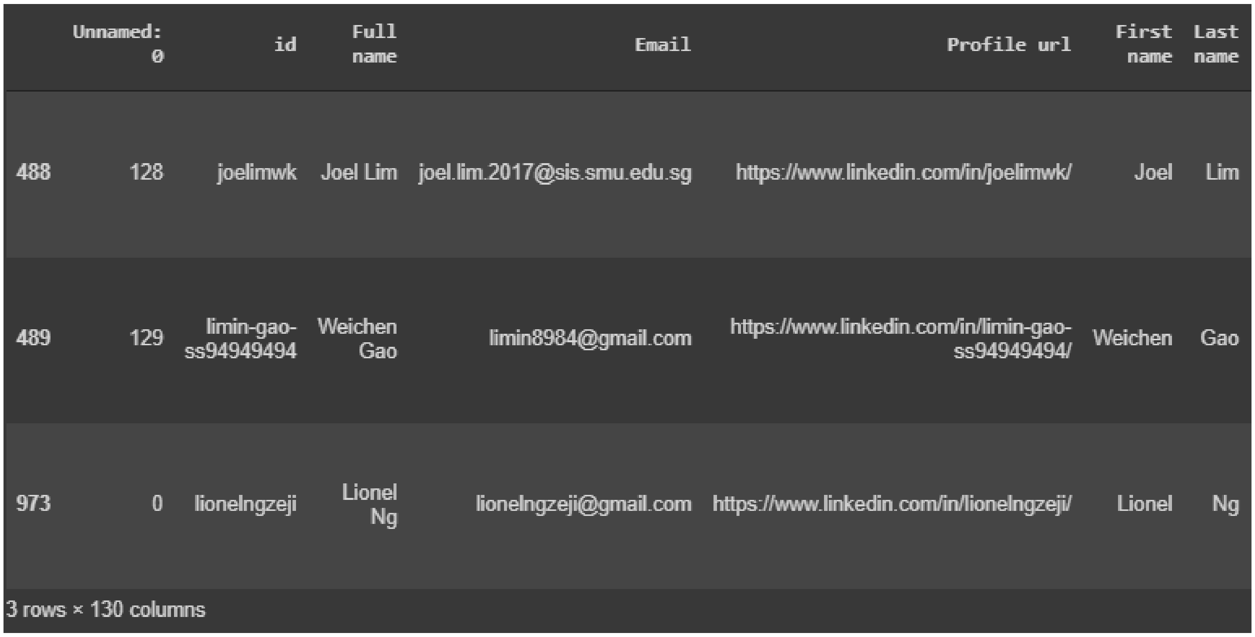
	Unnamed: 0	id	Full name	Email	Profile url	First name	Last name	Title	
488	128	joelimwk	Joel Lim	joel.lim.2017@sis.smu.edu.sg	https://www.linkedin.com/in/joelimwk/	Joel	Lim	Information Systems Undergraduate with an inte...	https://media.licdn.c

1 rows × 130 columns

Figure 39: Snippet of the results from searching for users that had the address column filled up

Subsequently, the email column has 73 records, which makes up less 5% of our overall scraped profiles. We will still proceed to include this column even it is below our threshold of 60% missing values, as we may still need their email, if available, to target them.

Lastly, the phone column may also be important to us. Hence, we did a check how many profiles have their phone number entered, but only 3 profiles were found. We will still proceed with dropping of this column.



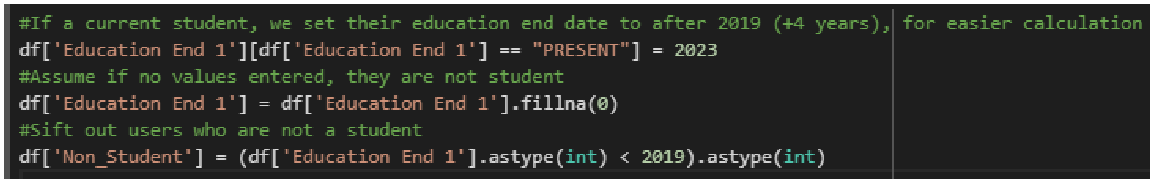
	Unnamed: 0	id	Full name	Email	Profile url	First name	Last name
488	128	joelimwk	Joel Lim	joel.lim.2017@sis.smu.edu.sg	https://www.linkedin.com/in/joelimwk/	Joel	Lim
489	129	limin-gao-ss94949494	Weichen Gao	limin8984@gmail.com	https://www.linkedin.com/in/limin-gao-ss94949494/	Weichen	Gao
973	0	lionelngzeji	Lionel Ng	lionelngzeji@gmail.com	https://www.linkedin.com/in/lionelngzeji/	Lionel	Ng

3 rows × 130 columns

Figure 40: Snippet of the results from searching for users that had the phone number column filled up

4.2.2.2 Student status column

To find out if a profile is still a student, we will extract the end date of their latest education. The column is in the format of “MM, YYYY”. Hence, we first remove the month in the column by retaining only the digits which corresponds to the year. Subsequently, if the end date of the extracted date column is before 2019, we will assume that the profile is not a student. Else, we will assume that he or she is a student.

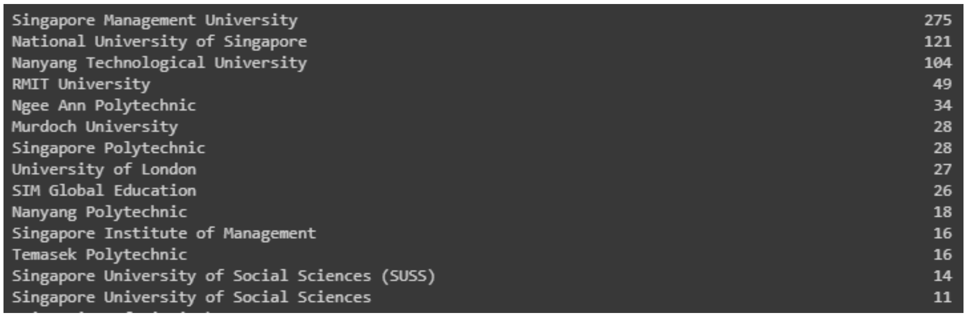


```python
#If a current student, we set their education end date to after 2019 (+4 years), for easier calculation
df['Education End 1'][df['Education End 1'] == "PRESENT"] = 2023
#Assume if no values entered, they are not student
df['Education End 1'] = df['Education End 1'].fillna(0)
#Sift out users who are not a student
df['Non_Student'] = (df['Education End 1'].astype(int) < 2019).astype(int)
```

Figure 41: Code snippet to extract non-student status

4.2.2.3 Education category column

To determine their education category for analysis, we extracted their latest education column, named ‘Education 1’.



```
Singapore Management University                          275
National University of Singapore                         121
Nanyang Technological University                         104
RMIT University                                           49
Ngee Ann Polytechnic                                      34
Murdoch University                                        28
Singapore Polytechnic                                     28
University of London                                      27
SIM Global Education                                      26
Nanyang Polytechnic                                       18
Singapore Institute of Management                         16
Temasek Polytechnic                                       16
Singapore University of Social Sciences (SUSS)            14
Singapore University of Social Sciences                   11
```

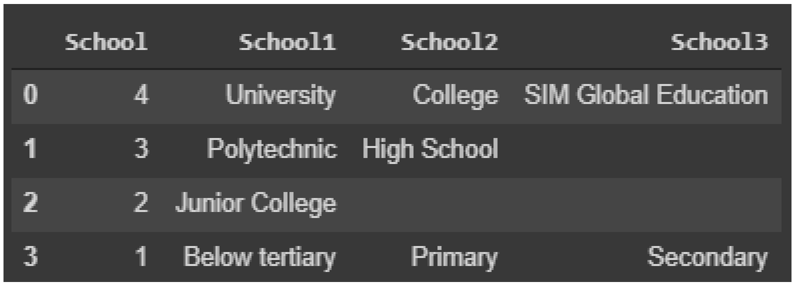
Figure 42: Snippet of results of searching for schools within the education columns that had more than 1 occurrence

After visualizing the distribution of the schools, most of the users had their highest level of education at university level. To split them into categories, we used fuzzy matching, a technique that can match strings that may be slightly different in spelling or strings that contain a spelling mistake, and it will return the matched string along with the score of the matched string.

For instance, a school with a Malay naming convention will be ‘Universiti’, which is similar to the English word of ‘University’. In this case, the fuzzy matching algorithm will match it to the word ‘University’ and give a relatively high matching score, since the difference between the 2 strings is 1 character.

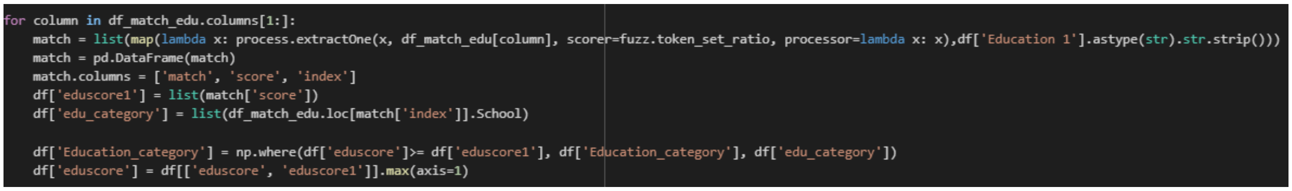
To match the schools to their respective categories, we first predefine a list of education levels. Universities and colleges will be assigned a higher score of 4, while Polytechnics and high schools are of a lower score. In the above table, we discovered that SIM Global Education does not contain the word ‘University’ in it, which doesn’t allow us to identify it as such. Hence, we have included it for matching.

The table can be referenced below. The score of 4 refers to universities and equivalent, 3 for polytechnics and high schools, 2 for junior colleges and 1 for the rest. If the name of the school is unable to match, it will be assigned a low matching score, and automatically have its assigned score to 1.



	School	School1	School2	School3
0	4	University	College	SIM Global Education
1	3	Polytechnic	High School	
2	2	Junior College		
3	1	Below tertiary	Primary	Secondary

Figure 43: Reference table to match each profile's education level to a score



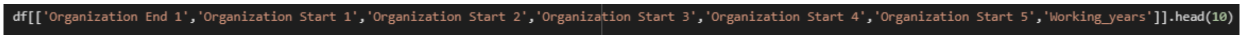
```
for column in df_match_edu.columns[1:]:
    match = list(map(lambda x: process.extractOne(x, df_match_edu[column], scorer=fuzz.token_set_ratio, processor=lambda x: x),df['Education 1'].astype(str).str.strip()))
    match = pd.DataFrame(match)
    match.columns = ['match', 'score', 'index']
    df['eduscore1'] = list(match['score'])
    df['edu_category'] = list(df_match_edu.loc[match['index']].School)

    df['Education_category'] = np.where(df['eduscore']>= df['eduscore1'], df['Education_category'], df['edu_category'])
    df['eduscore'] = df[['eduscore', 'eduscore1']].max(axis=1)
```

Figure 44: Code snippet to extract highest education

4.2.2.4 Years of Experience column

To retrieve their years of experience, we first converted all start and end working dates for all 5 of their past working experience columns (Organization Start, Organization End, having a postfix from 1 to 5) to years by removing the months, which are written in string. We will then need 2 values in determining their start and end year of their entire career. To get the start year of their career, we took the year of their first working experience, and subtracted it from the end year of their most current working experience.

If they are currently still working, i.e. column is marked as 'PRESENT', then we will assume the current year of 2019.

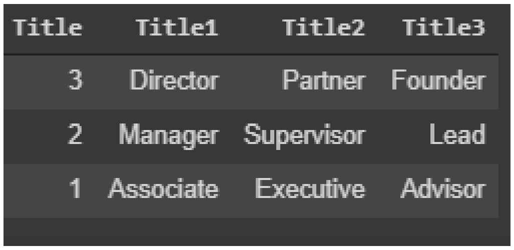


```
df[['Organization End 1','Organization Start 1','Organization Start 2','Organization Start 3','Organization Start 4','Organization Start 5','Working_years']].head(10)
```

Figure 45: Code snippet to extract number of years of experience

4.2.2.5 Corporate position

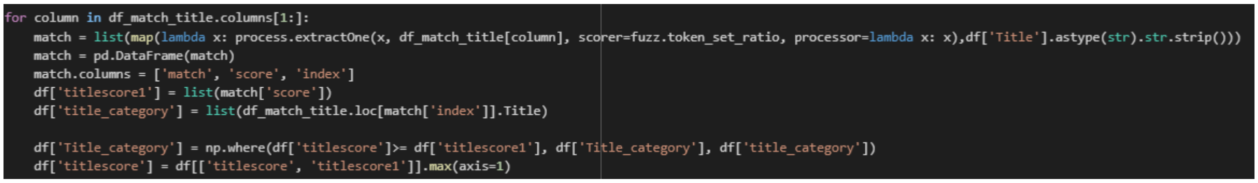
We extracted the corporate position of the profiles from the job titles (title column) and attempted to map it to 3 levels of corporate position. A higher level in their corporate positions would translate to a higher score.



Title	Title1	Title2	Title3
3	Director	Partner	Founder
2	Manager	Supervisor	Lead
1	Associate	Executive	Advisor

Figure 46: Reference table used for fuzzy matching

The titles to match are predefined and can be easily expanded. A position of director, partner and founder would translate to a score of 3, while a manager, supervisor or lead would have a score of 2. Lastly, associates, executives, advisors, and positions which we are unable match the name to, will have a score of 1. To map the job titles to the score, we performed a fuzzy matching to our predefined list.



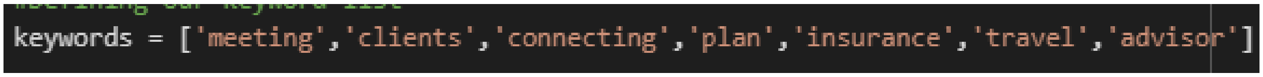
```
for column in df_match_title.columns[1:]:
    match = list(map(lambda x: process.extractOne(x, df_match_title[column], scorer=fuzz.token_set_ratio, processor=lambda x: x),df['Title'].astype(str).str.strip()))
    match = pd.DataFrame(match)
    match.columns = ['match', 'score', 'index']
    df['titlescore1'] = list(match['score'])
    df['title_category'] = list(df_match_title.loc[match['index']].Title)

    df['Title_category'] = np.where(df['titlescore']>= df['titlescore1'], df['Title_category'], df['title_category'])
    df['titlescore'] = df[['titlescore', 'titlescore1']].max(axis=1)
```

Figure 47: Code snippet to fuzzy match and extract the users' corporate position

4.2.2.6 Profile similarity

To get a profile similarity score, we will extract the text from their profile summary description, under column 'Summary'. We then load a pretrained Word2Vec model ('glove.6B.50d.txt'), trained on Wikipedia articles [6].

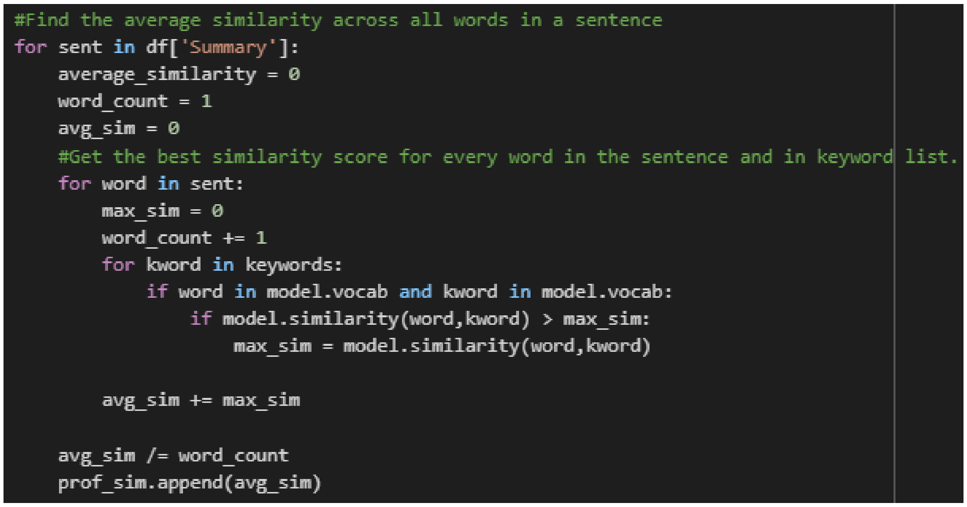
To get the similarity scores for each profile, we will match the similarity score of the profile to a list of keywords *(Refer to Figure 48)*. These keywords are designed for us to sift out profiles that are related to jobs that have a constant need to travel.



```
keywords = ['meeting','clients','connecting','plan','insurance','travel','advisor']
```

Figure 48: List of Keywords used to sift out profiles that are more likely to have the need to travel

We then match each of these predefined keywords to each word in the extracted text from each users' profile. The similarity score is generated by inferring our keywords with another word (from the user profiles) on the trained Word2Vec model, which then produces a similarity score based on how similar the words are to one another. This allows us to know how similar each profile is to our predefined keywords.



```
#Find the average similarity across all words in a sentence
for sent in df['Summary']:
    average_similarity = 0
    word_count = 1
    avg_sim = 0
    #Get the best similarity score for every word in the sentence and in keyword list.
    for word in sent:
        max_sim = 0
        word_count += 1
        for kword in keywords:
            if word in model.vocab and kword in model.vocab:
                if model.similarity(word,kword) > max_sim:
                    max_sim = model.similarity(word,kword)

        avg_sim += max_sim

    avg_sim /= word_count
    prof_sim.append(avg_sim)
```

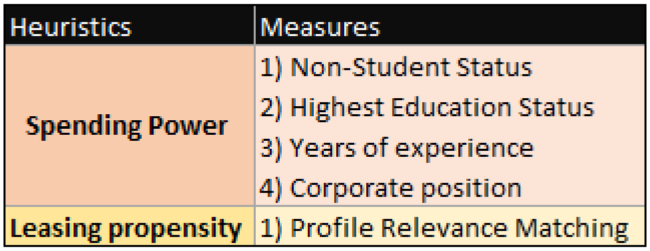
Figure 49: Code snippet of extracting relevance score from profiles

4.2.3. Analysis

After searching and identifying the target market, we needed to define the heuristics that were going to be used to determine the potential of each individual prospective customer identified. These prospective customers were ranked in accordance to how useful they were to DFM; the more likely they were to engage DFM's leasing services, the higher they scored.

