Final report

---- Categorizing User’s Feature through Twitter Data and Machine Learning

Dingle Zhang, Lu Wei, Xingxing Li

[Background 2](#_Toc501135715)

[Environment 2](#_Toc501135716)

[Approaches 3](#_Toc501135717)

[Get data from twitter 3](#_Toc501135718)

[Filter the useless words 3](#_Toc501135719)

[Expand the data 4](#_Toc501135720)

[Classify the expanded data into different topics 7](#_Toc501135721)

[Generate the diagram of the result 8](#_Toc501135722)

[Results 8](#_Toc501135723)

[Conclusion 12](#_Toc501135724)

[Test Setup Guide 13](#_Toc501135725)

[Install the required libraries 13](#_Toc501135726)

[Acquire Tokens from Twitter, MonkeyLearn, Bing 15](#_Toc501135727)

[Change the setting variable in code 17](#_Toc501135728)

# Background

Twitter has been growing rapidly since 2006. For example, Twitter had more than 500 million users in May 2015, including more than 332 million active users. According to statistics, when Leonardo won the 88th Academy Award for Best Actor in February 28 in 2016, the news of this discussion on Twitter reached 440,000 times per minute. It shows that the amount of data on Twitter is very large. In order to get more information about what people are talking about on Twitter, we need an effective way to get real-time Twitter hot topics. It can analyze the users’ feature through their tweets, favorites and bios. Machine learning is a good tool to help us realize this function. Using machine learning for topic classification is a method of categorizing text in a supervised machine learning approach.

# Environment

**Software:** PyCharm 2017.3 (Community Edition)

**Operating System:** Windows 10

**Python version:** Python 3.6.0

**Repository requirement:** see in the *requirements.txt*

# Approaches

## Get data from twitter

In our project, we need to get the data before we analyze it. At first, we think we can get external datasets from some websites, However, we think the data is too old to be representative. We need to get the in-time data so that this project can be practical. So we use the external Python library Tweepy to get the user data from Twitter.

1. Acquire the API token.

Go to <http://apps.twitter.com>, log-in using your twitter accounts. Register a new app. You will get a consumer key and a consumer secret. And you can also get an access token and an access token secret.

1. Authentication

Twitter use OAuh authentication we can use the token we have got to access the API.

1. Get the descriptions, timeline and favorites of a user.

Using the function get\_friends\_description, get\_tweets in the source code, we can get the descriptions, timeline and favorites of a user

## Filter the useless words

Since tweets can contains a lot of useless information which can slow down the classification step sharply, we choose to process the data first.

1. Choose the first 500 tweets with most importance

Considering users can have many tweets which are not necessarily representatively, so we want to reduce some unimportant tweets and data. So we give every tweets a score by adding the number of favorite and retweet times. And choose the first 500 tweets with most score.

1. Filter the word which is not English

Since our classifier can only recognize English, we filter non-English words in order to increase our accuracy using the language detector from python library NLTK, using NLTK we can detect the language of each word, then we calculate the ratio of English words in a tweet. Keep the tweets if the ratio of English words is the most.

1. Extract the keyword for each keyword

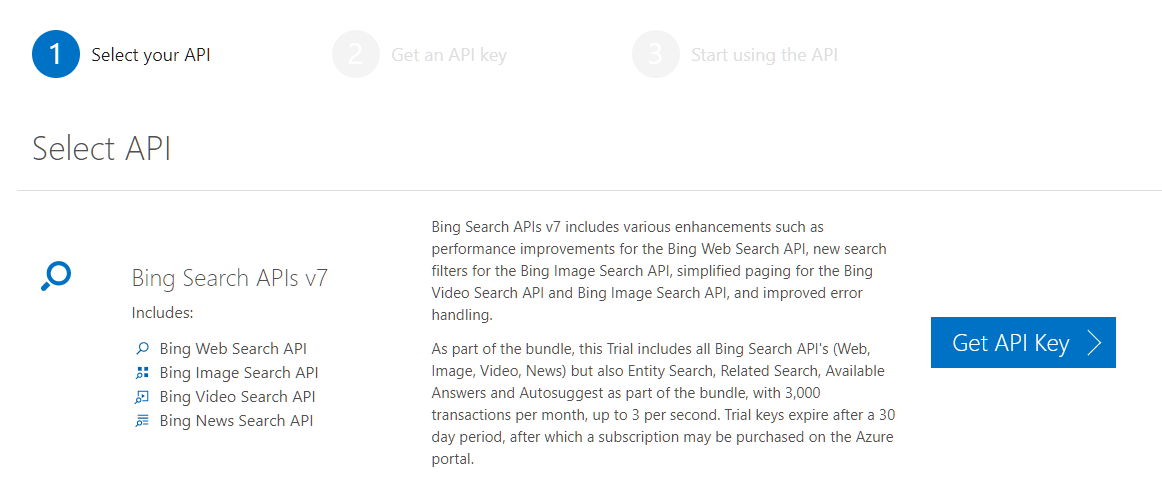
In order to improve the speed of processing, we extract the keyword of each tweets, filter the other information. We use the keyword extractor from MonkeyLearn. After this step, we have datasets which only contain keywords so that we can further use Bing search engine to expand the keywords.

