Keyphrase Detection and Text Summarization using TextRank Algorithm

Yisu.Tian (yxt172830)

Abstract

Humans are in a society of high-speed information transmission. People get full of all kinds of messy information in their daily life. From the news reading in the morning to the academic paper after class, we have to quickly obtain information in a large number of messy text information. In order to efficiently extract keywords and text summarization from the article, I used TextRank Algorithm to build a keyphrase detection machine. My machine could automatically extract key phrases and document summarization from any input articles.

1. Introduction

Graph-based ranking algorithms like Google's PageRank have been successfully used in citation analysis, social networks, and the analysis of the link-structure of the World Wide Web. PageRank works by counting the number and quality of links to a page to determine a rough estimate of how important the website is. The underlying assumption is that more important websites are likely to receive more links from other websites. [1] The TextRank algorithm is based on PageRank algorithm.

The TextRank algorithm constructs a word network (a graph contains vertices and edges) in order to find relevant keywords. This network is constructed by looking which words follow one another. If one word is followed by another, a link will set up between them. The link will also get a higher weight if these two words occur more frequently next to each other in the text.

For the testing corpus, I use online news. The news on the Internet is full of messy information. They are good samples of testing my machine.

2. Previous Work

There are many approaches to extract summarization. One of them is Frequency Based Approach.

Using frequency of occurrence in order to find which words are descriptive of the topic of the document; words that occur often in the document are likely to be the main topic of the document. The assumption is that, the important words in document will be repeated many times compared to the other words in the document.

Word probability is a technique that use frequency as a basic form of measure in text summarization[2]:

In word probability technique, we denotes probability f(w) of a word w as:

$$f(w) = \frac{n(w)}{N}$$

n(w) is the number of times word W shown in the text

N is the total number of words in the document

For the weight of sentence Sj, we use the function below:

$$weight(Sj) = \frac{\sum_{w \in Sj} f(w)}{|\{w|w \in Sj\}|}$$

We can use the sentence weight to extract the key sentences in the article.

3. Method

In my machine, I use steps below to extract keywords and identify most relevant sentences from articles.

3.1

I extract the input articles, and get a list of sentences including a word list in each sentence. Then I use regular expression to remove all non-alphanumeric words.

3.2

To keep only the potential keywords, stop words need to be removed in the list. Then do POS tagging to all the words. I only keep distinct nouns and adjectives words.

Before creating a graph, I present each of these nouns and adjectives as one vertex in the graph. Formally, we call the each vertex: a lexical unit.

3.3

Any relation that can be defined between two lexical units is a potentially useful connection (edge) that can be added between two such vertices. I use a co-occurrence relation, controlled by the distance between word occurrences: two vertices are connected if their corresponding lexical units co-occur within a window of maximum N words, where N can be set anywhere from 2 to 10 words. [3]To simplify this algorithm, I set N equals to 2. So every adjacent word in each sentence has an edge. If same adjacent pair appear more than once. The weight of the edge will be increased. Note that the graph does not have to be fully connected, as not all label pairs can be related by dependency.

3.4

Next step is to construct a weighted graph using vertices and edges formed in previous two steps. Let $G=（V,E）$ be a directed graph with the set of vertices *V* and set of edges *E*, where *E* is a subset of *V x V*. For a given vertex *Vi*, let *In(Vi)* be the set of vertices that points to it (predecessors), and let *Out(Vi)* be the set of vertices that vertex *Vi* points to (successors). The score of a vertex *Vi* is defined as follows[3]:

$$S（Vi）=(1-d)+d*\sum_{j\in In(Vi)}\frac{1}{|Out(Vj)|}S(Vj)$$

3.5

Consider a random walker who begins at a word (a vertex of the graph) and executes a random walk on the graph as follows. At each time step, the walker proceeds from the current word A to a randomly chosen word that A links to (has edge between them). Because the weight between A and all vertices connected to A are different. The probability of which node next is also different.

Iterate until the algorithm convergence. After running the algorithm, a score is associated with each vertex, which represents the importance of the vertex(word) within the graph.

