Natural Language Processing for Company Business Descriptions

Wei Ding, Wusheng Liu, Kwok Ping Ng
Technical University Munich
wei.ding@tum.de, kwokping.ng@tum.de, wusheng.liu@tum.de

Abstract

Finding similar companies or competitors of a new start-up company is important for investors when making their decision for investment. However, the number of start-up companies grows at such a rapid pace over the past 10 years that it significantly increased the difficulties of identifying their industries or similarity manually by looking at company descriptions from the websites. In our project, the websites' content were preprocessed by cleaning up nonverbal signs, removing duplicates and etc. After that, TF-IDF, Word2Vec and Doc2Vec techniques are applied to the words from each website to generate word vectors. Based on these information, industry categories of new websites were able to be predicted by supervised learning methods: Support Vector Machine, K Nearest Neighbours, Feed Forward Neural Network and unsupervised method: K-means. Apart from those machine learning methods, cosine similarity of the websites based on their TF-IDF, Word2Vec and Doc2Vec vectors were also explored to understand the industries' relationship in-between. In addition to the general industry category prediction, the keywords analysis of the categories and sub-categories of an industry were also investigated to reveal their characteristic. Last but not the least, a Flask App was built to allow user interaction and recommend similar companies based on the input company provided.

1 Introduction

Defining company category business boundary is central to the study of financial management. One example is the application of the SIC system and NAICS system in United State and UK where the classification of the industrial type of a company is standardized so that analysis can be performed and shared across organizations. However, these systems are not really aligned and standardized across countries. This makes it difficult to perform analysis on a global scale. With the evolvement of globalization, it seems more tempting to develop a new approach to allow categorization of companies regardless of their residing countries and the standards adopted domestically. If the number of companies requiring categorization is small, people can easily read their websites information and tell what it does directly. This is called manual classification. However, once the data size gets large, it is less desirable to perform this process manually. As a result, automatic classification of company industrial types based on website data is becoming an area of interest to lots of researchers recently. The website of a company normally contains valuable information about their company history, products, services, business and lots of other information. However, it remains challenging to extract necessary information from the website and derive the industry types of the companies automatically as the data are all mixed together and presented in a non-structured format. Hoberg and Philips[Hoberg and Phillips, 2010] has proposed the Text based Network industry classification approach to classify the industry types of the well established firms from SEC Edgar based on their product descriptions. However, till today, there are no research done on start-up companies whose websites may not be as structured and complete as well established companies. In this project, we applied different natural language processing techniques to automatically classify the industry types of 10-K start-up companies based on their website data. In Section 2, we performed cleaning and preprocessing steps to the raw website data provided. In Section 3 and 4, we explored different

vectorization techniques such as TF-IDF, word2vec and doc2vec as well as different supervised and unsupervised classification methods such as K-means, SVM, Neural Network, KNN, etc. Then, in Section 5 and Section 6, we carried out similarity comparison across different categories and performed further subcategory analysis. In Section 7, we built a flask app to recommend similar companies by combining word2vec and cosine similarity techniques. Eventually, we drew the conclusion and gave a direction which may improve the result of this project in Section 8.

2 Data cleaning and preprocessing

In this section, we performed cleaning and preprocessing steps to the website data provided and make it ready for the classification task in the subsequent sections.

2.1 Files Cleaning

2.1.1 Json Files

Each website is stored in a json file. Each page/tab under the website with its web content is stored as a key-value entry in this json file. Page/tab names are keys and texts from each tab are contents. Through a brief view of part of the data, we find that some json files are empty, some scrawled tabs and contents are invalid. So it is necessary to do the files cleaning.

2.1.2 Workflow

The workflow of file cleaning is shown in Figure 1. Firstly we removed the empty json files and tabs with empty content. We found that recorded tab keys have the pattern: Date_Different level of tabs_Webpage filename extension (for example *20161203_ipsc_about-us_executive-leadership_.html* as in Figure 2). So we used the regular expression to extract tabs which do not match this pattern.
Then we checked the content of these unmatched tabs. A lot of them has this error message as content. (*Error Messages: "The Wayback Machine has not archived that URL.\Help make the Wayback Machine more complete!\Save this url in the Wayback Machine..."*) So we only removed these kind of unmatched tabs. We also checked whether the matched tabs contain this error message and removed them. In addition, We also removed the empty files after cleaning these tabs.

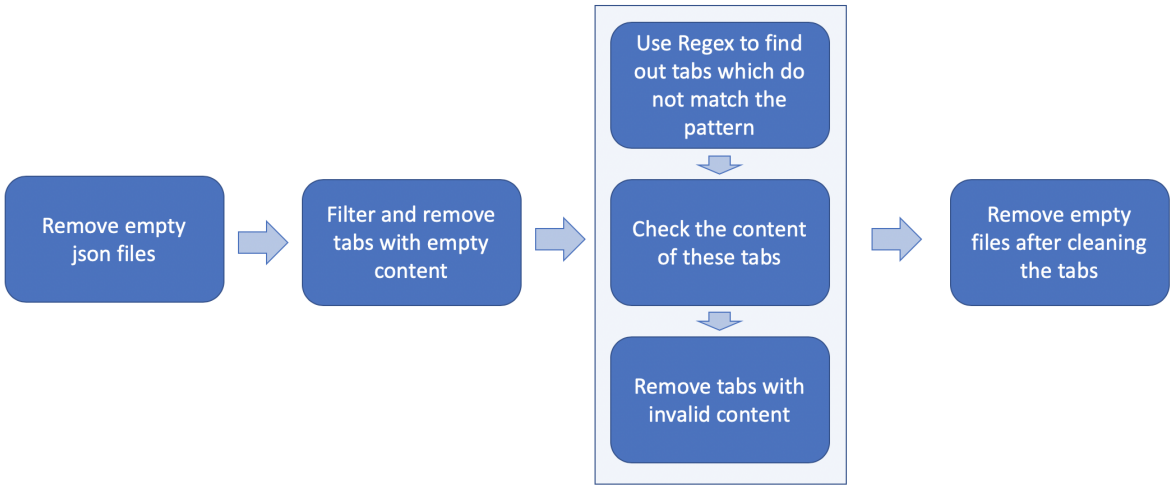


Figure 1: The workflow of file cleaning

2.1.3 Files Processing Results

Table 1 shows the number of valid files after each processing steps. We found that the number of final valid files from the *Sum* folder is less than half of the total files in that folder, which means that we need to focus on the files from *Sum_all* for the further classification task.

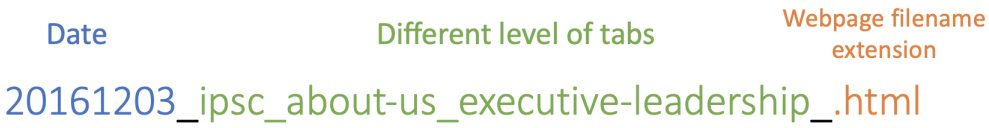


Figure 2: Name structure of each tabs

Table 2 is the statistics of number of tabs (for example *20161203_ipsc_about-us_executive-leadership_.html* as a tab) in the whole json files. These tabs may indicate the content of the webpages. From the numbers, we see each valid json file in *Sum* folder has average **3.64** valid tabs, while each valid file in *Sum_all* folder has around **27** valid tabs. Valid tabs in *Sum* folder has higher rate than that in *Sum_all* folder because files in *Sum* have been filtered by some business logic. Similarly, valid tabs per file in *Sum* are fewer than in *Sum_all*. We think that for classification purpose, we need to investigate the business logic to filter the tabs in *Sum_all* folder to achieve better classification results.

No.	File Type	No. in "Sum" folder	No. in "Sum_all" folder
a	Total Files	9,910	9,910
b	Empty Files	4,862	29
c	Empty Files after removing empty tabs and removing "The Wayback Machine" tabs*	5,036	2,057
d	Valid Files(d = a-c)	4,874	7,853
e	Valid Files %(e = d/a)	49.2%	79.2%

Table 1: Files Results in Cleaning * The tab contains error message "The Wayback Machine" is considered as an irrelevant tab

No.	Tab Type	No. in "Sum" folder	No. in "Sum_all" folder
A	Total Tabs	18,528	235,713
B	Empty Tabs	753	21,913
C	Empty Files after removing empty tabs	5,031	1914
D	Tabs with name not start with date and not end with (html$\|htm\|php\|org\|asp\|aspx$) **	20	2,465
E	Tabs with content "The Wayback Machine"*	6	419
F	Valid Tabs(F = A - B -E)	17,769	213,381
G	Valid Tabs %(G = F / A)	95.9%	90.5%
H	No of Valid Tabs per File(H = F/d)	3.64	27.17

Table 2: Tabs Results in Cleaning * The tab contains error message "The Wayback Machine" is considered as an irrelevant tab ** The valiad tab always has format as stated above and other file type like htm, php, org, asp and aspx. For those tab name does not fulfill this format will be analyzed independently

2.2 Tabs Names Preprocessing

2.2.1 Working Flow

Each tab has different level of sub-tabs. For example as in Figure 2, *executive-leadership* is the sub-tab of *about-us*, which is the sub-tab of company *ipsc*. So we split the tabs to get single sub-tabs according to "_". We then counted the occurrence of each sub-tabs. If a sub-tab appears several times in a website, it is still considered as once. After that we analyzed the occurrences and highlighted some important sub-tabs manually. Figure 3 illustrates the tab processing in this step.

2.2.2 Tab Analysing

In this preprocessing step, we are trying to analyze the name of the tabs of all the company website. There are two main purposes for this step. The first goal is to identify the tabs with high occurrence rate across the company websites. Meanwhile, the second goal is to identify the tabs with high correlation with the tasks we are going to perform at later steps, such as classifying company categories, extracting

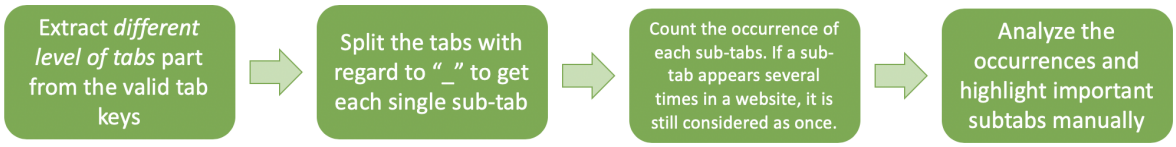


Figure 3: The workflow of tab processing

founders information, etc.
For the first goal, Figure 4 is the distribution of the company tabs with occurrence rate of over 200 for the data in the *Sum_all* folder and over 100 for the data in the *Sum* folder.