## Expand the data

To get more accurate results of classifying, we need to expand the corpus we get. We use Bing search engine provided by Microsoft to expand our corpus.

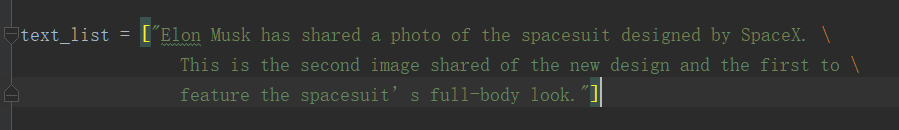
1. Get the authorization from Microsoft

To use the Bing search APIs, we need to apply for a token from Microsoft. The website of application is: <https://azure.microsoft.com/en-us/try/cognitive-services/?api=bing-web-search-api> The picture shows the entry of this service:

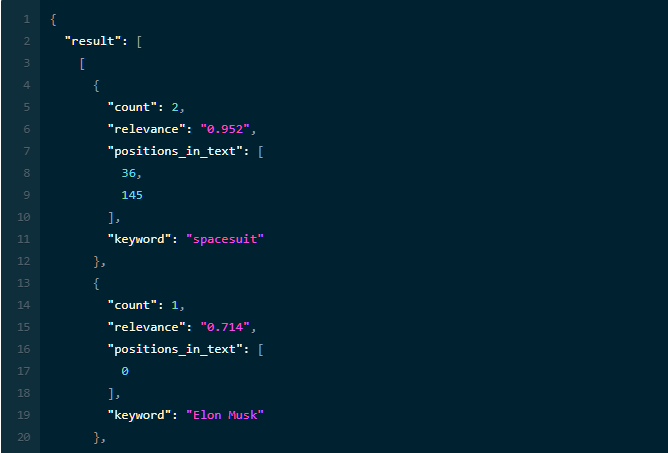


1. Extract key words of each tweets of the twitter user

We extract the key words of the tweet by using the topic extractor. For example, if we need to analyze a tweet like this paragraph:

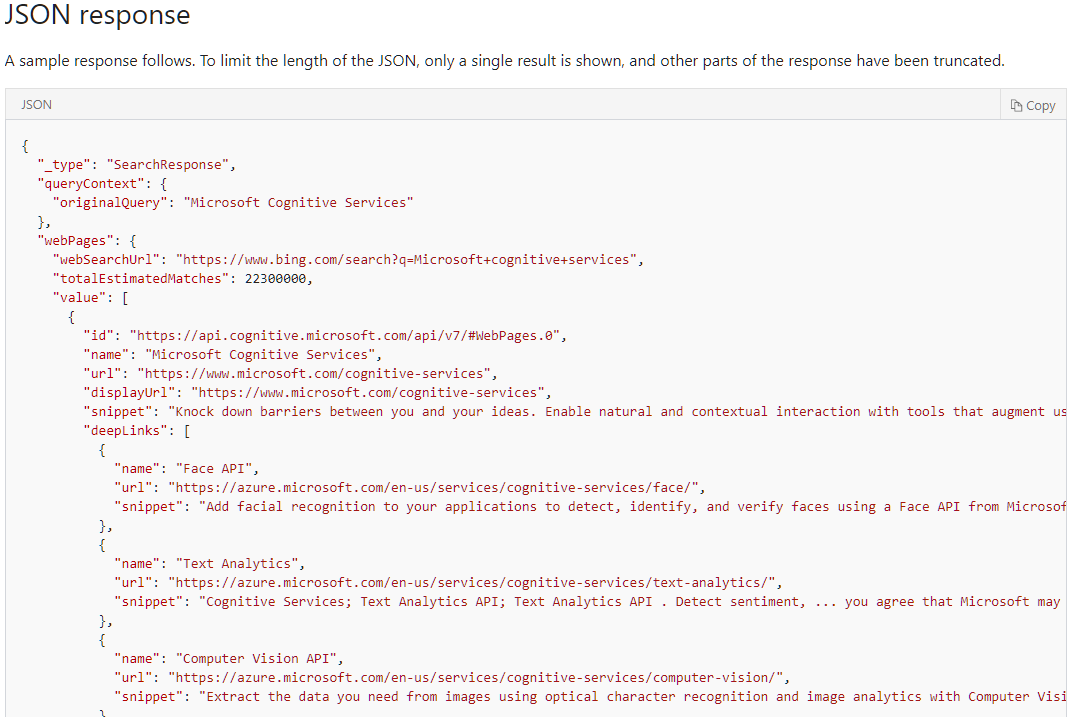


by using the extract, we can get a JSON object containing the keywords like this:



1. Using Bing search engine to search the extracted key words

After getting the key words, we can use Bing search engine to search these key words. The Bing search engine will return a JSON object like this:

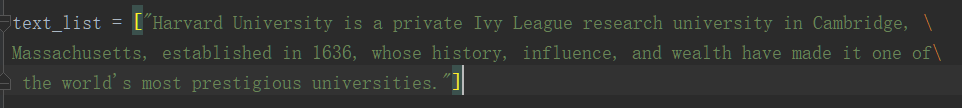


1. Using the top 5 webpages found by Bing to expand the corpus

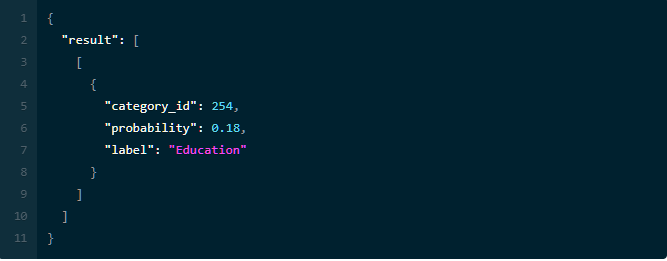
When we get the response, we can use the names and descriptions of the top 5 webpages to expand the text lists. This operation will give more information to our topic classifier to get a more accurate result.

## Classify the expanded data into different topics

After getting the corpus, now we can use the topic classifier to figure out the topics of each tweet we find. The topic classifier can analyze the text lists and give the topic of these sentence. For example, if we have this paragraph:



By using the topic classifier, we can get a JSON object containing the topic lists like this:



If a tweet involves several topics, we can easily assemble them together by concatenating the labels and multiplying the probabilities. Also, we will discard some topics since they have low probalilities.

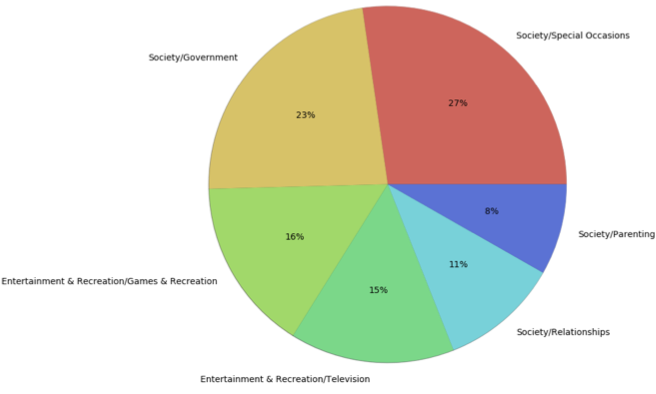
## Generate the diagram of the result

After getting the list of topics, we can count the occurrences of topics and then use the Pyplot APIs plot the diagram.