3.6

Sort vertices by their final score. Set top N words as the keywords in this text.

3.7

If there are adjacent words in the keywords, set them to key phrases.

3.8

For summarization of input article, we use the similar method to extract key sentences in the text by replacing the vertices to sentences in the graph. The overlap of two sentences can be determined simply as the number of common tokens between the lexical representations of the two sentences, or it can be run through syntactic filters, which only count words of a certain syntactic category. In my machine, I only count the number of nouns and adjectives overlap. Moreover, to avoid promoting long sentences, I use a normalization factor, and divide the content overlap of two sentences with the length of each sentence. Formally, given two sentences *Si* and *Sj*, the similarity of these two sentences is defined as:

$$Similarity(Si, Sj) = \frac{|Interaction\_words|}{\log|Si| + \log|Sj|}$$

4. Experiments & Result

Running command:

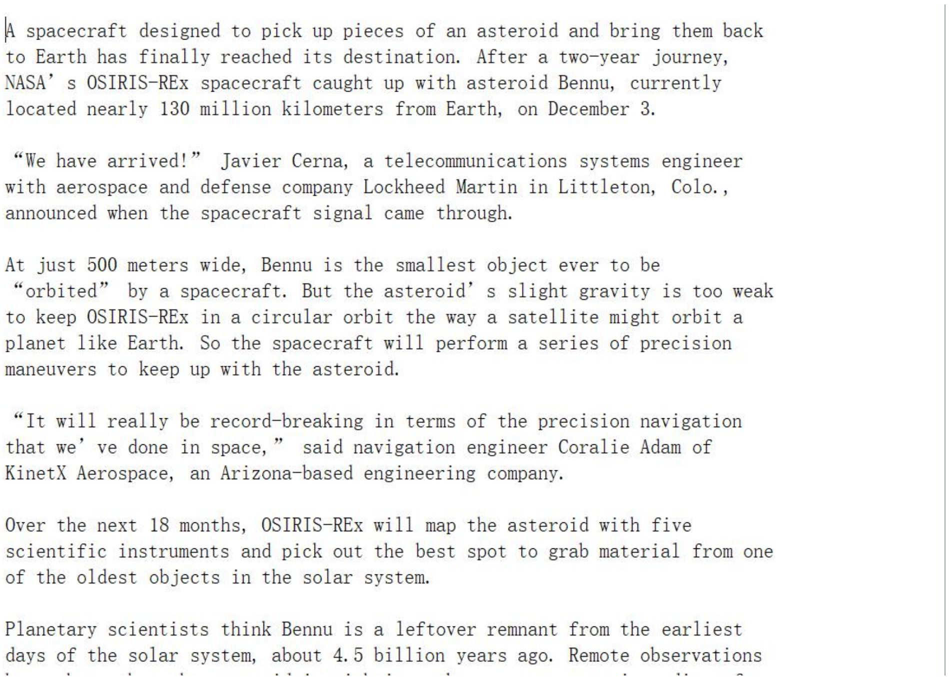
```
Python NLP_Final.py <pathOfTestDocument/docName> N M
```

N is the number of keywords we want to extract

M is the number of key sentences we want to put into our summarization

Here is an snapshot of a extraction result:

Input text:



A spacecraft designed to pick up pieces of an asteroid and bring them back to Earth has finally reached its destination. After a two-year journey, NASA’s OSIRIS-REx spacecraft caught up with asteroid Bennu, currently located nearly 130 million kilometers from Earth, on December 3.

“We have arrived!” Javier Cerna, a telecommunications systems engineer with aerospace and defense company Lockheed Martin in Littleton, Colo., announced when the spacecraft signal came through.