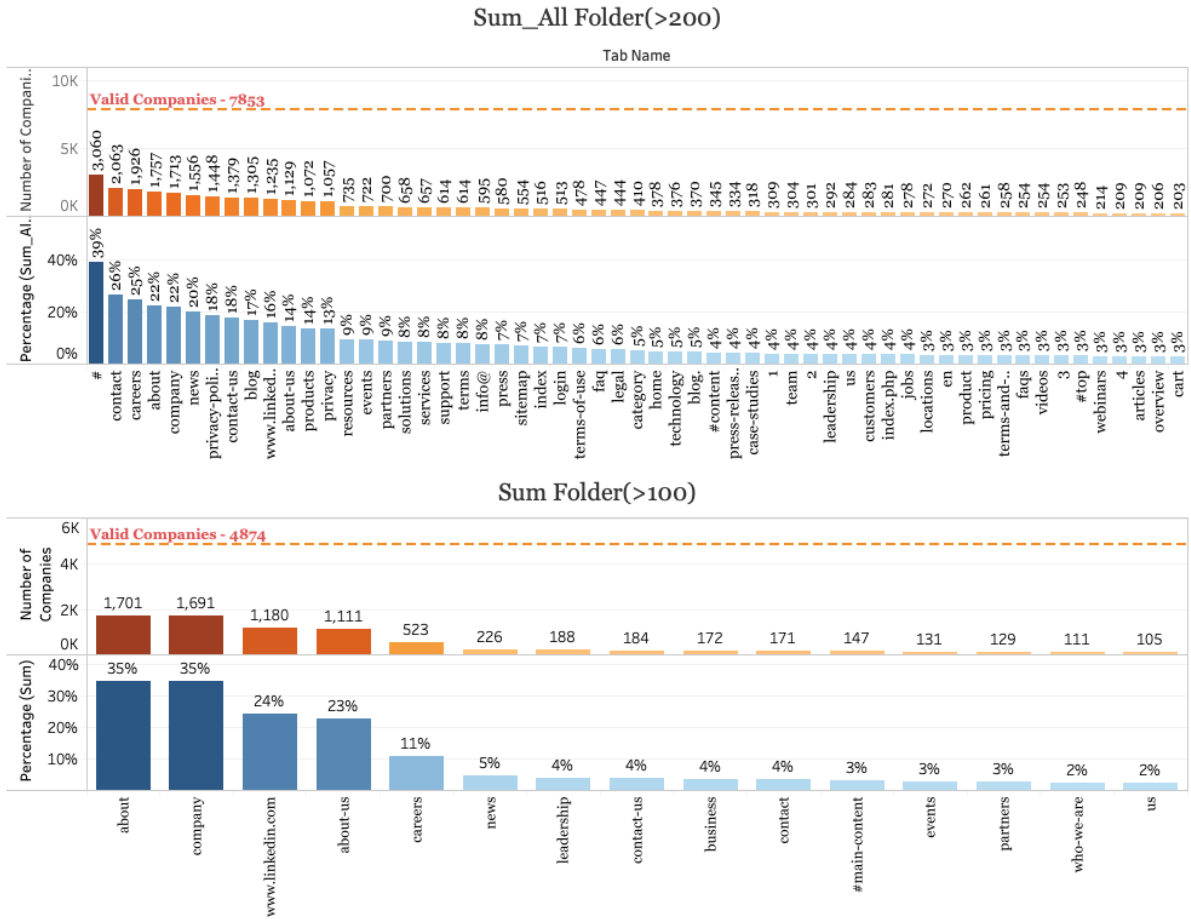


Figure 4: Tabs which have high occurrences. The upper figure shows these high frequent figures in *Sum_all* and the lower figure is for *Sum* folder. Orange color shows the occurrence number and blue color shows the percentage of the valid files. # stands for the main page of some companies.

The sub-tab names in *Sum* folder are more concentrated in fewer keywords than that in *Sum_all* folder files. There are only 5 sub-tabs which appear in over 10% percentage of the company in *Sum* folder. But there are 13 sub-tabs in *Sum_all* folder which appear in over 10% percentage of the company. This indicates we can focus less in *Sum* folder to conduct analysis.
To fulfill the second goal, the tabs names in Figure 5 and in Figure 6 are are identified manually as having high correlation with the tasks we are going to perform in the future. We consider each tab name with occurrence over 50 for *Sum* folder and over 100 for *Sum_all* folder to decide whether it contains key information for **company classification** or **founder information**. Note that some tab names may

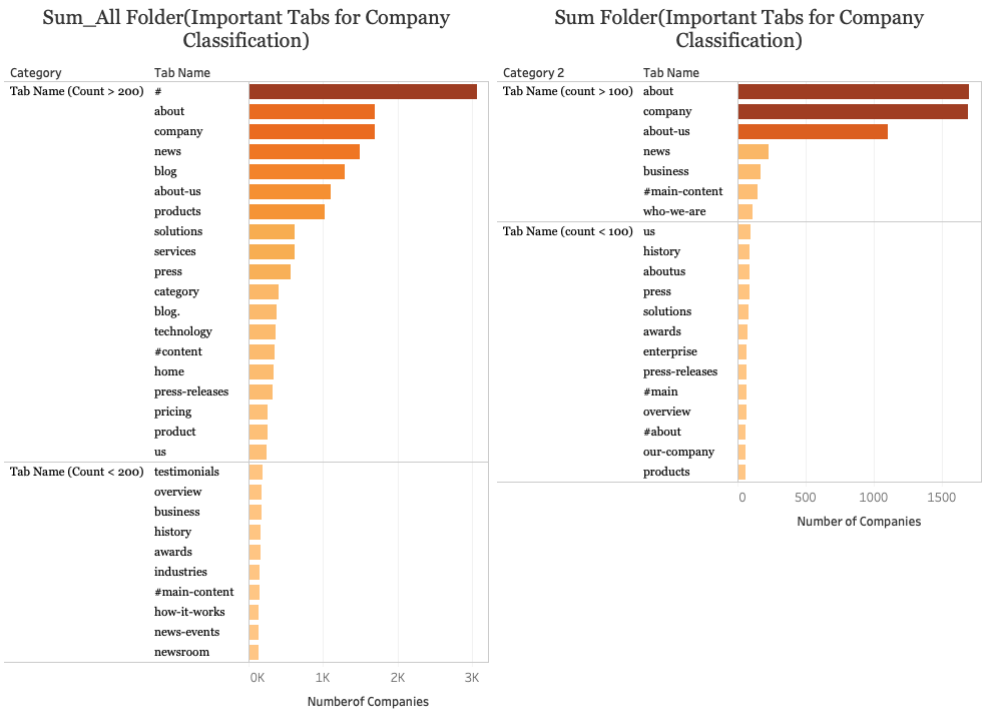


Figure 5: Important sub-tabs we have selected for the company classification. Left column shows the sub-tabs appeared in *Sum_all* folder while right column shows those appeared in *Sum* folder.

contain both.
We agree that some important tabs names that obviously contains company classification such "about-us", "company" or "news". For the important sub-tabs from *Sum_all* folder, most of them appear in more than 200 web pages, while most of the helpful sub-tabs in *Sum* folder appear less than 100 times.
For the founder information, tab names like "leadership", "about-us" are more likely to have such information.

2.3 Content Data Cleaning

Before applying any machine learning method to generate vectors for each website and doing the classification, we cleaned the webpage content as our first step. Figure 7 shows this text cleaning pipeline. Since all the content is crawled from the Internet, it contains HTML tags. So first of all we removed such tags using the package *Beautiful Soup*[1]. After that, we removed punctuation and numbers from the web content with *Regular expression package*[2], which may not be helpful for the category classification and we only kept alphabets. We also converted all the letters to lowercase to get more accurate occurrence rate of each word. Furthermore, we removed English stopwords such as 'a', 'is' etc using the *NLTK*[3] package. Then we checked the number of characters of each companies' content and removed those with fewer than 200 characters. Because content with such few characters contains very less information or states that the domain of that websie is for sale.

[1] https://www.crummy.com/software/BeautifulSoup/bs4/doc/
[2] https://docs.python.org/3/library/re.html
[3] https://www.nltk.org/

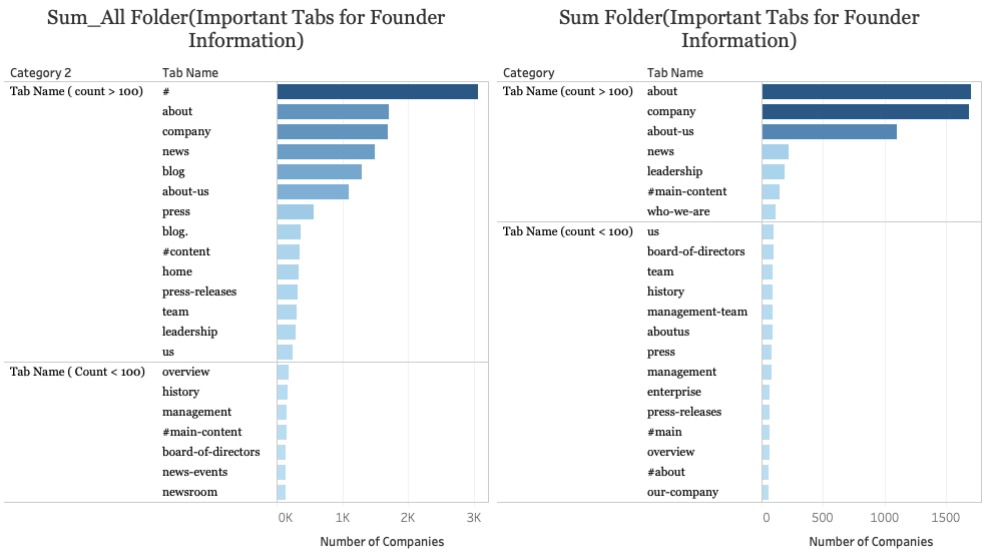


Figure 6: Important sub-tabs we have selected for selecting founder information. Left column shows the sub-tabs appeared in *Sum_all* folder while right column shows those appeared in *Sum* folder.

Figure 7: Pipeline of data cleaning for the website content

2.4 Data Regrouping

Each website belongs to industry groups and segments according to *VentureSource*[4] of Dowjones in the provided data file. The total number of segments are 28. The websites were essentially regrouped to 7 groups/categories according to VentureSource again. The reason is simplicity of data prediction. The labels are

- BUSINESS & FINANCIAL SERVICES
- CONSUMER GOODS GROUP
- CONSUMER SERVICES GROUP
- ENERGY & UTILITIES GROUP
- HEALTHCARE GROUP
- INDUSTRIAL GOODS & MATERIALS GROUP
- INFORMATION TECHNOLOGY GROUP

. Once we find out the best prediction model, we may use the 28 segments for the website prediction.

2.5 Imbalanced Data

Table 3 shows number of companies in each big category. It is very obvious that the number of companies in each category is very imbalanced. The largest category INFORMATION TECHNOLOGY GROUP has 8 times more companies than the smallest category ENERGY & UTILITIES GROUP.

[4] http://privatemarkets.dowjones.com/Deals/Help/VSPremium/Glossary_IndustryGroups.html

Category	Number of Companies
BUSINESS & FINANCIAL SERVICES	1579
CONSUMER GOODS GROUP	416
CONSUMER SERVICES GROUP	1140
ENERGY & UTILITIES GROUP	219
HEALTHCARE GROUP	1041
INDUSTRIAL GOODS & MATERIALS GROUP	477
INFORMATION TECHNOLOGY GROUP	1885

Table 3: Number of companies in each category

In order to test the influence of imbalanced classes, we tried three different ways to gain our training and testing data used for further processes:

- Group 1: directly using imbalanced seven categories
- Group 2: using the smallest category companies number (219) and selecting same number of companies with their web content from other category randomly
- Group 3: only choosing four categories which have more than 1000 companies and selecting randomly 1041 companies in each category.

