travel route planning

2021.09.20

Design and Implementation

**1.1 Twitter key account network analysis**

**1.1.1 why we do network analysis**

They extract data from social media, blogs, wikis, and other user-generated content to provide new ideas and methods for travel route planning. This public data can be combined with other data from such as location data and social network networks, and processed using artificial intelligence (deep learning sentiment analysis model) to provide updates on what’s happening in the city, as well as a series of help to develop Descriptive and predictive analysis of the itinerary and development of travel routes.

At a macro level, this can help determine travel route demand and behavior trends, enabling them to make more informed responses and planning. For example, if the system detects an increase in negative sentiment information from a certain area of ​​the city, it may be that artificial intelligence can classify and text photos with relevant geographic tags to find the problem-it may be that the street is in disrepair, or Someone deliberately vandalized-we should choose to avoid attractions that are too negative.

On the contrary, the increase in social media positive reviews and the heated discussion of newly renovated parks can be used as a measure of the success of the project. Because of social media, natural language processing, and sentiment analysis, they can now understand public opinions and behaviors in real time.

We mainly use networkx to draw the relationship diagram, meanwhile we use Matplotlib and numpy are used for data analysis and preprocessing. They use the python built-in library for data cleaning and simple pipe connection and processing. The data download is obtained through twitter api and my own crawler program.

For hot locations, we can first conduct data exploration and social network analysis on twitter to confirm that this is a normal transmission path, rather than some targeted social network advertising (some areas or scenic spots will do this) )

**1.1.2 methodology of twitter key account network analysis**

**networkx**

**图像**

As an open source package of Python, NetworkX is convenient for users to create, operate and learn complex networks. Using networkx, you can store networks in standardized and non-standardized data formats, generate a variety of random networks and classic networks, analyze network structures, establish network models, design new network algorithms, and perform network drawing, etc. In NetworkX, there are the following 4 basic graph types:

Graph: Refers to the undirected graph, that is, the direction of the edge between two nodes is ignored.

DiGraph: Refers to a directed graph (directed Graph), which considers the directionality of the edges.

MultiGraph: Refers to multiple undirected graphs, that is, the number of edges between two nodes is more than one, and the vertex is allowed to be related to itself through the same edge.

MultiDiGraph: a directed version of the multigraph

All graph classes allow hashable objects as nodes. Hashable objects include strings, tuples, integers, etc. Any edge attributes (such as weights and labels) can be associated with edges.

The internal data structure of the graph is based on the adjacency list representation, and the dictionary data structure is implemented using python. The graph adjacency structure is implemented as a python dictionary; the external dictionary is keyed by nodes to values, these values themselves are dictionaries, and the adjacent nodes are keyed to the edge attributes associated with the edge. This "dict of dicts" structure allows quick addition, deletion and search of nodes and neighbors in large graphs. Directly access the underlying data structure through the methods in the class definition (programming interface "api"). On the other hand, all functions only use these API methods instead of directly acting on data structures to manipulate graphics-like objects. This design allows replacing the dicts-based "dicts of dicts" data structure with alternative data structures that implement the same method.

**Numpy**

NumPy is the basic package of scientific computing in Python. It is a Python library that provides multi-dimensional array objects, various derived objects (such as masked arrays and matrices), and various APIs for fast array operations, including mathematics, logic, shape operations, sorting, selection, input and output , Discrete Fourier Transform, basic linear algebra, basic statistical operations and random simulation, etc.

The core of the NumPy package is the ndarray object. It encapsulates python's native n-dimensional array of the same data type. In order to ensure its excellent performance, many operations are executed after the code is compiled locally.

There are several important differences between NumPy arrays and native Python Arrays (arrays):

NumPy arrays have a fixed size when created, which is different from Python's native array objects (which can grow dynamically). Changing the size of the ndarray will create a new array and delete the original array.

The elements in the NumPy array all need to have the same data type and therefore have the same size in memory. Exception: When Python's native array contains NumPy objects, in this case arrays with elements of different sizes are allowed.

NumPy arrays facilitate advanced math and other types of operations on large amounts of data. Generally, these operations are performed more efficiently and less code than using Python's native arrays.

More and more Python-based science and mathematics packages use NumPy arrays; although these tools usually support Python's native arrays as parameters, they will still convert the input arrays to NumPy arrays before processing, and also Usually the output is a NumPy array. In other words, in order to efficiently use today's science/mathematics Python-based tools (most of the scientific computing tools), it is not enough that you only know how to use Python's native array types-you also need to know how to use NumPy arrays.

**Python and Matplotlib**

​ Nowadays, the two most commonly used programming languages for data analysis are R and Python. Among them, R, as an old data analysis language, integrates a large number of commonly used tools in mathematical statistics and economics, and is an indispensable programming language for data analysis. In recent years, with its powerful ecology, Python has many open source data processing modules, such as Numpy, Pandas, scipy, Matplotlib, SKlearn, etc., which have become the dark horse of data analysis programming languages.

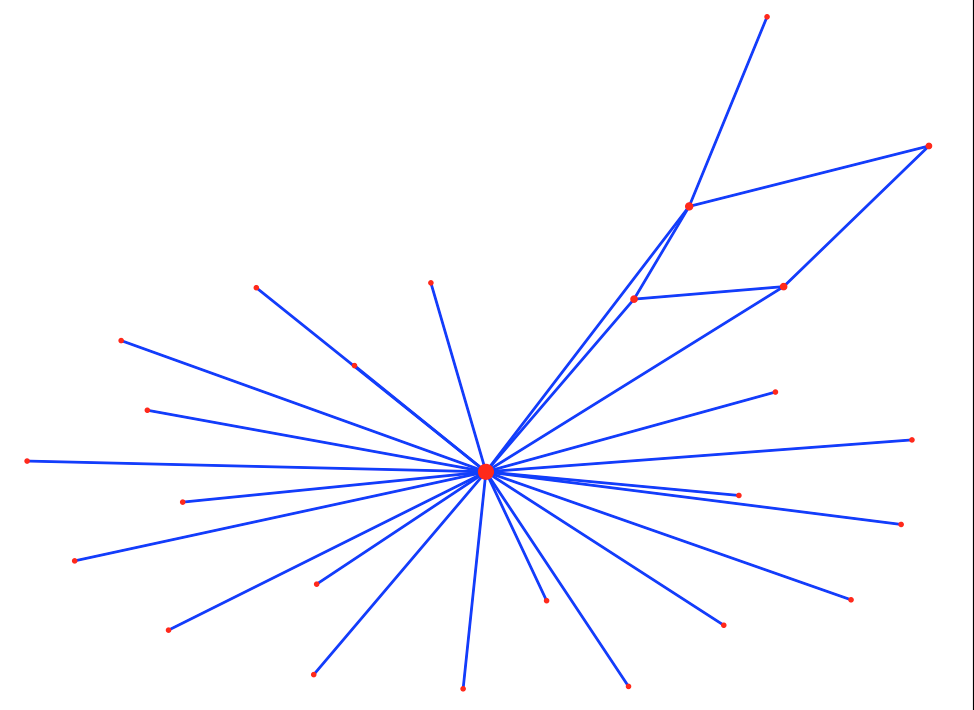
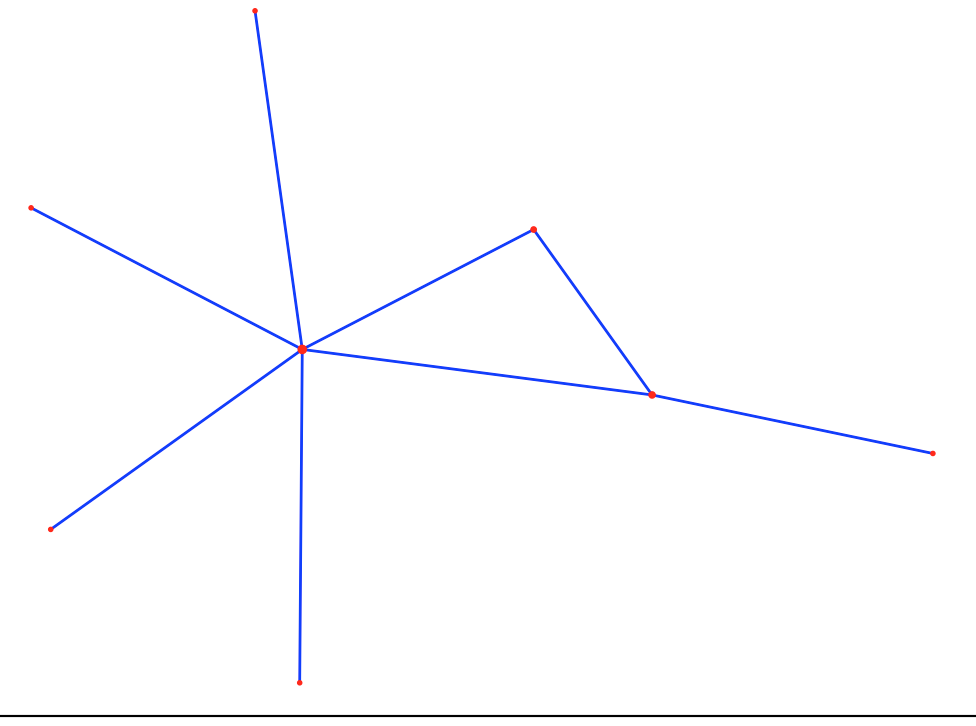
​ Speaking of using Python for data visualization, the most important module is Matplotlib. At the beginning, it was a drawing module developed in a way that imitated Matplotlib. After iteration, it can be maturely compatible with Numpy and Pandas, which is convenient and fast to use. On the other hand, more advanced modules like Seaborn are also developed on the basis of Matplotlib. Although it is more efficient, if you want to control every control of the picture more flexibly, you still need to master Matplotlib.

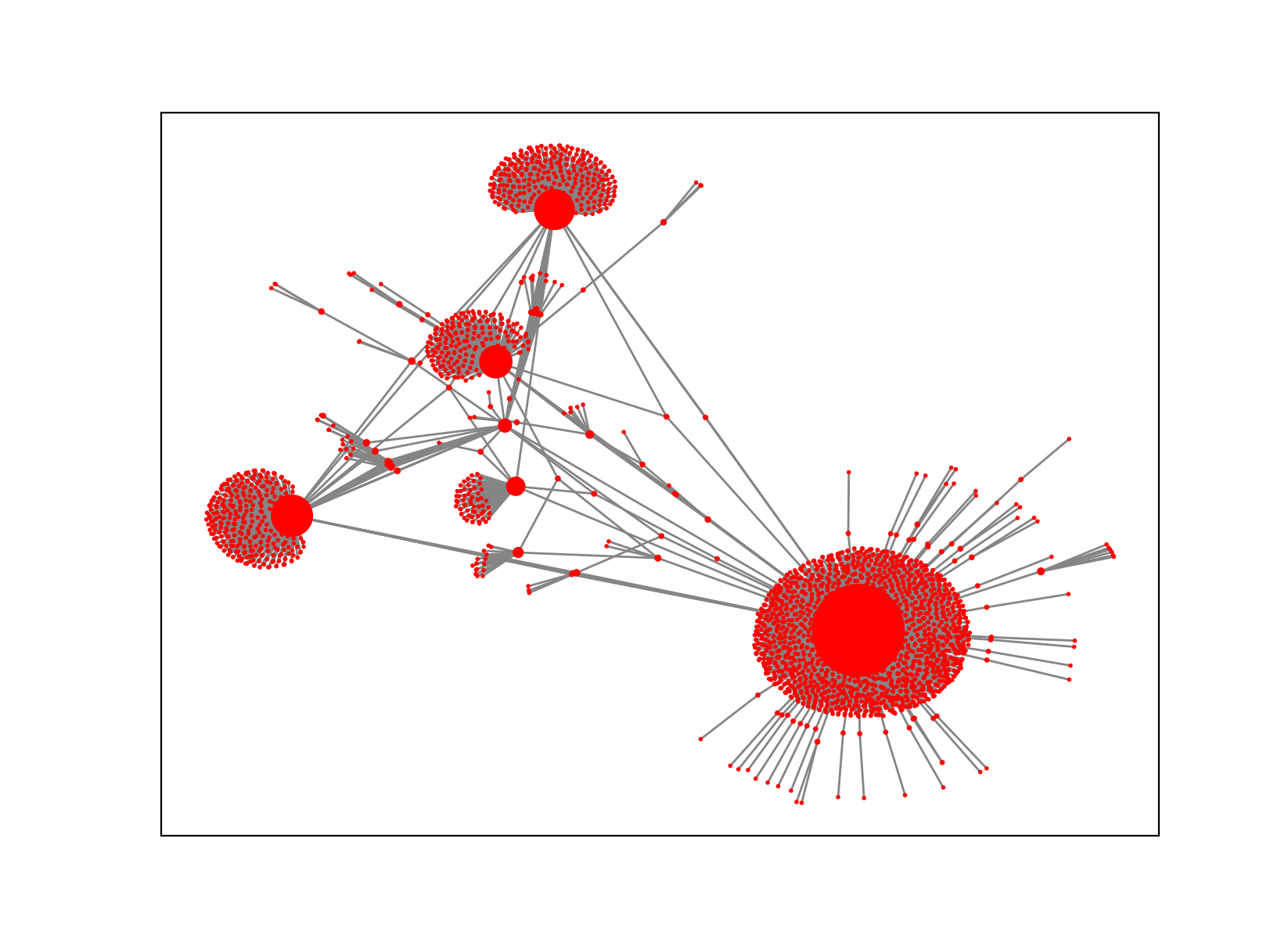
Matplotlib is a python-based plotting library that fully supports two-dimensional images and limited support for three-dimensional graphics. Matplotlib is a visual operation interface library of the python programming language and its data science extension package NumPy. It uses the adopted graphical user interface toolkit (such as Tkinter, wxPython, Qt, FLTK, Cocoatoolkits or GTK+) to provide an application programming interface (API) to the application embedded drawing. In addition, Matolotlib also has a pylab interface based on image processing libraries (such as graphics library OpenGL), and its design is very similar to MATLAB. SciPy is to use Matplotlib to draw graphics

