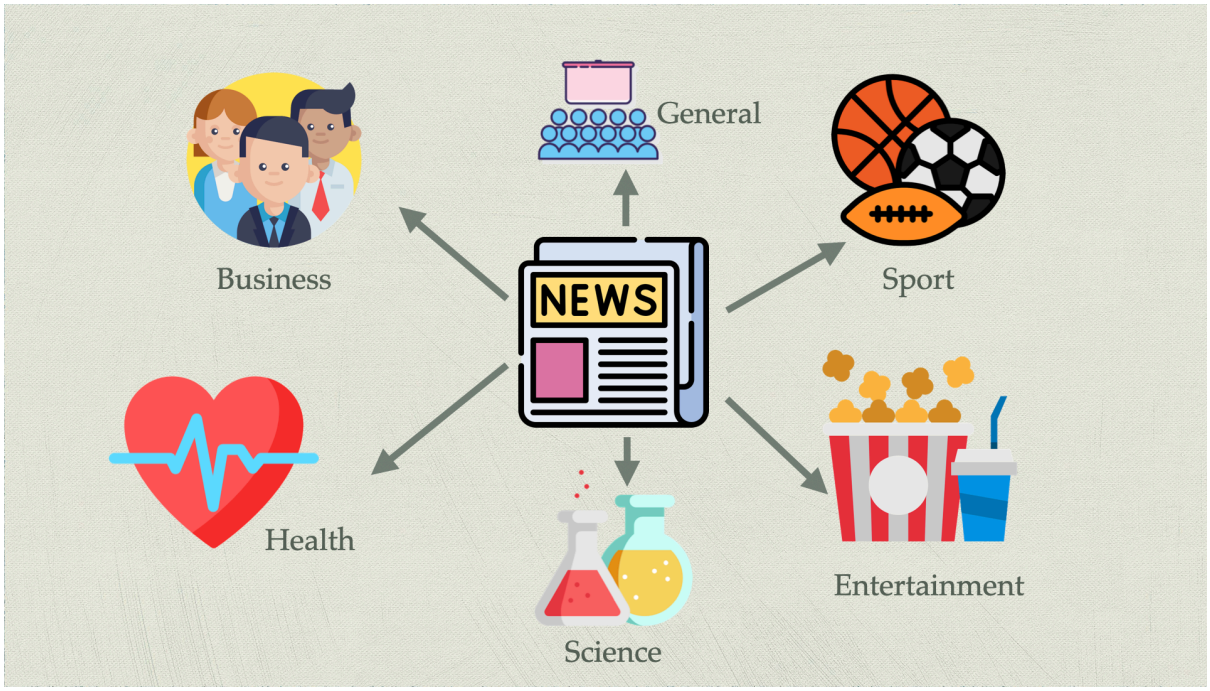
News Classification

Yu-Chieh Wang 04/01/2021

It is a system which loads daily news articles in 6 categories automatically.



Data Mining

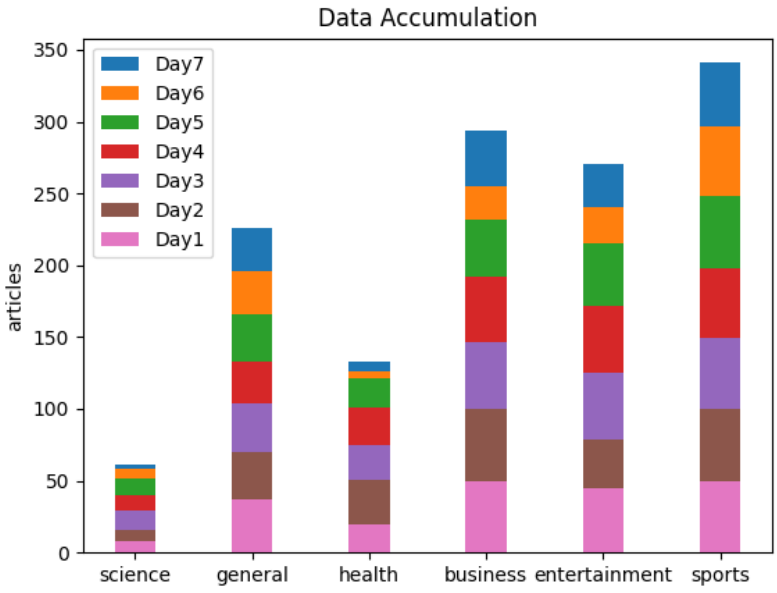
Through the news API, it is able to load real-world data by JSON format.