We then carried out experiments and did the classification on the data selected by these three ways.

3 Classification: TF-IDF with Traditional Machine Learning Methods

In this section, we employed both supervised and unsupervised methods to classify the companies into the seven business categories based on the vectors generated by TF-IDF.

3.1 Generated Vectors

3.1.1 TF-IDF

TF-IDF stands for term frequency-inverse document frequency. The TF is number of a word appears in a document over the total number of words in the document.

$$tf_{t,d} = \frac{n_{t,d}}{\sum_k n_{k,d}} at\ t^{th}\ word\ and\ d^{th}\ document.$$

The IDF is the logarithm of the total number of documents over the number of documents which the word appears.

$$idf_t = log(\frac{D}{D_t}).$$

Then TF-IDF means these 2 terms are multiplied together.

$$score_t = tf_{t,d} * idf_t.$$

Therefore, TF-IDF is a matrix of words vs documents. Its entries represents the score of the word appearing in the document. This higher score of a word means the word higher appearing frequency in the document(TF) but little appearing in other documents (IDF). Then this word has higher possibility to be recognized as an identifier in current document.

We use the TF-IDF as word vectors for classification and prediction by different prediction models.

3.1.2 Words With High Scores

Through computing TF-IDF scores for all words from the websites, we got a vector for each company. Length of the vector is number of all the words. The value is TF-IDF score of each word in that website content. By averaging TF-IDF scores with regard to the categories, below are some interesting keywords we found among the 50 keywords with highest scores for each category.

Labels	**Keywords**
BUSINESS & FINANCIAL SERVICES	marketing,need,right,account,based
CONSUMER GOODS GROUP	order,platform,product,available,provide
CONSUMER SERVICES GROUP	best,like,experience,new,personal
ENERGY & UTILITIES GROUP	work,shall,make,solutions,energy
HEALTHCARE GROUP	health,help,provide,online,personal
INDUSTRIAL GOODS & MATERIALS GROUP	work,care,technology,products,business
INFORMATION TECHNOLOGY GROUP	systems,development,mobile,software,security

Table 4: Keywords in each category

We selected 5 symbolic ones for each category among the 50 high value words. For example, "system", "software" are the keywords of INFORMATION TECHNOLOGY GROUP and "health" indicates the HEALTHCARE GROUP. However some high scores words are obtained in every category like "information". So only the keywords could not identify the category. The machine learning classifier would also consider the distribution of values in the vectors to identify the belonged category.

3.2 Experiments and Results

We performed K-Means,K Nearest Neighbour(K-NN) and Support Vector Machine(SVM) on group 1-3 data. Using accuracy and weighted F1 score to compare them. Following are the results:

	K-Means		KNN		SVM	
	Weighted F1	Accuracy	Weighted F1	Accuracy	Weighted F1	Accuracy
Group 1	0.28	0.32	0.37	0.36	**0.62**	**0.62**
Group 2	0.23	0.28	**0.65**	**0.64**	0.55	0.56
Group 3	**0.33**	**0.36**	0.57	0.57	**0.62**	**0.62**

Table 5: Weighted F1 score and accuracy of different prediction models

SVM performed best on group 1 and 3 data because their data volume were more sufficient. So the sample size influenced the accuracy of the prediction when using SVM. K-NN has the both highest accuracy and F1 score in group 2 data because each category has the same sample size. Due to same reason, each category of group 1 data is imbalance therefore it gave the lowest accuracy and F1 Score. Among 3 models, K-Means achieved the worst performance. It required to calculate the distance of the features/ word vectors to classify the categories.It can be seen that features are overlapping in Figure 9. Then the distance in-between cannot be utilized. Therefore K-Means failed the prediction.

3.2.1 Unsupervised Learning

3.2.1.1 K-Means K-Means is one of the most commonly used unsupervised learning algorithms. It works by partition n items into K clusters so that all the items are categorized to the cluster with the nearest mean. K-Means minimizes the within-cluster variances. For this project, we used the MiNi-BatchKmeans from sklearn library with default settings.

From Table 5 we can see the performance of K-means of Group 1 and Group 3 are quite similar. Group 3 outperformed Group 1 slightly. Although the result is not good, it still exceeds the result if

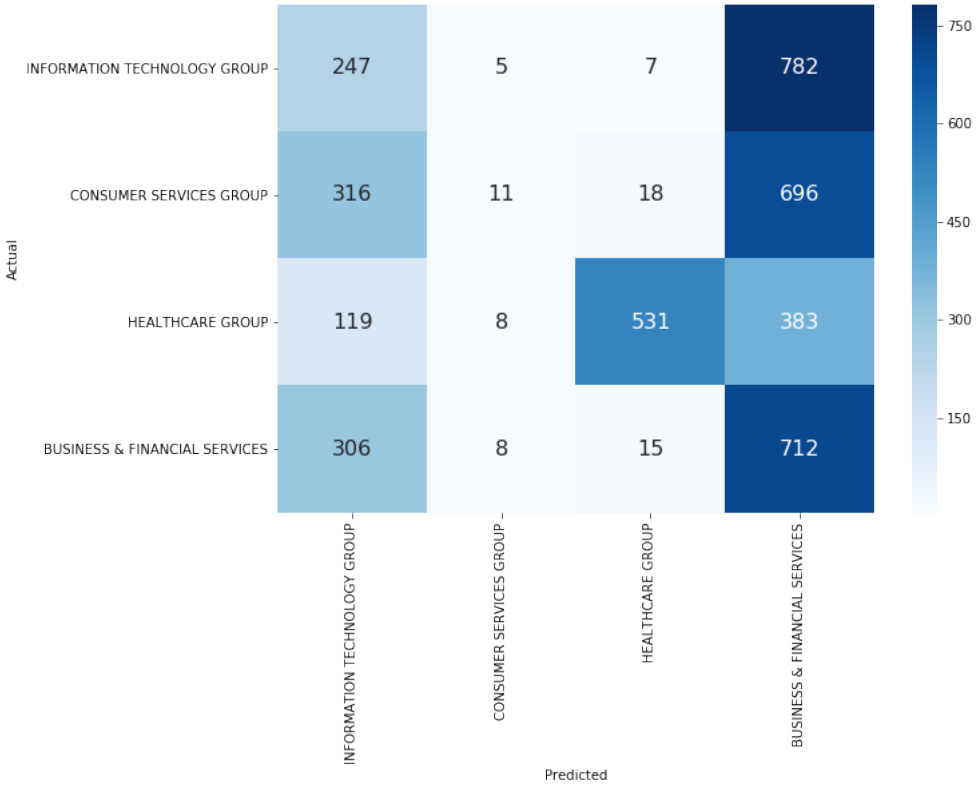
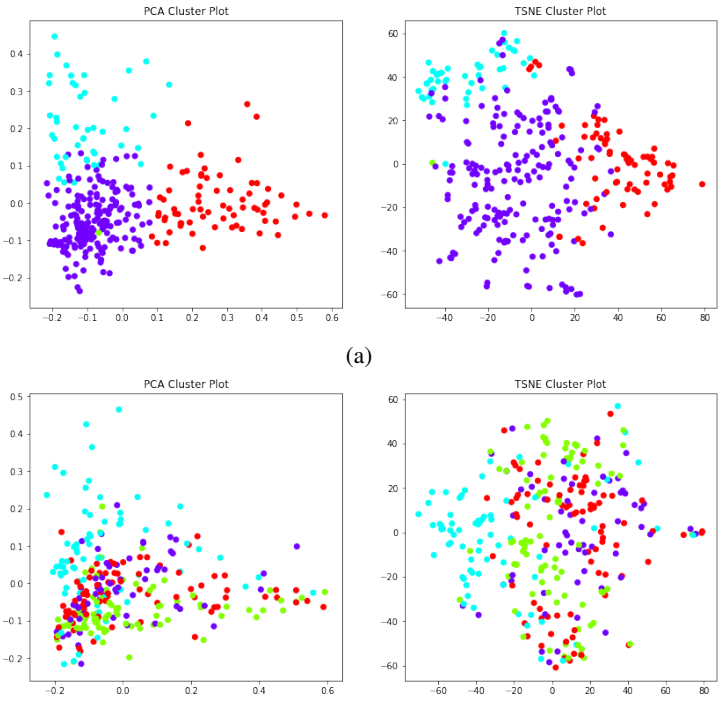


Figure 8: The confusion matrix of K-means result from Group 3



(a)

(b)

Figure 9: Visualization of TF-IDF vectors into two dimensions using PCA and T-SNE from Group 3(a) Different colors represent the predicted labels from K-means (b) Different colors show the actual labels

we classify the website randomly, which is 0.14 for Group 1 and 0.25 for Group 3. Figure 8 shows the confusion matrix of the K-means result from Group 3. From the matrix we may notice that K-

means tends to classify more companies into BUSINESS & FINANCIAL SERVICES and less data into CONSUMER SERVICES GROUP, while classifying for HEALTHCARE GROUP is relative reasonable.

In order to analyse why the results are not ideal, we also plotted the data into two dimensional space using PCA and T-SNE. Both methods are dimensionality reduction methods. PCA is better at capturing global structure of the data while T-SNE explains better the relations between neighbor. We showed scatterplots with 300 randomly sampled website of their TF-IDF score in Figure 9. The colors in the above figure represent the predicted categories by K-means while the colors in the below figure show the actual category with the content distribution. We can see there are a lot of overlapping data points in the below figure. This is also the main reason for the unsatisfied performance of K-means. Since K-means only calculated the mean distance, it is hard to tell nearby data points apart.

3.2.2 Supervised Learning

For supervised learning, three methods were being tried out, which are K Nearest Neighbours, Support Vector Machine and Neural Network. For K Nearest Neighbours, Support Vector Machine, the whole labeled data were separated with a fraction of 80% for training and 20% for testing.

3.2.2.1 K Nearest Neighbours(KNN) K Nearest Neighbours algorithm works by finding the K nearest neighbours in training data and use the labels of the nearest neighbours to predict the labels for the test cases. There are a few options to choose for the distance calculation. For a start, we will use the default distance function 'Minkowski' of the sklearn library. Further tunning and grid search should be performed as next step to improve the performance.