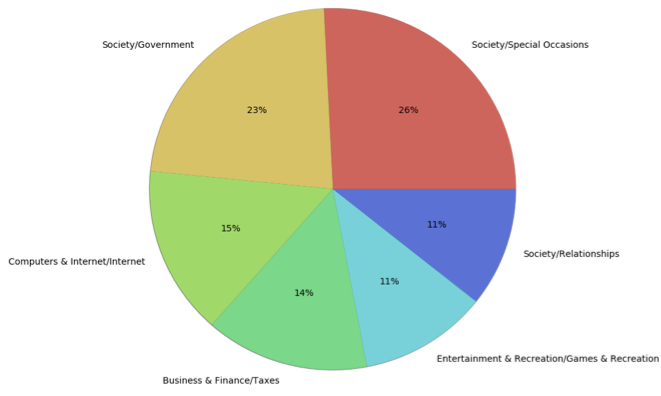
# Results

We select the 3 Twitter accounts and different data to compare the unexpanded result and the result after expanding.

The first account is @realDonaldTrump. We chose the first 100 person that he follows, his first 200 tweets and the first 200 tweets he likes. We get the result as follows. The figure 2 has one more topic classifier called Business & Finance/Taxes than figure 1. We examined his Twitter later and we did find a lot of related topic about Business & Finance/Taxes. Then we believe that after expanding, we get more accurate data than without expanding.

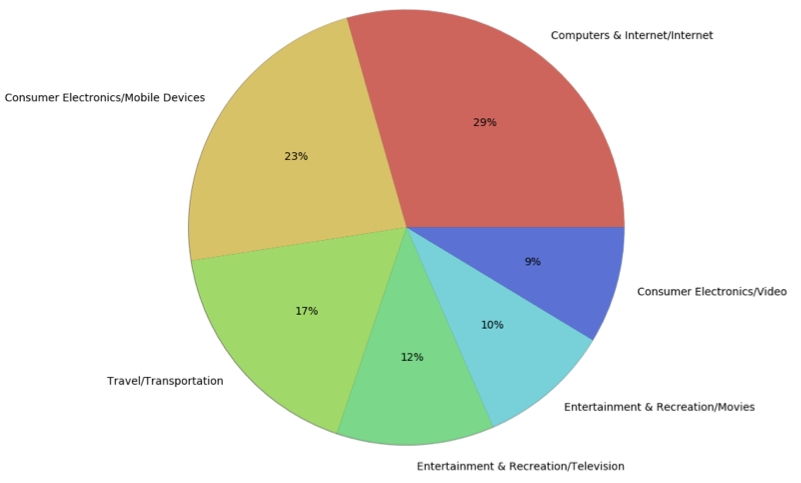


**Figure 1. Unexpanded data of Donald Trump**

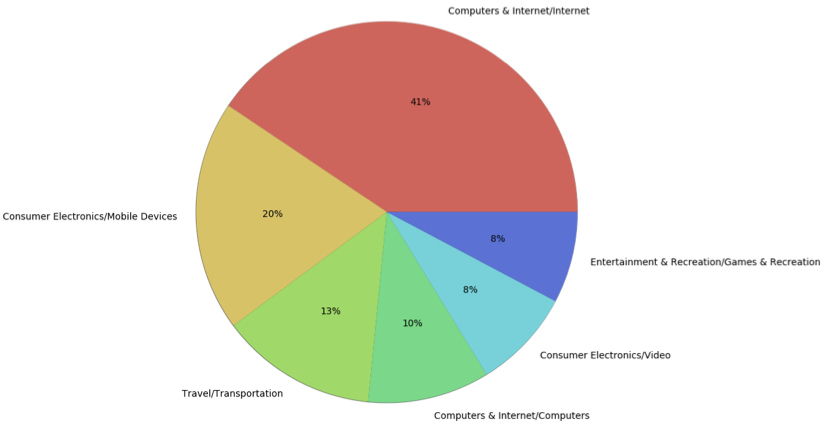


**Figure 2. Expanded data of Donald Trump**

The second account we chose is @verge. As we all know, the verge is a famous technology media. It provides news, product reviews, podcasts, videos and other information. We chose the first 100 person that it follows, its first 100 tweets and the first 200 tweets it likes. We got the data as follows:



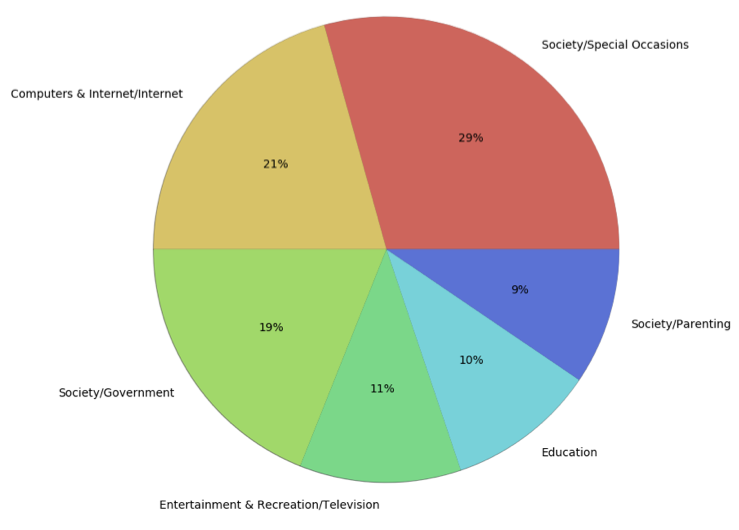
**Figure 3.** **Unexpanded data of The Verge**



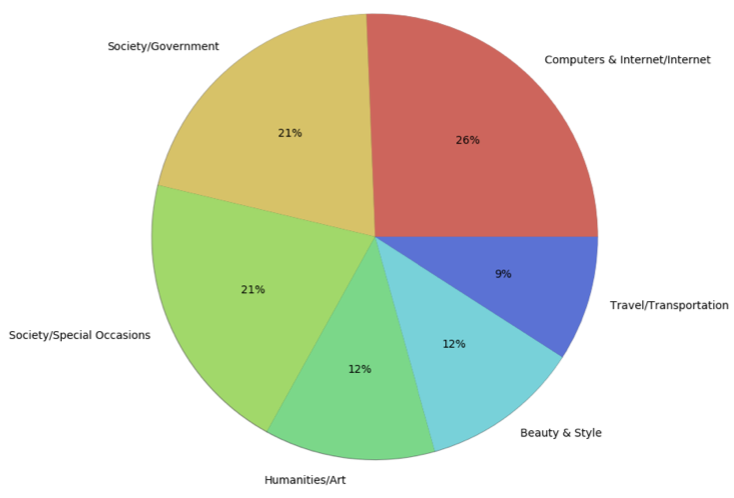
**Figure 4. Expanded data of The Verge**

As we can see, the percentage of Computers & Internet has 12% more than the chart before expanding. Also, we checked its twitter and made statistics after that, we found Computers & Internet did take a large proportion. Then we believe that the data after expanding is more accurate than before.

The last account is @nytimes. We chose the New York Times because it has more kinds of information and it has a considerable influence. We selected the first 100 person that it follows, its first 100 tweets and the first 100 tweets it likes. The pie charts of unexpanded and expanded show below. According to the observation, we found after expanding, we can get more information we don't have before and greatly improved the accuracy.



**Figure 5.** **Unexpanded data of The New York Times**



**Figure 6.** **Expanded data of The New York Times**

# Conclusion

Why do we need topic classification? Twitter as a blog platform, it's tweets are full of a wealth of point of view. The topic classification also has a broad application scenario, such as the following aspects:

First, topic classification can help users make the decision whether to buy. For example, when a consumer hesitates to purchase a product, it would be natural for the consumer to see what other people are saying about this product. If there is a big number of "likes," the consumer may make the purchase, whereas if the "unlike" is the majority, then the consumer will not make the purchase normally. It would be more convenient for users to conduct a topic analysis on social media such as Twitter, which is both time-consuming and widely-spoken.

Also, topic classification can help companies conduct market research. With the introduction of a new product, companies can derive useful information from a multitude of user reviews, such as what users like and do not like, positively and negatively impact the company's products and services. So that enterprises can understand their own advantages and disadvantages, they can better formulate appropriate measures to improve the service to occupy the initiative in the fierce market competition.