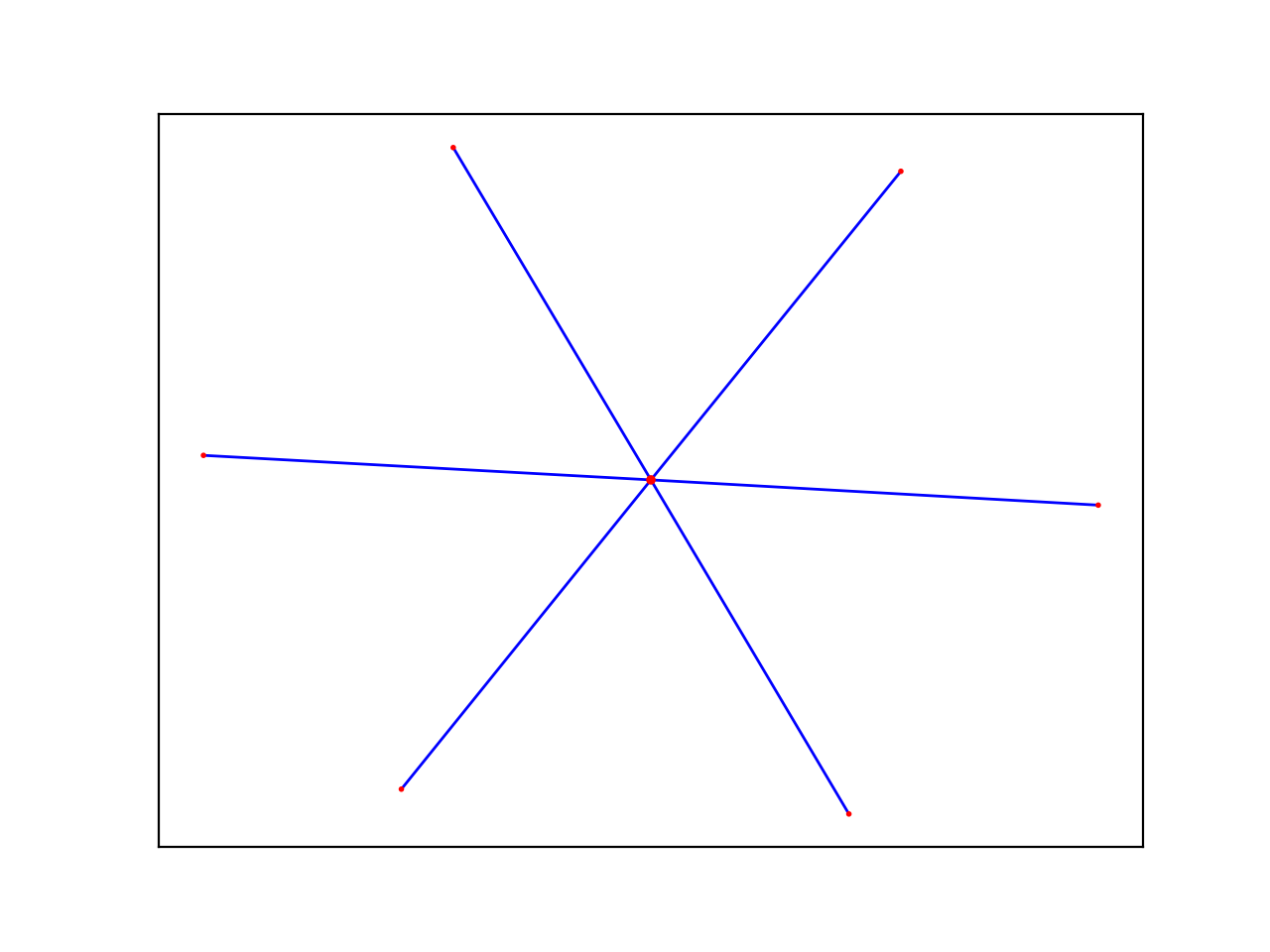
**2.0 Findings**

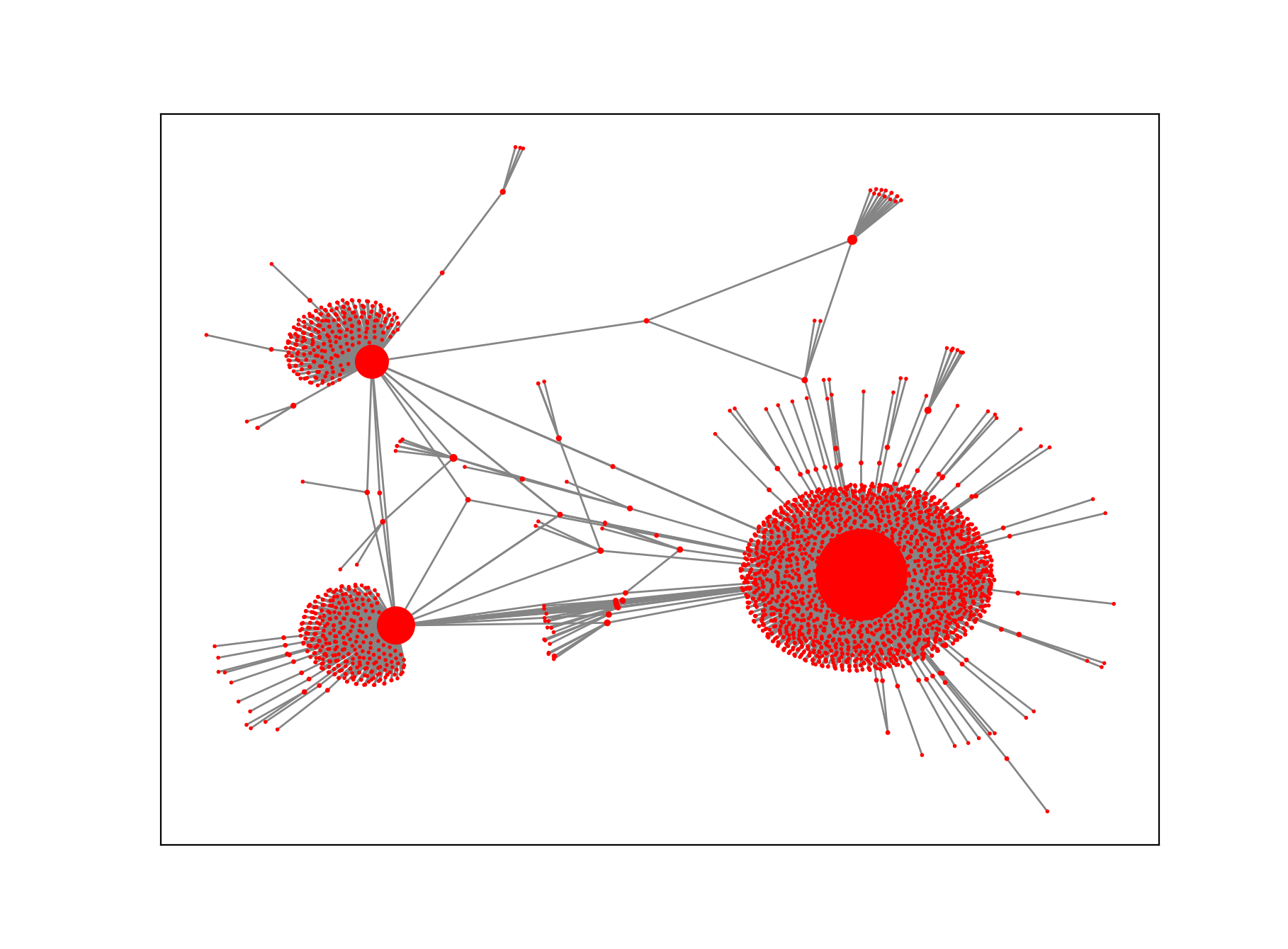
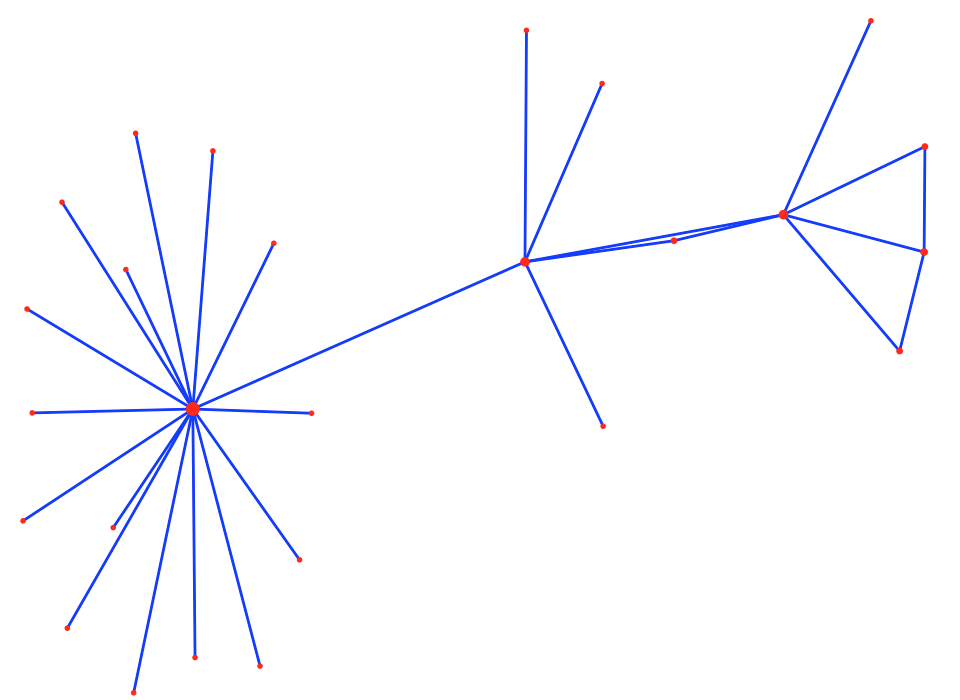
**social network analysis**

First let use see London Eye eye tweets ’s week/month/month social network analysis

****



Then let use see Leadenhall Market ’s week/month/month social network analysis

****

With the advent of the era of big data, people's awareness of accumulating data has become stronger and stronger, so how to obtain the information we want from the accumulated data has attracted more and more attention. Public comments on social networks, twitter or facebook, company e-mail records, disease development, website visits, etc. More and more data that can be connected to the network is saved, so learning to analyze these data can help us find the information we need in the messy connection relationship

We first use networkX to grab the data of Twitter to obtain the key data of 2 on Twitter. The results show that the network density is 0.0904, indicating that the actors in the network have a certain degree of relevance. There are a total of all nodes in the network. There are 3287 lines, the longest distance between nodes is 6, and that is, the longest distance between any two nodes is 6 lines. The distance reflects the length of the interval between actors, and the smaller the distance, it means that each other the closer the connection. In other words, on Twitter, the media organizations can be connected to each other through up to 5 nodes, which is less than the average distance between any two people in the world that is only 5 nodes apart from the "six degrees of separation" theory emphasizes. In addition, the average distance between the network nodes is 2.937, which means that any two media agencies only need to pass through the intermediary of 1 to 2 nodes to become the interconnected nodes in the network.

The London Eye’s network ’s more then 8000 edges, meanwhile the Leadenhall Market ’s network’s 5000 edges. The dataset we uses is about year of 2020 till now.

If we regard Social Network Analysis as an application as a research method, sometimes it can also be used as a research perspective (perspective). Of course, some middle-level theories have also been produced. The more common ones are Granovetter's weak connection theory, Burt's structural hole theory, Watts' small world model, and Barabasi's Power Law.

The previous social sciences often focused on the characteristics of individuals (or actors, such as companies, individuals), and ignored the relationships between individuals. The study of social networks is precisely the method and perspective of studying relationships. The biggest feature is that it considers the interdependence between individuals, which is closer to the real society. Displaying these relationships with the pictures shown in the title, you can intuitively see the position of each actor in the network and the overall structure of the network. Looks very dazzling and very advanced.