Our heuristics involved identifying 2 aspects of an individual: Spending power and Leasing propensity.

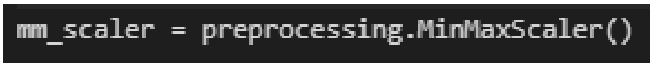
Each measure is scored from a score of 0 to 1, and each measure in the defined heuristics (spending power and leasing propensity) is summed up and standardized to a score of 0 to 1. In other words, spending power will have a total score of one and leasing propensity will have a total score of one. Hence, the total score we can obtain here is a total of 2, with an equal weightage to both spending power and leasing propensity. We breakdown our scoring algorithm and identify which measures fall under which heuristic *(Refer to Figure 50)*.



Heuristics	Measures
Spending Power	1) Non-Student Status 2) Highest Education Status 3) Years of experience 4) Corporate position
Leasing propensity	1) Profile Relevance Matching

Figure 50: The measures and heuristics to rank our users

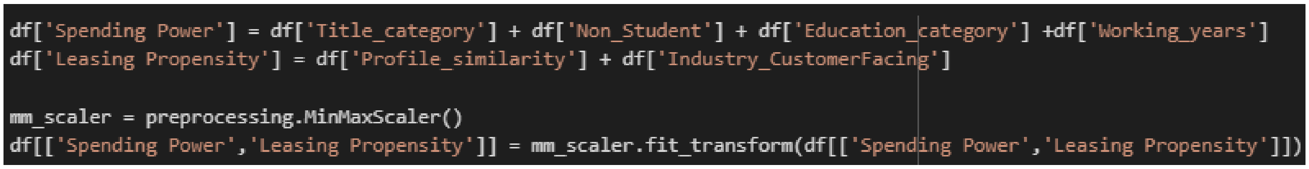
As mentioned, we scale all the measures (4 for spending power and 1 for leasing propensity), to a value of 0 to 1.



```
mm_scaler = preprocessing.MinMaxScaler()
```

Figure 51: Applying Min Max Scaler to normalize the scores

Then, the measures are summed up for the respective heuristics and normalized again to have a score of 1 for each heuristic.

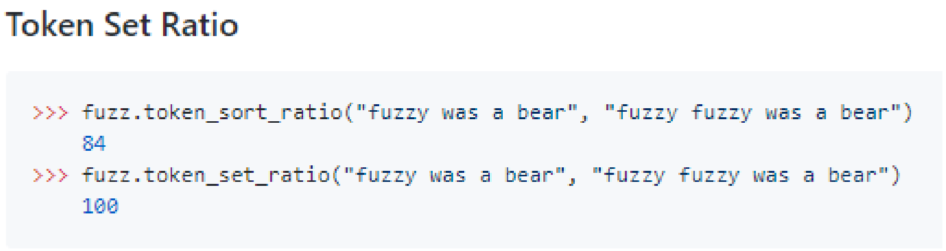


```
df['Spending Power'] = df['Title_category'] + df['Non_Student'] + df['Education_category'] +df['Working_years']
df['Leasing Propensity'] = df['Profile_similarity'] + df['Industry_CustomerFacing']

mm_scaler = preprocessing.MinMaxScaler()
df[['Spending Power','Leasing Propensity']] = mm_scaler.fit_transform(df[['Spending Power','Leasing Propensity']])
```

Figure 52: Code Snippet to sum up relevant measures to each heuristic and normalize their scores

For the following measures: Highest education status and corporate position, we performed fuzzy matching, a technique that allows us to match text in their profile with our own defined set of keywords. This technique allows us to standardize our profiles. The resulting output will be a feature in their profile (job title or education) being mapped to our keywords and a similarity matching score out of a maximum of 100. The similarity score allows us to know how similar their profile is to our keywords (in percentage). In this fuzzy matching process, we are using the token set ratio, which will give a score of

100 even if the strings don’t match completely, but our text is a subtext belonging to a keyword we want to match *(Refer to Figure 53).*

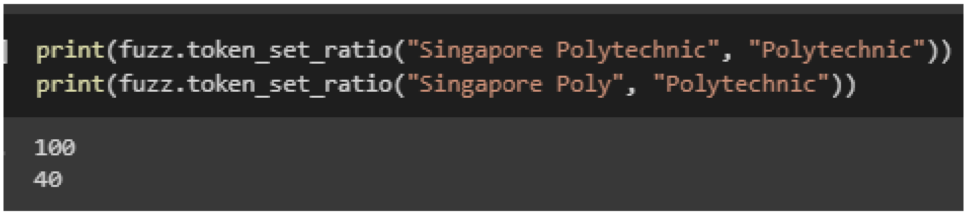


Token Set Ratio

```
>>> fuzz.token_sort_ratio("fuzzy was a bear", "fuzzy fuzzy was a bear")
    84
>>> fuzz.token_set_ratio("fuzzy was a bear", "fuzzy fuzzy was a bear")
    100
```

Figure 53: Code snippet to show the differences between 2 algorithms of fuzzy matching. We are using the latter

We acknowledge that if the column is unmatchable (I.e. noisy text), it will return an inaccurate match. Hence, we will multiply the score with weights, and these weights are the matching scores given by the fuzzy matching library. If the education column of profile 1, for example, is ‘Singapore Polytechnic’, and we want to match to a keyword of ‘Polytechnic’, then it will give a score of 100. If profile 2’s education states ‘Singapore Poly’, it will give a lower score *(Refer to Figure 54)*.



```
print(fuzz.token_set_ratio("Singapore Polytechnic", "Polytechnic"))
print(fuzz.token_set_ratio("Singapore Poly", "Polytechnic"))
```

```
100
40
```

Figure 54: Code snippet of fuzzy matching comparing 2 strings

Hence, by multiplying the similarity scores with an assigned score, and taking 2 for an example, we will get a score of 2 * 100 = 200 for the first profile, while the second profile receives a 40 * 2 = 80. This serves as a confidence level for us, as we do not want to mistakenly match a wrong profile to a high score, which will result in a waste of time and effort.

4.2.3.1 Spending Power

Spending power refers to the degree to which an individual has money to purchase products and services. In our context, we can know to what extent they are likely to lease from us. Identifying their spending power gives us an idea of what type of cars they can afford to lease, or if any at all. The spending power is represented by a score from 0 to 1, with 1 having the highest spending power. In this section, we have identified 4 measures to gauge their spending power: Non-student status, Highest education status, years of experience and corporate position.

4.2.3.1.1 Non-Student Status

If an individual is a student, he/she will be more likely to have a lower spending power. We find out if the user is a student or not by taking their latest education profile. If the end date of their education has past current year of 2019, we will mark them as students. Else, they would not assume the status of a student.

We observe the distribution of students to non-students. This measure is stored under the column 'Non_student', where a value of 1 indicates the profile is not a student, and 0 if the profile is a student. We observe that there are about 500 students to 700 non-students.

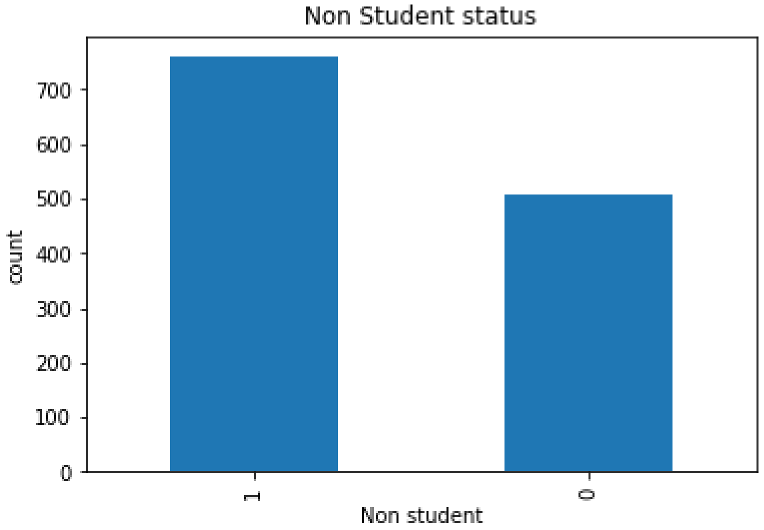
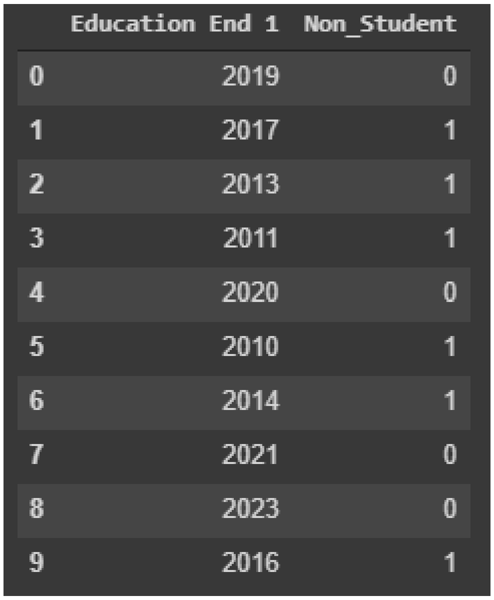


Figure 55: Comparison of the number students to non-students in our dataset



	Education End 1	Non_Student
0	2019	0
1	2017	1
2	2013	1
3	2011	1
4	2020	0
5	2010	1
6	2014	1
7	2021	0
8	2023	0
9	2016	1

Figure 56: Sample of 10 profiles with their education end date and student status

4.2.3.1.2 Highest Education Status

We aim to find out the highest education that the user has obtained. This will roughly gauge the potential of their career which translates to their rough remuneration. We performed fuzzy matching to match their level of education using keywords such as university, polytechnic or junior college, to their latest education profile. In determining this measure, we assume that higher education means higher career estimated progress, which will translate to a better remuneration and spending power. Hence, a higher education status (4 levels) will give us a score of 1 to 4, with 4 being the highest education status which will reflect as a higher score on spending power.

We can observe that university as the highest education obtained being the most prominent, with around 900(70%) of our profiles being in this category, followed by around 200 (15%) having a category of 1, which is classified as primary, secondary education or others. Category of 3 (High school or polytechnic equivalent) and 2(Junior colleges) have a lesser amount.

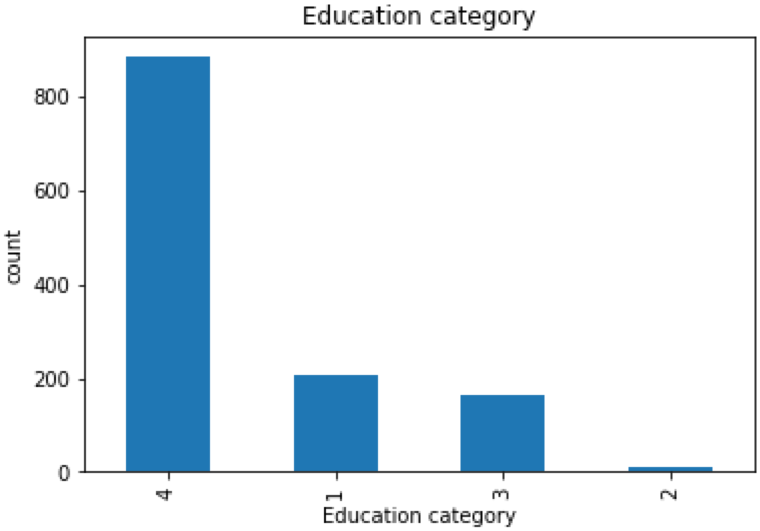
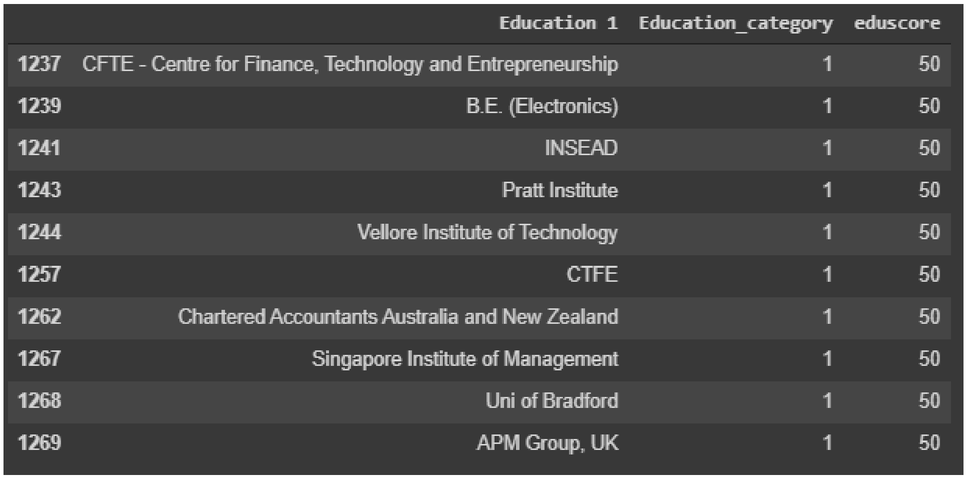


Figure 57: Distribution of education categories

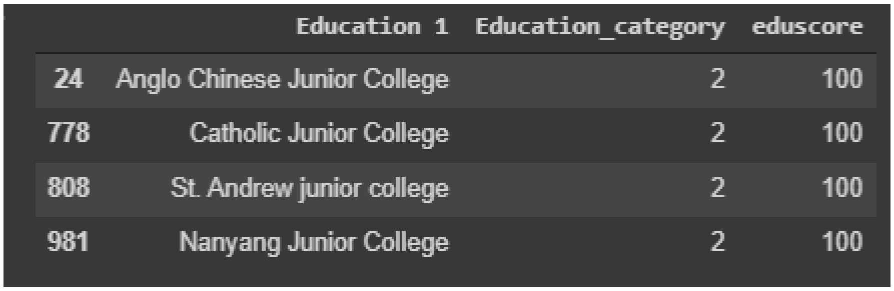
We look at the last 10 values from each of the education category to check if the assignment of education category is reasonable. For Education category 1, we observe that the matching scores are low (below 40). Observing the list, we can see that most of them are schools that are not as prominent or popular in the context of Singapore or are wrongly entered school names.



	Education 1	Education_category	eduscore
1237	CFTE - Centre for Finance, Technology and Entrepreneurship	1	50
1239	B.E. (Electronics)	1	50
1241	INSEAD	1	50
1243	Pratt Institute	1	50
1244	Vellore Institute of Technology	1	50
1257	CTFE	1	50
1262	Chartered Accountants Australia and New Zealand	1	50
1267	Singapore Institute of Management	1	50
1268	Uni of Bradford	1	50
1269	APM Group, UK	1	50

Figure 58: Sample of 10 profiles of Education category 1

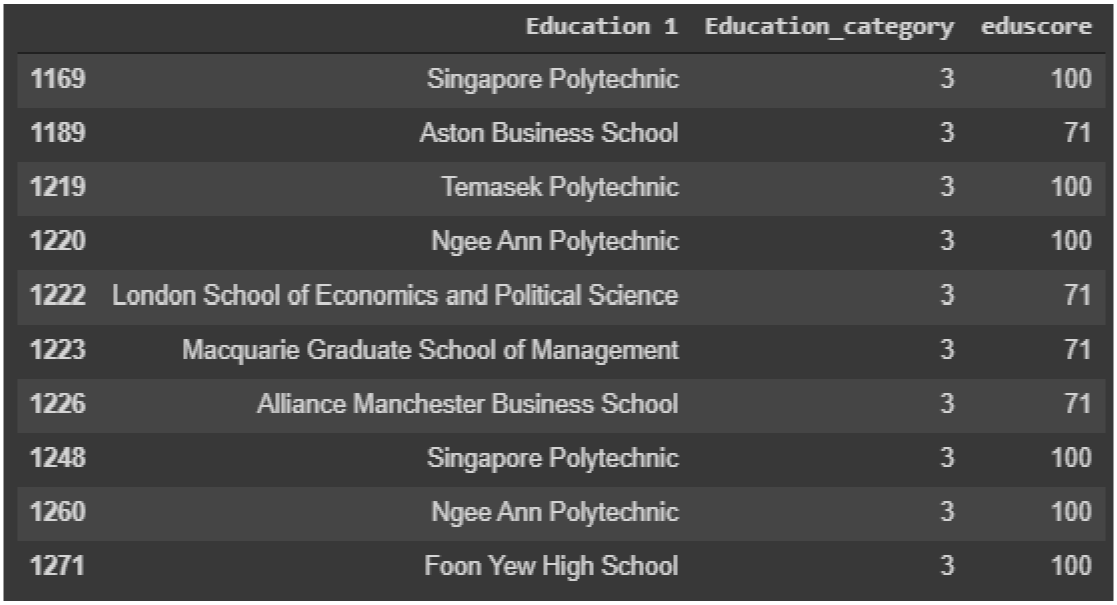
Education category 2 shows an exact match of score 100 for all 4 schools. Only 4 profiles have ‘Junior colleges’ as their highest education, which is reasonable because junior college students usually go on to study university.