Among all the three groups, Group 2 has the best test performance with an accuracy of 64% and F1

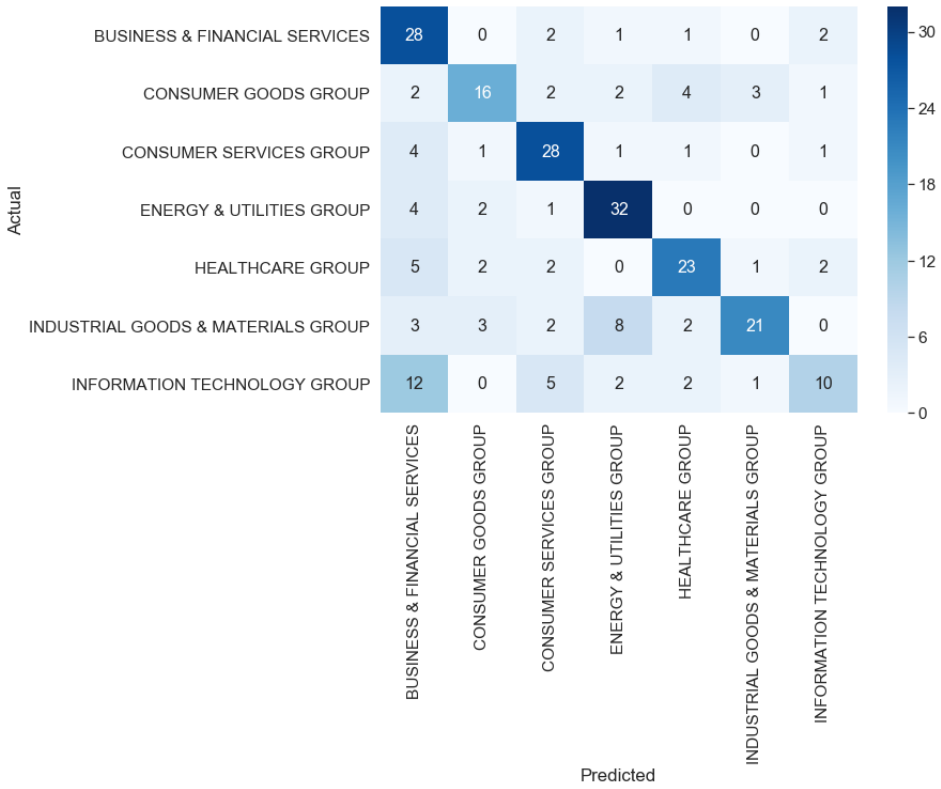


Figure 10: The confusion matrix of K Nearest Neighbours result from Group 2

score of 65%. The confusion metrics for Group 2 is shown in Figure 4 above. It can be seen from the heatmap that the highest numbers for each categories are mostly on the diagonal entries, which suggested that they are being correctly predicted. The only case where the model had a less than 50% accuracy is on the INFORMATION TECHNOLOGY GROUP.

3.2.2.2 Support Vector Machine(SVM) Support Vector Machine (SVM) is a supervised machine learning algorithm capable of performing classification, regression and even outliers detection. In 2-dimensional space, the linear support vector machine works by drawing a line to separate two classes and make sure that it is as far away from the closest samples as possible. In higher dimensions, SVM works by drawing hyperplanes to separate data so that the hyperplane has the largest distance to the nearest training-data point of any class. For simplicity, we will use the SVM with a linear kernel as a start. Different kernels would be tried out at the next step to include non-linearity and fine tune the model.

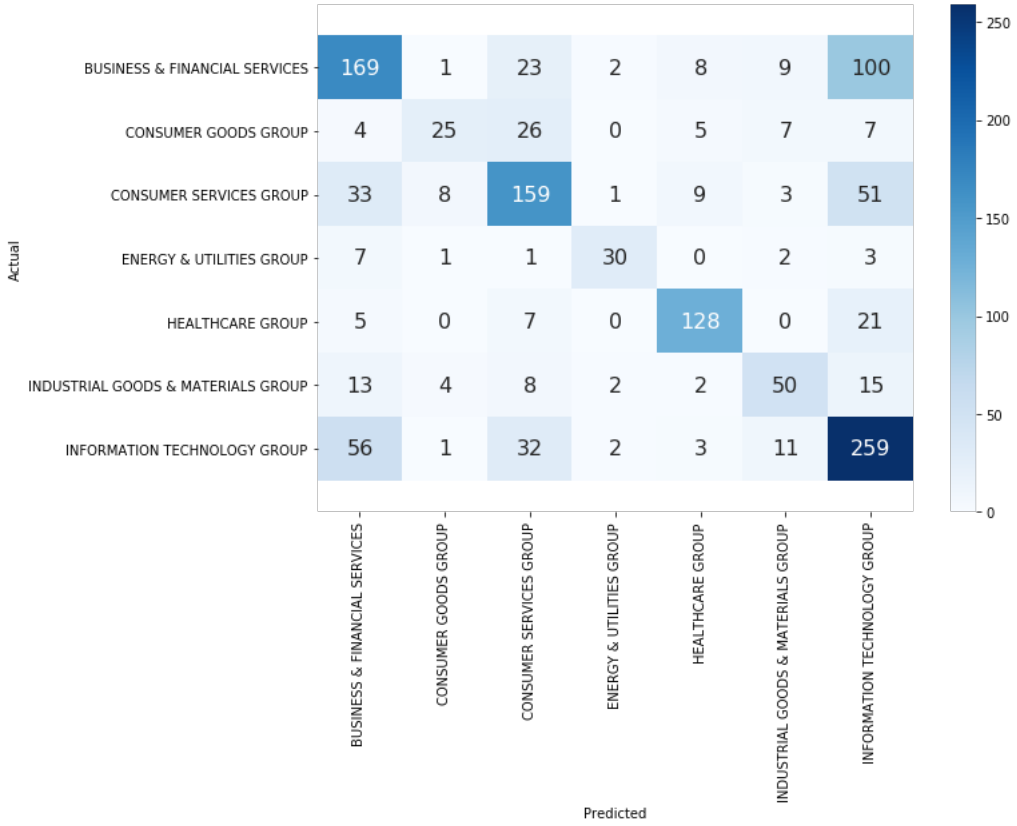


Figure 11: The confusion matrix of SVM result from Group 1

Performing SVM on group 1 imbalance data has the best performance among all the prediction models because it has the highest accuracy and F1 score(both are around 62%). The diagonal of the confusion matrix (Fig 11) is showing number of corrected predicted categories. Most of them are the largest entry among the row or column except for CONSUMER GOODS GROUP due to lack of data in this category. The sample size are influencing the accuracy.So INFORMATION TECHNOLOGY GROUP has the most corrected predicted values due to sufficiency of data volume.

3.2.2.3 Neural Network Classifier A neural network is a deep learning technique for prediction. It is built by multiple nodes and several hidden layers to stimulate a model for data to predict some results like category of a website in our case. According to universal approximation theorem, a feed forward neural network can approximate any functions or models with appropriate activation functions in nodes and parameters.

We split the data into 60% training, 20% validation and 20% testing. We use TF-IDF vectores for all seven categories from training data as the training input data for our simple neural network model. We tried different number of layers with different number of nodes in each layer. We also tried different activation function and different optimizer with some optimization tricks like weight decay.

The best accuracy from the neural newwork model is a model with 3 hidden layers(Size of each layer is 4096,1024 and 256), using ReLu activation function and Adam optimizer. However, the best prediction accuracy is 56% which is still lower than the results from SVM classifier. So we didn't explore more about the neural network method.

3.3 Category Top Score Keywords

Since the top keywords mentioned in the section 3.1.2 are not so representative, we made a deeper search into the TF-IDF scores of each category. We first selected N words with the highest TF-IDF keywords in each category. Some words, like "information", "service", "data", appeared in multiple categories. We considered these common high score keywords shared by multiple categories also as stopwords and removed them from the original web contents. Then we selected 8,000 words with highest term frequency as a basis of our vectors to each website and recalculated TF-IDF scores for each website. For the value of N, we chose 30, 50,100 and 200.

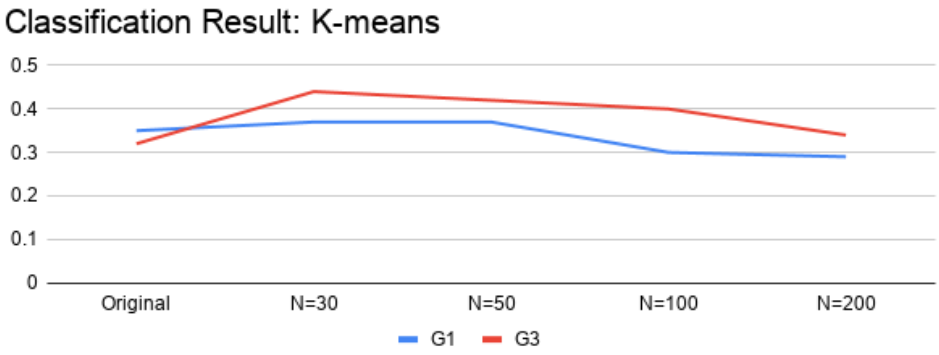
Table 6 shows top 10 words with the highest average TF-IDF scores in each categories after removing overlapping words appeared in top 50 keywords in each category. It means that these words contain most information about each category. From this table we can see quite clearly that these keywords are relative straightforward for indicating each categories.

Labels	**Keywords**
BUSINESS & FINANCIAL SERVICES	clients,financial,companies,platform,cloud,sales, media,insurance,agreement,learn
CONSUMER GOODS GROUP	food,order,ingredients,foods,quality,water, design,free,available,shall
CONSUMER SERVICES GROUP	app,free,agree,rights,shall,order, available,users,day,agreement
ENERGY & UTILITIES GROUP	energy,solar,gas,oil,power,fuel, water,renewable,wind,drilling
HEALTHCARE GROUP	medical,patients,clinical,care,patient,healthcare, cancer,treatment,dr,drug
INDUSTRIAL GOODS & MATERIALS GROUP	manufacturing,packaging,quality,equipment,design, high, steel,materials,aerospace,production
INFORMATION TECHNOLOGY GROUP	cloud,users,network,solution,platform,applications, app,sales,learn,enterprise

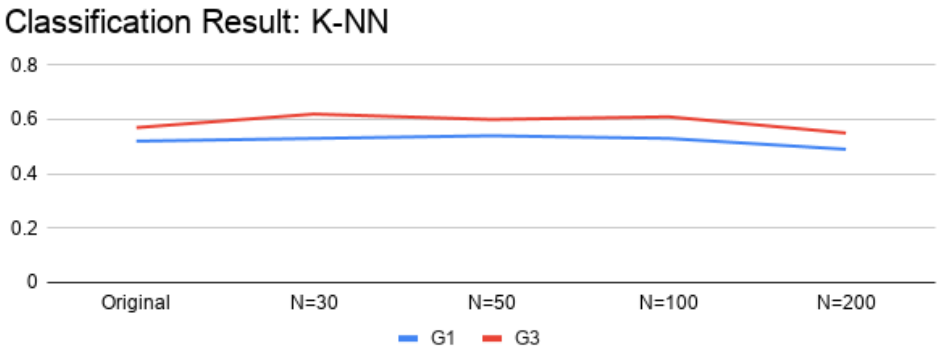
Table 6: Top 10 Keywords in each category after removing the common words shared in first 50 highest keywords in each category

We also used the updated TF-IDF scores to carry out the experiments again. Previous results suggested that G2 is having the worst performance over all the three models. This might be due to the fact that the sample size of G2 is too small. As a result, we decided to continue the experiment with dataset G1(original data with all categories) and G3(only 4 categories with each more than 1000 companies) only moving forward.