A network can be defined from the following three aspects:

**sentiment analysis on MVP of #Place twitter account**

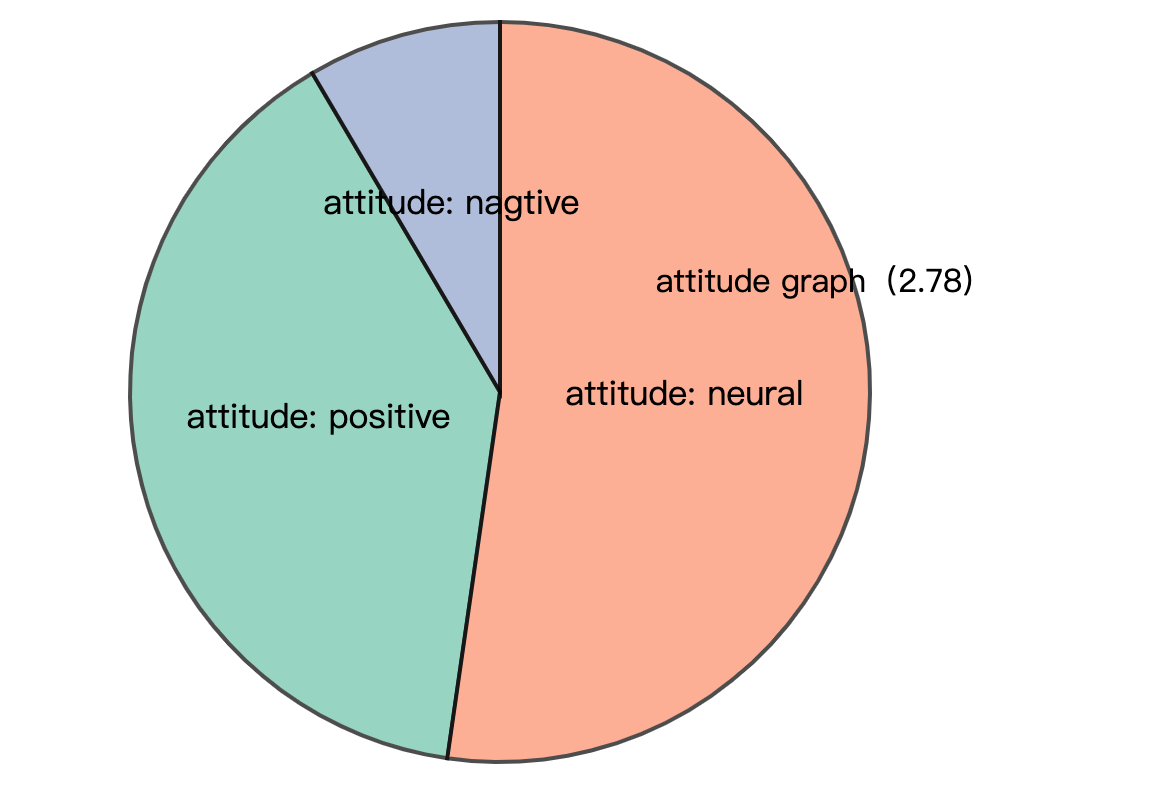
****

Figure 1 below is the sentiment analysis statistics of London Eye's sent tweets and reposts, and Figure 2 is the sentiment analysis statistics of paul's sent tweets and reposts

Figure 1

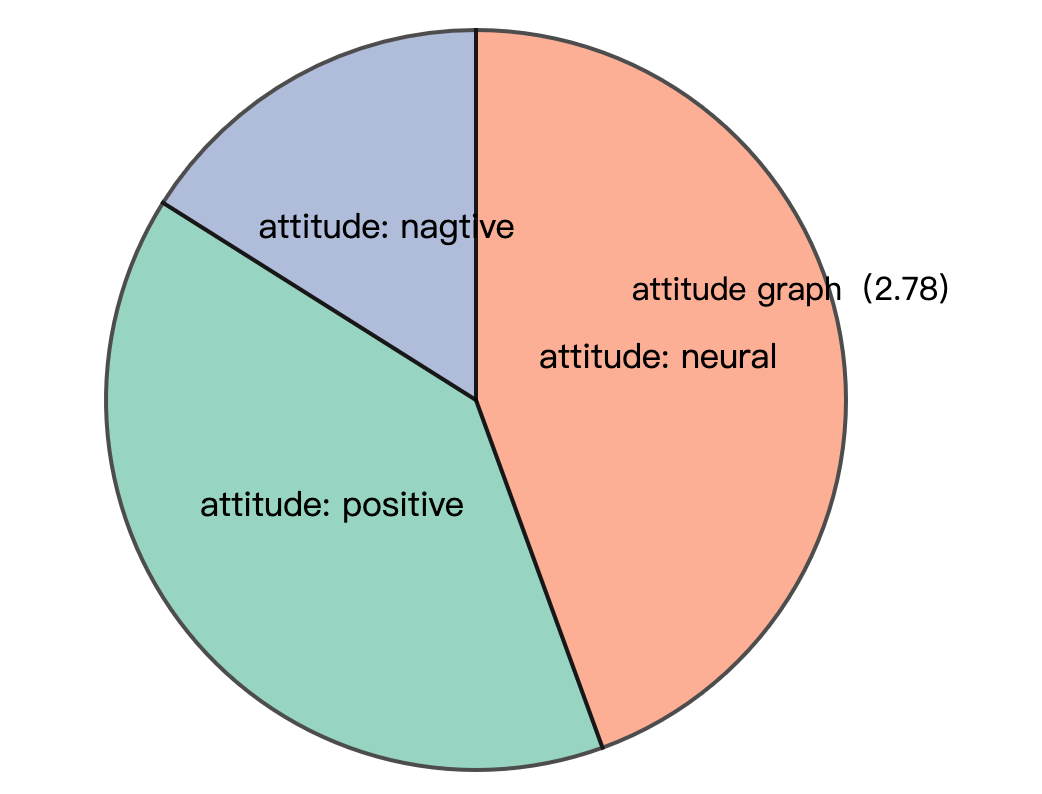


Figure 2

The explosive development of social network services such as Twitter has also brought great opportunities for researchers. Researchers can analyze the public's emotional changes by analyzing a large amount of emotional data, and have an impact on government management, economics, and entertainment. From the perspective of the government and managers, the United Nations has developed an application for global emotional fluctuation monitoring, Global Pulse, and Beihang’s research team launched the first online emotional system for Chinese Weibo, MoodLens; Romney and Obama during the 2012 US general election Intense propaganda was launched on Twitter to influence ordinary people and journalists, becoming a typical case of Internet participation in the presidential election. In terms of financial applications, many research institutions apply sentiment analysis technology to stock analysis and forecasting systems. For example, Stock Sonar displays daily positive and negative sentiment information for the stock next to the hotness of each stock, providing investors with instant information For reference, the Twitter sentiment analysis developed by UIC is used to predict and track the rise and fall of the stock market. In the entertainment field, Alibaba Cloud’s artificial intelligence system "Little Ai" successfully predicted Coco Lee's victory in the "I Am a Singer" program. It also relied on live data and comment data on social networks to analyze and predict. Text sentiment analysis technology. It can be seen that the monitoring, analysis and prediction of social media's emotional big data continue to influence government decision-making and public choice.

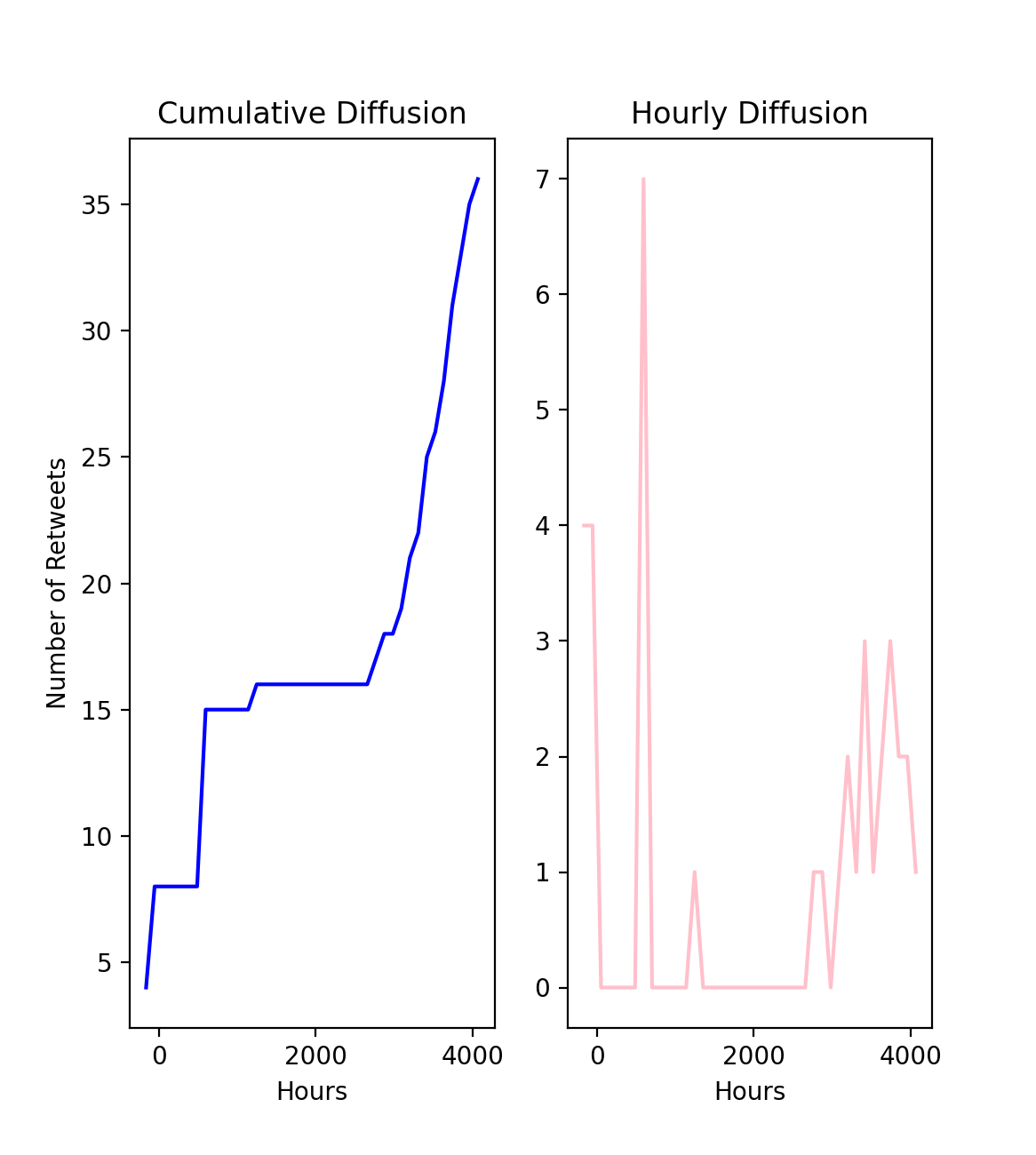
The time when the text is published can usually be obtained using simple rules. Therefore, the goal of sentiment analysis is usually to automatically analyze the three elements of Holder, Target, and Polarity from unstructured text. . Holder is the issuer of the opinion; Target is the object of the opinion evaluation (such as the entity or entity's attributes, or topic); Polarity is the emotion type expressed. Due to different tasks, the emotion category system will be different, usually including praise and criticism. , Joy, anger, sorrow, joy, sorrow, and emotional scoring (such as 1-5 points) classification system The emotions in the text are divided into explicit emotions and implicit emotions. Explicit emotions refer to emotional texts that contain obvious emotional words (such as happy and beautiful), and implicit emotions refer to emotional texts that do not contain emotional words, such as "this There is a layer of ash on the table." Since implicit sentiment analysis is more difficult and relies on background knowledge and common sense knowledge, many current works focus on the research of revealed sentiment analysis. In this report ,we are just use revealed sentiment analysis of (Polarity)

We do it at the sentence level, it determines whether its sentiment is positive, negative or neutral for each sentence of the document. Unlike the previous document level, some descriptive sentences actually have no emotion, so there is a neutral category here, which means there is no emotion. This question has something to do with subjectivity classification-subjectivity classification is to judge whether a sentence is subjective or objective. Usually neutral sentences are objective, while positive or negative sentiment sentences are subjective. But they are not exactly the same. For example, "We bought the car last month and the windshield wiper has fallen off" is an objective sentence, but it describes an undesirable thing, so it implies negative emotions. Although the sentence "I think he went home after lunch" is subjective, it has no positive or negative emotions