	Education 1	Education_category	eduscore
24	Anglo Chinese Junior College	2	100
778	Catholic Junior College	2	100
808	St. Andrew junior college	2	100
981	Nanyang Junior College	2	100

Figure 59: Sample of 10 profiles of Education category 2 that only had 4 of such users

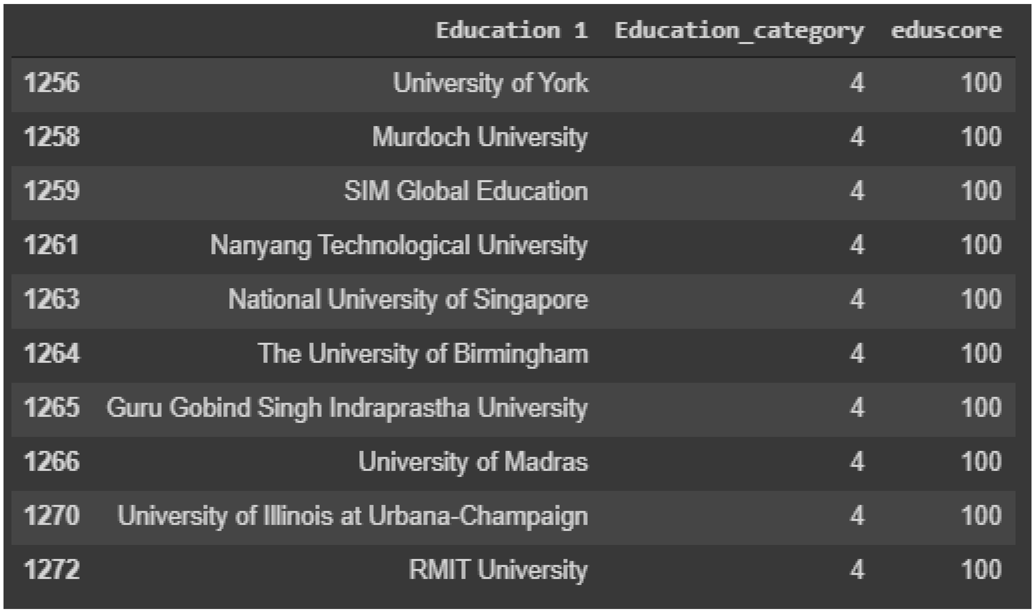
For education category 3, we observe most of the high school and polytechnic matches. For the 4 records of matching score 71, we observe that they should be categorized as universities. However, since the word ‘school’ appear in their names, they are mapped to 3 instead (similar to our defined keyword of ‘high school’). Since an education category of 3 and 4 do not differ much, and there are not a lot of such cases, we will ignore it.



	Education 1	Education_category	eduscore
1169	Singapore Polytechnic	3	100
1189	Aston Business School	3	71
1219	Temasek Polytechnic	3	100
1220	Ngee Ann Polytechnic	3	100
1222	London School of Economics and Political Science	3	71
1223	Macquarie Graduate School of Management	3	71
1226	Alliance Manchester Business School	3	71
1248	Singapore Polytechnic	3	100
1260	Ngee Ann Polytechnic	3	100
1271	Foon Yew High School	3	100

Figure 60: Sample of 10 profiles of Education category 3

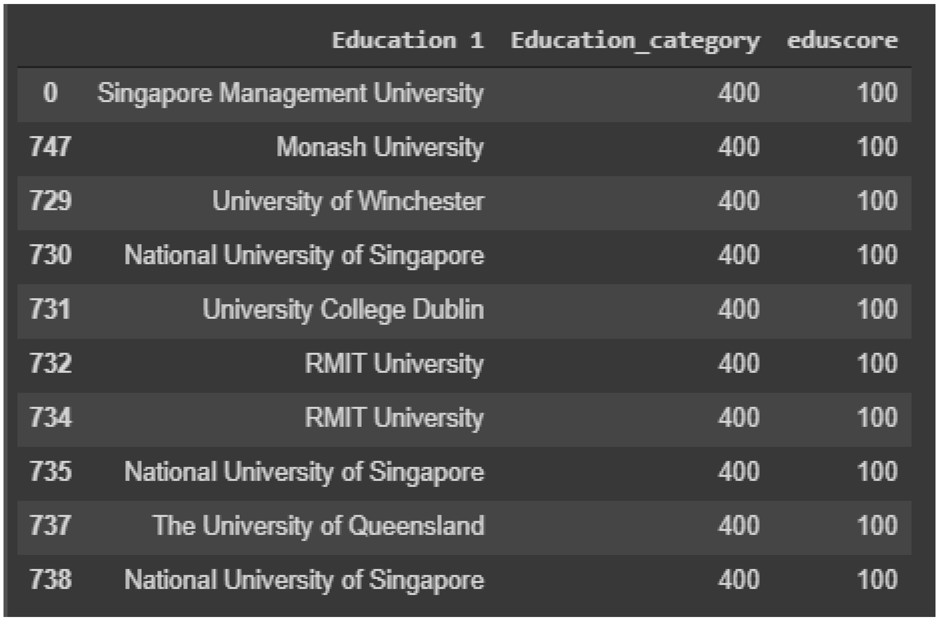
Education 4 sees a full matching score of 100 and all schools identified are at the university level.



	Education 1	Education_category	eduscore
1256	University of York	4	100
1258	Murdoch University	4	100
1259	SIM Global Education	4	100
1261	Nanyang Technological University	4	100
1263	National University of Singapore	4	100
1264	The University of Birmingham	4	100
1265	Guru Gobind Singh Indraprastha University	4	100
1266	University of Madras	4	100
1270	University of Illinois at Urbana-Champaign	4	100
1272	RMIT University	4	100

Figure 61: Sample of 10 profiles of Education category 4

As mentioned in the last paragraphs of 4.2.3 Analysis, we will apply weights on the scores to give us a sense of how confident the algorithm is, in predicting the education scores. In the diagram below, we can observe that if their highest education is university and the matching score is high, they will be assigned a max score of 400.



	Education 1	Education_category	eduscore
0	Singapore Management University	400	100
747	Monash University	400	100
729	University of Winchester	400	100
730	National University of Singapore	400	100
731	University College Dublin	400	100
732	RMIT University	400	100
734	RMIT University	400	100
735	National University of Singapore	400	100
737	The University of Queensland	400	100
738	National University of Singapore	400	100

Figure 62: Sample of 10 profiles which showcases the maximum score attainable based on Education

4.2.3.1.3 Years of Experience

Find out the number of years of experience by subtracting present time or last worked year, by the year the user first worked in, i.e. their first job. We are able to gauge the level of their spending power by looking at the years of experience in the workforce. The higher the years of experience, the greater the likelihood of them having a higher spending power and this allows us to differentiate the profiles.

We observe that majority of them have a working experience of less than 10 years. We then bin it into a range of 5 and observed that around 550 (42.7%) have a working experience of 0 to 5 years, and 400 (31.1%) is between 5 to 10 years, and around 150(11.7%) is between 10 to 15 years. Hence, around 85% of our data has 15 years of experience and below, showing the 80-20 rule at work. The rest of the profiles have working years of 15 to 40 years, which then takes up the remaining 15% of our profiles.

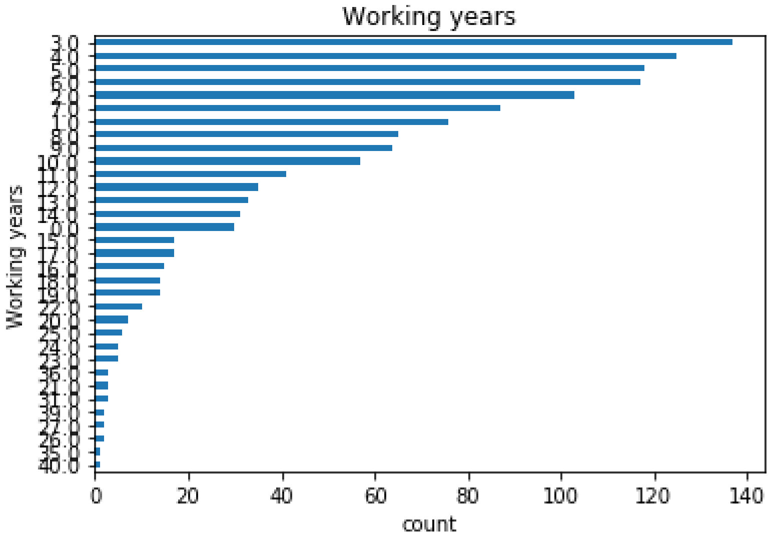


Figure 63: Working years by count

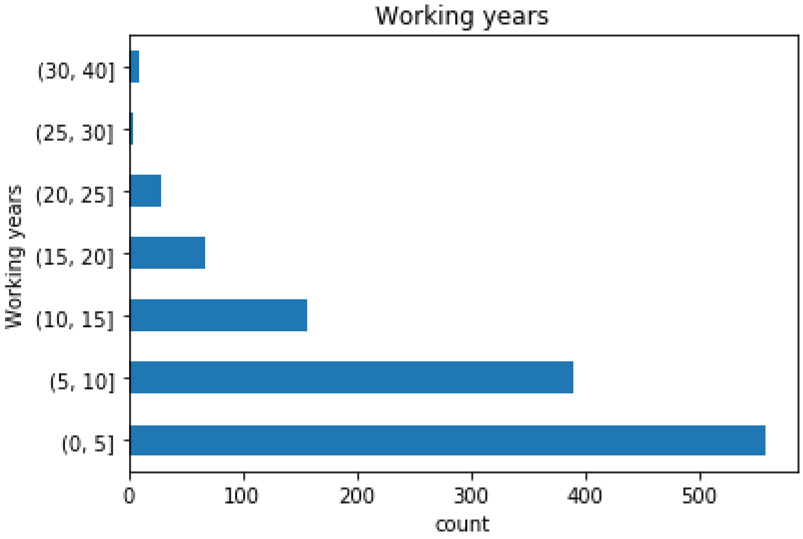
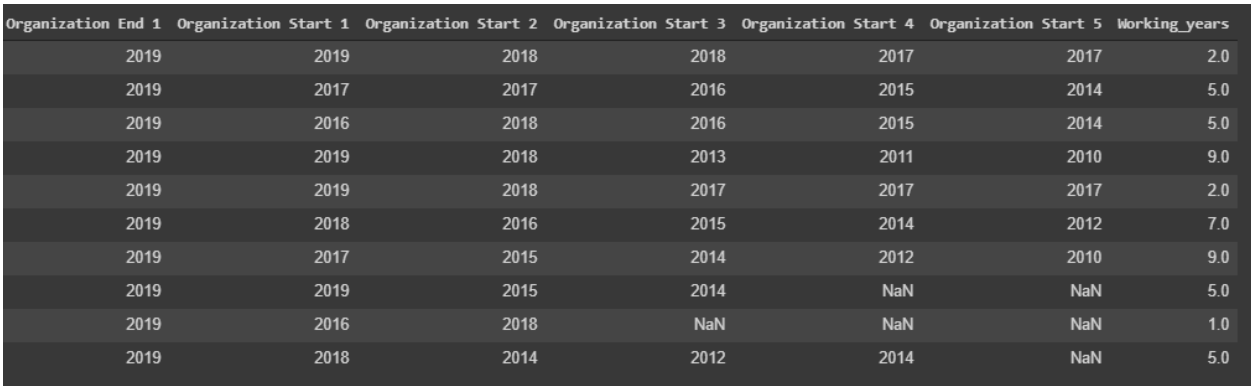


Figure 64: Working years count by binned working years



Organization End 1	Organization Start 1	Organization Start 2	Organization Start 3	Organization Start 4	Organization Start 5	Working_years
2019	2019	2018	2018	2017	2017	2.0
2019	2017	2017	2016	2015	2014	5.0
2019	2016	2018	2016	2015	2014	5.0
2019	2019	2018	2013	2011	2010	9.0
2019	2019	2018	2017	2017	2017	2.0
2019	2018	2016	2015	2014	2012	7.0
2019	2017	2015	2014	2012	2010	9.0
2019	2019	2015	2014	NaN	NaN	5.0
2019	2016	2018	NaN	NaN	NaN	1.0
2019	2018	2014	2012	2014	NaN	5.0

Figure 65: Sample of 10 profiles with their working years

4.2.3.1.4 Corporate Position

Find the job title and standardize it to 3 levels: Director, manager or associate level. A higher corporate position will likely to translate to higher spending power as well. Hence, a higher corporate position will be assigned a score of 3, while the lowest position will be assigned a score of 1. There are 3 levels to allow us to distinguish between the higher spending powers with the lower spending power profiles.

We can observe that most of the corporate position are grouped into 1, which means the lowest ranking in the corporate world. This aligns with our findings of having 85% of our profiles having 15 working years' experience and less.

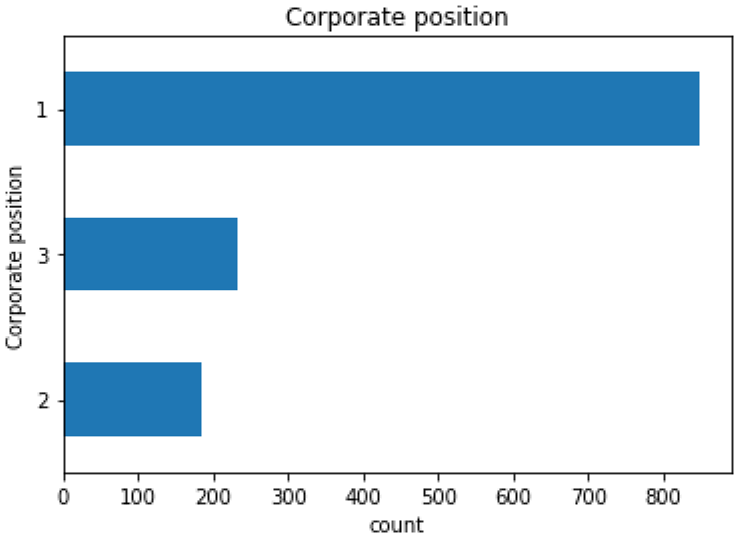
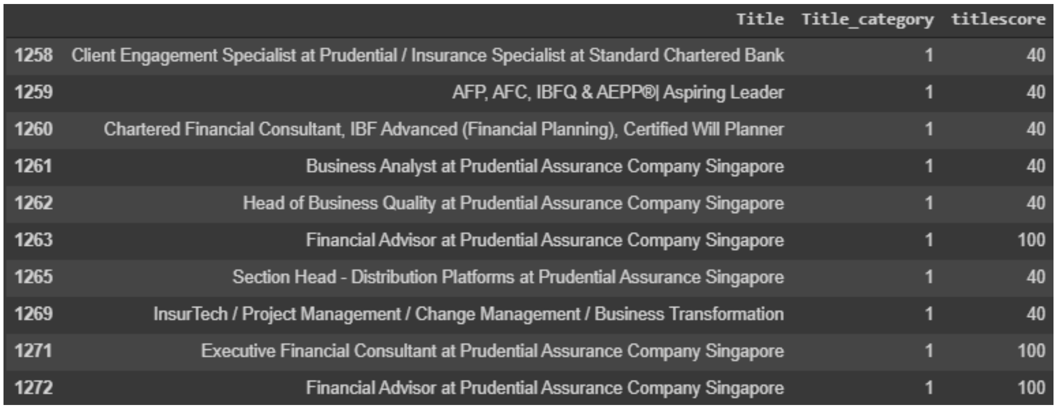


Figure 66: Corporate Positions by Count

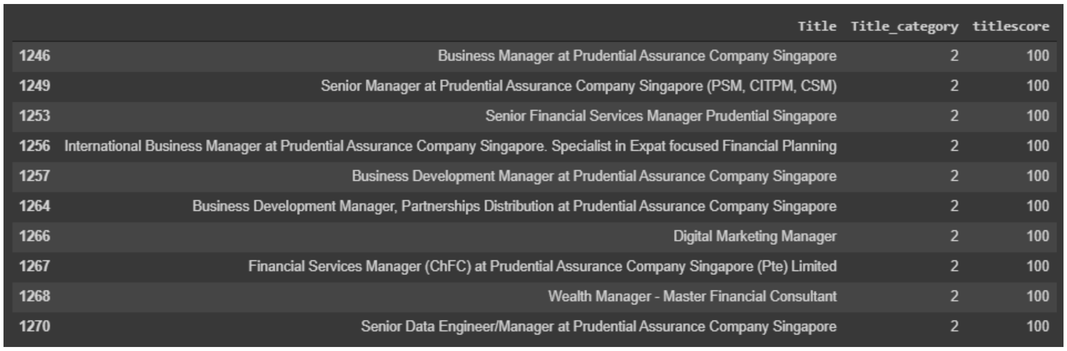
After exploring how many of our profiles get categorized into corporate position category 1, we observed that most of the profiles have a low matching score of 40. This means that there are many profiles out there that are not matchable, so we will assume that they have a score of 1.