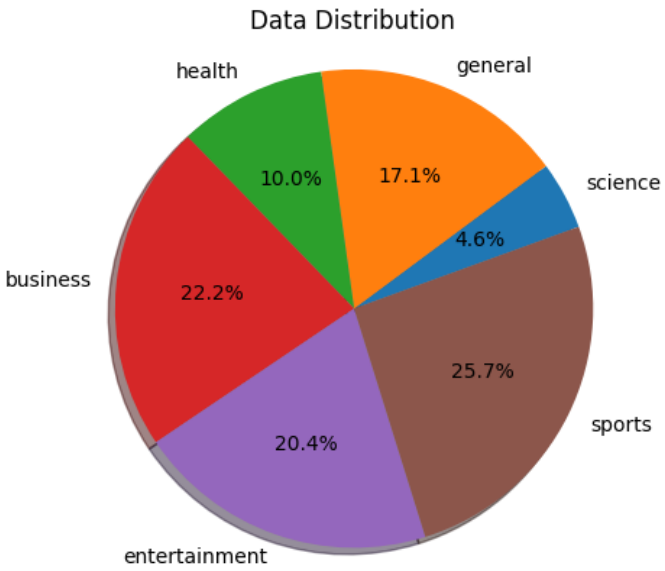
```
source : {'id': None, 'name': 'CNBC'}
author : Saheli Roy Choudhury
title : Papua New Guinea's coronavirus cases spike, health system 'at risk of collapsing' - CNBC
description : The situation in PNG is said to be dire and international organizations have warned of an imminent collapse of the
url : https://www.cnbc.com/2021/04/01/papua-new-guinea-coronavirus-cases-spike-health-system-on-the-brink.html
urlToImage : https://image.cnbcfm.com/api/v1/image/106861969-1617180959399-gettyimages-1231885263-AFP_96K4B7.jpeg?v=1617180707
publishedAt : 2021-04-01T01:15:00Z
content : Australian officials carry boxes containing some 8,000 initial doses of the AstraZeneca vaccine following their arrival
```

The system aggregates the data obtained daily into a json file to facilitate subsequent calls.

```
Loading.. 2021-03-31 13:44:34
Get news articles from newsapi.NewsApiClient in 6 categories
Articles: {'science': 13, 'general': 34, 'health': 24, 'business': 46, 'entertainment': 46, 'sports': 49}
Finish loading -> machine learning is available now.
```

After 7-day loading, the system accumulates about 1300 articles in 6 categories.





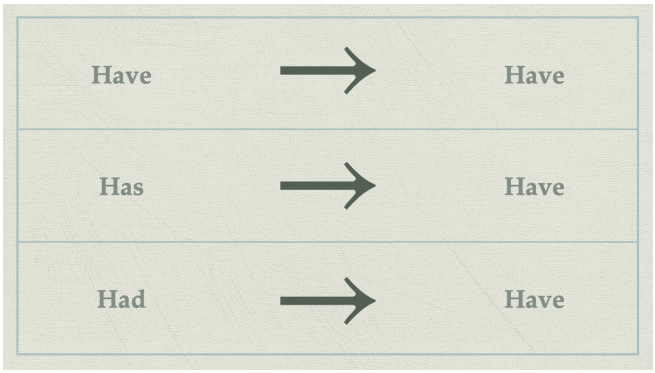
Data Preprocessing

Considering that the classification system uses keywords in articles to distinguish, before starting machine learning, the words must be processed well.

First, extract the words in the article one by one, and then integrate the words with the same meaning into one. Therefore, here uses NLTK library to help to lemma words, which means that it checks out the type of each word, and then restore the word to a simple form.

For example,

```
def get_wordnet_pos(tag):
    if tag.startswith('J'):
        return wordnet.ADJ
    elif tag.startswith('V'):
        return wordnet.VERB
    elif tag.startswith('N'):
        return wordnet.NOUN
    elif tag.startswith('R'):
        return wordnet.ADV
    else:
        return None
```



Before v.s. After

By that time, health had been damaged.
By that time, health have be damage.

Next, after restoring words, the system counts the number of times each word appears in all articles, and records the most frequently occurring words into a dictionary for future machine learning.

No.	...	6	7	8	9	10	11	12	...
Word	...	Time	Health	Vegetable	Meat	Sport	Football	Superstar	...
Frequency	...	6	3	2	2	5	2	4	...

Finally, the text in the article is filtered by stop words and vectorized word by word. If it can correspond to the word in the dictionary, convert the word to its number in the dictionary, otherwise use 0 to fill the gap.

```
tfidf_vect = TfidfVectorizer(sublinear_tf=True, norm='l2', ngram_range=(1, 2), stop_words='english')
```

For example,

Vegetable	Meat	Make	Body	Become	Health
8	9	0	0	0	7

Data Modeling

In this project, 3 supervised learning models are considered:

Naive Bayes, SVM, and Logistic Regression.