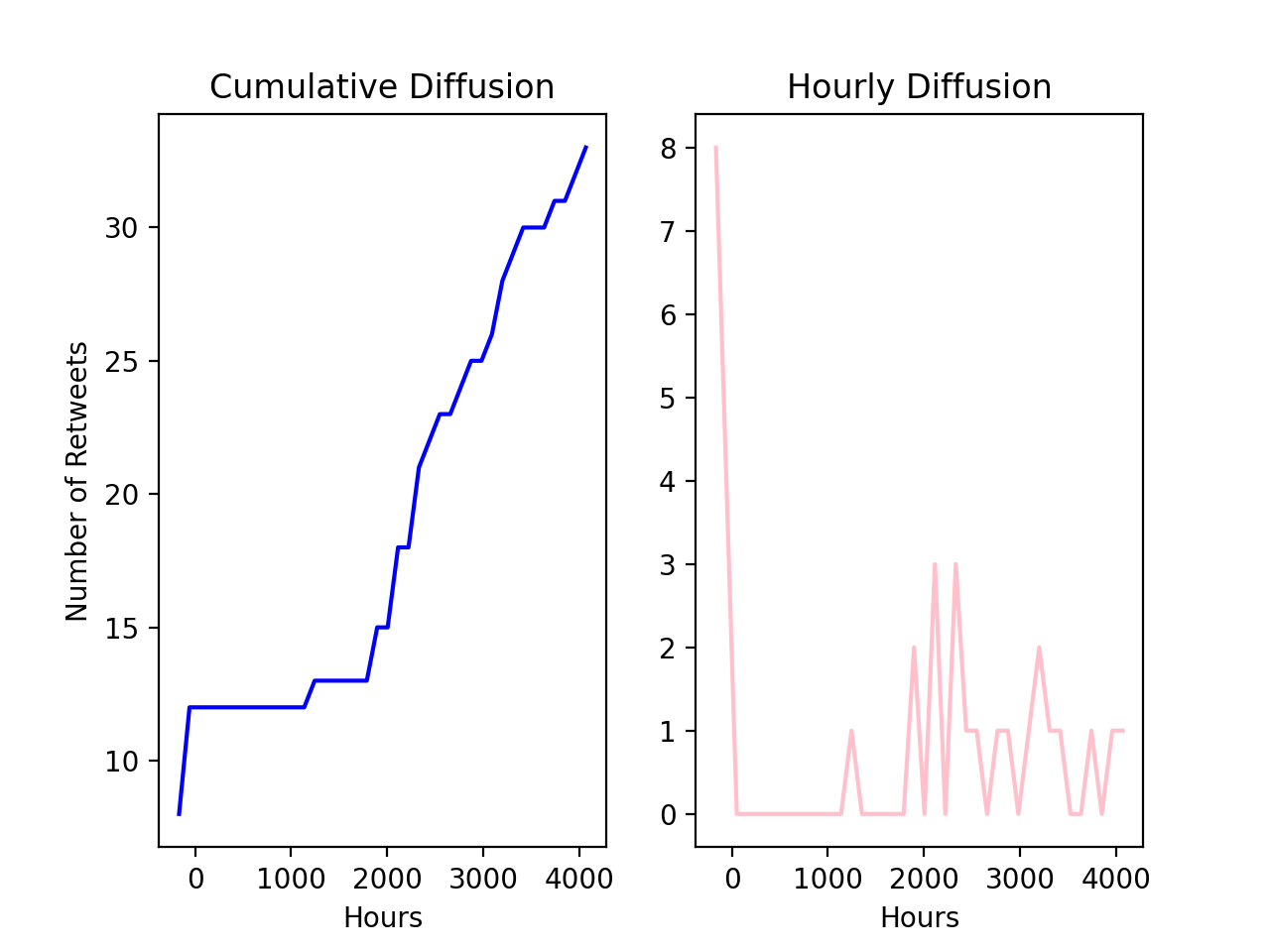
We can see :

London Eye have or retweet more postive tweets than Leadenhall Market, but in common, the are on the Internet, most topics (more than 1/2) are neutral.

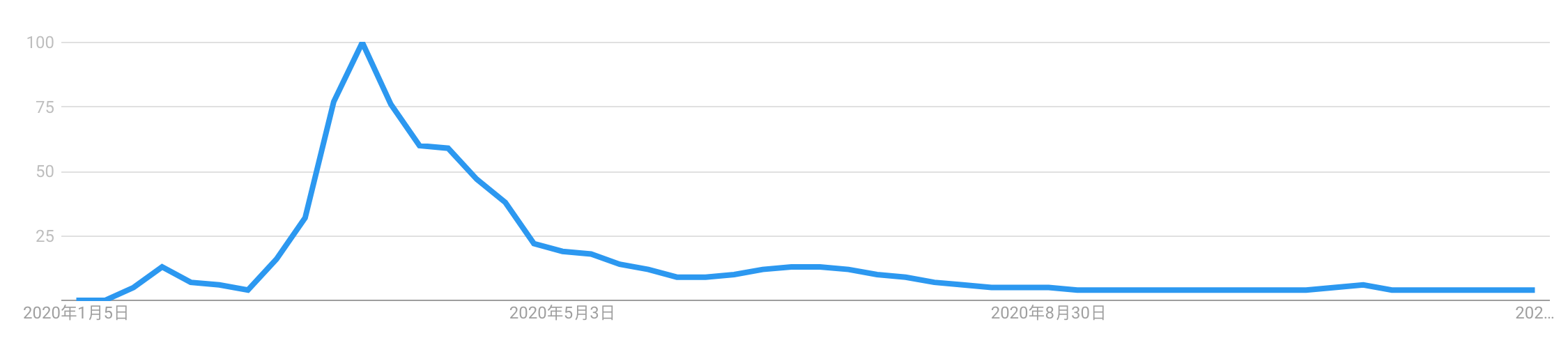
**Google search analysis**

The “Audiences" report in Google Analytics can view the details of visitors. This information can help you tailor content to attract audiences. If you use paid advertising, you can use this report to optimize your targeting .First we anlysis their reteet rules from the time and frequence. Then we count the most send / reteet topic . We then search topics on google trends to find the Most relevant.

London Eye frequencly of send



paul frequencly of send

****

London Eye’s tweet interaction rules are more random, but there are no loosening rules for a few months in the middle, and the popularity is not affected; and Paul uses tweets to interact with the audience with similar probability throughout the year.

Through the Google search correlation analysis, it can be seen that the main audience of London Eye is most concerned about the bitcoin hotness and blockchain-related news.

Pual’s audience is most relevant to some news and discussions about the new crown

Recommendation and conclusion

The 21st century is a century in which mankind is highly dependent on data and deeply integrated into the information society. In the information society, online social networks have built a huge functional platform. People express opinions and make friends and interact on social networks, and hundreds of millions of information are generated every day. Online social interaction is changing people's behavior patterns and social patterns. Online social network data is also becoming the most mature big data. Through research and Analyzing this technology, people are expected to have an unprecedented depth of understanding of user behavior and social phenomena behind online social network big data.

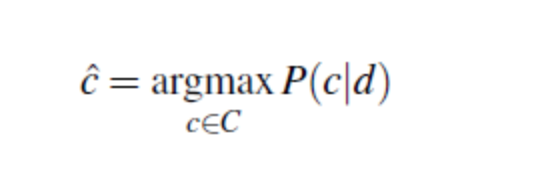
This analysis is very enlightening

Common place:

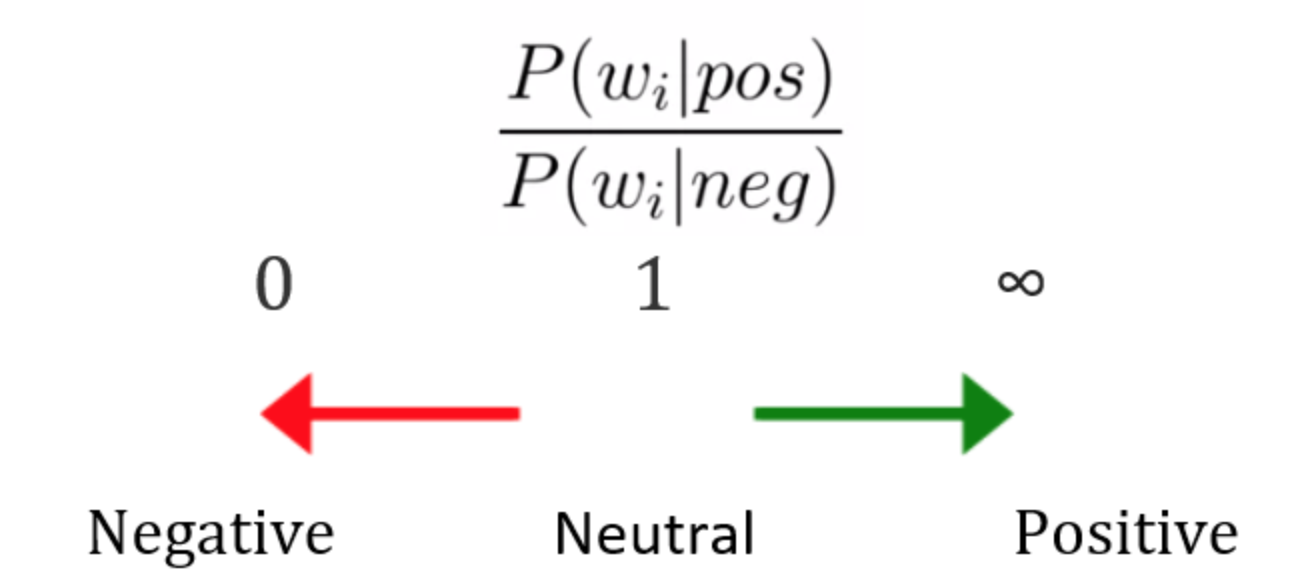
1. Stay hot and interact with your audience frequently. Of course, the most important thing is to be sincere and discuss the issues that you really care about (for London Eye is the future of mankind, journey to Mars, testla, reduce global warming), for Paul, he is most concerned about how to help the initial entrepreneurship People, build alliances for them, news about the new crown pneumonia and how to help everyone tide over the difficulties
2. Both individuals have maintained close communication with their most important partners. It can be seen from the visualization of social networks that there are basically 3 and 4 communication objects and groups, and not all topics and groups are involved. It is important to choose your own groups and interest groups
3. Be objective, sincere, and enthusiastic. Through sentiment analysis, it can be found that the two people still maintain a neutral attitude on most topics, and post meaningful opinions and reposts mainly in areas that they are familiar with.

**1.2 sentiment Model for all #Place tweets**

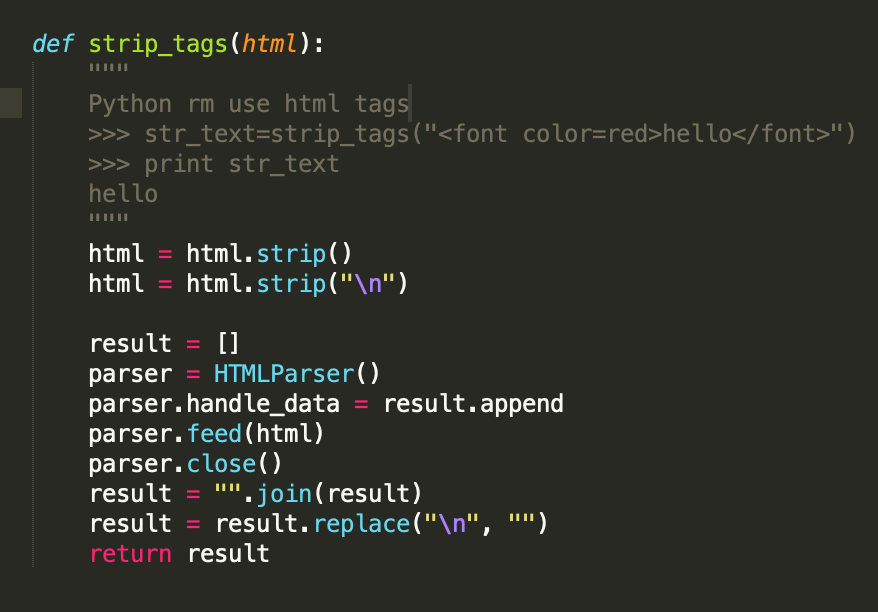
**Native Bayes model and compare**

**Naive Bayes is a probabilistic classifier, which means that for document d, among all categories [formula], the classifier returns the category [formula] with the greatest posterior probability of a given document.**

**In Naive Bayes, we will discover how each word affects emotion, which can be calculated by the ratio of the probability of occurrence of two types of words. Let us take an example. We can see that the word "negative" is more likely to appear in the negative category than in the positive category. Therefore, we will find the ratio of these probabilities for each word through the formula:**