	Title	Title_category	titlescore
1258	Client Engagement Specialist at Prudential / Insurance Specialist at Standard Chartered Bank	1	40
1259	AFP, AFC, IBFQ & AEPP®| Aspiring Leader	1	40
1260	Chartered Financial Consultant, IBF Advanced (Financial Planning), Certified Will Planner	1	40
1261	Business Analyst at Prudential Assurance Company Singapore	1	40
1262	Head of Business Quality at Prudential Assurance Company Singapore	1	40
1263	Financial Advisor at Prudential Assurance Company Singapore	1	100
1265	Section Head - Distribution Platforms at Prudential Assurance Singapore	1	40
1269	InsurTech / Project Management / Change Management / Business Transformation	1	40
1271	Executive Financial Consultant at Prudential Assurance Company Singapore	1	100
1272	Financial Advisor at Prudential Assurance Company Singapore	1	100

Figure 67: 10 Profiles that were categorized into corporate position category 1

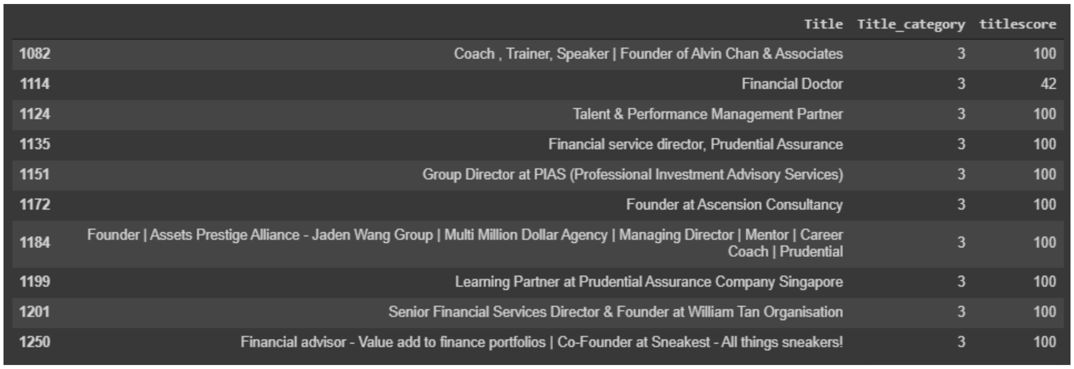
A score of 2 shows that most profiles are matchable with a score of 100, and they are of the managerial level.



	Title	Title_category	titlescore
1246	Business Manager at Prudential Assurance Company Singapore	2	100
1249	Senior Manager at Prudential Assurance Company Singapore (PSM, CITPM, CSM)	2	100
1253	Senior Financial Services Manager Prudential Singapore	2	100
1256	International Business Manager at Prudential Assurance Company Singapore. Specialist in Expat focused Financial Planning	2	100
1257	Business Development Manager at Prudential Assurance Company Singapore	2	100
1264	Business Development Manager, Partnerships Distribution at Prudential Assurance Company Singapore	2	100
1266	Digital Marketing Manager	2	100
1267	Financial Services Manager (ChFC) at Prudential Assurance Company Singapore (Pte) Limited	2	100
1268	Wealth Manager - Master Financial Consultant	2	100
1270	Senior Data Engineer/Manager at Prudential Assurance Company Singapore	2	100

Figure 68: 10 Profiles that were categorized into corporate position category 2

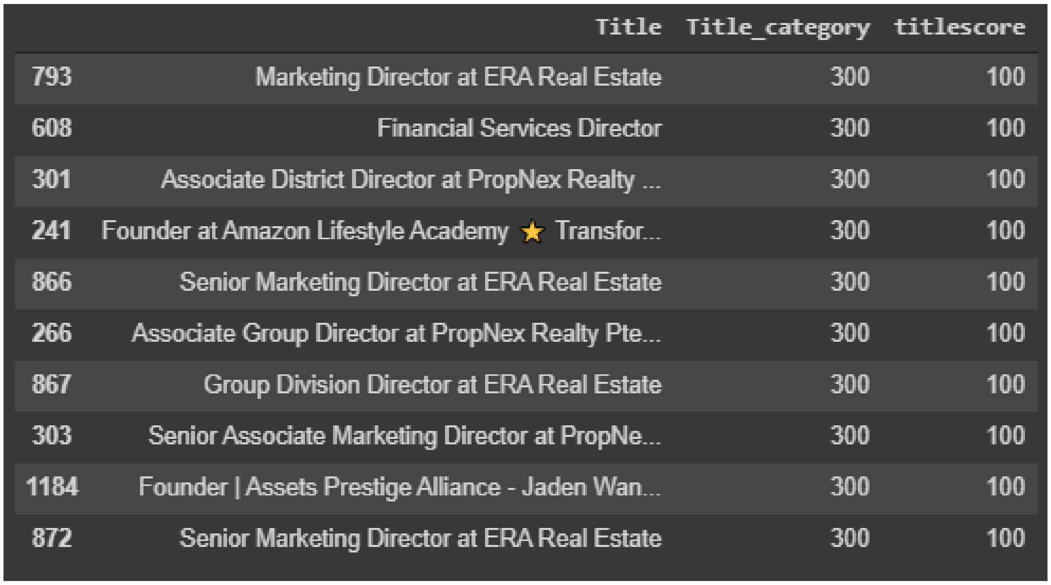
Lastly, all scores of 3 shows that the profiles are of director, founder or partner level, with most matching score of 100.



	Title	Title_category	titlescore
1082	Coach , Trainer, Speaker | Founder of Alvin Chan & Associates	3	100
1114	Financial Doctor	3	42
1124	Talent & Performance Management Partner	3	100
1135	Financial service director, Prudential Assurance	3	100
1151	Group Director at PIAS (Professional Investment Advisory Services)	3	100
1172	Founder at Ascension Consultancy	3	100
1184	Founder | Assets Prestige Alliance - Jaden Wang Group | Multi Million Dollar Agency | Managing Director | Mentor | Career Coach | Prudential	3	100
1199	Learning Partner at Prudential Assurance Company Singapore	3	100
1201	Senior Financial Services Director & Founder at William Tan Organisation	3	100
1250	Financial advisor - Value add to finance portfolios | Co-Founder at Sneakest - All things sneakers!	3	100

Figure 69: 10 Profiles that were categorized into corporate position category 3

Likewise, in highest education status, we multiply the scores by their weight to get a score that reflects the confidence levels. In the diagram below, we observe a maximum score of 300 if they are of a director level and have a 100 matching score.



	Title	Title_category	titlescore
793	Marketing Director at ERA Real Estate	300	100
608	Financial Services Director	300	100
301	Associate District Director at PropNex Realty ...	300	100
241	Founder at Amazon Lifestyle Academy ★ Transfor...	300	100
866	Senior Marketing Director at ERA Real Estate	300	100
266	Associate Group Director at PropNex Realty Pte...	300	100
867	Group Division Director at ERA Real Estate	300	100
303	Senior Associate Marketing Director at PropNe...	300	100
1184	Founder | Assets Prestige Alliance - Jaden Wan...	300	100
872	Senior Marketing Director at ERA Real Estate	300	100

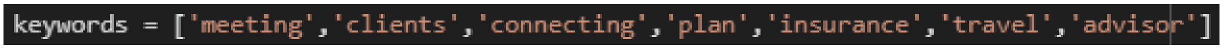
Figure 70: 10 Profiles that obtained the maximum weighted score of 300

4.2.3.2 Leasing propensity

The leasing propensity describes the likelihood that they will want to lease cars. This likelihood is determined by their needs. The leasing propensity is represented by a score from 0 to 1, with 1 having the highest leasing propensity.

4.2.3.2.1 Profile Matching

We also did a similarity score matching of the profiles, to see if any predefined keywords *(Refer to Figure 71)* are being mentioned in their profiles. For instance, if their profiles mention 'meeting' and 'connecting' more often, then we can infer that they are likely to need a vehicle as their job scope involves them having to travel to meet clients.

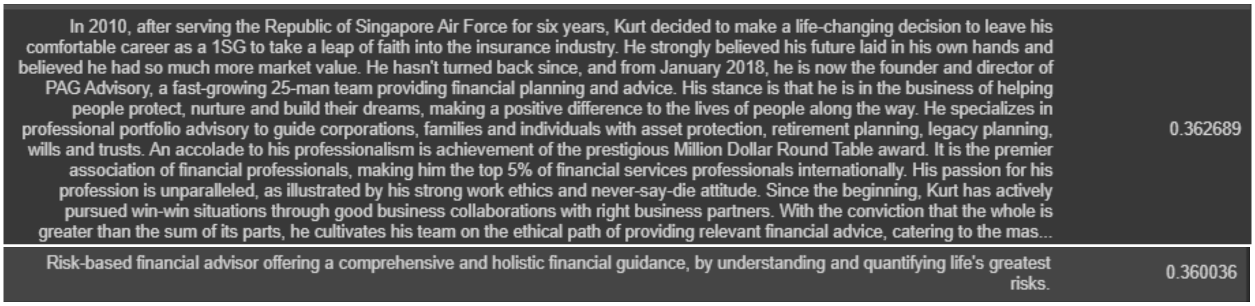


```
keywords = ['meeting','clients','connecting','plan','insurance','travel','advisor']
```

Figure 71: List of predefined keywords used for similarity score matching of profiles

A higher score will mean that their bio is more relevant to what we are interested in, particularly individuals who meet people often. We took a pre-trained word embeddings model, which allows us to identify the cosine similarities between our identified keywords and their profile. Each word in their profile is matched with our keyword, and the average is taken.

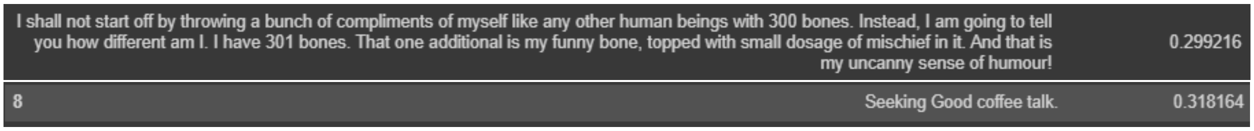
For profiles with a higher similarity score, we observe the following 2 profiles and observed that they have mentioned terms like ‘insurance’, ‘financial planning’, ‘financial advice’, which will all be highly related to our keywords, which suggests that they are financial or insurance agents and will have the need to travel around and connect frequently.



In 2010, after serving the Republic of Singapore Air Force for six years, Kurt decided to make a life-changing decision to leave his comfortable career as a 1SG to take a leap of faith into the insurance industry. He strongly believed his future laid in his own hands and believed he had so much more market value. He hasn't turned back since, and from January 2018, he is now the founder and director of PAG Advisory, a fast-growing 25-man team providing financial planning and advice. His stance is that he is in the business of helping people protect, nurture and build their dreams, making a positive difference to the lives of people along the way. He specializes in professional portfolio advisory to guide corporations, families and individuals with asset protection, retirement planning, legacy planning, wills and trusts. An accolade to his professionalism is achievement of the prestigious Million Dollar Round Table award. It is the premier association of financial professionals, making him the top 5% of financial services professionals internationally. His passion for his profession is unparalleled, as illustrated by his strong work ethics and never-say-die attitude. Since the beginning, Kurt has actively pursued win-win situations through good business collaborations with right business partners. With the conviction that the whole is greater than the sum of its parts, he cultivates his team on the ethical path of providing relevant financial advice, catering to the mas...	0.362689
Risk-based financial advisor offering a comprehensive and holistic financial guidance, by understanding and quantifying life's greatest risks.	0.360036

Figure 72: Illustration showing the scores of different bios

We have also observed profiles with relatively lower similarity score, and they are profiles with words that are completely unrelated to our keywords.



	I shall not start off by throwing a bunch of compliments of myself like any other human beings with 300 bones. Instead, I am going to tell you how different am I. I have 301 bones. That one additional is my funny bone, topped with small dosage of mischief in it. And that is my uncanny sense of humour!	0.299216
8	Seeking Good coffee talk.	0.318164

Figure 73: Illustration showing bios with relatively lower profile similarity scores

4.2.4 Interesting Finding Insights

We observe that the final scores follow a normal distribution *(Refer to Figure 69)*, with scores of 1 to 1.25 being the most common, with a frequency count of 350 (27%). We decide to **target the top 10%** of the users based on their final scores, which is equivalent to around 128 profiles. The number of **profiles in bins 1.5 to 1.75 and 1.75 to 2 adds up to 153,** so we will focus on these 2 bins to narrow down our search for high value prospective customers.

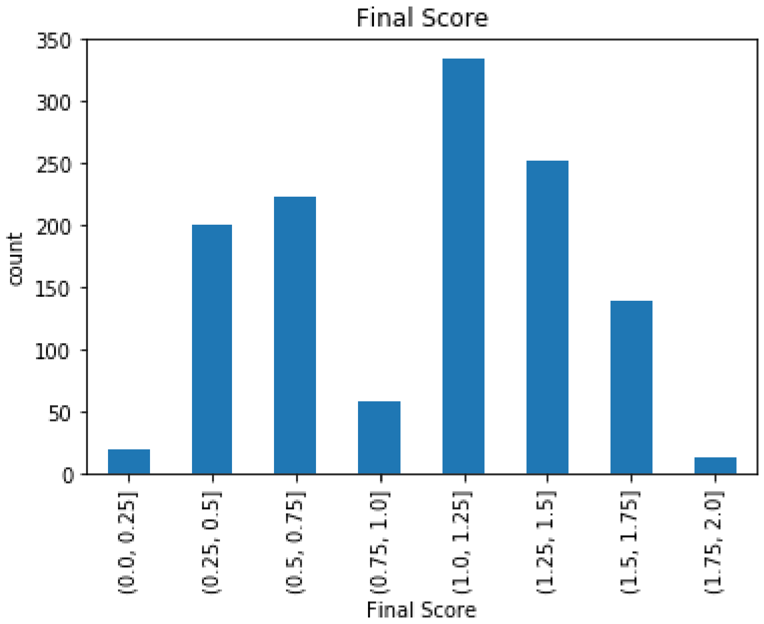


Figure 74: Distribution of final scores of users

After extracting the top 155 profiles (score of 1.5 and above), we proceed to analyze their characteristics.

Organization

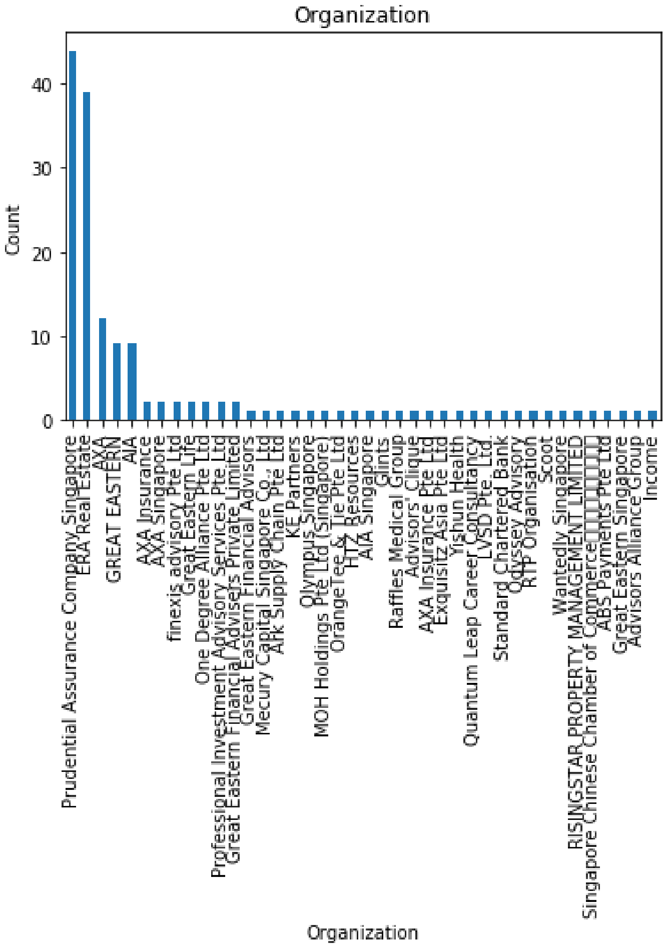


Figure 75: Distribution of the different organizations within the upper score bins

We observe **most of the organizations are from insurance companies**. To focus on which companies to target, we select the top 5.

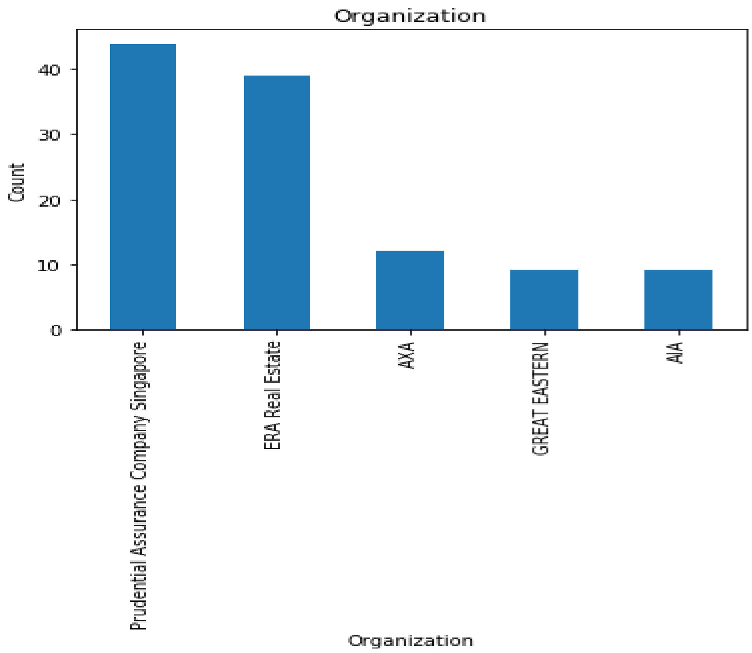


Figure 76: Top 5 companies within the top scoring profiles

Around 43 profiles belong to Prudential, which makes up about 50% of our top profiles. ERA also comes close, with around 38 profiles. The rest of the profiles belong to AXA, Great Eastern and AIA. Since only Prudential and ERA has a significant number compared to the rest, we will choose these 2 companies to target.