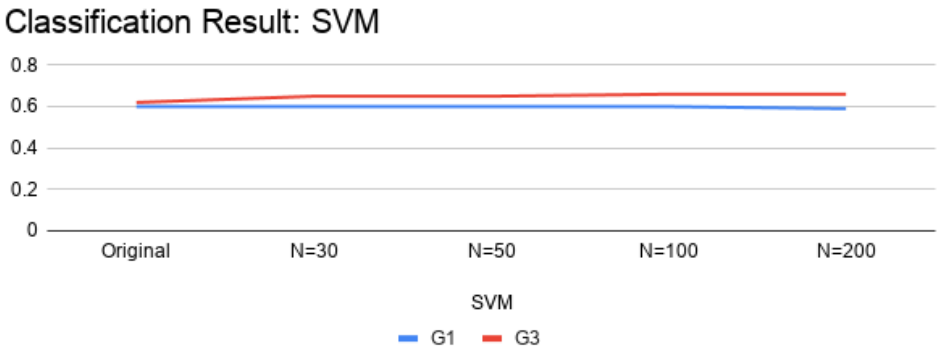
Figure 12 shows the comparison of the model accuracy by removing the common words from N top key words for dataset G1 and G3. As you can see, the model accuracy did improve after removing the common words. The accuracy is highest for K-means and KNN when N=30 or 50 while the performance of SVM is relatively stable regardless of the value of N.
Meanwhile, we could observe from Table 7 that the size of vectors for each website also influences the classification results. Especially for group 1, the classification accuracy using K-NN has increased more than 15% if we shrink the size of vectors from the size of the website content unique words(150,000) to 8,000 high term frequency words.



(a) Classification results of K-means



(b) Classification results of K-NN



(c) Classification Results of SVM

Figure 12: Comparing the accuracy of different methods after removing the common words from N top keywords for dataset G1 and G3

3.4 Other Experiments

3.4.1 Result After Choosing Tabs

As we analysed the tab names in the beginning of this project, occurrence of words in tabs are also taken account into the prediction. The tabs are selected by filtering the occurrence of words larger than 100 and the length of the word larger than 1. The original of company number were 6698. After filtering, companies left are 6388.

By comparing the previous group 1 original data, effect of filtering important tabs are not observable. It is because K-means results of filtering tabs are slightly better than original data but K-NN and SVM are a little bit worse. Then our assumption about important tab containing key information of the company is not true. Other tabs could also contain relevant information which is able to classify companies.

	Vector Size	
	150,000	8,000
Accuracy	0.36	0.52

Table 7: Comparison of Group 1 with different vector size using K-NN method

	K-Means		KNN		SVM	
	Weighted F1	Accuracy	Weighted F1	Accuracy	Weighted F1	Accuracy
Group 1 Original	0.28	0.32	**0.51**	**0.51**	**0.62**	**0.62**
Group 1 Choosing Tabs	**0.33**	**0.33**	0.5	0.5	0.6	0.6

Table 8: Weighted F1 score and accuracy of group 1 data whether choosing tabs

3.4.2 Result After K-folder Cross Validation

In order to make sure that the result from the models can be generalized to other data, we decided to adopt a technique called K-folder cross validation for the model training and validation. K-folder cross validation is widely used in applied machine learning to estimate how a model would perform on unseen data. Empirical results have suggested that a K value of 5 or 10 would yield a test error rate that suffer neither from excessively high bias nor from very high variance. For our project, we decided to use k value of 5. The result for the SVM model after applying 5-folder cross validation is shown in Table 4 below. You can see that the k-folder cross validation result is close to our original model test result(62%), which suggested that our model can be well generalized on unseen data.

	SVM	
	Weighted F1	Accuracy
Group 1 Cross Validation	0.6	0.6
Group 3 Cross Validation	0.65	0.65

Table 9: Weighted F1 score and accuracy of group 1 and group 3 of SVM Model

4 Other Vectorization Techniques - Word2vec/Doc2vec

Beside the TF-IDF method to get vectors for different website, we have also explored another two techniques to generate vectors: one is using the Word2Vec model based on the results of TF-IDF. The other is using the Doc2Vec model.

4.1 TF-IDF with Word2Vec

Word2Vec[Mikolov et al., 2013] model uses to generate vector embedding for each word. This model was developed by Tomas Mikolov in 2013 at Google. It is claimed that the embedding is capable of capturing context of a word in a document, semantic and syntactic similarity, relation with other words, etc.

However, Word2Vec model only generates vectors for each word. In order to use this model, we first analyse the result of TF-IDF model. We chose N words with the highest TF-IDF value in each website. Then we generated embedding for these N words using Word2Vec and calculated the average of these vectors to get final vector for each company. The size of vectors for each website is 300. After that, we put the vectors into SVM classifier(80% training data and 20%testing data) and get the classification result.

For choosing the best N, we tried 10, 50, 100 and 200. Our model to get vectors using **100** top TF-IDF

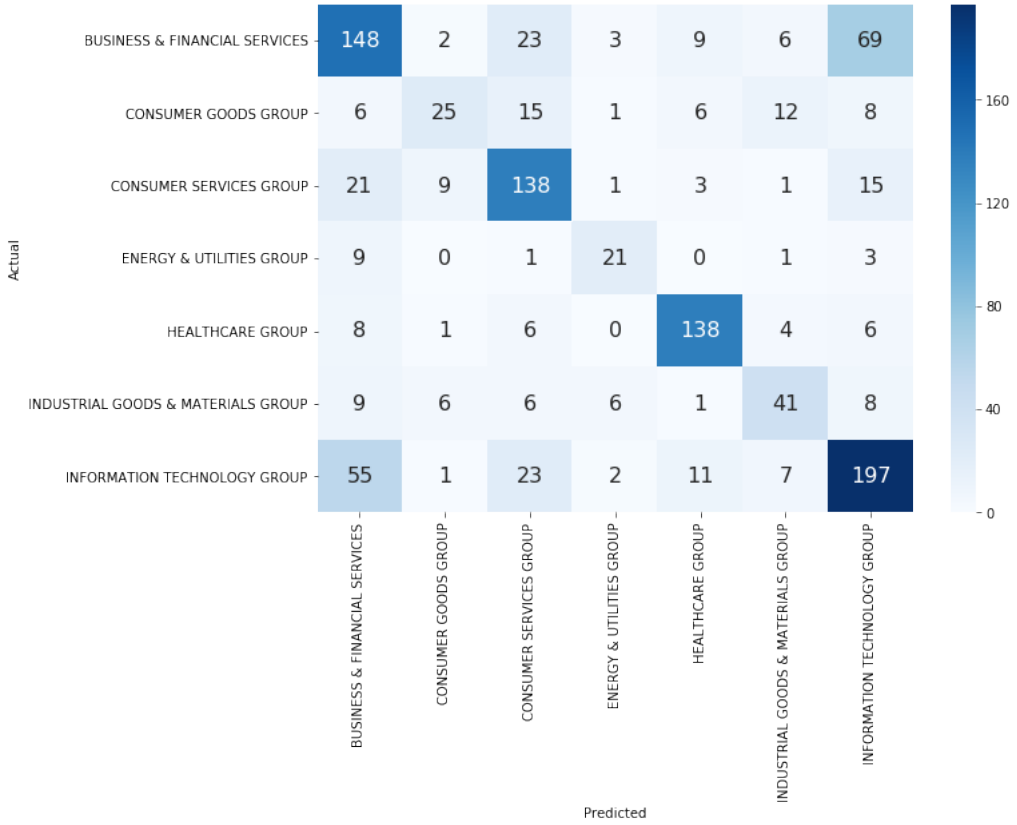


Figure 13: Confusion matrix of the classification accuracy using TF-IDF with Word2Vec model

score words achieves the accuray **65%**, which is the best so far in our all experiment. Figure 13 shows the confusion matrix of this classification result. We may notice that this model classify the most cases correctly.

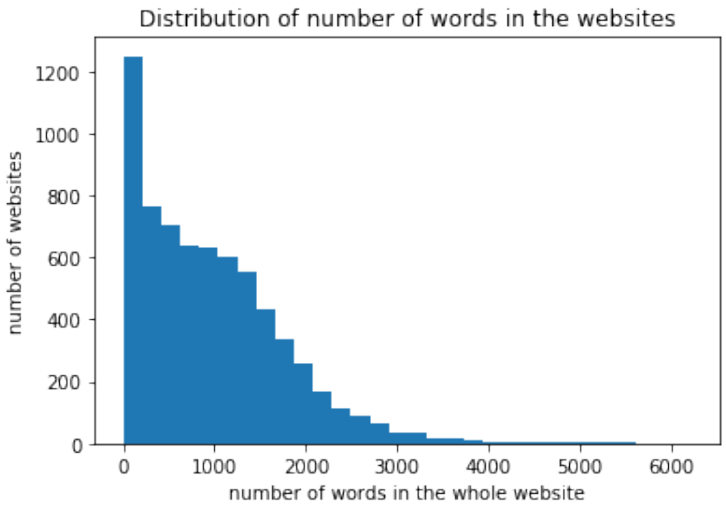


Figure 14: Historgram of word count in each website

We think the main reason for this good result is that we filtered all the words in the website to find the most representative words. Also Word2Vec is a powerful model to generate vectors for these words. Besides. we also involved a an additional data cleaning step in this model. Figure 14 shows the histogram of word count in each website. x-axis labels the number of words in each website, and y-axis

means the number of companies. We can see that more than 1000 companies have less than 200 words in the whole website. So we removed these websites with words less than 200. Figure 15 shows the number of websites in each category with the category percentage in the whole corpus.

We randomly selected 300 vectors of the website generated by the TF-IDF with Word2Vec model and plotted them into two dimensions as shown in Figure 16. From the plot using PCA to reduce dimensions, although there are a lot of overlapping among the points, but we may observe that the HEALTHCARE GROUP is in the middle upper position(hell green), CONSUMER SERVICES GROUP lies in the button position(yellow). INDUSTRIAL GOODS & MATERIALS GROUP is in the middle upper location(dark blue) while CONSUMER GOODS GROUP is in the middle bottom(red). This plot is much more clear than the two-dimensional representation for only TF-IDF scores(in Figure 17).