****

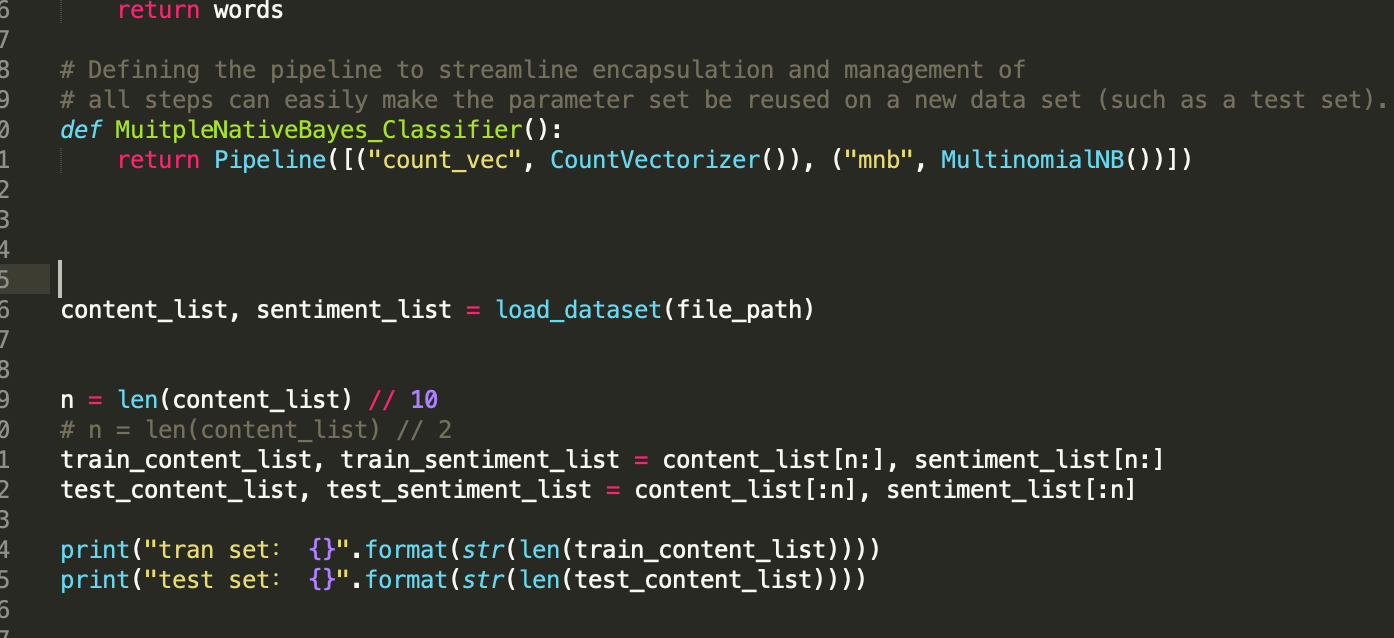
**Before we started train dataset , first we should do data pre-processing to remove useless tags:**

****

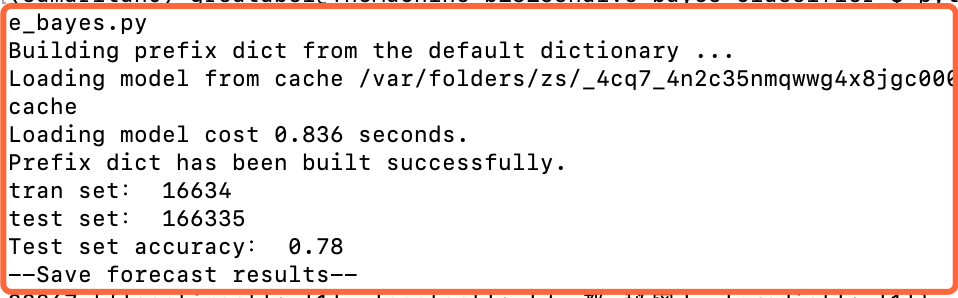
**Then we need to choose stop words used in Native-bayes model:**

****

**Then we make proper pipe line to load our model and split dataset to train/test to exmine and run many epochs:**

****

**Our ultimate goal is to deploy the trained model to the real environment. We hope that the trained model can get a good prediction effect on real data. In other words, we hope that the model predicts less error on real data. We get our model run on the test dataset , the best is 0.78, which is still lower than our deep learning model ,wchi accuracy is 0.7978, and even faster than Native Bayes model. We will introduce the Deep learning model in later.**

****

**1.2.1 What model we choose and why**

Sentiment analysis or opinion mining is a computational study of people's opinions, emotions, emotions, evaluations and attitudes about products, services, organizations, individuals, issues, events, topics and their attributes. The beginning and rapid development of this field is consistent with the development of social media, such as comments, forums, blogs, Weibo, Twitter and social networks, because this is the first time in human history that there are such a large number of digitally recorded views data. As early as 2000, sentiment analysis became one of the most active research fields in NLP. It has been extensively studied in data mining, web mining, text mining and information retrieval. In fact, because of its overall importance to business and society, it has expanded from computer science to management and sociology, such as marketing, finance, political science, communication, health science, and even history. The reason for this development is that viewpoints are at the core of almost all human activities and are important factors influencing human behavior. Our beliefs, our perception of reality, and the decisions we make depend to a large extent on the way others see and evaluate the world. Therefore, when we make decisions, we usually seek the opinions of others. Not just individuals, but also organizations.

Existing research has produced a large number of techniques that can be used for multiple tasks in sentiment analysis, including supervised and unsupervised methods. Among the supervised methods, early papers used all supervised machine learning methods (such as support vector machines, maximum entropy, naive Bayes, etc.) and feature combinations. Unsupervised methods include different methods using sentiment dictionaries, grammatical analysis, and syntactic patterns. There are many review books and papers, covering the early methods and applications extensively.

About ten years ago, deep learning became a powerful machine learning technology, which produced the current best results in many application fields, including computer vision, speech recognition, NLP, etc. The application of deep learning to sentiment analysis has also become popular recently. This article first summarizes deep learning, and then reviews sentiment analysis based on deep learning.

In this report comprehensively describes deep learning to achieve sentence-level sentiment analysis (positive, neutral, optimistic), mainly introduces the department we hope to use deep learning to predict sentiment, try to surpass the sentiment analysis performance of traditional machine learning algorithms, and analyze the problem in detail. The functions that need to be completed, and the system development process, development tools and techniques used are explained. After completing the model training, the machine learning model is engineered and turned into an AI web service, which can be used by the website. From the four angles of system requirement analysis, outline design, detailed design, and realization of functional modules, the development process of the system is explained in detail, and the main characteristics of the system are introduced.

**1.2.2 Development Tech Stack we choose**

Introduction of development technology and tools

a. tensorflow development framework

TensorFlow is an end-to-end open source machine learning platform. Provide comprehensive and flexible professional tools to enable individual developers to easily create machine learning applications, help researchers to promote the development of cutting-edge technology, and support enterprises to establish stable large-scale applications.



Since its release in 2015, TensorFlow has had 41 million downloads worldwide. As a widely used machine learning framework around the world, TensorFlow has passed its third anniversary, and TensorFlow has gradually become an end-to-end mature platform with a complete ecosystem. With the release of TensorFlow 2.0, it marks the arrival of a new era of TensorFlow, which is easier to use, more flexible and powerful, and more usable in production environments. I hope TensorFlow can become a machine learning platform suitable for all users and can help everyone use it. Machine learning solves real-world problems.

Machine learning is mainly divided into two steps: training and deployment.

In the training phase, TensorFlow not only supports Python, but also provides support for Swift and JS languages. You can choose the language you are familiar with for development.

In the deployment phase, TensorFlow models can run on different platforms, support TensorFlow Serving deployed on the server side, TensorFlow Lite deployed on end-side platforms such as Android, iOS and embedded devices, and TensorFlow.js deployed on browsers and Node servers. And many languages ​​including C language, Java language, Go language, C# language, Rust and R

B. keras

Keras is a neural network library written in pure Python, focusing on deep learning, running on TensorFlow or Theano.

TensorFlow and Theano are currently two popular deep learning libraries, but they are relatively complicated for beginners.

Keras is simple to use and has a clear structure. The underlying computing platform can be based on TensorFlow or Theano with powerful functions.

Keras can run in Python 2.7 or 1.5 environment, perfectly combined with GPU and CPU, released under the MIT license.

Keras is developed and maintained by Google engineer François Chollet.

The following are the design principles of Keras:

o Modularity: A model can be understood as an independent sequence or graph. The models are independent of each other and can be freely combined.

o Minimalism: Each module should be as concise as possible. Each piece of code should appear intuitive and easy to understand when it is first read. There is no black magic, because it will cause trouble for iteration and innovation.

o Extensibility: Adding a new module is super simple and easy. You only need to imitate the existing module to write a new class or function. The convenience of creating new modules makes Keras more suitable for advanced research work.

o Collaboration with Python: Keras does not have a separate model configuration file type (for comparison, caffe has), the model is described by Python code, making it more compact, easy to debug and easy to expand.

C. NumPy

NumPy is the basic package of scientific computing in Python. It is a Python library that provides multi-dimensional array objects, various derived objects (such as masked arrays and matrices), and various APIs for fast array operations, including mathematics, logic, shape operations, sorting, selection, input and output , Discrete Fourier Transform, basic linear algebra, basic statistical operations and random simulation, etc.

The core of the NumPy package is the ndarray object. It encapsulates python's native n-dimensional array of the same data type. In order to ensure its excellent performance, many operations are executed after the code is compiled locally.

There are several important differences between NumPy arrays and native Python Arrays (arrays):

• NumPy arrays have a fixed size when created, which is different from Python's native array objects (which can grow dynamically). Changing the size of the ndarray will create a new array and delete the original array.

• The elements in the NumPy array all need to have the same data type, so they have the same size in memory. Exception: When Python's native array contains NumPy objects, in this case arrays with elements of different sizes are allowed.

• NumPy arrays facilitate advanced math and other types of operations on large amounts of data. Generally, these operations are performed more efficiently and less code than using Python's native arrays.

• More and more Python-based science and mathematics software packages use NumPy arrays; although these tools usually support Python's native arrays as parameters, they will still convert the input arrays to NumPy arrays before processing, and It is also usually output as a NumPy array. In other words, in order to use today's science/mathematics Python-based tools (most of the scientific computing tools) efficiently, it is not enough that you only know how to use Python's native array types-you also need to know how to use NumPy arrays.

D. Jupyter Notebook

Jupyter Notebook is a web-based application for interactive computing. It can be applied to the whole process of calculation: development, document writing, running code and displaying results.

In short, Jupyter Notebook is opened in the form of a web page, and you can write and run the code directly on the web page. The running result of the code will also be directly displayed under the code block. If you need to write a description document during the programming process, you can write it directly on the same page, which is convenient for timely description and explanation.

A web application is a tool based on a web page format that combines writing instructions, mathematical formulas, interactive calculations, and other rich media forms. In short, web applications are tools that can implement various functions

Documents are all the input and output of interactive calculations, writing instructions, mathematical formulas, pictures, and other rich media forms in Jupyter Notebook, which are all embodied in the form of documents.

These documents are saved as JSON format files with the suffix .ipynb, which is not only convenient for version control, but also convenient for sharing with others. In addition, the document can also be exported to: HTML, LaTeX, PDF and other formats

Main features of Jupyter Notebook

① It has the functions of syntax highlighting, indentation and tab completion during programming.

② You can run the code directly through the browser, and display the running result under the code block at the same time.

③ Display calculation results in rich media format. Rich media formats include: HTML, LaTeX, PNG, SVG, etc.

④ Support Markdown syntax when writing documentation or sentences for the code.

⑤ Support the use of LaTeX to write mathematical descriptions.

**1.2.3 Development Tech Stack we choose**

Experiment introduction and data reading

We hope to use deep learning to predict sentiment and try to surpass the sentiment analysis performance of traditional machine learning algorithms.