<u>Working years</u>

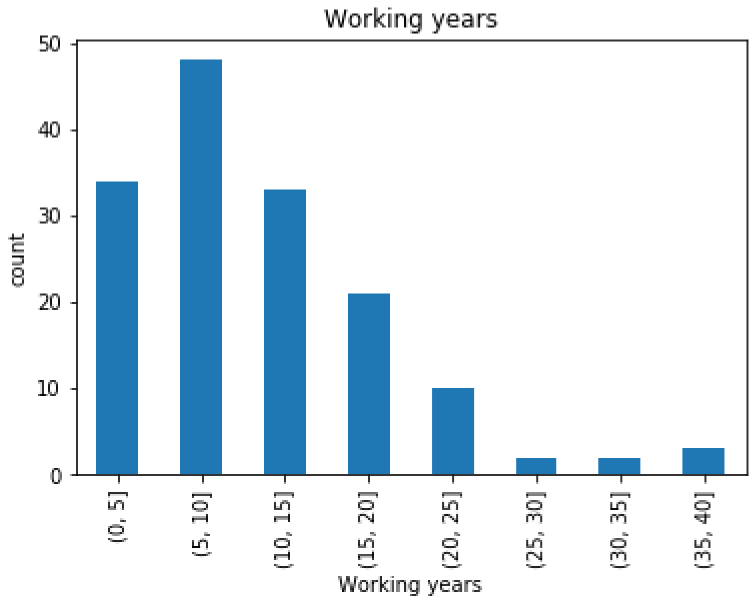


Figure 77: Distribution of the working years of the profiles

We plotted the working years of these top profiles by binning the working years in bins of 5, and majority of them have 5 to 10 years of working experience, with about 60 profiles (38%). Then, this is followed by profiles who have 0 to 5 working years and 10 to 15 working years. Since we know majority of our target audience have a working experience of 0 to 15 years, we can assume that they will be in their **early twenties to late thirties.** Hence, DFM can cater its vehicle fleet accordingly when targeting this age group and keeping the estimated spending power of this age group in mind. Profiles in this age group suggests that they may prefer a modern or sporty looking vehicle among the tier of car class that their spending power can afford.

Corporate position

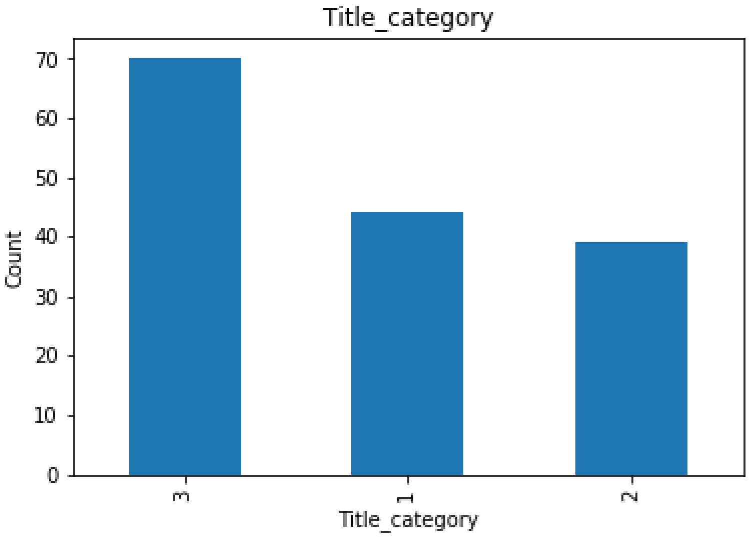
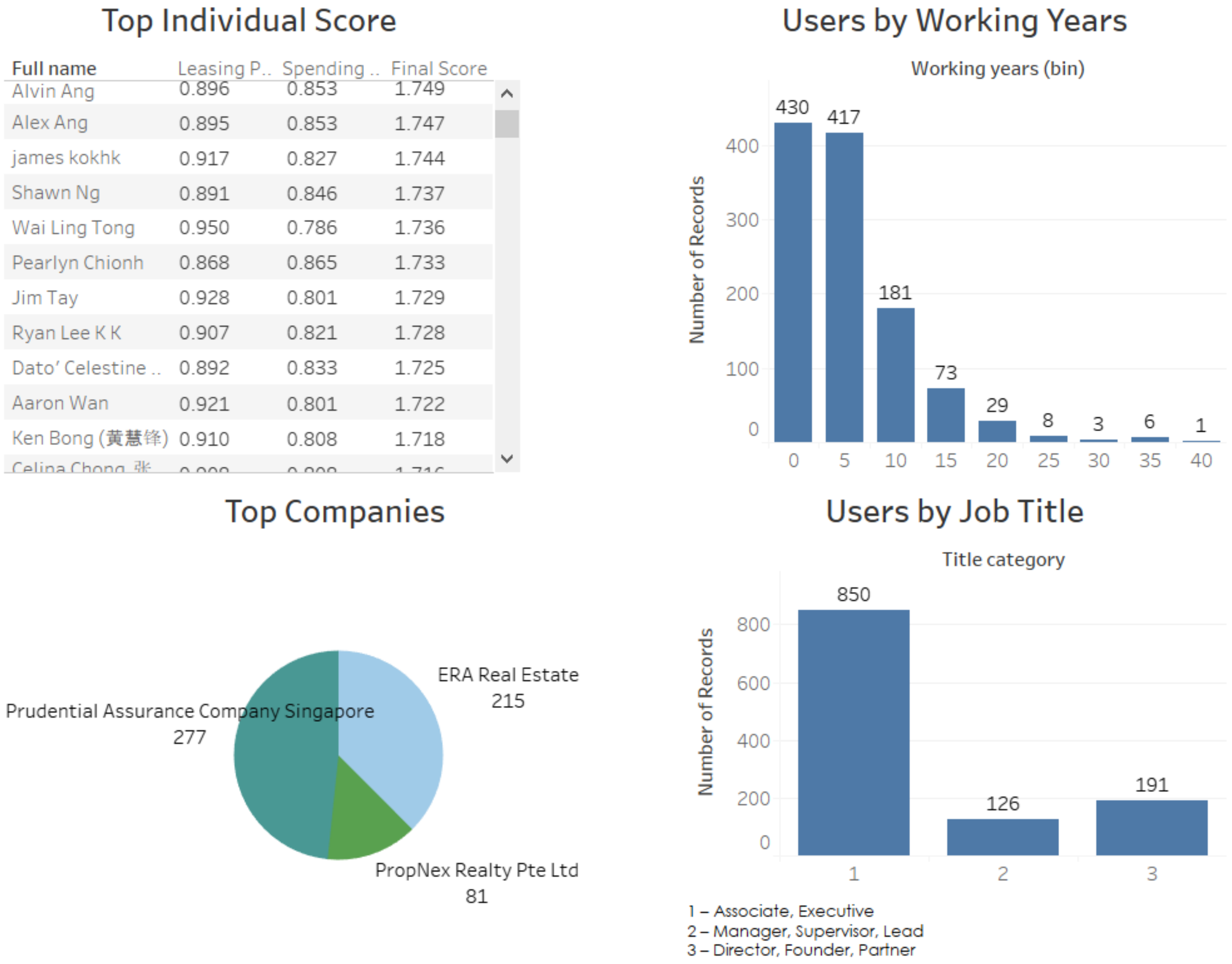


Figure 78: Distribution of the Corporate positions assigned to the users

We also explored the title categories of these profiles, with 1 being Associate or executives, 2 being managers or supervisors, and 3 being directors, founders or partners. We observe that most of our profiles belong to 3, which means that they will **have the spending power** to afford leasing slightly more expensive models.

After doing all these analyses, we wanted to help DFM employees be able to visualize these findings better, thus we created a Tableau dashboard for them to use *(Refer to Figure 79).* DFM will be able to see the individual heuristic scores to identify which are potential high valued customers for their leasing services. In addition, DFM can look at an overview of the job titles, working years and companies of these users



Top Individual Score

Full name	Leasing P..	Spending ..	Final Score
Alvin Ang	0.896	0.853	1.749
Alex Ang	0.895	0.853	1.747
james kokhk	0.917	0.827	1.744
Shawn Ng	0.891	0.846	1.737
Wai Ling Tong	0.950	0.786	1.736
Pearlyn Chionh	0.868	0.865	1.733
Jim Tay	0.928	0.801	1.729
Ryan Lee K K	0.907	0.821	1.728
Dato' Celestine ..	0.892	0.833	1.725
Aaron Wan	0.921	0.801	1.722
Ken Bong (黄慧锋)	0.910	0.808	1.718
Celina Chong 张	0.908	0.808	1.716

Figure 79: Tableau Dashboard showcasing insights from LinkedIn analysis

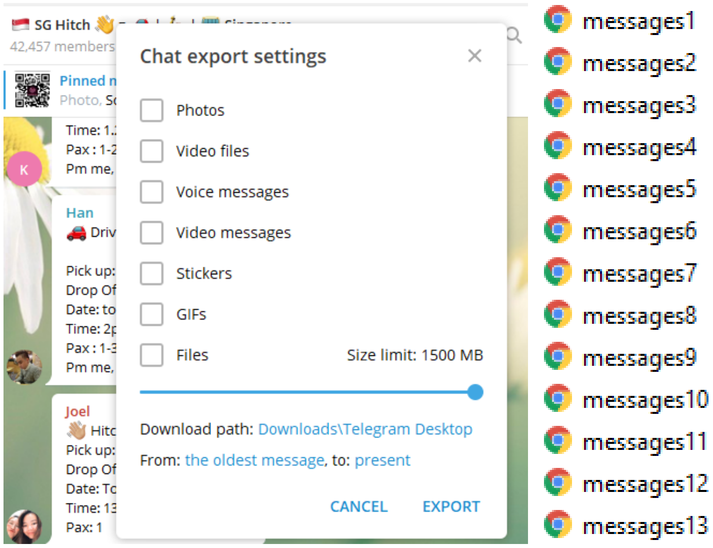
4.3. Telegram

Telegram has many public groups consisting of large multitudes of audiences with differing characteristics. These people often join these interest groups for a specific purpose or interest, thus forming a possible community of prospective customers.

For example, from the telegram group SGHITCH, we were able to identify prospective customers with a **higher propensity of car leasing as they are frequent users of car hitching services.** From this group, we were able to retrieve details of the hitches posted in the group such as the Pickup location, Dropoff location, Date, Time and Number of Pax. Based on the aforementioned information, we were able to calculate useful metrics which can give insights into the prospective customer's characteristics such as how many times the user posted for a hitching service and the distance travelled between the rides.

4.3.1. Collecting

We used Telegram Desktop to extract all the chat history of SGHITCH. The chat history spans over 3 months and are in html files. We created a script to save these data into a csv file first for easier data cleaning and analysis.



Chat export settings

□ Photos
□ Video files
□ Voice messages
□ Video messages
□ Stickers
□ GIFs
□ Files   Size limit: 1500 MB

Download path: Downloads\Telegram Desktop
From: the oldest message, to: present

CANCEL   EXPORT

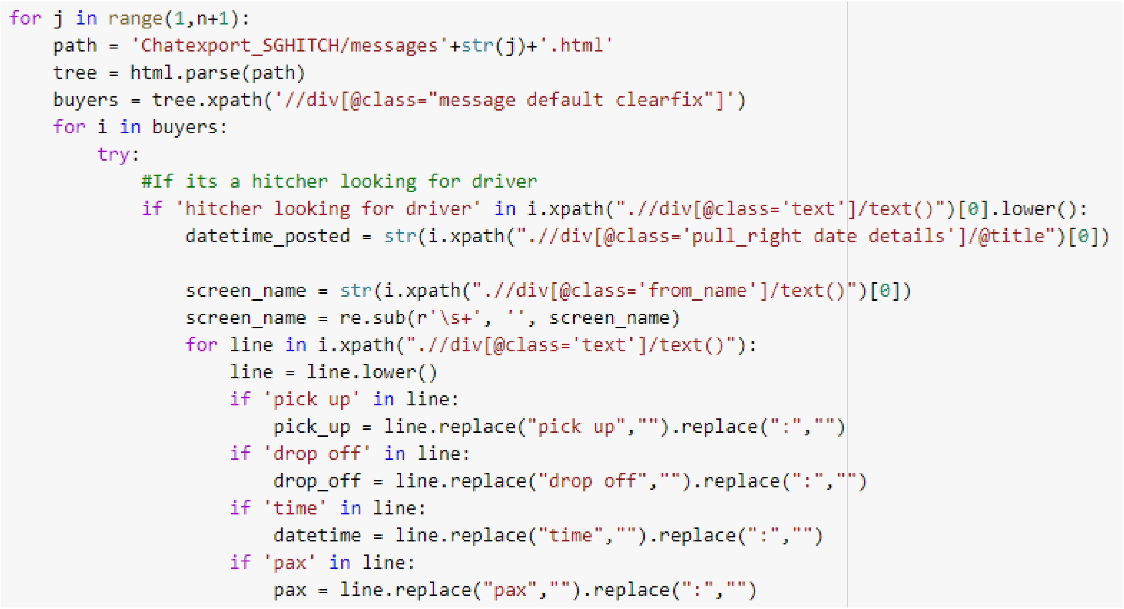
messages1
messages2
messages3
messages4
messages5
messages6
messages7
messages8
messages9
messages10
messages11
messages12
messages13

Figure 80: Exporting the Telegram group chat history through its chat export settings

4.3.2. Cleaning

4.3.2.1. Extracting, Transforming & Loading Data into CSV format

For each file, we used xpath to detect each user's telegram handle and extract their pickup location, drop off location, the date and time posted and estimated number of pax.

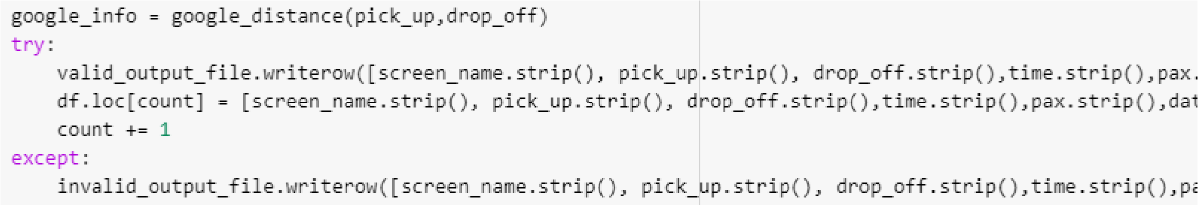


```
for j in range(1,n+1):
    path = 'Chatexport_SGHITCH/messages'+str(j)+'.html'
    tree = html.parse(path)
    buyers = tree.xpath('//div[@class="message default clearfix"]')
    for i in buyers:
        try:
            #If its a hitcher looking for driver
            if 'hitcher looking for driver' in i.xpath(".//div[@class='text']/text()")[0].lower():
                datetime_posted = str(i.xpath(".//div[@class='pull_right date details']/@title")[0])

                screen_name = str(i.xpath(".//div[@class='from_name']/text()")[0])
                screen_name = re.sub(r'\s+', '', screen_name)
                for line in i.xpath(".//div[@class='text']/text()"):
                    line = line.lower()
                    if 'pick up' in line:
                        pick_up = line.replace("pick up","").replace(":","")
                    if 'drop off' in line:
                        drop_off = line.replace("drop off","").replace(":","")
                    if 'time' in line:
                        datetime = line.replace("time","").replace(":","")
                    if 'pax' in line:
                        pax = line.replace("pax","").replace(":","")
```

Figure 81: Code Snippet used to convert data into csv format

4.3.2.2. Filtering Valid & Invalid Addresses

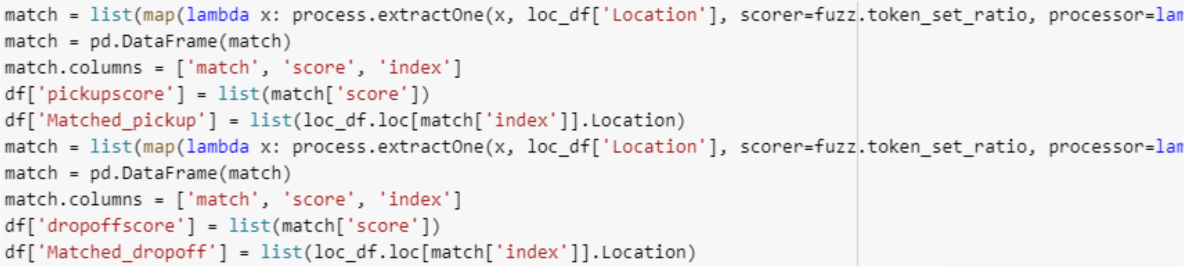
In order to verify whether an address is genuine, we used Google Distance API to validate the addresses provided. Valid addresses will be given a distance and estimated time of travel which will be stored for further analysis.



```
google_info = google_distance(pick_up,drop_off)
try:
    valid_output_file.writerow([screen_name.strip(), pick_up.strip(), drop_off.strip(),time.strip(),pax.
    df.loc[count] = [screen_name.strip(), pick_up.strip(), drop_off.strip(),time.strip(),pax.strip(),dat
    count += 1
except:
    invalid_output_file.writerow([screen_name.strip(), pick_up.strip(), drop_off.strip(),time.strip(),pa
```

Figure 82: Code Snippet used to call Google Distance API to validate addresses provided

For the invalid addresses, we fuzzy matched against a list of street names **[8]** to standardize our pickup and drop off locations. After that, we used the Google Distance API to re-process these data again.