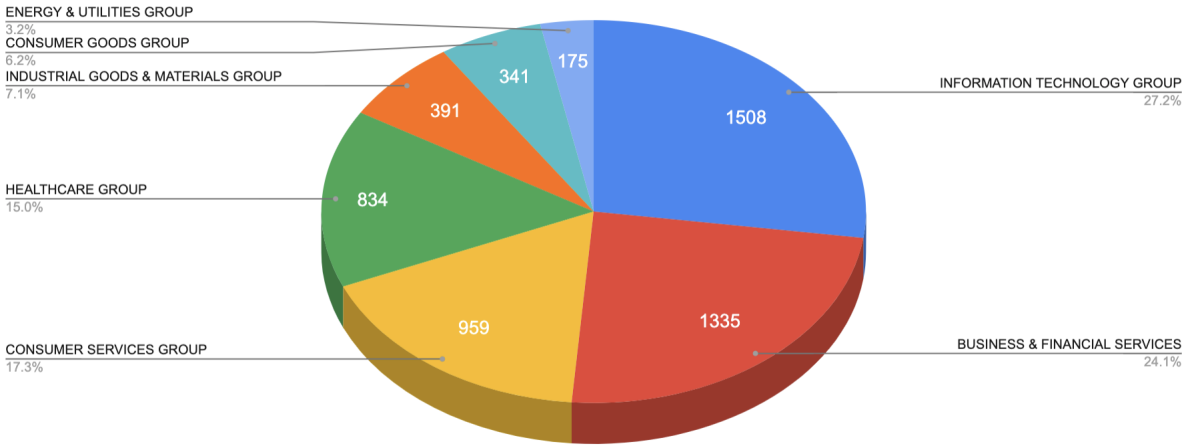


Figure 15: Websites in each category and their percentage

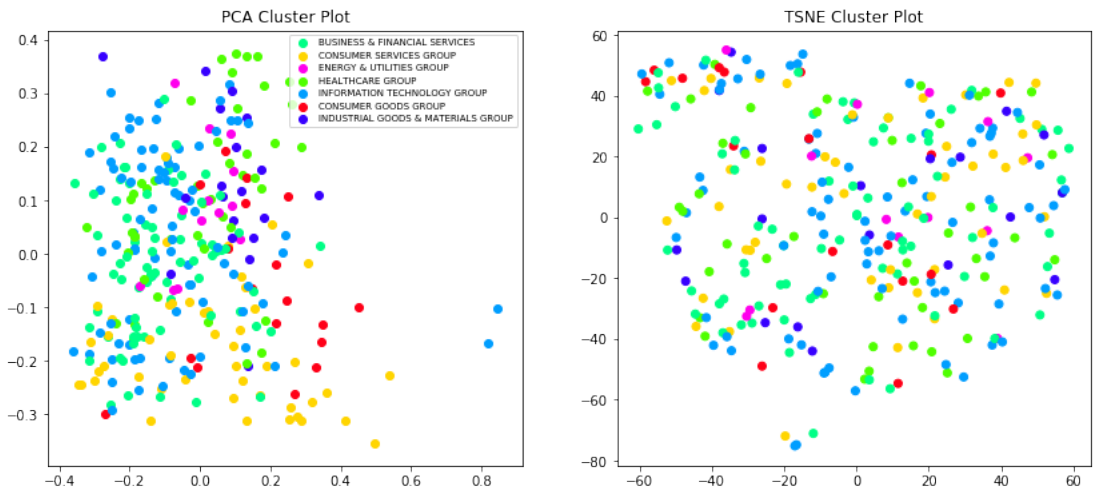


Figure 16: Plot the vectors generated by TF-IDF with Word2Vec into two dimensions using PCA and T-SNE

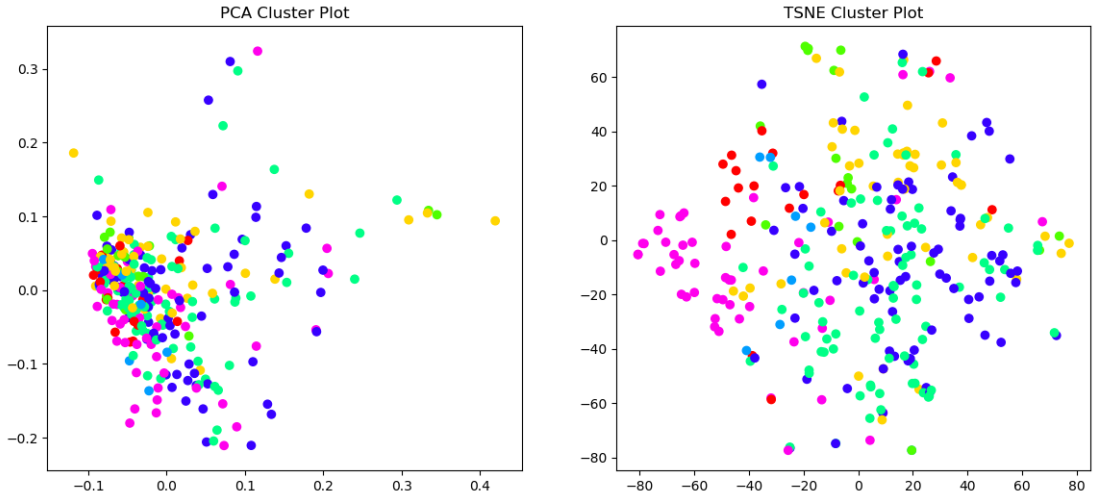


Figure 17: Plot the vectors generated by TF-IDF scores into two dimensions using PCA and T-SNE

4.2 Doc2Vec

Doc2vec is a NLP technique to represent a complete document as a vector and is the generalized version of word2vec method. In our project, we will be using the doc2vec techniques in Gensim. The process flow is as follows:

Text preprocessing →Build the Doc2Vec Model →Infer the feature vector →Build the classifier .

The *text preprocessing* step involves removing punctuation, tokenization, creating tagged content, etc. For the *Doc2Vec* step, there are two main algorithms which are Distributed Bag of Words(DBOW) and Distributed Memory(DM). In our project, we have decided to pair these two algorithms as empirical result has shown improvement in accuracy by combining these two algorithms. In the *feature vector* step, a inference training process is conducted to find a good vector to predict the website's word. Then at the last step, a SVM classifier is built to utilize the vector obtained from previous step to classify the website business categories.

Figure 18: Plot the vectors generated by Doc2Vec into two dimensions using PCA and T-SNE

Figure 18 shows the plot of the vectors generated by Doc2Vec. It can be seen that the clustering is even clearer with this approach compared with TF-IDF and Word2Vec in the last section. On the other hand,

the company categories classification result on Group 1 data is shown in Figure 19. An accuracy of 53% is achieved which is slightly lower than the TF-IDF and Word2Vec approach.

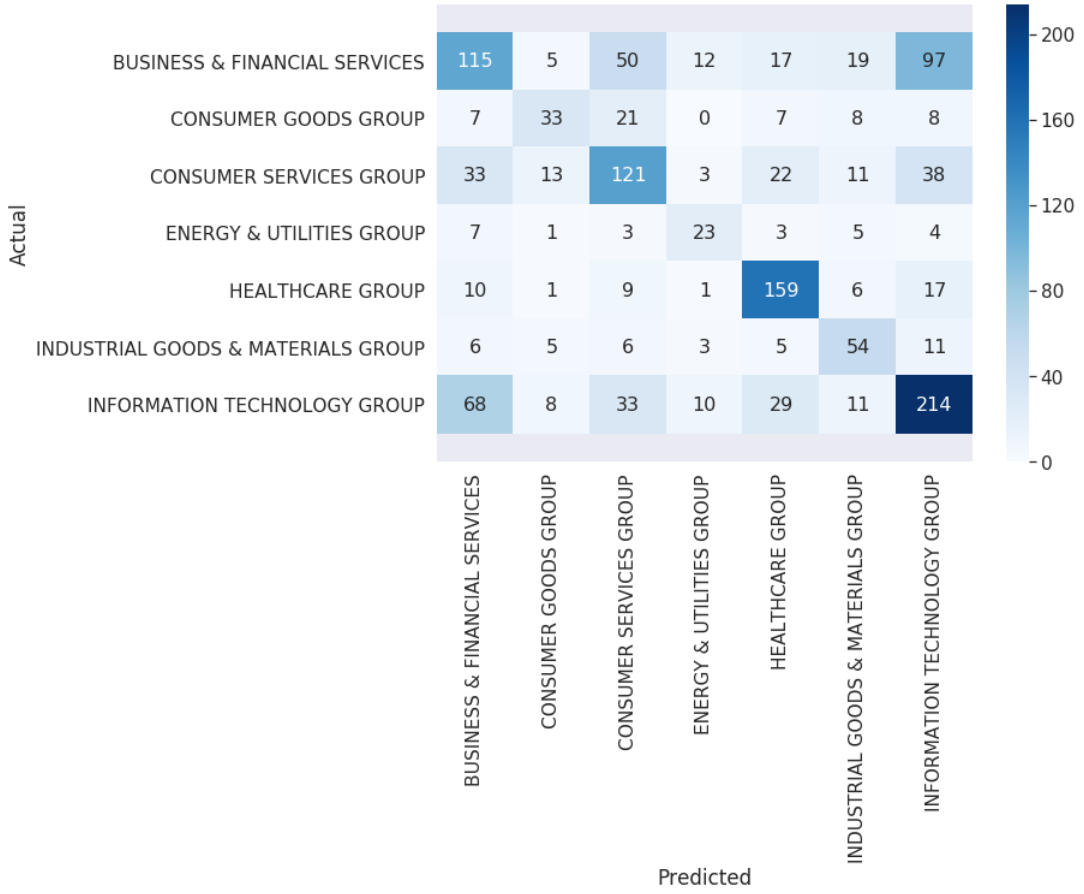


Figure 19: Confusion matrix of the classification accuracy using Doc2Vec model

5 Similarity Comparison

In this section, we explored the similarity across different business categories. Text Similarity is one of the essential NLP techniques to find the closeness of different groups of texts. This allows the user to extract semantically similar questions, searching similar documents or recommending similar articles, etc. In our project, we have decided to adopt this technique to compare the similarity within different business categories as well as across them. Hopefully, this can help generate insights to find companies with similar business models. In order to perform text similarity calculation, a classic process flow is:

$$\text{Text preprocessing} \rightarrow \text{Feature extraction} \rightarrow \text{Vector similarity calculation} \quad .$$

As mentioned in previous sections, *text preprocessing* step includes removing html tags, removing stop words, etc. *Feature extraction* can be done in several ways such as TF-IDF, word2vec, doc2vec, etc while *vector similarity calculation* can be achieved by using euclidean distance, word mover's distance, cosine similarity, etc. Among them, our project decided to use the cosine similarity method, which is also the most widely used vector similarity calculation method.

5.1 Similarity with TF-IDF

Similarity matrix for the feature vectors extracted by TF-IDF is shown in Figure 20. It can be observed that the diagonal values are the biggest for each row and each column. This indicates that the similarity within the business categories is always higher than across the categories. However, it is also noticed

that the similarity values are relatively low with a feature vector of size 8000 for each website. This is because that the similarity scores decreases as the feature vector size increases and the magnitude of the tfidf value of each word doesn't impact much the cosine similarity.