We mainly explore the following aspects:

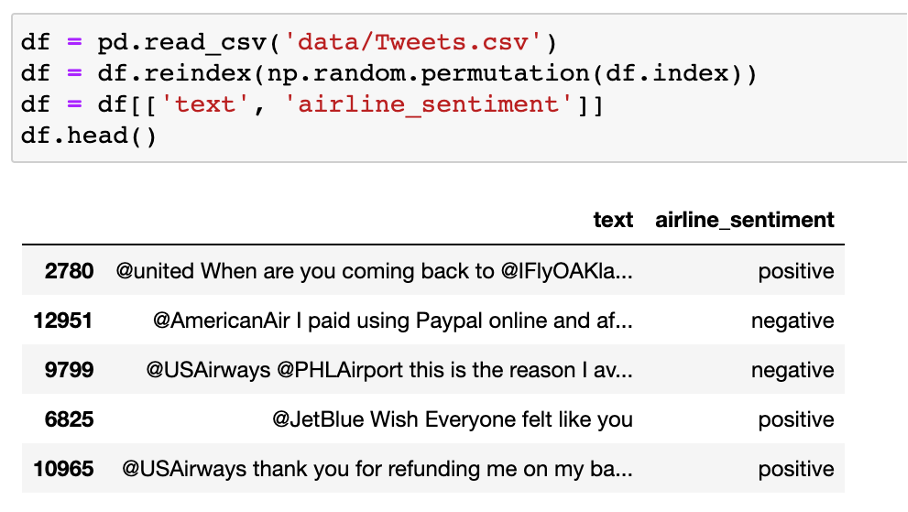
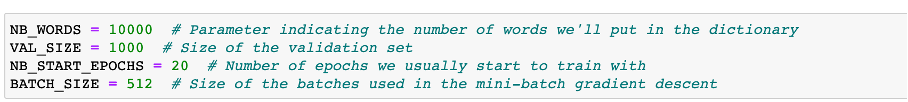
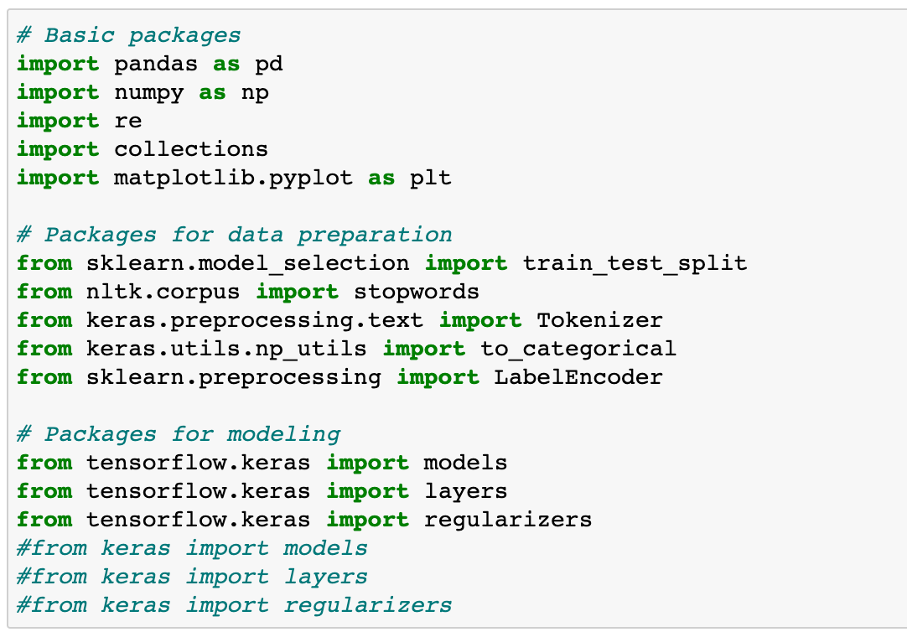
\*Use adaptive deep learning model

\*Identify and deal with overfitting

\*Use word embedding

\* Built on a pre-trained model

We use the tweet data to read the csv and perform randomization of the data. A good practice is to shuffle the data before splitting between the training and test sets. In this way, the sentiment level is evenly distributed on the training set and the test set.



We only use the text column as input, and the airen\_sentiment column as the output target.

1.2 Data preparation and data clean

Data cleaning refers to the last process of discovering and correcting identifiable errors in data files, including checking data consistency, dealing with invalid and missing values, etc. Different from the questionnaire review, the data cleaning after entry is generally done by computer rather than manually. Data cleaning-The process of re-examining and verifying data with the purpose of removing duplicate information, correcting existing errors, and providing data consistency.

Data cleaning can also tell from its name to "wash out" the "dirty", which refers to the last procedure for discovering and correcting identifiable errors in data files, including checking data consistency, dealing with invalid and missing values, and so on. Because the data in the data warehouse is a collection of data oriented to a certain subject. These data are extracted from multiple business systems and contain historical data. In this way, it is unavoidable that some data is wrong data, and some data is mutually interrelated. Conflict, these erroneous or conflicting data are obviously undesirable, called "dirty data". We have to "wash out" the "dirty data" according to certain rules. This is data cleaning. The task of data cleaning is to filter out the data that does not meet the requirements, and submit the results of the filtering to the business department to confirm whether it is filtered or amended by the business unit before extracting. The data that does not meet the requirements is mainly divided into three categories: incomplete data, wrong data, and repeated data. Data cleaning is different from questionnaire review. Data cleaning after input is generally done by computer rather than manually.

Consistency check

Consistency check is to check whether the data meets the requirements based on the reasonable value range and mutual relationship of each variable, and find out the data that is out of the normal range, logically unreasonable or contradictory. For example, if the variable measured with a scale of 1-7 has a value of 0, and the weight has a negative number, it should be regarded as out of the normal range. Computer software such as SPSS, SAS, and Excel can automatically identify each variable value out of range according to the defined value range. Logically inconsistent answers may come in many forms: For example, many respondents said that they drive to work and report that they do not have a car; or respondents report that they are heavy buyers and users of a certain brand, but at the same time they are familiar with A very low score was given on the degree scale. When inconsistencies are found, the questionnaire serial number, record serial number, variable name, error category, etc. should be listed for further verification and correction.

Handling of invalid and missing values

Due to survey, coding, and entry errors, there may be some invalid and missing values ​​in the data, which need to be dealt with appropriately. Commonly used processing methods are: estimation, whole case deletion, variable deletion and pair deletion.

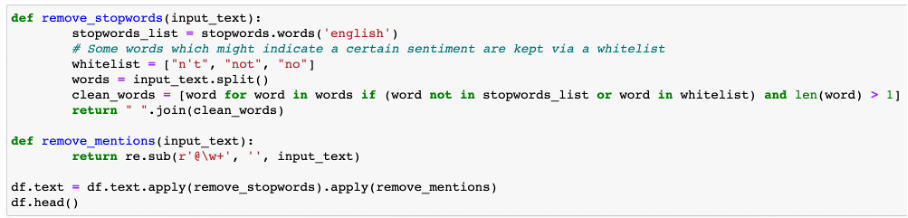
Estimation. The simplest way is to replace invalid and missing values ​​with the sample mean, median, or mode of a variable. This method is simple, but does not fully consider the existing information in the data, and the error may be large. Another way is to estimate through correlation analysis or logical inferences between variables based on the answers to other questions of the respondents. For example, the ownership of a certain product may be related to household income, and the possibility of owning this product can be estimated based on the household income of the survey respondents.

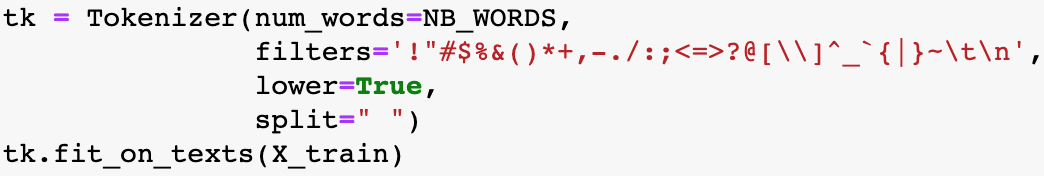
Casewise deletion is to remove samples with missing values. Since many questionnaires may have missing values, the result of this approach may greatly reduce the effective sample size and make it impossible to make full use of the collected data. Therefore, it is only suitable for cases where key variables are missing, or the proportion of samples with invalid or missing values ​​is small.

Variable deletion (variable deletion). If a variable has many invalid and missing values, and the variable is not particularly important to the problem being studied, you can consider deleting the variable. This approach reduces the number of variables for analysis, but does not change the sample size.

Pairwise deletion uses a special code (usually 9, 99, 999, etc.) to represent invalid and missing values, while retaining all variables and samples in the data set. However, in specific calculations, only samples with complete answers are used. Therefore, different analyses involve different variables and their effective sample sizes will vary. This is a conservative approach that maximizes the available information in the data set.

The use of different processing methods may affect the results of the analysis, especially when the occurrence of missing values ​​is not random and the variables are clearly correlated. Therefore, in the survey, invalid and missing values ​​should be avoided as much as possible to ensure the integrity of the data



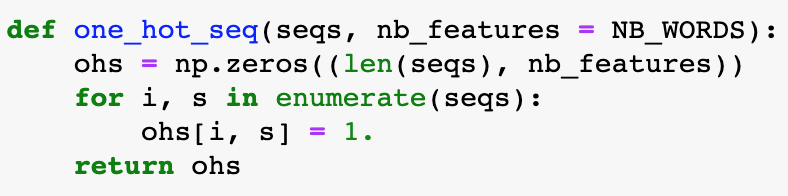
The first thing we'll do is removing stopwords. These words do not have any value for predicting the sentiment. Furthermore, as we want to build a model that can be used for other airline companies as well, we remove the mentions.

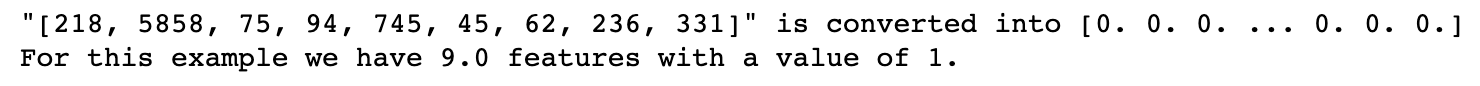
filter\_out\_flight\_related = 'late|early|soon|minutes|phone'

df = df[df[“text"].str.contains(filter\_out\_flight\_related)==False]

To use the text as input for a model, we first need to convert the tweet's words into tokens, which simply means converting the words to integers that refer to an index in a dictionary. Here we will only keep the most frequent words in the train set.

We clean up the text by applying filters and putting the words to lowercase. Words are separated by spaces





We clean up the text ，then we have done a good job of data conversion function for deep learning model training

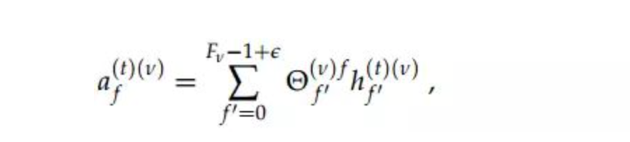
Deep learning model

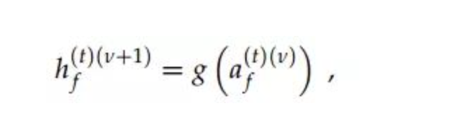
Feedforward neural network

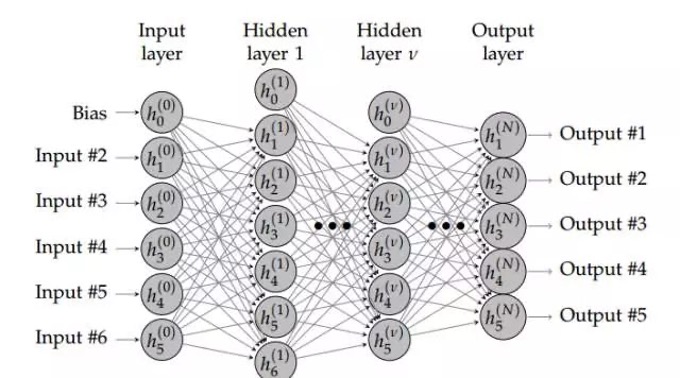
The conventional feedforward neural network (FNN) does not consider any specific structure that the input data may have. Nevertheless, it is still a very powerful machine learning tool, especially when used with advanced regularization techniques. These regularization techniques help solve the training problems that people encounter when dealing with "deep" networks: neural networks have a large number of hidden layers, which are very difficult to train (vanishing gradients and overfitting problems). A neural network with N + 1 layers (N − 1 hidden layers). The shallow network architecture uses only one hidden layer. Deep learning requires the use of multiple hidden layers, usually containing the same number of hidden neurons. The number is approximately the average of the number of input and output variables.

FNN consists of an input layer, one (shallow network) or multiple (deep network, hence called deep learning) hidden layers, and an output layer. Each layer (except the output layer) is connected to the next layer. This connection is the key to the FNN architecture and has two main features: weighted average and activation function.

The weighted average process is to multiply the excitation value of the previous layer to the neuron and the corresponding weight matrix to obtain the input value of the next neuron. This process is shown in the figure below. We can say that the weight of the previous layer of neurons The sum is the input of the next layer of neurons.



Every layer is:



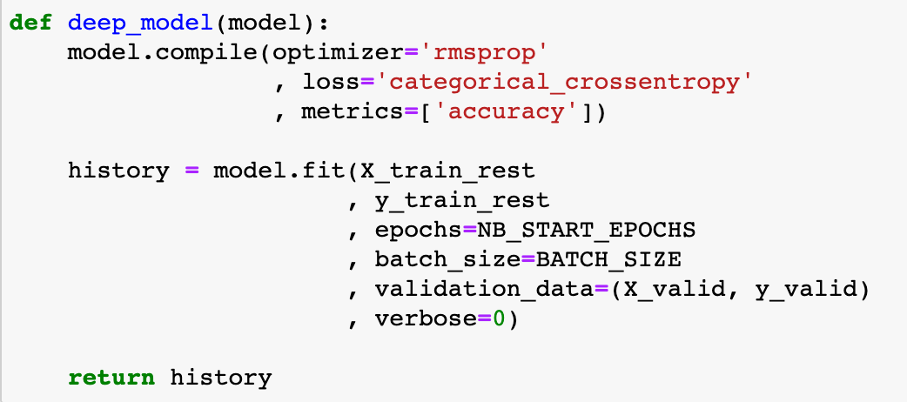
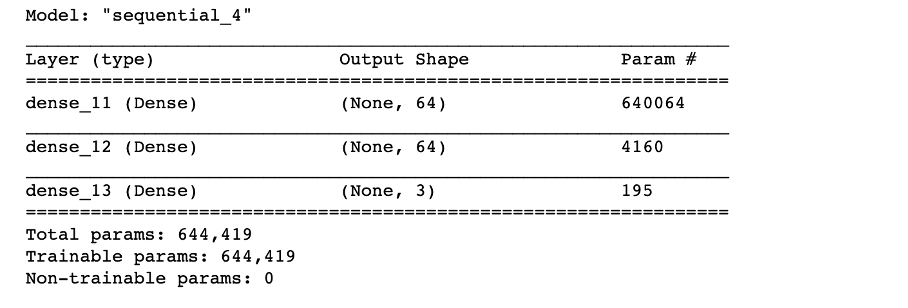


Since we need to predict 3 different emotion categories, the last layer contains 3 hidden elements. The softmax activation function ensures that the sum of the three probabilities is 1.