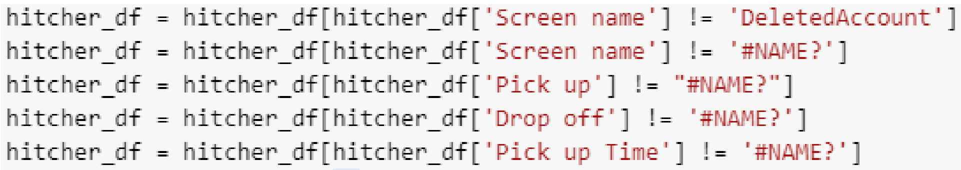


```
match = list(map(lambda x: process.extractOne(x, loc_df['Location'], scorer=fuzz.token_set_ratio, processor=lam
match = pd.DataFrame(match)
match.columns = ['match', 'score', 'index']
df['pickupscore'] = list(match['score'])
df['Matched_pickup'] = list(loc_df.loc[match['index']].Location)
match = list(map(lambda x: process.extractOne(x, loc_df['Location'], scorer=fuzz.token_set_ratio, processor=lam
match = pd.DataFrame(match)
match.columns = ['match', 'score', 'index']
df['dropoffscore'] = list(match['score'])
df['Matched_dropoff'] = list(loc_df.loc[match['index']].Location)
```

Figure 83: Code Snippet used to fuzzy match invalid addresses to a list of street names

4.3.2.3. Cleaning Invalid Records

After re-processing some invalid addresses, we removed users whose screen name is 'DeletedAccount' as these users have removed their Telegram accounts and we would not have a way to track them anymore. There are also records which contains value such as '#NAME?'. These are misrepresented data and should be removed.



```
hitcher_df = hitcher_df[hitcher_df['Screen name'] != 'DeletedAccount']
hitcher_df = hitcher_df[hitcher_df['Screen name'] != '#NAME?']
hitcher_df = hitcher_df[hitcher_df['Pick up'] != "#NAME?"]
hitcher_df = hitcher_df[hitcher_df['Drop off'] != '#NAME?']
hitcher_df = hitcher_df[hitcher_df['Pick up Time'] != '#NAME?']
```

Figure 84: Code Snippet used to remove invalid records

4.3.2.4. Cleaning Duplicated Records

As users might make multiple requests in the chat if there are no one picking them up, duplicate data might exist. Therefore, we sorted the columns by Screen names and the date time posted. After which, we use a series of 'if-else' statements to check if the record is authentic. The guidelines are as follows:

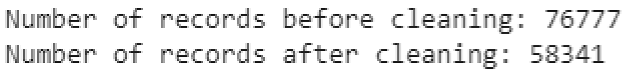
1) If Screen name of current record is not the same as previous record, it is authentic, else continue to next point

2) If date of current record is not the same as previous record, it is authentic, else continue to next point

3) If pick up of current record is not the same as previous record, it is authentic, else continue to next point

4) If the absolute difference between the time posted of current record and the drop off time of the previous record is less than the travel duration estimated by Google, it is not authentic, else the record is authentic

Following which, records that were deemed as not authentic were removed, which led to many rows of records being eliminated *(Refer to Figure 85).*

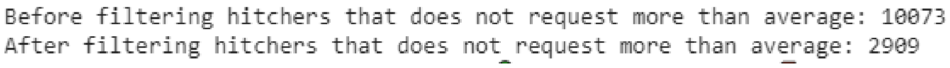


```
Number of records before cleaning: 76777
Number of records after cleaning: 58341
```

Figure 85: Number of records before and after removing duplicated records

4.3.2.5. Narrowing Dataset by Filtering Hitchers with more than the Average Requests

As there are 10,073 users from this huge dataset, it will be computationally expensive and time consuming to quickly identify potential customers. Hence, we decided to narrow the list to users who are making more than the average number of requests. This quickly narrow down to 2,909 users, this allowed us to analyze only 6084 records out of the 58,341 records.



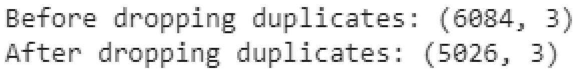
```
Before filtering hitchers that does not request more than average: 10073
After filtering hitchers that does not request more than average: 2909
```

Figure 86: Number of users before and after applying filter to narrow down search

4.3.2.6. Further Narrowing Dataset by Filtering Hitchers who Travels to or from the Same Location Thrice

Based on these users, we further narrow down the list to find users who hitch from the same pick up or drop off location for more than 3 times. This is because we will be assuming that those locations are our users' residence since it is likely that a user would request for a hitch from his residence multiple times a month. However, it might be possible that we have the same users having 2 locations which he/she travelled to and from more than thrice. For example, User A has been travelling to Serangoon for more than 3 times and has been travelling from Bugis for more than 3 times. There might also be a possibility that they are hitching a ride to or from places like schools, offices or industrial places. Therefore, we will still have a form a manual filtering when deciding the high value users.

This list will later be used to fuzzy match with a property price list from Data.gov.sg to infer the spending power of the users. Therefore, the final dataset we are analyzing is only 5026 records.



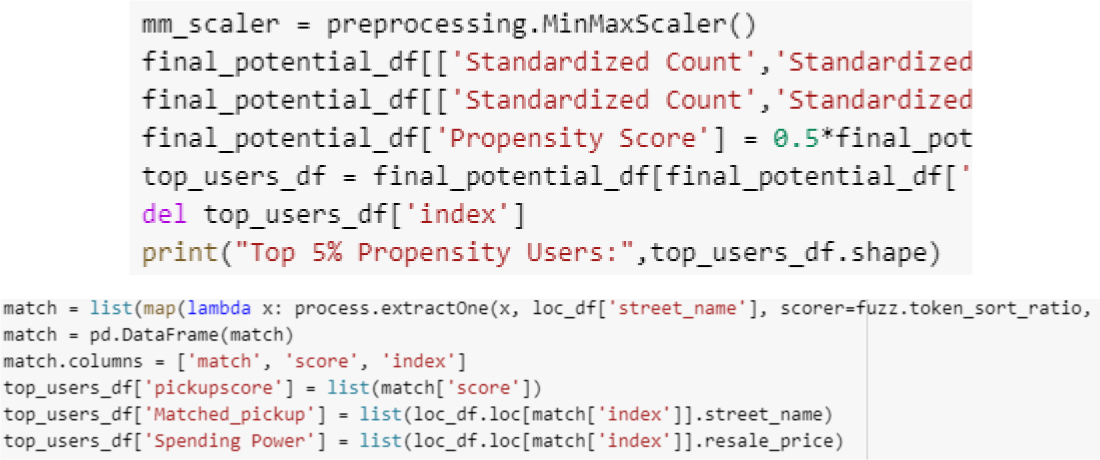
```
Before dropping duplicates: (6084, 3)
After dropping duplicates: (5026, 3)
```

Figure 87: Number of records left after filtering users who have the same pick up or drop off location for more than 3 times

4.3.2.7. Fuzzy Matching users' pick up or drop off locations and assigning spending power.

Due to the computational power and resource required to fuzzy match the 5,026 records to the property price list which has 58,154 records, we divided the datasets into smaller sets to run the algorithm.

For each batch of dataset, we will infer the spending power of the user from the property price list using the fuzzy match algorithm.



```
mm_scaler = preprocessing.MinMaxScaler()
final_potential_df[['Standardized Count','Standardized
final_potential_df[['Standardized Count','Standardized
final_potential_df['Propensity Score'] = 0.5*final_pot
top_users_df = final_potential_df[final_potential_df['
del top_users_df['index']
print("Top 5% Propensity Users:",top_users_df.shape)
```

```
match = list(map(lambda x: process.extractOne(x, loc_df['street_name'], scorer=fuzz.token_sort_ratio,
match = pd.DataFrame(match)
match.columns = ['match', 'score', 'index']
top_users_df['pickupscore'] = list(match['score'])
top_users_df['Matched_pickup'] = list(loc_df.loc[match['index']].street_name)
top_users_df['Spending Power'] = list(loc_df.loc[match['index']].resale_price)
```

Figure 88: Code snippets to fuzzy match the locations by running the fuzzy match algorithm in batches of smaller datasets

As we are predicting the user's spending power by assuming their residence area if they have 3 of the same pick up or drop off point, we will map each area of residence to a list of towns in Singapore, with their respective average HDB resale price for that area. This allows us to infer their spending power as staying in an estate like 'Sixth Avenue' or 'Orchard Road' will be highly likely to have a higher spending power than a user staying in a neighbourhood heartlands area such as 'Toa Payoh' or 'Woodlands'. Hence, this gives us a good gauge on their spending power which can suggest their capability to afford a car leasing program.

As our fuzzy match algorithm will try to match the location of the user to a location from the property price list, a matching score will be provided. A higher matching score means that the match of the location is very similar. Furthermore, based on the matched location, we will also take its respective property price to assign it as the user's spending power.

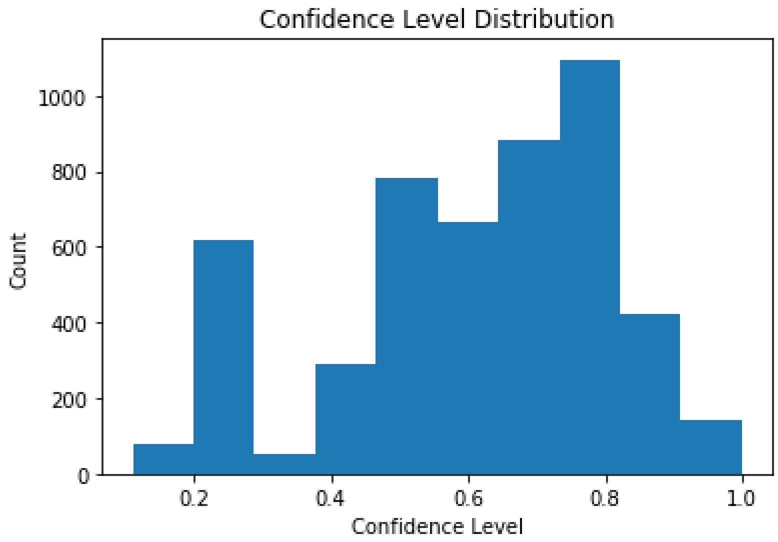
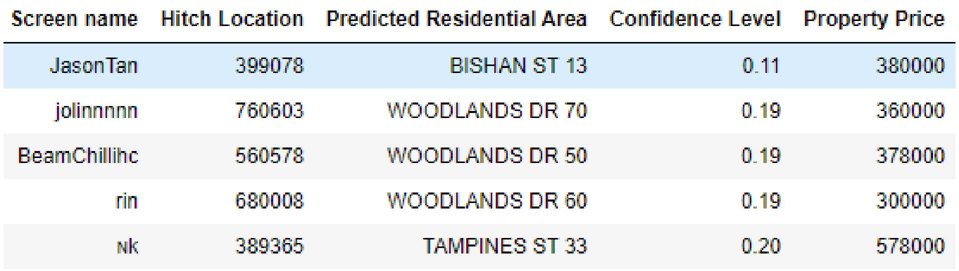


Figure 89: Distribution of Fuzzy Match's Confidence

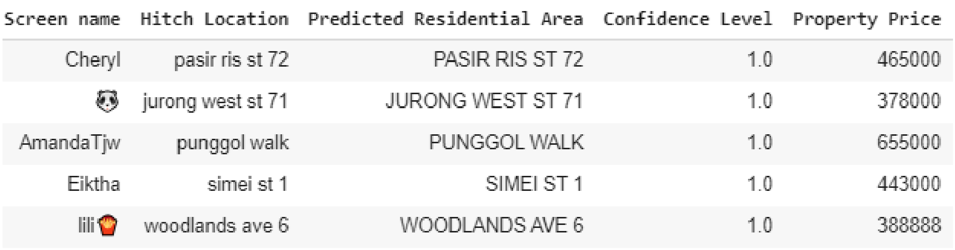
After looking at the distribution of how accurate our fuzzy match algorithm is, we can reasonably say that the algorithm is accurate. However, we still have a significant number of locations which the algorithm feels that it is not confident. One of the possible reasons is that users might have inputted postal codes instead of a text address (Refer to Figure 90). As our property price list do not show postal code, the algorithm couldn't find the best match to the user's location which explains the low confidence score.



Screen name	Hitch Location	Predicted Residential Area	Confidence Level	Property Price
JasonTan	399078	BISHAN ST 13	0.11	380000
jolinnnnn	760603	WOODLANDS DR 70	0.19	360000
BeamChillihc	560578	WOODLANDS DR 50	0.19	378000
rin	680008	WOODLANDS DR 60	0.19	300000
NK	389365	TAMPINES ST 33	0.20	578000

Figure 90: Example showing users putting postal codes and fuzzy match's result

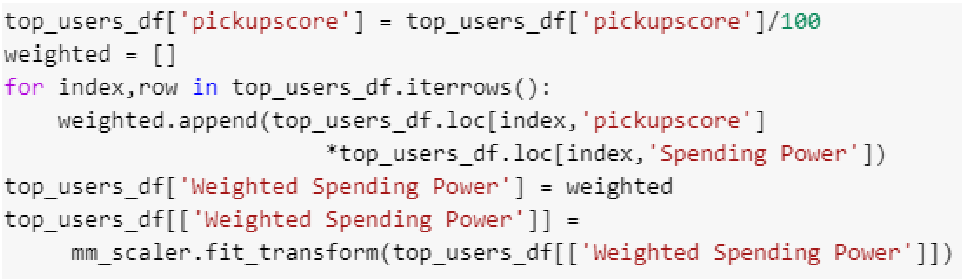
However, if users have inputted a valid address which the fuzzy match algorithm can match, a higher confident score will be assigned. Below is a screenshot which shows the fuzzy match algorithm matches the location and have a confidence score of 1 as the locations are the exact match.



Screen name	Hitch Location	Predicted Residential Area	Confidence Level	Property Price
Cheryl	pasir ris st 72	PASIR RIS ST 72	1.0	465000
🐼	jurong west st 71	JURONG WEST ST 71	1.0	378000
AmandaTjw	punggol walk	PUNGGOL WALK	1.0	655000
Eiktha	simei st 1	SIMEI ST 1	1.0	443000
lili🐚	woodlands ave 6	WOODLANDS AVE 6	1.0	388888

Figure 91: Example showing users putting postal codes and fuzzy match's result

Therefore, one solution to reflect the algorithm’s confidence as part of heuristic is to factor it in the to get a more accurate overview of the users’ spending power.

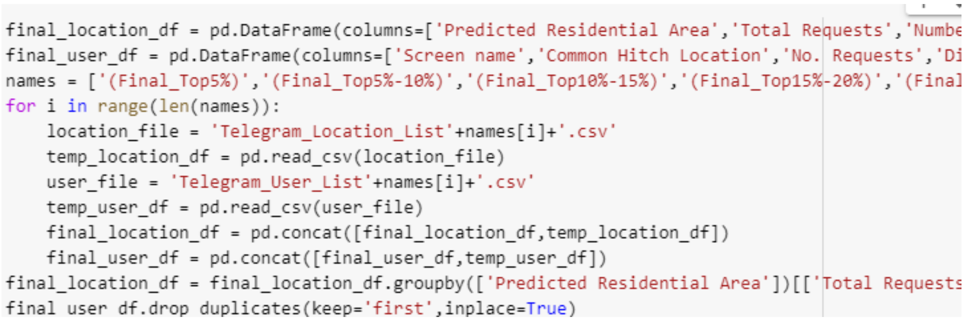


```
top_users_df['pickupscore'] = top_users_df['pickupscore']/100
weighted = []
for index,row in top_users_df.iterrows():
    weighted.append(top_users_df.loc[index,'pickupscore']
                    *top_users_df.loc[index,'Spending Power'])
top_users_df['Weighted Spending Power'] = weighted
top_users_df[['Weighted Spending Power']] =
     mm_scaler.fit_transform(top_users_df[['Weighted Spending Power']])
```

Figure 92: Example showing users putting postal codes and fuzzy match’s result

4.3.2.8. Merging the small datasets into a single file

After fuzzy matching all the small datasets, we merge them again into a single view for easier analysis and data cleaning.