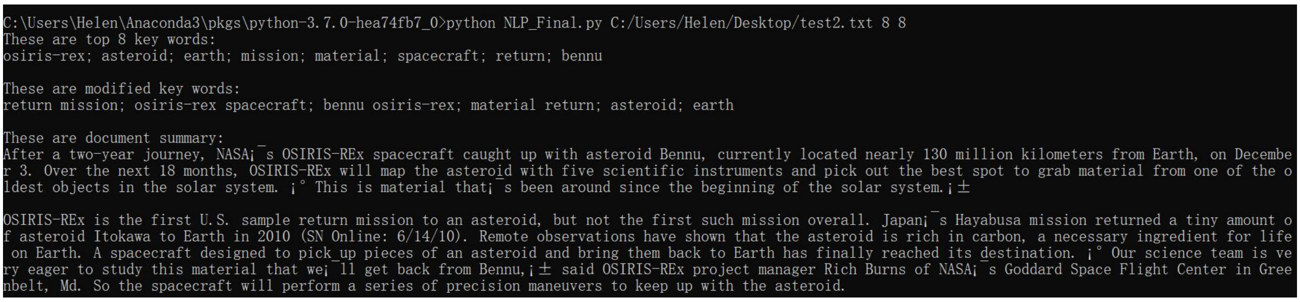
At just 500 meters wide, Bennu is the smallest object ever to be “orbited” by a spacecraft. But the asteroid’s slight gravity is too weak to keep OSIRIS-REx in a circular orbit the way a satellite might orbit a planet like Earth. So the spacecraft will perform a series of precision maneuvers to keep up with the asteroid.

“It will really be record-breaking in terms of the precision navigation that we’ve done in space,” said navigation engineer Coralie Adam of KinetX Aerospace, an Arizona-based engineering company.

Over the next 18 months, OSIRIS-REx will map the asteroid with five scientific instruments and pick out the best spot to grab material from one of the oldest objects in the solar system.

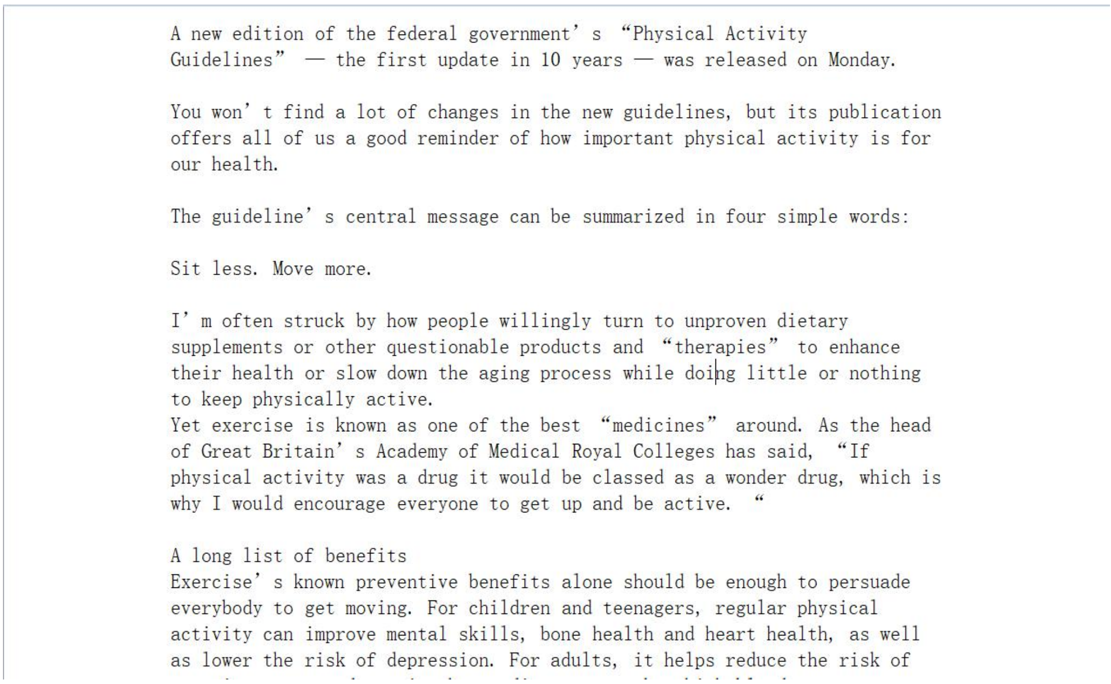
Planetary scientists think Bennu is a leftover remnant from the earliest days of the solar system, about 4.5 billion years ago. Remote observations