In the first layer, we need to estimate the 640064 weight. It depends on (nb inputs \* nb hidden elements) + nb bias terms, or (10000 x 64) + 64 = 640064

In the second layer, we estimate (64 x 64) + 64 = 4160 weight

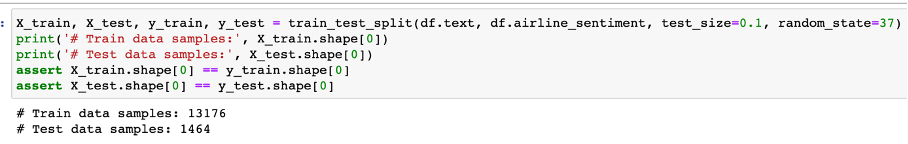
In the last layer, we estimate (64 x 3) + 3 = 195 weights



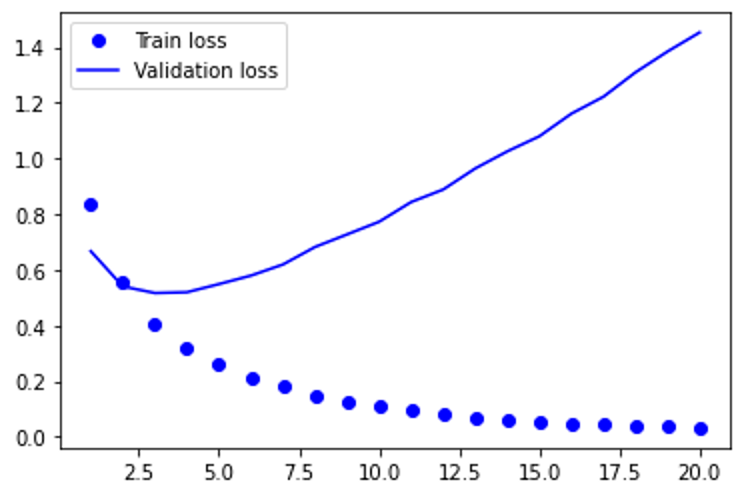
Critical Evaluation

**2.1 Training data test data set split of sentiment dataset**

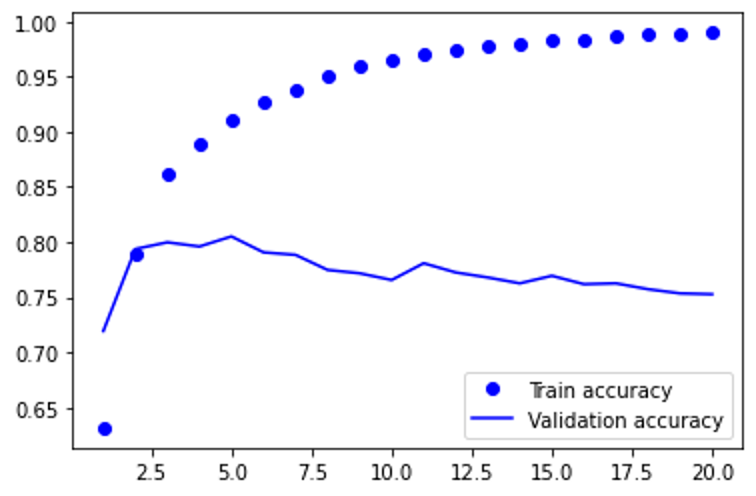
The evaluation of model performance needs to be carried out in a separate test set. In this way, we can estimate the degree of generalization of the model. This is done through the train\_test\_split method of scikit-learn.

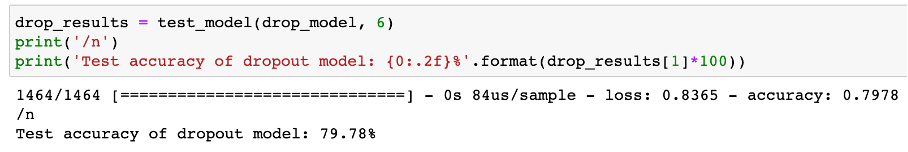


The verification loss has increased since the 4th generation. The training loss continues to decrease, which is normal because the model is trained to fit the training data as much as possible



Just like verification loss, verification accuracy will peak at an earlier period. After that, it will drop slightly. To sum up, we can say that the model has been overfitting since epoch 2.



Results in the test set:

**2.2 Training data test data set split of original hotness dataset**

First transform the data, turn the day into a number, and then visualize the hotness data from nasdaq after the conversion to find the law. First transform the data, turn the day into a number, and then visualize the effective data after the conversion to find the law.

We have to choose an algorithm to predict future visits, which is obviously supervised learning.

Before building our first model, we need to design an evaluation function to judge what kind of model is good. That is, the error function, which can be calculated like this,

def error(f, x,y):

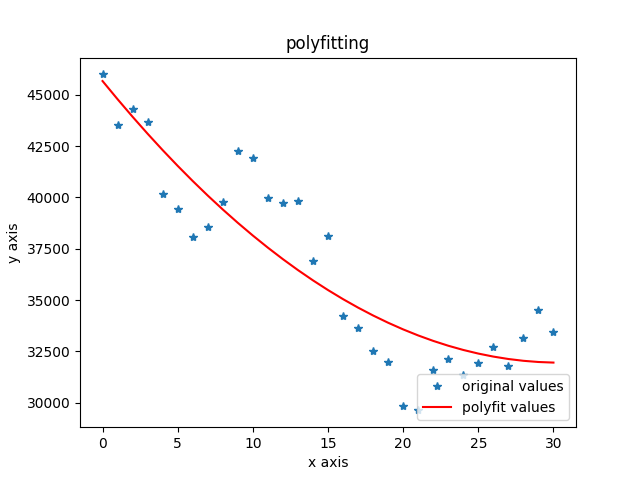
return sp.sum( (f(x) - y)\*\*2 )

and evaluated by the square of the difference between the predicted value of the model and the true value (the training sample has been provided)

Linear regression

**Linear regression**：It is easy to know that this is actually a fitting problem. Fit these data to the best model (ie a function, and then use this function to predict new data). Starting from the simplest case, we first go to a straight line to fit the data. SciPy provides the function polyfit(), as long as the data x and y and the order of the polynomial are given (a straight line is a function of 1st order), it can find the function of the model so that the previously defined error function is minimized (only the smallest error is the surface The best model). Type in: fp1,residuals,rank,sv,rcond=sp.polyfit(x,y,1,full=True)

**Polynomial regression**：A straight line is a first-order function, obviously not optimized enough, and then we start to consider the second-order curve, the third-order

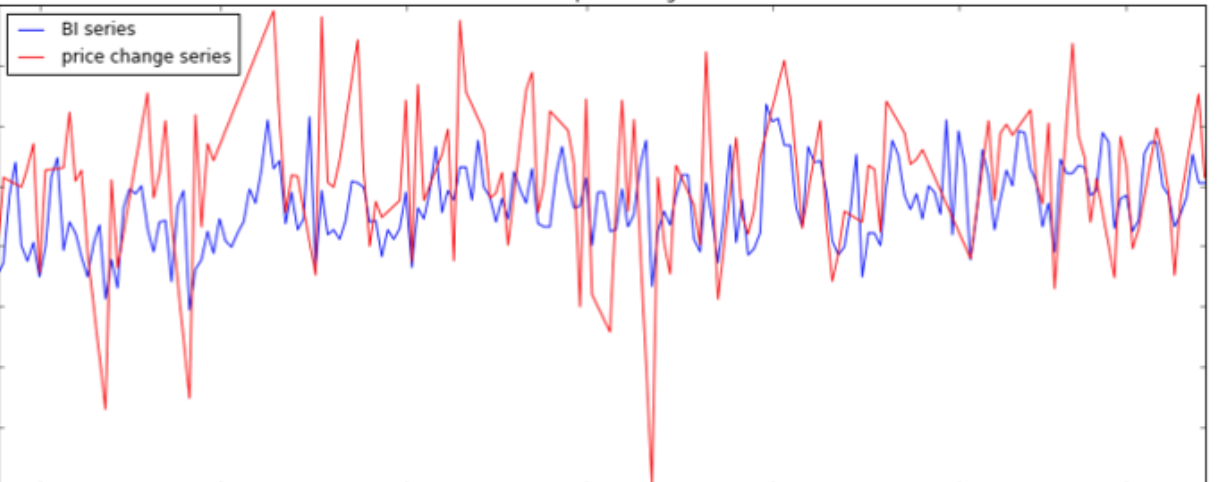
****

**2.3 Training data test data set split of original hotness dataset with consideration of sentiment of tweets of whole twitter and MVP account of #Place**

Building a time series forecasting model

Index selection: The first is the choice of indicators. For sentiment indicators, this article chooses a bullish index. Calculate the time series data of sentiment indicators in units of days. As shown in the following formula

We first do Correlation analysis of hotness and our indicators we choose.



The ARIMA model consists of 3 parts, namely autoregressive (AR), traction (I), and moving \* average (MA). They are:

ARIMA(p, d, q) consists of three parts:

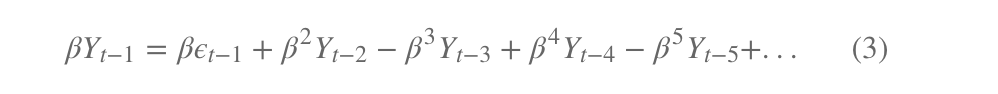
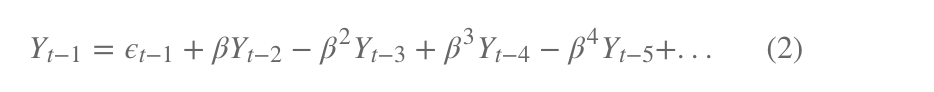
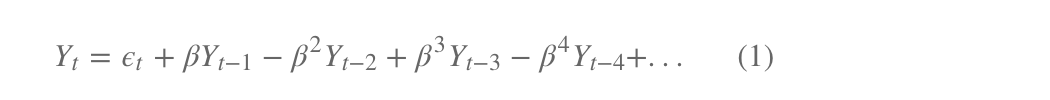
AR(p): AR is the abbreviation of autoregressive, which means autoregressive model, meaning that the value of the current time point is equal to the regression of the value of several time points in the past-because it does not depend on other explanatory variables, only depends on the past The historical value is called autoregressive; if it relies on the last p historical values ​​in the past, the order is called p, and it is recorded as the AR(p) model.

I(d): I is the abbreviation of integrated, which means that the model differentiates the time series; because time series analysis requires stationarity, unstationary series need to be transformed into stationary series by certain means, and the generally used means is difference; d Represents the order of difference, the value at time t minus the value at time t-1, the new time series is called the first-order difference sequence; the first-order difference sequence of the first-order difference sequence is called the second-order difference sequence, and so on ; In addition, there is a special difference is the seasonal difference S, that is, some time series reflect a certain period T, let the value at time t minus the value at time tT to get the seasonal difference series.

MA(q): MA is the abbreviation of moving average, which means moving average model, meaning that the value of the current time point is equal to the regression of the forecast error at several time points in the past; forecast error = model predicted value-true value; if the sequence depends on the past The most recent q historical prediction error values, called the order q, are recorded as the MA(q) model.

I(d) is well understood, and the non-stationary sequence difference is obtained as a stationary sequence, and the table is omitted. Assume that our current time series is already stationary.

I(d) is well understood, and the non-stationary sequence difference is obtained as a stationary sequence, and the table is omitted. Assume that our current time series is already stationary.

The AR(p) model is well understood. Generally speaking, time series variables have time series correlation. For example, the speed of a road, when the time interval is small enough, if the speed at the previous point in time is slow, the next point in time is often very slow. This inherent correlation allows us to predict the value of the next few time points based on the observations of the most recent time points.

Predicted value of Y = a constant and/or a weighted sum of one or more recent values of Y and/or a weighted sum of one or more recent values of the errors. If the predictors consist only of lagged values of Y, it is a pure autoregressive (“self-regressed”) model, which is just a special case of a regression model and which could be fitted with standard regression software. For example, a first-order autoregressive (“AR(1)”) model for Y is a simple regression model in which the independent variable is just Y lagged by one period (LAG(Y,1) in or Y\_LAG1 in RegressIt). If some of the predictors are lags of the errors, an ARIMA model it is NOT a linear regression model, because there is no way to specify “last period’s error” as an independent variable: the errors must be computed on a period-to-period basis when the model is fitted to the data.