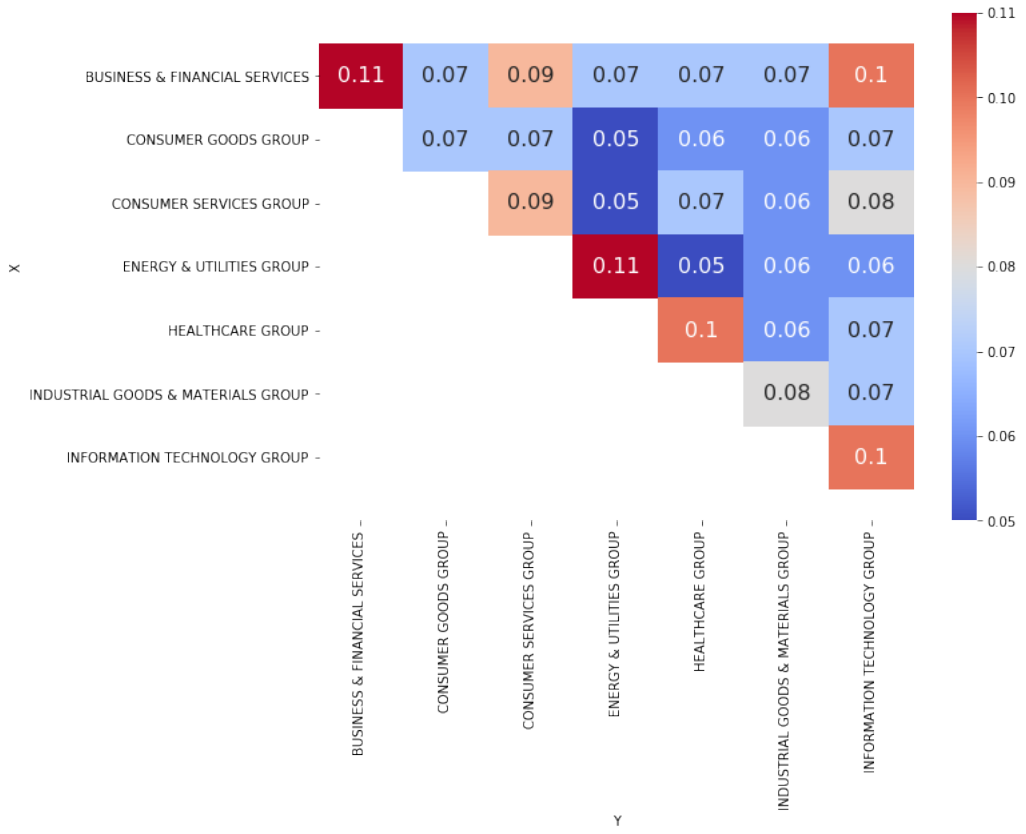


Figure 20: Similarity Matrix of TF-IDF

5.2 Similarity with Word2Vec

Top 100 tf-idf words were selected before vectorization because top 100 tf-idf gave the best accuracy as mentioned above. The words were vectorized by GoogleNews-vectors-negative300[goo, 2013]. The average word2Vec values of all words were calculated in each website. We sorted all categories and compared them by computing their similarity.

In the similarity matrix, the diagonal entries are the optimal value in each row or column. So it can concluded that each category always has the most similarity compared to itself.
Once a new website appears, we can compute its similarity with each category respectively. Except for BUSINESS & FINANCIAL SERVICES and CONSUMER SERVICE, the category of the website can be estimated by checking the largest similarity of the category it compared with.

5.3 Similarity with Doc2Vec

Similarity matrix for the feature vectors extracted by Doc2Vec is shown in Figure 22. It can be observed that the diagonal values are again the biggest for each row and each column. Moreover, it seems also to suggest that BUSINESS & FINANCIAL SERVICES and INFORMATION TECHNOLOGY GROUP has a close relationship as the similarity values across these two categories are very close to the similarity values within their categories.

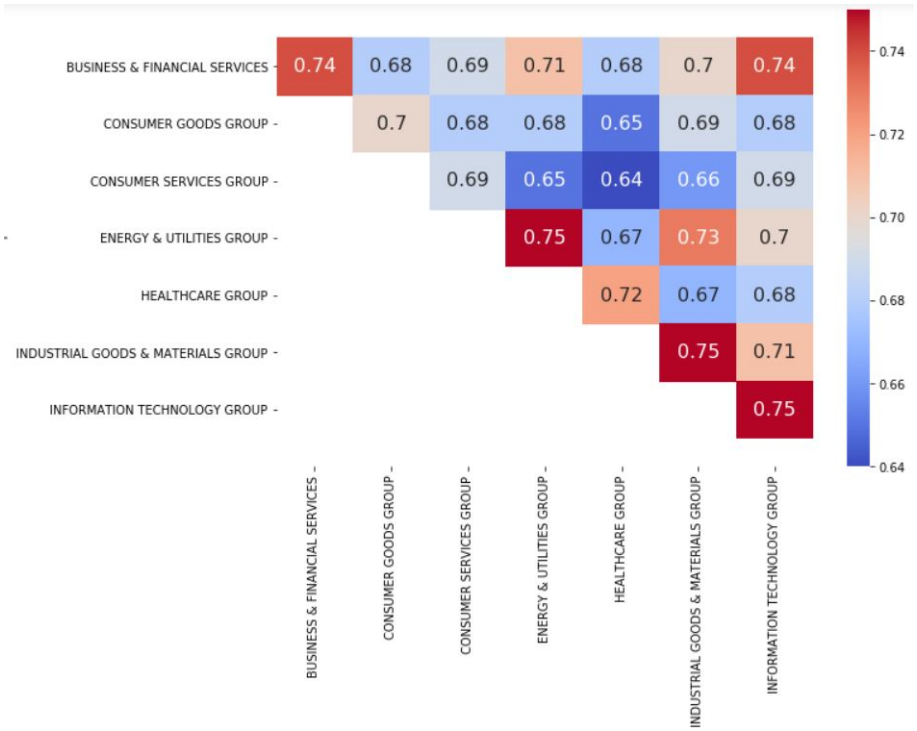


Figure 21: Similarity Matrix of word2vec

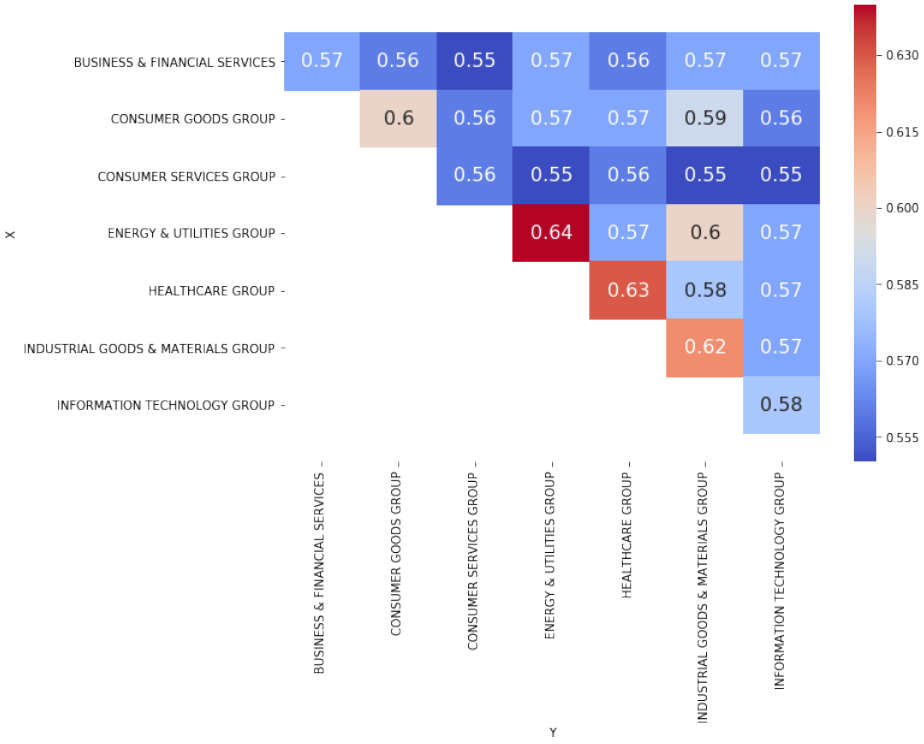


Figure 22: Similarity Matrix of Doc2Vec

6 Subcategory Analysis

In this section,We have explored the subcategories of several big categories with the unsupervised method - K-means. We chose the TF-IDF value as input for K-means instead of Word2Vec, because it is quite straight forward to see the high frequency key words with regard to the TF-IDF value to verify the

classification result. Our process flow is:

Clean Data →Find Optimal Number of Cluster →Classify Using K-Means →Analyse Results .

We analysed subcategories of 5 groups with relative more data - INFORMATION TECHNOLOGY GROUP, BUSINESS & FINANCIAL SERVICES, HEALTHCARE GROUP, CONSUMER SERVICES GROUP, and ENERGY & UTILITIES GROUP. In general, we found that the sub-classification for the category 'CONSUMER SERVICES GROUP' is most meaningful. This class has also the most balanced data. For other categories, the results suffer from the imbalance of the data. We analysed the result in Section 6.1 and gave a summary for other categories in Section 6.2.

6.1 Category "CONSUMER SERVICES GROUP"

Table 10 gives an overview of the given information of the Category "CONSUMER SERVICES GROUP". From the table we can observe that this category has four subcategories. Each subcategory has about 200-400 companies. The top keywords for each subcategory are also representative.

Subcategory	# Companies	Top Keywords
Consumer Information Services	407	website,terms,time,content,site
Media and Content	261	education,new,students,learning,content
Travel and Leisure	220	items,service,products,shipping,order
Retailers	201	personal,service,website,services,travel

Table 10: An overview of the Category "CONSUMER SERVICES GROUP" with the given subcategories, number of companies in each subcategory and the top TF-IDF keywords.

Firstly, we need to decide the number of clusters. We put the TF-IDF scores of each website into the k-means classifier. We iterated number of groups from two to ten. Then we got the SSE score from the K-means classifier, which calculates the distance between the website and the center of the assigned group. So the lower the score is, the better that number of group is. The given number of subcategories is 4. However, from the Figure 23, we can see that K-means suggests that we should choose 8 as the number of clusters for classifying subcategories in "Consumer Services Group".

Figure 24 shows the top TF-IDF keywords after classifying all the companies from "Consumer Services Group" into 8 subcategories. This classification mainly represented the original 4 given subcategories, while they also explored more details inside the given 4 classes. For instance, under the subcategory "Media and Content", we have now two groups, one is in the education direction, the other is in the content or video direction. Also for the subcategory "Retailers", our classifier finds a group with sweets like candy or chocolate. With the classification, we know more details about the websites in "Consumer Services Group". This classification is quite successful.

6.2 Other Categories

Other categories are very imbalanced in their subcategories. For example in INFORMATION TECHNOLOGY GROUP, most of websites inclined to the Software subcategory. So the clustering results are not as obvious as the "CONSUMER SERVICES GROUP".
Similarly, the "BUSINESS & FINANCIAL SERVICES" does have extraordinary clustering results as well but some noticeable keywords for **insurance industry** from 44 websites. It does not have such subcategory for it. So in this case we can add extra subcategory for insurance industry.