It is worth noting that the Naive Bayes model is able to consider unigrams and bigrams at the same time, which means that, in the dictionary, the classification system groups keywords one by one and two by two, and then consider these two situations together.

For example,

Unigrams	Bigrams
He	He Have
Have	Have Eat
Eat	Eat Salad
Salad	Salad For
For	For Dinner
Dinner	Dinner Tonight
Tonight	

This special method may help improve the accuracy of the overall model.

Data Training & Testing

After seven days of operation, the system successfully loads daily data and trains and tests the accuracy of the model every day.

The following pictures show how it works.

Collecting Data

```
Loading.. 2021-03-29 13:44:21
Get news articles from newsapi.NewsApiClient in 6 categories
Articles: {'science': 8, 'general': 37, 'health': 20, 'business': 50, 'entertainment': 45, 'sports': 50}
Finish loading -> machine learning is available now.
```

Training & Testing

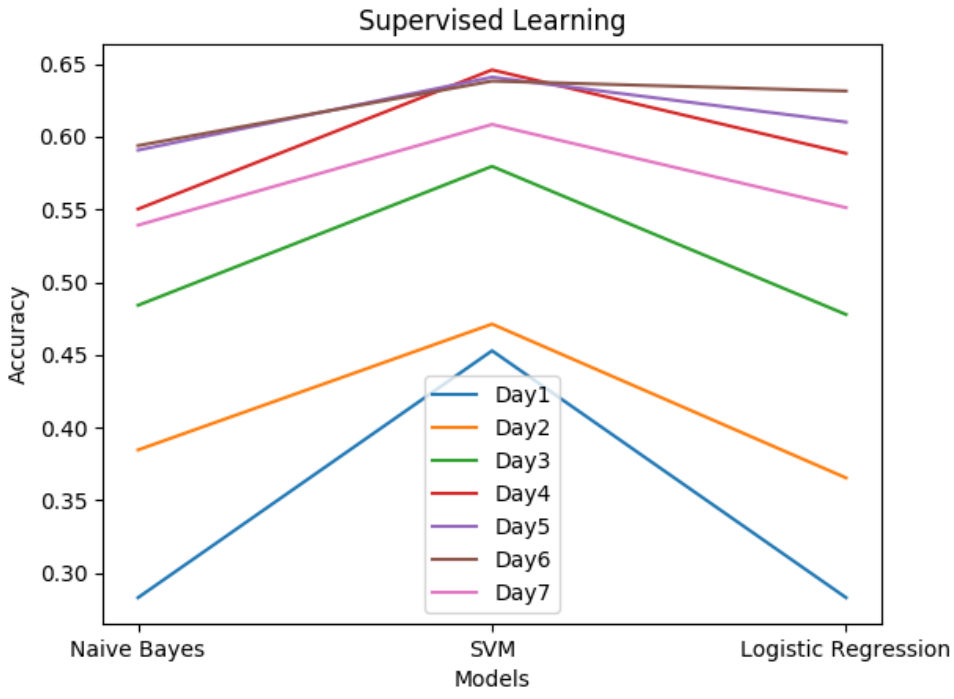
```
Articles: {'science': 8, 'general': 37, 'health': 20, 'business': 50, 'entertainment': 45, 'sports': 50}
Get lemma data successfully! training size: 157 testing size: 53
Final accuracy: 0.2830188679245283
```

Since the amount of data in each catalog is very different, a random selection function is used to separate training and test data respectively before doing machine learning.

Here, it gets 75% of data as training data and 25% as testing data.

```python
# make training data and testing data
from sklearn.model_selection import train_test_split
trainX, testX, trainY, testY = train_test_split(df[:]['content'], df[:]['label'], test_size=0.25, random_state=1000)
```

Finally, the result of 7-day accuracy is plotted as below.



Evaluation

Through the 7-day experiment, about 1300 articles can be obtained, and the distribution of 6 categories can be seen from the plot ahead.

According to the results, the SVM model performs best when the amount of data is small, and its accuracy is 0.60. Followed by logistic regression, whcih is 0.55, and finally is Naive Bayes, which is 0.53.

It can be found from the confusion matrix that among the six categories, sports articles have the highest recognizability, and its accuracy is as high as 96.6%. Followed by business and entertainment news, their discrimination is as high as 68.6% and 62.7%.

The remaining three categories are due to insufficient data, it is difficult to distinguish the content of the article leading to a decrease in overall accuracy.