AR stands for Auto Regression,

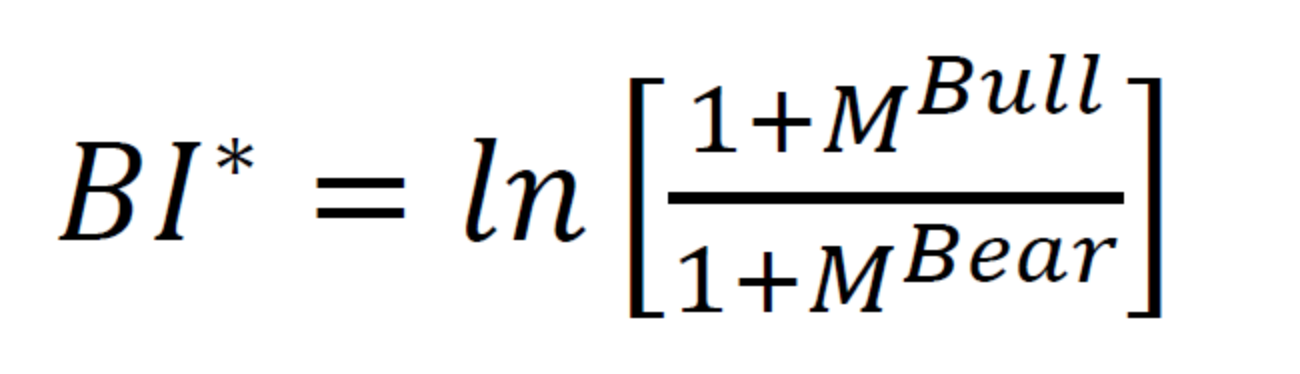
I mean a single integral order (Integration), the time series must be \*stable in order to establish a stable model. After being transformed into a stable sequence, the following orders are single integers;

MA stands for Moving Average

In the ARIMA model, denote ARIMA(p, d, q), where p is the mean number of autoregressive terms; q is the number of steps to make \*, and d is the number of steps to make \* stable ordinal. The key step, the data predicted by the IMA model must be stable \* stable), stable, it is impossible to capture AR mode data

**9-12 replace ################**

1. **Index selection**

**The first is the choice of indicators. As shown in the following formula:**

**my\_index =ln( frac{m+M^{up}}/{m+M^{down}})**

**Where m we use math.e\*base\*100 , the base is empirical value we get from trial.**

**Get the daily unit hotness and #place hotness as time series data.**

**Where**

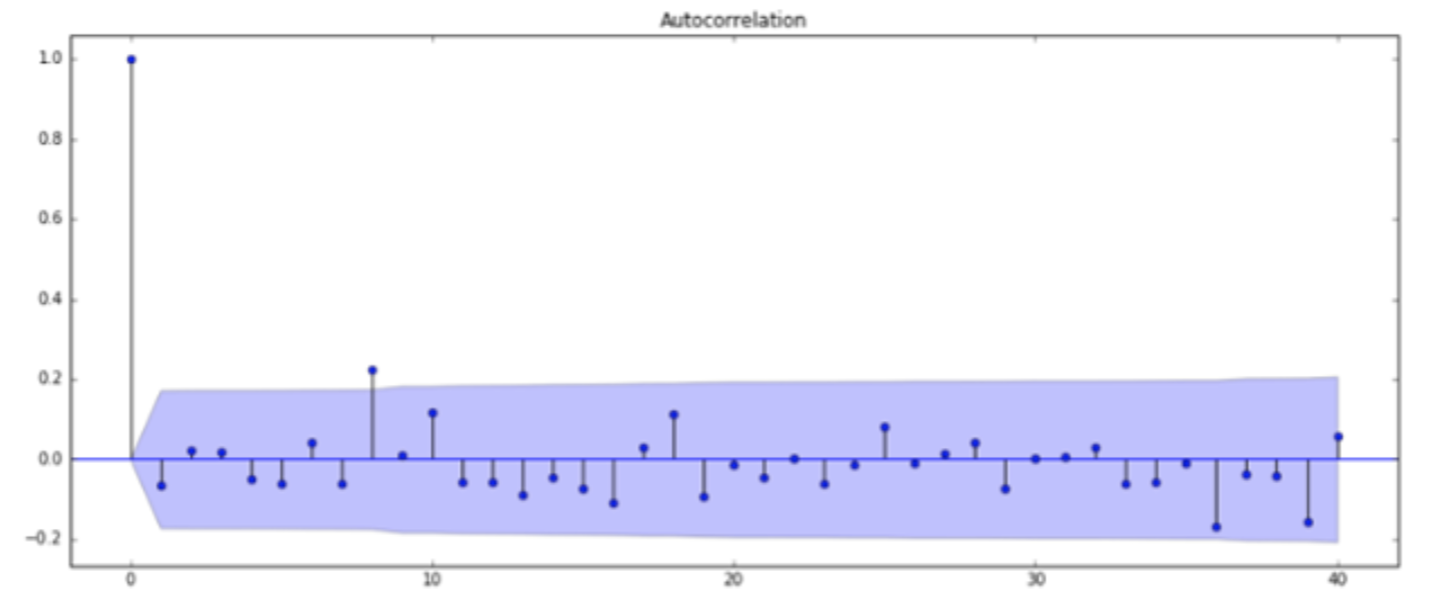
**up = the weight we assign \* the amount of optimism analyzed by the twitter tag grab data set of the day + the weight we assign a key person and a small circle of key people \* the corresponding amount of optimism**

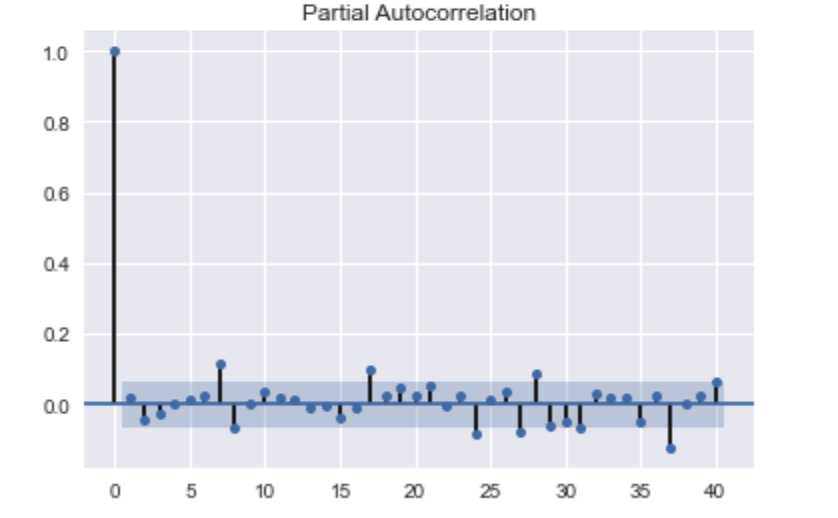
**down = the weight we assign \* the number of pessimistic sentiments analyzed by the Twitter tag grabbing data set of the day + the weight we assigned to a key person and a small circle of key people \* the corresponding number of pessimistic sentiments**

**2)parameters selection**

**ARMA(p,q) has two parameters, p and q. According to the tailing feature of the autocorrelation graph, q=1, and p=2 according to the tailing feature of the partial autocorrelation graph. Further use the AIC criterion for parameter selection. The smaller the AIC, the better the model. Using the AIC criterion also obtains q=1, p=2**

**The chart below provides a brief guide on how to read the autocorrelation and partial autocorrelation graphs to select the proper terms. The big issue as with all models is that you don’t want to overfit your model to the data by using too many terms.**

****

****

**3**）**why is improved hotness predict forecast**

**Exogenous variables refer to variables that are affected by external factors in the economic mechanism and can affect internal variables. The stock review sentiment index can be regarded as a comprehensive reflection of many exogenous variables, reflecting the macroeconomic, #place-related open source organizations or blockchain community fundamental information, policies, major events and many other external factors of #place hotness changes. We are not sure what the specific factors are, but we believe it is reflected in the hotness index. Therefore, the addition of exogenous variable model considerations improves the accuracy of hotness forecasts.**

**4) predict hotness use Arima to hotness, help adjust curve fitting**

**Fit Auto Arima: Fit the model on the univariate series**

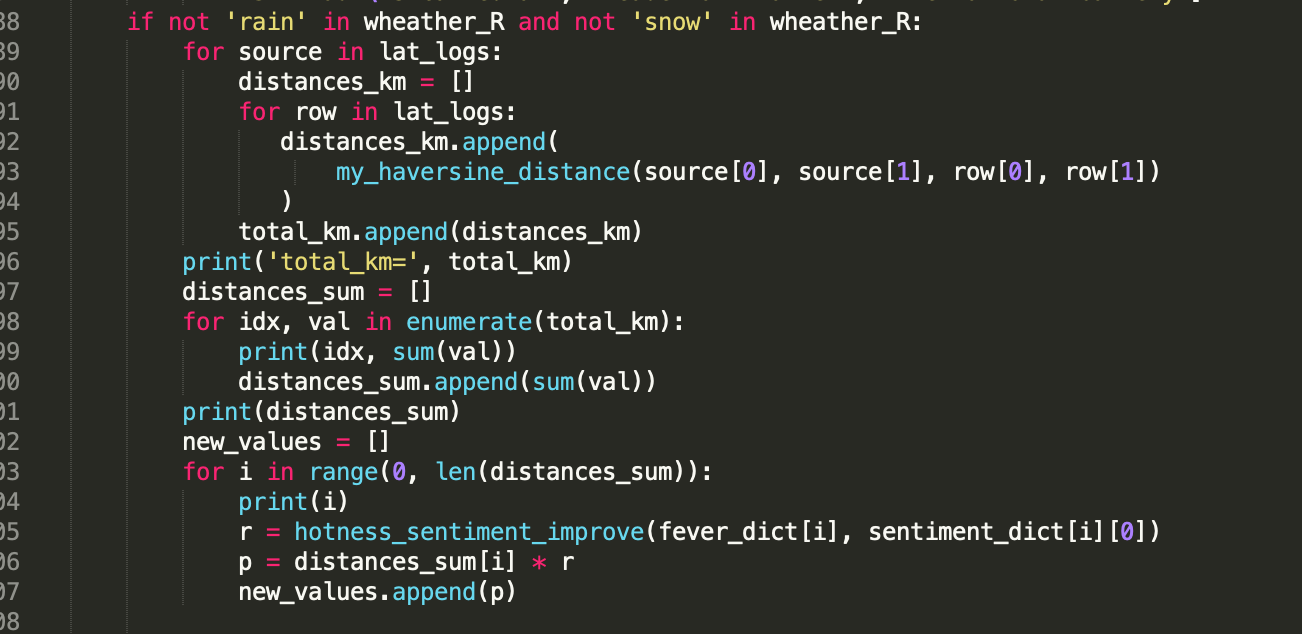
**Adjust curve fitting of simple use the new Y( adjust by our Arima model)**

**Predict values on validation set: Make predictions on the validation set**

For the **#Place** indicator, select the closing hotness and the fluctuation range for research. Get the closing hotness and the time series data of the rise and fall in units of days

we have also achieved a pure distance planning, without considering the weather factor, social network popularity and emotional memory, and the planning is not very simple. This kind of planning is impossible to consider recent regional related events (for example, roads have been dug or large-scale events are held in the park), and there is no way to plan ahead.

As compare , we use an advance way to plan routines : we use external model（Arima like), and then integrate the latitude and longitude distances, and at the same time integrate the popularity of social networks, the polarity and quantity of positive emotions into a multi-dimensional matrix for consideration and calculations to generate a comprehensive travel plan

****

Conclusion

About the sentiment model:

As we have seen above, the model with the dropout layer performs best on the test data. However, this is the same as the result achieved with LogisticRegression and Countvectorizer in our previous kernel. But there, the transformation of the input data is different from here.

Deep learning machine learning models are used in sentiment analysis because they provide more information than text. Deep learning models map input to some feature spaces, and different forms of input from multimodal data can also be projected into some joint latent spaces or representations by these models. Therefore, the trend of using deep learning models to process multi-modal data continues to grow.

About hotness predict:

First, build a deep learning model for sentiment analysis classification to realize rapid judgment of the sentiment tendency of tourist keyword (place/good or bad for tourist) comments.

Second, comprehensive monitoring of keyword dynamic information on tweets, as well as keyword-related tweets information

Third, the network propagation structure of big v is analyzed, and the main nodes of big v interaction are also considered in the final prediction algorithm.

(For the sentiment index/hot index/big v index, we designed the bullish/bearish index ourselves.

Travel behavior is an individualized choice, and the travel psychology is subjective, and the travel distribution model must reflect the travel characteristics and scale of travelers, not only qualitatively but also quantitatively. Psychology cannot be quantified, but the factors that affect psychology can be quantified. Therefore, identifying the core factors that affect travel psychology and quantifying the relationship between it and travel behavior has become a breakthrough in quantifying travel psychology. I think we are on the right way.