Extraction:



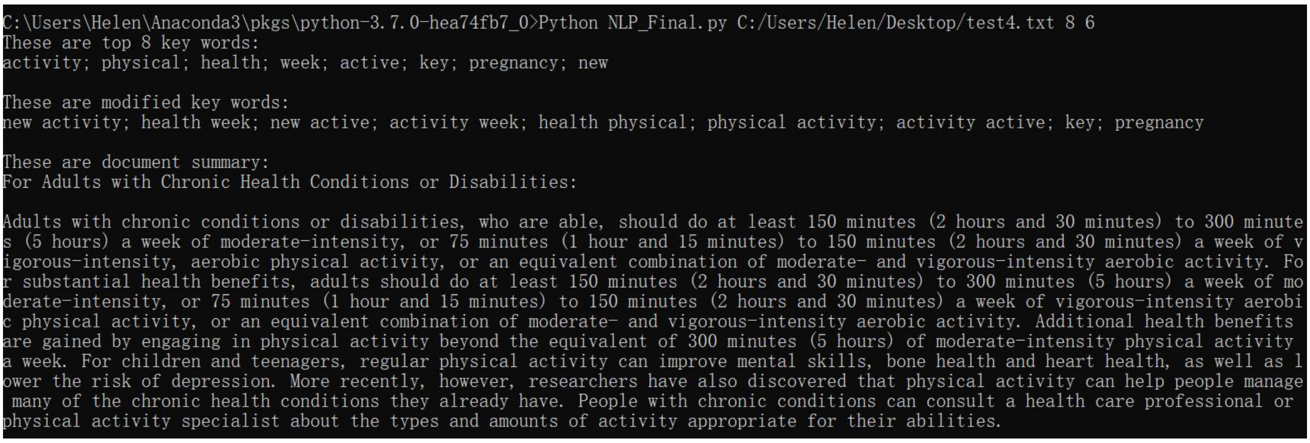
```
C:\Users\Helen\Anaconda3\pkgs\python-3.7.0-hea74fb7_0>python NLP_Final.py C:/Users/Helen/Desktop/test2.txt 8 8
These are top 8 key words:
osiris-rex; asteroid; earth; mission; material; spacecraft; return; bennu

These are modified key words:
return mission; osiris-rex spacecraft; bennu osiris-rex; material return; asteroid; earth

These are document summary:
After a two-year journey, NASA¡¯s OSIRIS-REx spacecraft caught up with asteroid Bennu, currently located nearly 130 million kilometers from Earth, on Decembe
r 3. Over the next 18 months, OSIRIS-REx will map the asteroid with five scientific instruments and pick out the best spot to grab material from one of the o
ldest objects in the solar system. ¡° This is material that¡¯s been around since the beginning of the solar system.¡±

OSIRIS-REx is the first U.S. sample return mission to an asteroid, but not the first such mission overall. Japan¡¯s Hayabusa mission returned a tiny amount o
f asteroid Itokawa to Earth in 2010 (SN Online: 6/14/10). Remote observations have shown that the asteroid is rich in carbon, a necessary ingredient for life
 on Earth. A spacecraft designed to pick up pieces of an asteroid and bring them back to Earth has finally reached its destination. ¡° Our science team is ve
ry eager to study this material that we¡¯ll get back from Bennu,¡± said OSIRIS-REx project manager Rich Burns of NASA¡¯s Goddard Space Flight Center in Gree
nbelt, Md. So the spacecraft will perform a series of precision maneuvers to keep up with the asteroid.
```