As users are relatively free to express their opinions on social media, this makes social media an important way to generate and disseminate public opinion topics. Through the topic classification of social media, we can understand the public opinion for the government and guide public opinion to provide effective tools. For negative news, you can timelier appease good people's emotions, to avoid further deterioration of the situation. At the same time, the government can also develop appropriate strategies to improve existing services.

With the development of the Internet, more and more people are willing to post their opinions about an event on the social platforms like Twitter, Facebook and etc. A typical example is the recent ALi artificial intelligence using neural networks, emotional perception and other technologies on the TV show "I'm a singer" to make a successful prediction. In addition, Twitter is often used to predict the election, stocks and so on.

# Test Setup Guide

## Install the required libraries

Use pip command to install all the required libraries. Here attached all the required libraries:

|  |  |
| --- | --- |
| Library name | Version |
| Jinja2 | 2.7.3 |
| MarkupSafe | 0.23 |
| PyMySQL | 0.7.11 |
| argparse | 1.2.1 |
| backports.ssl-match-hostname | 3.4.0.2 |
| certifi | 14.5.14 |
| idna | 2.6 |
| ipython | 2.3.1 |
| matplotlib | 1.4.2 |
| mock | 1.0.1 |
| monkeylearn | 0.3.7 |
| nltk | 3.2.5 |
| nose | 1.3.4 |
| numpy | 1.13.3 |
| pyparsing | 2.0.3 |
| pyreadline | 2.1 |
| python-dateutil | 2.4.0 |
| pytz | 2014.10 |
| pyzmq | 14.4.1 |
| requests | 2.4.3 |
| requests-oauthlib | 0.4.1 |
| scipy | 1.0.0 |
| six | 1.7.3 |
| tornado | 4.0.2 |
| tweepy | 3.1.0 |
| wsgiref | 0.1.2 |

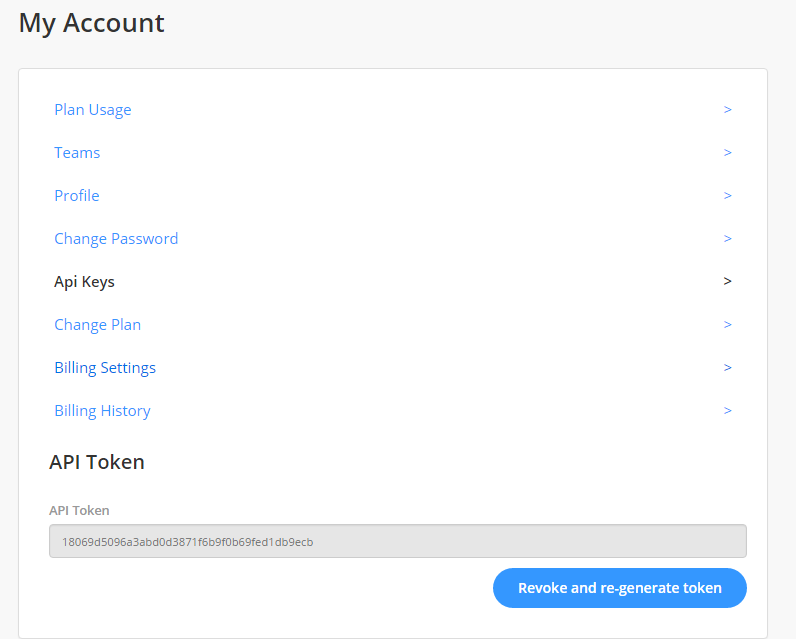
## Acquire Tokens from Twitter, MonkeyLearn, Bing

1. Twitter

The first step is the registration of your app. In particular, you need to point your browser to <http://apps.twitter.com>, log-in to Twitter (if you’re not already logged in) and register a new application. You can now choose a name and a description for your app (for example “Mining Demo” or similar). You will receive a consumer key and a consumer secret: these are application settings that should always be kept private. From the configuration page of your app, you can also require an access token and an access token secret. Similarly, to the consumer keys, these strings must also be kept private: they provide the application access to Twitter on behalf of your account. The default permissions are read-only, which is all we need in our case, but if you decide to change your permission to provide writing features in your app, you must negotiate a new access token.