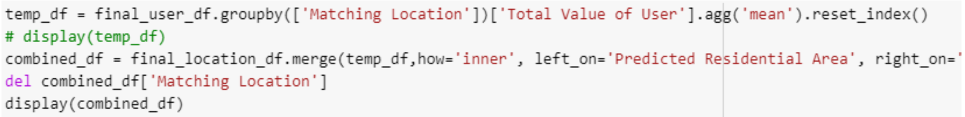


```
final_location_df = pd.DataFrame(columns=['Predicted Residential Area','Total Requests','Numbe
final_user_df = pd.DataFrame(columns=['Screen name','Common Hitch Location','No. Requests','Di
names = ['(Final_Top5%)','(Final_Top5%-10%)','(Final_Top10%-15%)','(Final_Top15%-20%)','(Final
for i in range(len(names)):
    location_file = 'Telegram_Location_List'+names[i]+'.csv'
    temp_location_df = pd.read_csv(location_file)
    user_file = 'Telegram_User_List'+names[i]+'.csv'
    temp_user_df = pd.read_csv(user_file)
    final_location_df = pd.concat([final_location_df,temp_location_df])
    final_user_df = pd.concat([final_user_df,temp_user_df])
final_location_df = final_location_df.groupby(['Predicted Residential Area'])[['Total Requests
final_user_df.drop_duplicates(keep='first',inplace=True)
```

Figure 93: Code snippets to merge back all the datasets post fuzzy match

4.3.2.9. Averaging the total value of the users per location

We calculated the average value of the users based on their leasing propensity and spending power (calculations will be explained in Section 4.3.3.).

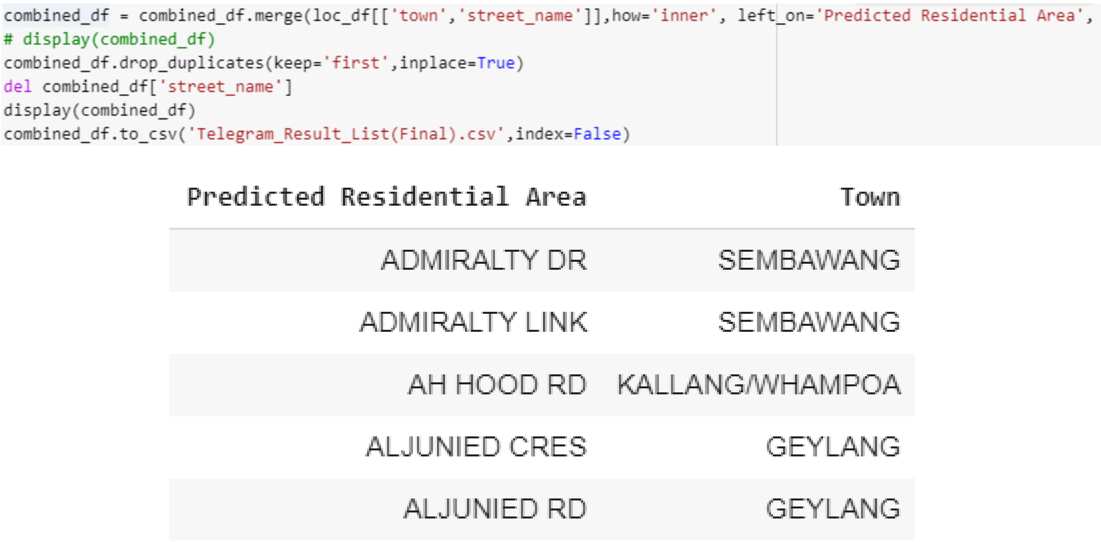


```
temp_df = final_user_df.groupby(['Matching Location'])['Total Value of User'].agg('mean').reset_index()
# display(temp_df)
combined_df = final_location_df.merge(temp_df,how='inner', left_on='Predicted Residential Area', right_on='
del combined_df['Matching Location']
display(combined_df)
```

Figure 94: Code snippets to calculate average total value of users in each location

4.3.2.10. Assigning the users’ location to a generic location

We assigned the user’s location to a generic location as we would like to visualize the data in a form of a heatmap using Tableau. The generic location is also from the property price list. For example, if we predict that the user’s location is at Admiralty Dr, then based on the property price list, the assigned town is Sembawang.



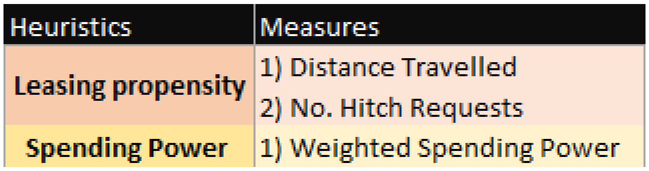
```
combined_df = combined_df.merge(loc_df[['town','street_name']],how='inner', left_on='Predicted Residential Area',
# display(combined_df)
combined_df.drop_duplicates(keep='first',inplace=True)
del combined_df['street_name']
display(combined_df)
combined_df.to_csv('Telegram_Result_List(Final).csv',index=False)
```

Predicted Residential Area	Town
ADMIRALTY DR	SEMBAWANG
ADMIRALTY LINK	SEMBAWANG
AH HOOD RD	KALLANG/WHAMPOA
ALJUNIED CRES	GEYLANG
ALJUNIED RD	GEYLANG

Figure 95: Code snippets and results of allocating residential areas to nearest Town

4.3.3. Analysis

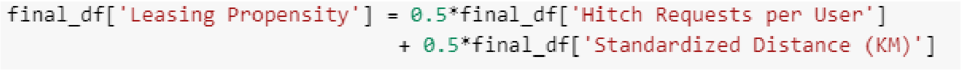
After retrieving and cleaning the data from telegram, we ranked these users based on their inferred spending power and how likely they were to engage DFM's leasing services.



Heuristics	Measures
Leasing propensity	1) Distance Travelled 2) No. Hitch Requests
Spending Power	1) Weighted Spending Power

Figure 96: Heuristics that were used and the measures under each heuristic

Leasing propensity is referred to as how likely the users from the location would lease a vehicle from DFM. The leasing propensity of a location is determined by a sum of the location’s estimated distance travelled and number of requests per user in the location. This score is normalized to 1. The higher the value implies that the users in the location are more likely to lease a car from DFM.

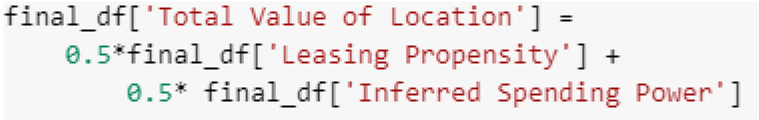


```
final_df['Leasing Propensity'] = 0.5*final_df['Hitch Requests per User']
                                 + 0.5*final_df['Standardized Distance (KM)']
```

Figure 97: Code snippets to calculate leasing propensity

Spending power is referred to how likely the users from the location have the capability to lease a vehicle from DFM. The spending power of a location is determined by summing up all the users’ spending power in the location and standardizing to a score of 0 and 1. This was covered in Section 4.3.2.9.

Therefore, by summing up both the spending power and leasing propensity score, we will be able to recommend a suitable location for DFM to target.



```
final_df['Total Value of Location'] =
    0.5*final_df['Leasing Propensity'] +
        0.5* final_df['Inferred Spending Power']
```

Figure 98: Code snippets to calculate total value of location

4.3.3.1 Leasing Propensity

4.3.3.1.1 Distance Travelled

Retrieving the distance travelled can allow us to gauge how much the user in each location travels. If we have a location where there are people travelling a lot, we may infer that these locations could either be inaccessible by public transport or their work or school may be far from their residence. Thus, these locations might be suitable for DFM to target for car leasing programs.

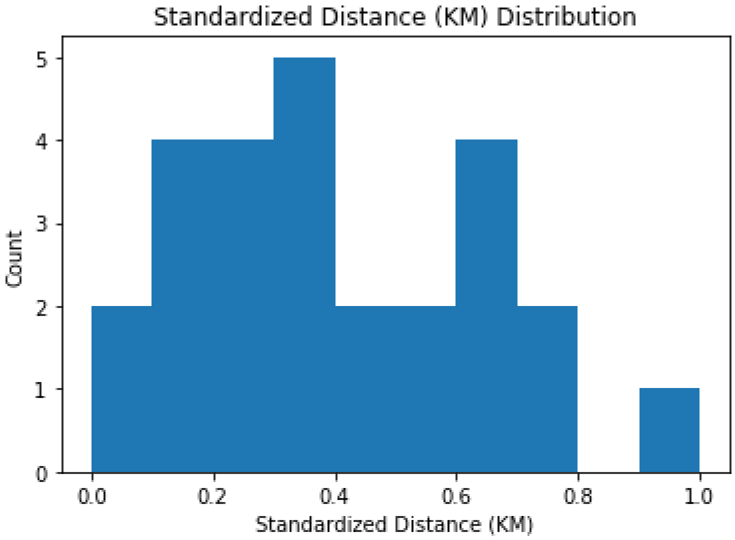


Figure 99: Distribution of Standardized Distance travelled by location

4.3.3.1.2 No. of Hitch Requests

The number of hitch requests per user in the location allows us to understand if there are a few common users in the location who often makes multiple hitch requests. This would suggest that the location have some users who move around frequently, maybe due to their job scope, which then gives them a reason to consider a car leasing as an option. A higher score would mean that there are hitch requests are requested by a small group of people in a location. Based on the distribution, we could say that there are a significant number of locations have hitch requests requested by a large group of people.

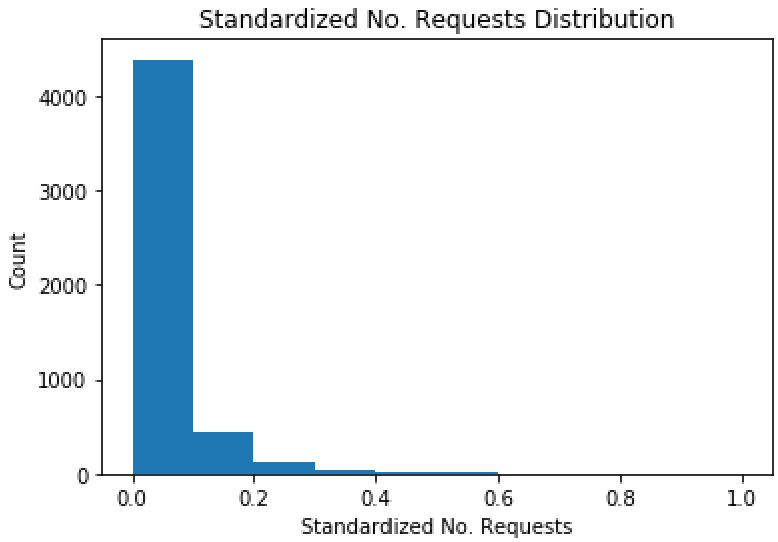


Figure 100: Distribution of Standardized number of requests per users by location

4.3.3.2 Spending Power

4.3.3.2.1 Weighted Spending Power

The spending power is inferred from the average price of the property that is assumed to be of their residence. Also, as mentioned above, this is calculated by the inferred spending power based on our fuzzy match algorithm and its confidence level. This helps us to get a more accurate prediction of the users' spending power. This has been summed and standardized among their respective location which can be seen below. Since a higher average property price could signify a higher spending power, we are able to filter and distinguish locations that have a higher potential to adopt DFM's leasing program as well as allowing us to recommend different types of cars to them.

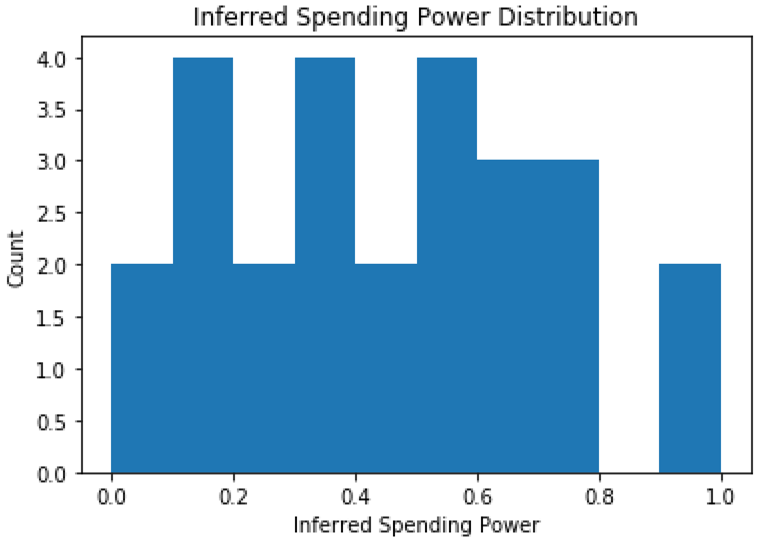


Figure 101: Distribution of weighted spending power of users by location

4.3.4 Interesting Finding Insights

After identifying Telegram users who frequently hitch, we find out the location of their pickup/drop-off and was able to plot out the most common hitch locations.

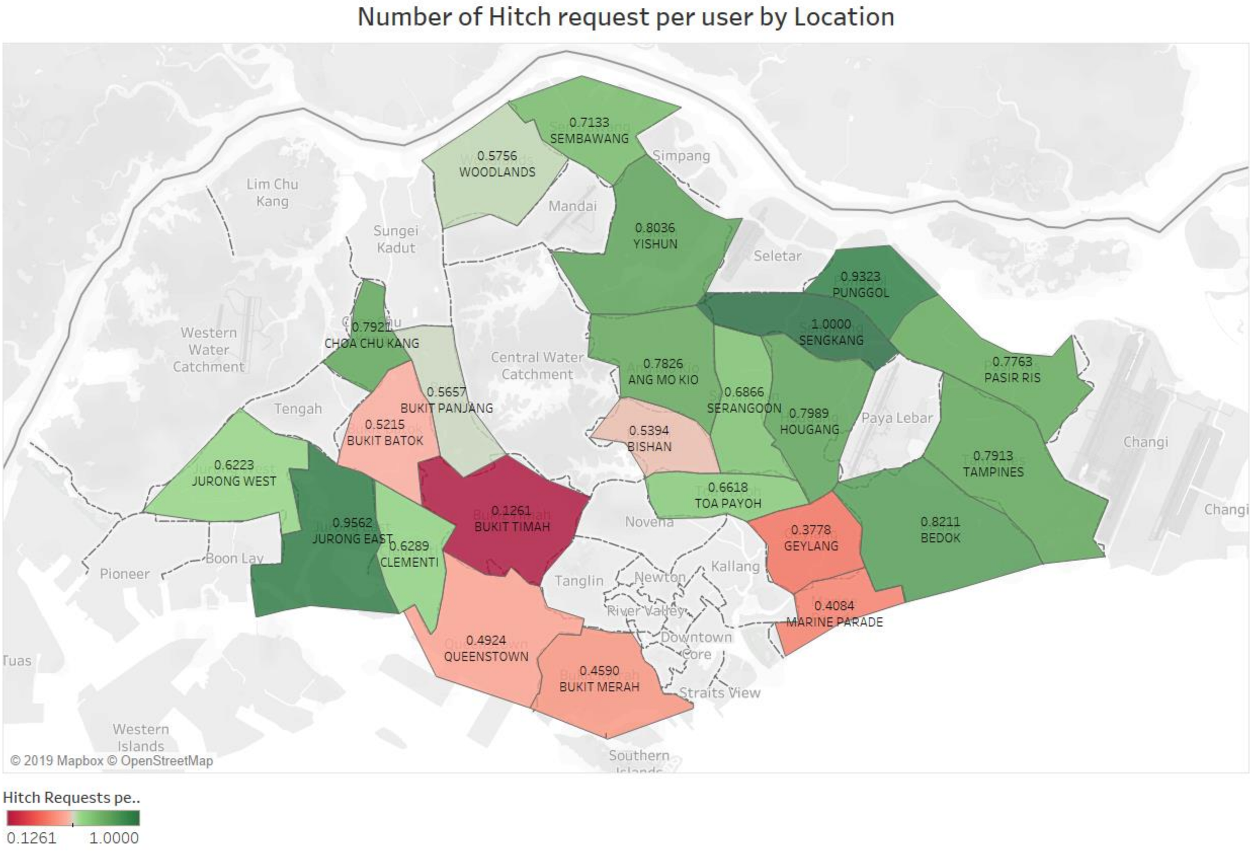


Figure 102: Heatmap of locations with intensity based on number of hitch requests per user

This map *(Refer to Figure 102)* shows the number of hitch requests per user in the location after standardizing it to a score between 0 and 1. We can observe that there are a couple of dark green locations which means that they have a high number of hitch requests per user in the location. This could be **due to the accessibility of public transports in their respective geographic locations.**

For location such as Sengkang and Punggol, these locations are still in development as the government is still launching new HDB flats around this area. As these areas are not fully developed, public transport might still be inaccessible or not available. This might explain why there is a high number of hitch requests per user in those location. For Jurong East, it might be because Jurong East can be considered a remote location and do not have any access to public transport to reach home easily.

Bukit Timah is reflected as red which means that they have a low hitch requests per user. This is not surprising, considering that the people living in those area has a higher spending power which is also an assumption we made ourselves. People living in those areas generally can afford a car and would least likely be interested to lease one.

Central areas such as Ang Mo Kio and Serangoon have a moderate score. This might be because they are areas which are very accessible with public transports.