Input text:



```
A new edition of the federal government’ s “Physical Activity
Guidelines” — the first update in 10 years — was released on Monday.

You won’ t find a lot of changes in the new guidelines, but its publication
offers all of us a good reminder of how important physical activity is for
our health.

The guideline’ s central message can be summarized in four simple words:

Sit less. Move more.

I’ m often struck by how people willingly turn to unproven dietary
supplements or other questionable products and “therapies” to enhance
their health or slow down the aging process while doing little or nothing
to keep physically active.
Yet exercise is known as one of the best “medicines” around. As the head
of Great Britain’ s Academy of Medical Royal Colleges has said, “If
physical activity was a drug it would be classed as a wonder drug, which is
why I would encourage everyone to get up and be active. “

A long list of benefits
Exercise’ s known preventive benefits alone should be enough to persuade
everybody to get moving. For children and teenagers, regular physical
activity can improve mental skills, bone health and heart health, as well
as lower the risk of depression. For adults, it helps reduce the risk of
```

Extraction:



```
C:\Users\Helen\Anaconda3\pkgs\python-3.7.0-hea74fb7_0>Python NLP_Final.py C:/Users/Helen/Desktop/test4.txt 8 6
These are top 8 key words:
activity; physical; health; week; active; key; pregnancy; new

These are modified key words:
new activity; health week; new active; activity week; health physical; physical activity; activity active; key; pregnancy

These are document summary:
For Adults with Chronic Health Conditions or Disabilities:

Adults with chronic conditions or disabilities, who are able, should do at least 150 minutes (2 hours and 30 minutes) to 300 minute
s (5 hours) a week of moderate-intensity, or 75 minutes (1 hour and 15 minutes) to 150 minutes (2 hours and 30 minutes) a week of v
igorous-intensity, aerobic physical activity, or an equivalent combination of moderate- and vigorous-intensity aerobic activity. Fo
r substantial health benefits, adults should do at least 150 minutes (2 hours and 30 minutes) to 300 minutes (5 hours) a week of mo
derate-intensity, or 75 minutes (1 hour and 15 minutes) to 150 minutes (2 hours and 30 minutes) a week of vigorous-intensity aerobi
c physical activity, or an equivalent combination of moderate- and vigorous-intensity aerobic activity. Additional health benefits
are gained by engaging in physical activity beyond the equivalent of 300 minutes (5 hours) of moderate-intensity physical activity
a week. For children and teenagers, regular physical activity can improve mental skills, bone health and heart health, as well as l
ower the risk of depression. More recently, however, researchers have also discovered that physical activity can help people manage
 many of the chronic health conditions they already have. People with chronic conditions can consult a health care professional or
physical activity specialist about the types and amounts of activity appropriate for their abilities.
```

5. Conclusion

In this project, I implement the Keyphrase Detection and text summarization using TextRank Algorithm. I learned how to tag words in sentences with POS tags using NLTK and how to implement the TextRank Algorithm.

TextRank can successfully extract the core information from the text itself. Unlike other supervised systems, which attempt to learn what makes a good summary by training on collections of summaries built for other articles, TextRank is fully unsupervised, and relies only on the give text to derive an extractive summary, which represents a summarization model closer to what humans are doing when producing an abstract for a given document.

There are a lot of things we can do to improve the TextRank algorithm. If we use a more complex and precise approach for word similarity calculation, the accuracy of key phrases extraction may higher. For example, we can use Levenshtein Distance as the relation between text units.[4]

6. References

[1] Page Ranking: https://en.wikipedia.org/wiki/PageRank

[2] Yogan Jaya Kumar, Ong Sing Goh, Halizah Basiron, Ngo Hea Choon and Puspalata C Suppiah “A Review on Automatic Text Summarization Approaches” ,2016

[3] Mihalcea, Rada, and Paul Tarau. "Textrank: Bringing order into text." Proceedings of the 2004 conference on empirical methods in natural language processing. 2004.

[4] Jan Wijffels (2017, Dec 18). *Textrank for summarizing text.* Retrieved from https://cran.r-project.org/web/packages/textrank/vignettes/textrank.html