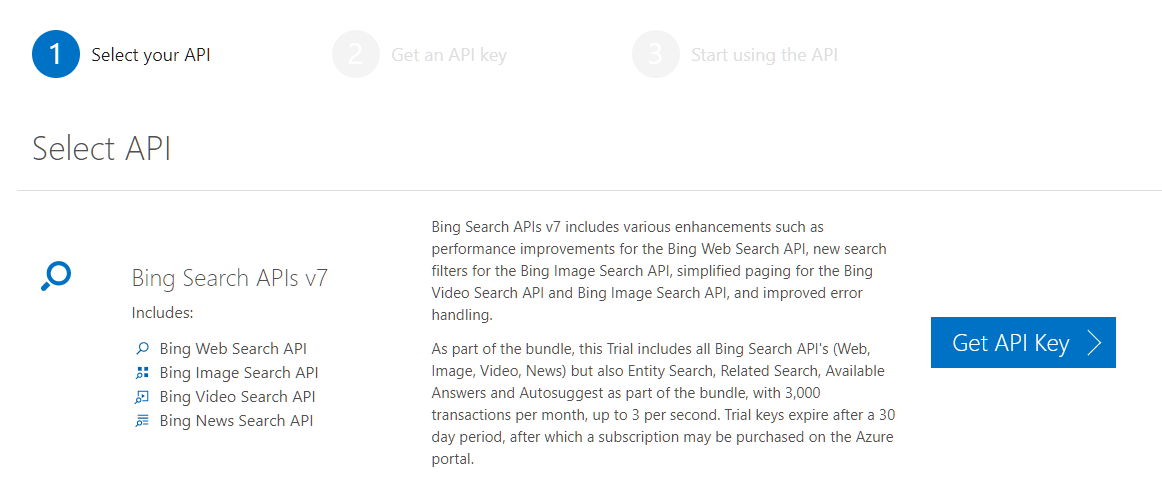
1. MonkeyLearn

Go to [https://monkeylearn.com/](https://monkeylearn.com/%20), sign up then go to my account, acquire the keys,

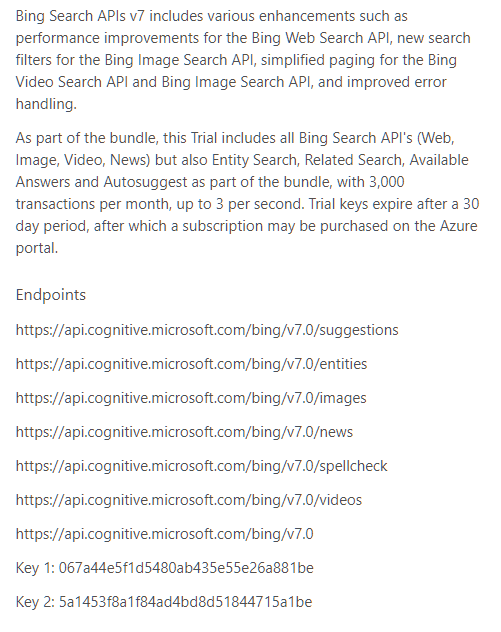


1. Bing

To use the Bing search APIs, we need to apply for a token from Microsoft. The website of application is: <https://azure.microsoft.com/en-us/try/cognitive-services/?api=bing-web-search-api> The picture shows the entry of this service:



After application, we can get the keys:



Then by accessing the endpoints with the keys, we can directly use the Bing search engine in our code.

## Change the setting variable in code

|  |  |
| --- | --- |
| Variable name | Description |
| TWITTER\_CONSUMER\_KEY | Used for Twitter consumer key |
| TWITTER\_CONSUMER\_SECRET | Used for Twitter consumer secret |
| TWITTER\_ACCESS\_TOKEN\_KEY | Used for Twitter access token key |
| TWITTER\_ACCESS\_TOKEN\_SECRET | Used for Twitter access token secret |
| TWITTER\_USER | Enter the user’s id who will be analyze |
| MONKEYLEARN\_TOKEN | Used for MonkeyLearn token |
| EXPAND\_TWEETS | This variable determines if we need to use Bing search to expand the collected tweets |
| BING\_KEY | This variable is used to access the Bing search engine |