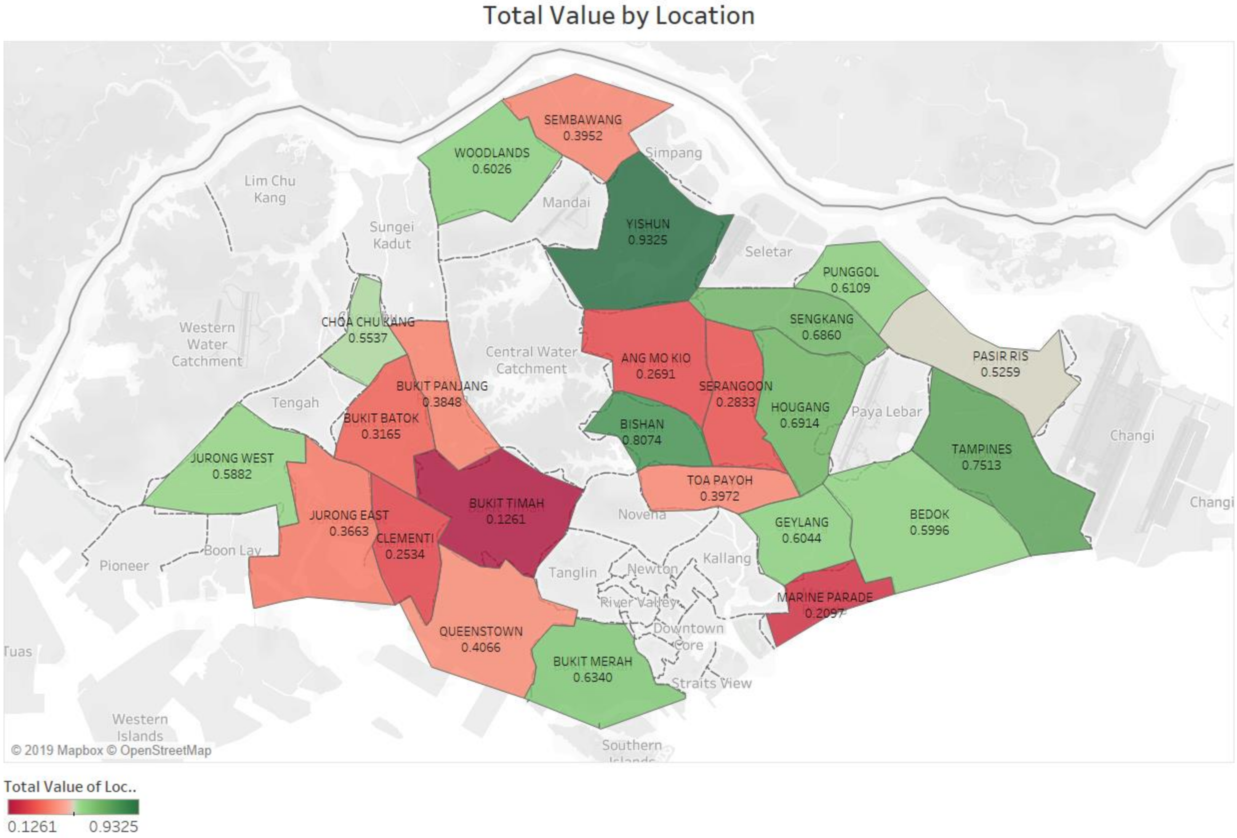


Figure 103: Heatmap of locations with intensity based on the total value

This map *(Refer to Figure 103)* shows the value of the locations with relation to spending power and leasing propensity. Basically, **the locations with the highest score consists of high valued individuals in terms of their spending power and leashing propensity**. This score also factored in the accuracy score of the fuzzy match algorithm. This means that we might have predicted some locations wrongly and assign them to the wrong town. If so, a lower score would be calculated and added into the value of these locations. This makes the heatmap more accurate for decision-making.

Unlike the previous heat map, we can observe that Sengkang and Punggol does not have the highest score. This might be because they are developing areas where new flats are being built. This would mean that young married couples (around 20 to 30 years old) are more likely to BTO in such locations. Furthermore, their spending power might be low as they might not have much working experience, having a lower chance to have a high paying job.

To conclude, if DFM wants **to focus on "quantity" and outreach,** they can refer to the first heat map which shows the number of hitch requests per user. If DFM wants **to focus on "quality" and likelihood of acquiring a customer for their services,** they should refer to the second heat map which shows the total value of the location. With this information at hand, we propose that DFM can set up event booths or roadshows in these locations to market their car leasing services.

5.0 Assumptions & Limitations

Assumptions

In performing topic modelling for our tweets to extract keywords and topics from competitors, we assumed that the tweets are made up of 1 topic per tweet. This is necessary for the topic to form meaningful clusters when using the GSDMM algorithm for topic modelling.

In performing our profile crawling on LinkedIn, we assume that users have filled out their profile correctly and exhaustively. This allows us to predict their working experiences and other measures.

For the extraction of users on Telegram group "SG HITCH", we assume that when the same pick up or drop off location is requested more than or equals to 3 times, then that will be their place of residence. This is an important assumption as we will be deriving their predicted spending power based on their residence. We also assume that they are staying in HDB flats as majority of the Singaporeans do, and we are matching their residence to the average price of HDB resale flat prices in that area.

Limitations

In generating our list of social media network influencers, there is the limitation of targeting a specific geographical location when crawling for tweets. Also**, online buzz on Twitter in Singapore is limited due to the decrease in popularity of Twitter among Singaporeans**. We are then limited to using the tweets of users in other countries and gather influencers that are outside of Singapore, and eventually the influencers are unable to reach certain countries where Daimler is in such as Singapore.

When generating insights from LinkedIn, there are many **fields that are not filled in that would allow us to better target and contact the users**. For example, their personal information such as address, phone number or email are usually left blank. In addition, LinkedIn does not provide any information that allows us to infer this information, such as the geo location of their posts. This leads us to having to try other ways to target the users such as through the companies, which may not be the most effective if the company is unwilling to work with us. Also, when we target the profiles individually, we are unable to do it in an effective manner because of the lack of personal contact information. Hence, we can only select the top few profiles and match it to DFM's list of existing customer database for further actions.

Another limitation of crawling for profiles on LinkedIn via LinkedHelper is the limitation of crawling profiles based on your connections. The **profiles collected will be skewed towards profiles that have direct connections or indirect connections** as they are prioritized on the search result list, where the LinkedHelper crawls their data from. Hence, we may not get fully representative profiles of users in our targeted industry.

Similarly, for Telegram, there is **a lack of contact details that can be extracted out which DFM can use to target these users.** There only exists the users' telegram handle and rarely, their phone number if it is available for public to view, which might restrict the ways that DFM can reach out to them whilst maintaining a professional approach. Hence the insights from Telegram is mostly geared towards providing DFM with insights into which are the best locations to target depending on the intentions behind their marketing campaigns.

6.0 Recommendations

Twitter

Twitter was primarily used for 2 objectives, topic modelling and identifying potential influencers.

For topic modelling, DFM **can use it as a form of competitive analysis** where they can crawl from tweets from their competitor and analyze their tweets. By doing such analysis, DFM can analyze their competitors' moves and what they are doing. This will help to improve DFM's competitive advantage as they are able to adjust their moves accordingly to best suit the current market trends.

After identifying an area to improve for DFM based on our competitive analysis, we started to identify potential influencers based on the keywords which we have identified using topic modelling. Influencers will be ranked based on their relevance and influence. Other than identifying influencers based on competitive analysis, **DFM can also use their own sets of keywords to search for potential influencers to work and collaborate with**. With DFM's long standing history in this industry, they can utilize their knowledge of the certain traits and characteristics associated with their customers, which could yield greater results.

LinkedIn

After analyzing the prospective customers on LinkedIn, Daimler can **run a match of their existing database of current customers to the information we have identified**. For customers whom we are able to get a match, Daimler is able to contact them before their current lease expires and follow up with a personalized marketing content based on the characteristics obtained in their profile, such as their job scope, job position and working years. This information can be combined with those that are already existing in their customer database, such as age, gender, and previous history of cars they have leased. They can then provide offerings that are tailored to the needs of their users, which will save both marketing cost as well as yield a higher rate of success as the proposed offerings are more aligned with the wants of these customers.

On the other hand, if they are not in Daimler's current database of customers, **DFM can target the companies of the top profiles of our users**. This can be achieved by reaching out to the companies and offering a partnership, in a form **of offering DFM's corporate car leasing solution** that is specifically catered to their employees at a corporate rate. This allows Daimler to indirectly reach out to the users that we have identified as well as other employees in the company that is likely to have the need for the leasing services as well.

Leasing Solutions for different models

According to our market research, DFM offers a variation of Mercedes Benz car models for their car leasing solutions. The price range varies depending on the class of the vehicle (from A to S) whereby the vehicles from a lower class such as A is usually smaller and cheaper than that of vehicles from S class. They also have different options within each class that cater to the different preferences of customers such as sportier models like the coupe and cabriolet and luxury cars like Maybach.

Since we have more customers that fall in the corporate position category “3” which are **Directors or partners, DFM can promote more on expensive and luxurious options from their E and S class models.** Conversely, other customers that has a low number of years of working experience and having **a more junior position can be offered a class A model** which is relatively cheaper than the other models and more aligned to the plausible spending power that these customers have.

We also have observed that majority of our customers have 0 to 15 years of working experience. Hence, we can assume that they are in their early twenties to late thirties. We can then offer them sportier variants of the models mentioned above, such as a cabriolet or coupe variant.

Telegram

After analyzing the requests on SG Hitch, we can map out which geographical location, specifically which street and town in Singapore have the most potential. In determining the exact locations to target, we have identified 3 different metrics that Daimler can view for each of the towns: Average score, number of requests, and number of unique users. This location will be their assumed place of residence. We are then able to **target these areas by placing booths at shopping malls or places with high traffic to advertise the leasing services of Daimler**. Since we have the street names of users who presumably live in that area, we can also **distribute flyers in that area to promote our services to make their current efforts of using traditional print media more effective**.

The range of models that we can focus on advertising can also be personalized based on the average resale prices of that area, which can be used to infer the spending power of residents in that area. If a town that we identified is of a higher-valued estate, it will be best to offer higher-end models for the residents, while a town that is lower-valued can be offered the lower end models that DFM offers.

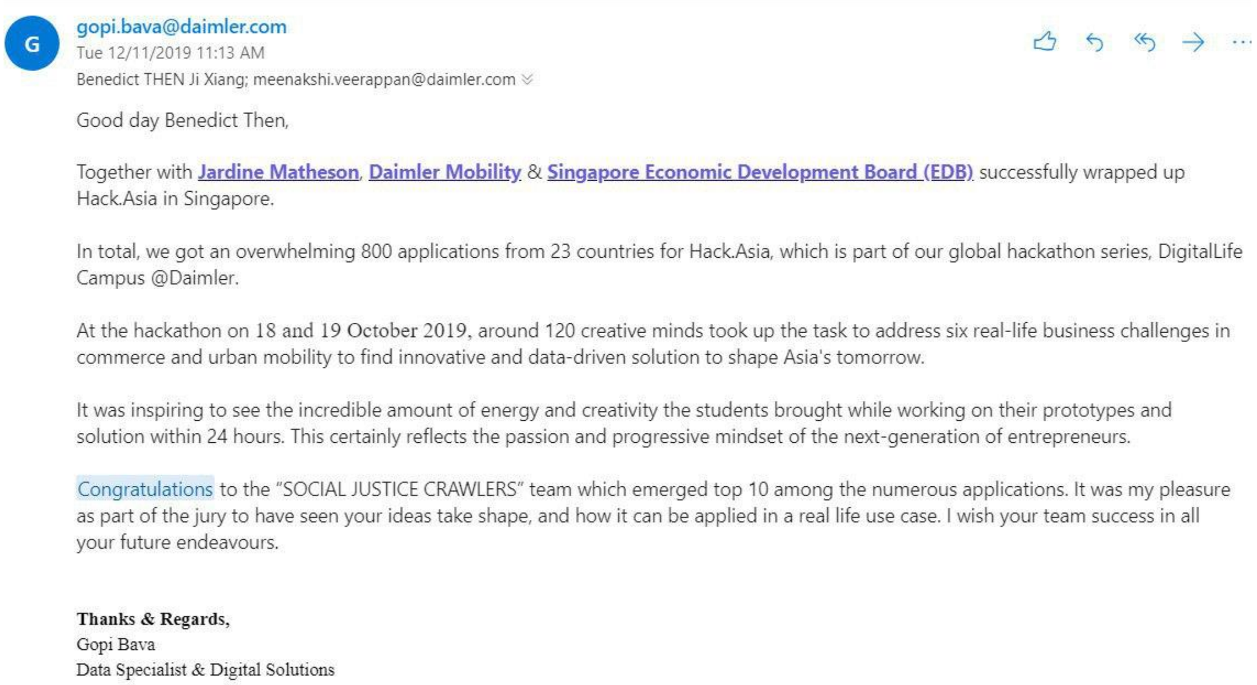
7.0 Conclusion

Our group thinks that this project has been a challenging but definitely rewarding experience for us. Through the embarkation of this project, we were also able to participate in a 2D1N hackathon at Hack.Asia of which one of the main sponsors was DFM. This hackathon gave us the chance to interact with the staff of DFM as well as the challenge owner who **gave us insights into the roots of DFM’s problem in this car leasing industry** that helped us to tailor the angles of our solutions to better suit their company along the way.

We tackled the problems that were shortlisted by identifying prospective customers who are likely to be interested in DFM’s services from social media data and improving DFM’s current offerings. We decided to **use the SHIT approach for each social media platform we decided to target** (Twitter, LinkedIn and Telegram). After analyzing our results, we identified ways in which DFM can leverage on our analysis as well as plausible methods on how DFM can improve our approach so that they are able to improve their analysis for decision-making.

This would be able provide DFM with a base solution which they can improve in the future through improving the algorithms by providing their own proprietary knowledge and contacts based on their existing customer database thus yielding better results. These results and analyses will be **beneficial in the long run as they are able to transit from a generalized approach to a more targeted one** which is lower in marketing costs as well as more effective in have successful customer acquisitions.

Representative of DFM’s Statement



G

gopi.bava@daimler.com
Tue 12/11/2019 11:13 AM
Benedict THEN Ji Xiang; meenakshi.veerappan@daimler.com

Good day Benedict Then,

Together with **Jardine Matheson**, **Daimler Mobility** & **Singapore Economic Development Board (EDB)** successfully wrapped up Hack.Asia in Singapore.

In total, we got an overwhelming 800 applications from 23 countries for Hack.Asia, which is part of our global hackathon series, DigitalLife Campus @Daimler.

At the hackathon on 18 and 19 October 2019, around 120 creative minds took up the task to address six real-life business challenges in commerce and urban mobility to find innovative and data-driven solution to shape Asia's tomorrow.

It was inspiring to see the incredible amount of energy and creativity the students brought while working on their prototypes and solution within 24 hours. This certainly reflects the passion and progressive mindset of the next-generation of entrepreneurs.

Congratulations to the “SOCIAL JUSTICE CRAWLERS” team which emerged top 10 among the numerous applications. It was my pleasure as part of the jury to have seen your ideas take shape, and how it can be applied in a real life use case. I wish your team success in all your future endeavours.

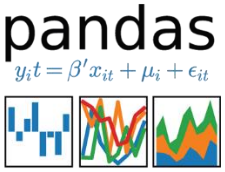
Thanks & Regards,
Gopi Bava
Data Specialist & Digital Solutions

8.0 Tools & Technology Used

Data Crawling

All files are stored on our local machines after they are crawled.

Data Visualization



Coding

9.0 Team Contribution

Everyone played their part to crawl data from the 3 social media platforms we focused our analysis on. This was to hasten our process.

Coding for Twitter Topic modelling was primarily done by Junrong, coding for Twitter Influencers was primarily done by Delin coding for LinkedIn was primarily done by Junrong and coding for Telegram was primarily done by Delin and Junrong. Tableau visualization and slides was done by Benedict. Ryan oversaw this report, making sure that every detail is present. Poster was done by Janell. Janell and Ryan ensured that the report is sound and cohesive. Everyone reminded each other what is left so that we can always continue progressing with work. Everyone also helped in the analysis of the data. We discussed the results and findings of each analysis to suggest recommendations to DFM. This also helped us to validate and check on each other's work. We met 3 days per week.

Therefore, everyone deserves the full score of 5.

10. Bibliography

1. Yin, J. and Wang, J. (2019). A dirichlet multinomial mixture model-based approach for short text clustering.
https://www.semanticscholar.org/paper/A-dirichlet-multinomial-mixture-model-based-for/d03ca28403da15e75bc3e90c21eab44031257e80

2. GitHub. (2019). rwalk/gsdmm. Implementation of Topic Modelling Algorithm
https://github.com/rwalk/gsdmm

3. Lo, S. L., Chiong, R., & Cornforth, D. (2016, March 2). Ranking of high-value social audiences on Twitter.
https://www.sciencedirect.com/science/article/pii/S0167923616300215

4. NealSchaffer. (2019, September 20). Twitter Following vs Followers: What is the Ideal Ratio?
https://nealschaffer.com/twitter-followers-following-quality-or-quantity/

5. Bell, L., & Bell, L. (2019, February 23). Influencer Marketing and Reach Versus Relevance: What You Need to Know.
https://www.grouphigh.com/blog/influencer-marketing-reach-versus-relevance-need-know/

6. Pennington, J. (n.d.). Implementation of Vectors for word representation
https://nlp.stanford.edu/projects/glove/

7. Courtney, W. S. (2019, November 3). AMG Wants to Attack Porsche Head-On-By Going Electric
https://gearpatrol.com/2019/10/29/mercedes-amg-electric-first-future-plan-to-attack-porsche/

8. Geographic, Street
https://geographic.org/streetview/singapore/index.htmlhttps://geographic.org/streetview/singapore/index.html