Apart from that, groups sometimes are sharing similar business. For example, the border of the

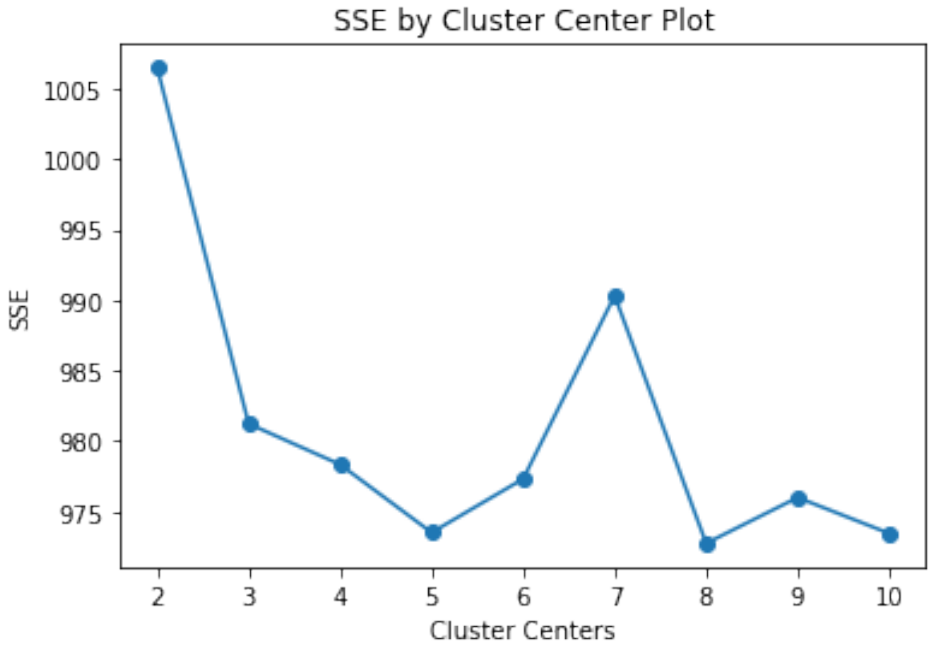


Figure 23: SSE scores for different number of groups for Category "CONSUMER SERVICES GROUP"

Subcategory	**#Company**
Software	1246
Communications and Networking	222
Electronics and Computer Hardware	218
Semiconductors	89

Table 11: An overview of the Category "INFORMATION TECHNOLOGY GROUP" with the given subcategories, number of companies in each subcategory and the top TF-IDF keywords.

business support service and financial services are quite blur. Their websites have similar keywords then it is hard to cluster them by using TF-IDF.

7 Website Recommendation

In this section, we compared the recommendation performance based on different vectorization methods and then built a Flask App to allow the user to interact and get recommended similar companies.

7.1 Recommendation Systems Comparison

We implemented the cosine similarity to each category in the last report. We found that each group has highest similarity value to its own group. So an idea comes up so that we can recommend similar websites to an incoming new website also by using cosine similarity. Our process flow is:

Generate Vectors (tfidf/doc2vec/wordvec) →Compute Similarity →Pick 5 Websites with highest Score .

We tested a website "wwww.sbamerica.com" into our recommendation system. This website is promoting healthy food. The contexts of the websites are food and beverage related. Then we used three methods to recommend our existing websites. Here are the results:

Most of the websites recommended by three methods belongs to CONSUMER GOODS GROUP. Among the three methods, word2vec approach gives the best result. We believe the reason is that Word2Vec model uses to generate vector embedding for each word. The embedding is capable of

```
Cluster 0    Travel and Leisure
cruise,cruises,river,travelers,trip,tours,tour,trips,stride,travel

Cluster 1    Consumer Information Services
personal,privacy,content,website,terms,service,site,services,use,information

Cluster 2    Consumer Information Services
investment,fleet,instructors,place,funding,business,training,driving,instructor,red

Cluster 3    Retailers
chicago,information,catering,site,beef,llc,hot,theme,dogs,il

Cluster 4    Media and Content
content,powhow,yahoo,team,creators,videos,live,streaming,rdf,video

Cluster 5    Retailers
corp,parkway,com,gourmet,va,la,chocolate,foods,source,candy

Cluster 6    Consumer Information Services
business,like,services,site,service,com,time,use,new,information

Cluster 7    Media and Content
teaching,use,information,university,college,student,school,learning,education,students
```

Figure 24: Top TF-IDF keywords from classified subcategories

```
In [181]: #TFIDF Similarity
          recommend(Input_Company, top_k, X_tfidf, y_tfidf, Website_tfidf)

          Website:  www.republicind.com  Category: CONSUMER GOODS GROUP  Similarity: 0.35
          Website:  www.sirkensingtons.com  Category: CONSUMER GOODS GROUP  Similarity: 0.23
          Website:  www.wholesomesweeteners.com  Category: CONSUMER GOODS GROUP  Similarity: 0.21
          Website:  www.pure360.com  Category: BUSINESS & FINANCIAL SERVICES  Similarity: 0.15
          Website:  www.bakewisebrands.com  Category: CONSUMER GOODS GROUP  Similarity: 0.15

In [191]: #Doc2vec Similarity
          recommend(Input_Company, top_k, X_doc2vec, y_doc2vec, Website_doc2vec)

          Website:  www.lyndale.co.uk  Category: CONSUMER GOODS GROUP  Similarity: 0.91
          Website:  www.perosbio.com  Category: HEALTHCARE GROUP  Similarity: 0.91
          Website:  www.huskietools.com  Category: INDUSTRIAL GOODS & MATERIALS GROUP  Similarity: 0.91
          Website:  www.kerznercareers.com  Category: CONSUMER SERVICES GROUP  Similarity: 0.91
          Website:  www.futuresbtc.com  Category: HEALTHCARE GROUP  Similarity: 0.91

In [182]: #Word2vec Similarity
          recommend(Input_Company, top_k, X_word2vec, y_word2vec, Website_word2vec)

          Website:  www.thanasi.com  Category: CONSUMER SERVICES GROUP  Similarity: 0.92
          Website:  www.mainstreetgourmet.com  Category: CONSUMER GOODS GROUP  Similarity: 0.91
          Website:  www.bellisiofoods.com  Category: CONSUMER GOODS GROUP  Similarity: 0.91
          Website:  www.caesarspasta.com  Category: CONSUMER GOODS GROUP  Similarity: 0.9
          Website:  www.rusticcrust.com  Category: CONSUMER SERVICES GROUP  Similarity: 0.9
```

Figure 25: Recommendation for www.sbamerica.com by three methods

capturing context of a word in a document, as well as semantic and syntactic similarity with other words. As a result, it was able to generate the most similar vectors for words with semantic similar meanings. When calculating cosine similarity of the vectors of the websites top key words, it can give a good representation of the similarity of the business nature of the companies.

In TF-IDF, although most of the websites recommended belong to CONSUMER GOODS GROUP, some websites are not food and beverage related. The first one www.republicind.com is an interior design company, whose area is actually different.

In Word2Vec, all the websites recommended are food and beverage related. For example, www.thanasi.com is a fast-moving consumer-goods company focusing on instant food product.

7.2 Flask App

In order to make sure that the user can interact with the recommendation system easily, we have built a graphic user interface based on the Flask web framework which is a python written micro framework

and doesn't require particular libraries or tools to function. Below is the architecture of the Flask App:

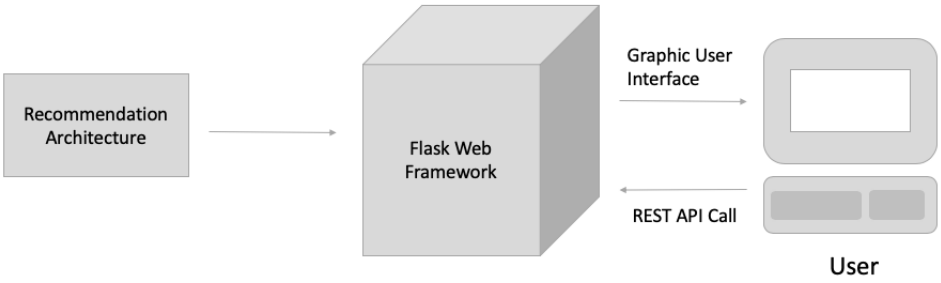


Figure 26: Architecture of Flask App

On the GUI, the user will just need to select the number of companies to recommend and input the website of the company to be recommended for. The app will calculate similarity in the background and output the name of the recommended companies and the category it is belonged to. Below is an example of the GUI:

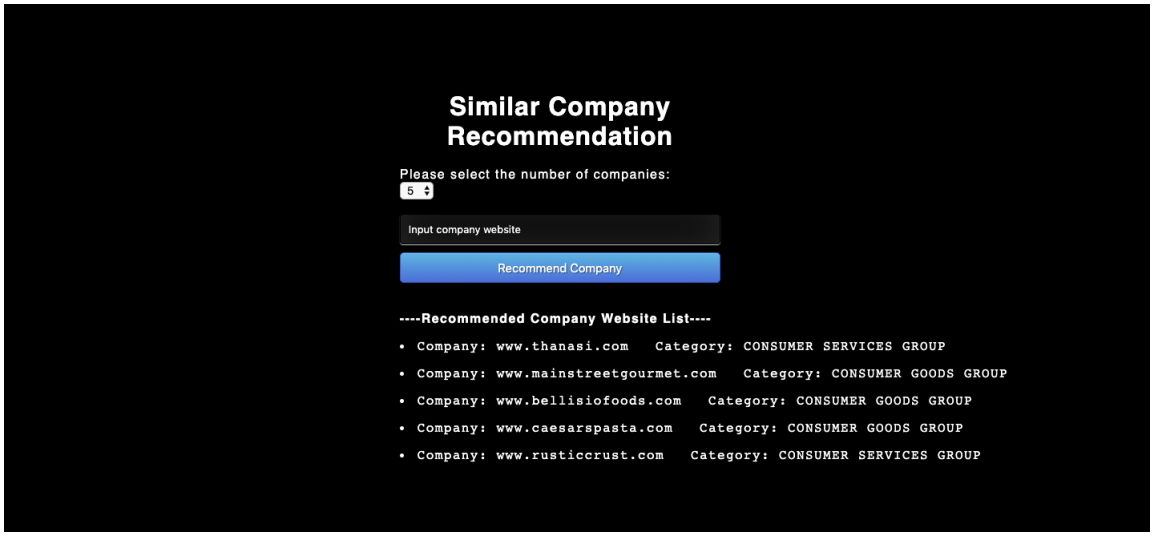


Figure 27: GUI of Flask App

8 Conclusion and Future Work

In order to find the similarity of different start-up companies, we started from the webpages of each company. We cleaned the raw data and analyzed the occurrence of tabs from each website to get insight into important webpage tabs. We used three different methods - TF-IDF, Word2Vec based on TF-IDF and Doc2Vec - to generate the vectors for different start-up companies, which could represent the content of each company's website. After that, we tried different machine learning classification methods including K-Means, K-Nearest Neighbour, SVM, and swallow Feed Forwarding Neural Network to classify the start-up companies into different industry categories. After carrying out a variety of experiments, we found out that using Word2Vec for top key-words of each company based on their TF-IDF scores gave us the best classification accuracy of 65%. Also, we calculated the cosine similarity

among different companies and different categories using the generated vectors. We found out that the Word2Vec methods gave us the highest average similarity among the same group of companies and the largest difference among different categories of companies. Furthermore, we also explored the website under one big industry category to see whether there are different subcategories among them. Moreover, we also wrote a small recommendation application using Flask to return the most similar websites when giving a website as input.

For future works, it is still worth trying the state-of-art natural language processing deep neural network models like Bert[Devlin et al., 2018] to produce the vectors for company webpage contents. Besides, using pre-trained neural network encoder-decoder models like T5[Raffel et al., 2019] from google and applying transfer learning to classify the start-up companies may also help to reach better classification scores. A more sophisticated user interface for the recommendation application could also be designed and implemented in the future.

References

Google code archive - long-term storage for google code project hosting., Jul 2013. URL https://code.google.com/archive/p/word2vec/.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Gerard Hoberg and Gordon M Phillips. Text-based network industries and endogenous product differentiation. Working Paper 15991, National Bureau of Economic Research, May 2010. URL http://www.nber.org/papers/w15991.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019.