Application of Big Data in Social Science

week 7

Heidi Hyeseung Choi
Fall 2022
HYSIS
heidichoi@hanyang.ac.kr

Announcements

~~9.1~~	~~1~~	~~Introduction~~
~~9.8~~	~~2~~	~~Web Scraping~~
~~9.15~~	~~3~~	
~~9.22~~	~~4~~	~~Natural Language Processing~~
~~9.29~~	~~5~~	
~~10.6~~	~~6~~	**Text Analysis** (recorded lecture on week 7)
10.13	7	
10.20	8	Mid term exam (as school schedule)
10.27	9	Social Network Analysis
11.3	10	
11.10	11	Machine Learning: Supervised Learning
11.17	12	
11.24	13	Machine Learning: Unsupervised Learning
12.1	14	
12.8	15	Data Visualization
12.15	16	Final Exam

Please don't come to class this week.

On recorded lecture!

- I will be dividing today's lecture into two.
- Sentiment analysis will be a rather short video.
- This current sentiment analysis will be included in the mid term and attendance for this lecture video will be taken (until 19th).

- The other video on word cloud, is NOT included in the mid term, BUT will be needed to complete your homework assignment 1.
  - I will post this section of the video by.. mid term week
  - I will not put time limit on WORD CLOUD, just watch it before you do your homework assignment 1.

Announcements

- Mid term exam: October 20th 1:00pm – 2:15pm
- Mid term will **not** include:
  - Lecture 1: OT
  - Selenium
  - Word cloud
- On mid term exam:

1. LMS portion: PDF lecture notes (50 points)
   - Multiple choice : 3 marks or 4 marks
   - True or false: 2 marks
2. Coding portion: Jupyter Notebook (50 points)
   - Coding: 2-4 marks

   = Total 100

My tentative plan

	points	#	worth
LMS	2	5	10
	3	8	24
	4	4	16
CODING	2	4	8
	3	6	18
	4	6	24
Total			100

ANY QUESTIONS, EMAIL ME!

Correction from last week's jupyter notebook!

What we covered last week.

Bag-of-Words model

This is a good day.
Is this a good day?

- A bag of words is a representation of text that describes the occurrence of words within a document.
- We just keep track of word counts and disregard the grammatical details and the word order.
- It is called a "*bag*" of words, because any information about the order or structure of words in the document is discarded.
- The model is only concerned with whether known words occur in the document, not *where* in the document.
- BoW depends on absolute term frequencies.

⇒ It is one of the simplest feature extraction* technique used for text data.

Feature?

- Column represents a measurable piece of data can be used for analysis.

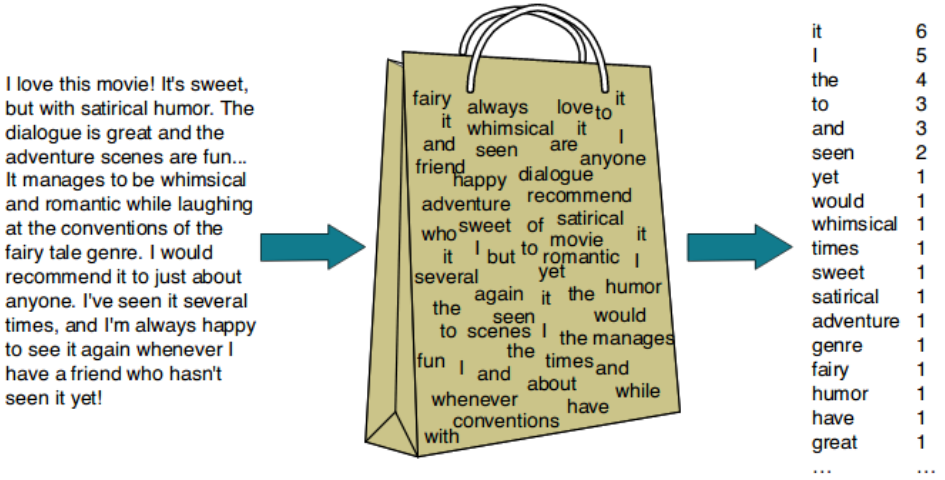
Eg) Name, age, income etc. (Variable)

Feature extraction?

The process of transforming raw data into numerical features that can be processed while preserving the information in the original data set.

Bag-of-Words model

- **1) Tokenize** each document and give an integer id for each token.
- **2) Count** the occurrences of tokens in each document.
- **3) Convert** into numeric vector so that each document is represented by one vector in the feature matrix and each column represents a unique word.
- Result in a sparse matrix containing many 0s



sentence 1(or document 1) : The weather is nice today.
sentence 2(or document 2) : The weather seems to be quite cold today.

	The	weather	is	nice	today	seems	to	be	quite	cold
Doc 1	1	1	1	1	1	0	0	0	0	0
Doc 2	1	1	0	0	1	1	1	1	1	1

TF-IDF Model (Term Frequency-Inverse Document Frequency)

$$w_{i,j} = tf_{i,j} \times idf_i$$

- Where w_{ij} is TF-IDF score for word i in document j,
- tf_{ij} is term frequency for word i in document j, and idf_i is IDF score for word i.

- TF-IDF: words that are common in every document, such as 'the', 'a', 'this', 'if' ranks low(close to 0) even though they may appear many times.
- In other words, if word is very common and appears in many documents, the number would approach 0. Otherwise it would approach 1.
- Word with high tf-idf in a document, it is most of the times occurred in given documents and must be absent in the other documents, thus, 'signature word'.

Usages?

- TF-IDF is useful extracting keywords from texts.
- The highest scoring words of a document are the most relevant to the document. Thus, considered as keywords.

Sentiment analysis

What is Sentiment analysis?

- Sentiment: a view of or attitude toward a situation or event; an opinion.
- Sentiment analysis: the use of NLP, text analysis to systematically identify, extract, quantify and study affective states and subjective information.
- Polarity: quantifying sentiment with positive or negative value.
- Subjectivity: generally refers to personal opinion, emotions or judgments.
- usages:
  - monitoring and measuring sentiment for social media posts and reviews
  - enhance customer experience
  - market research etc

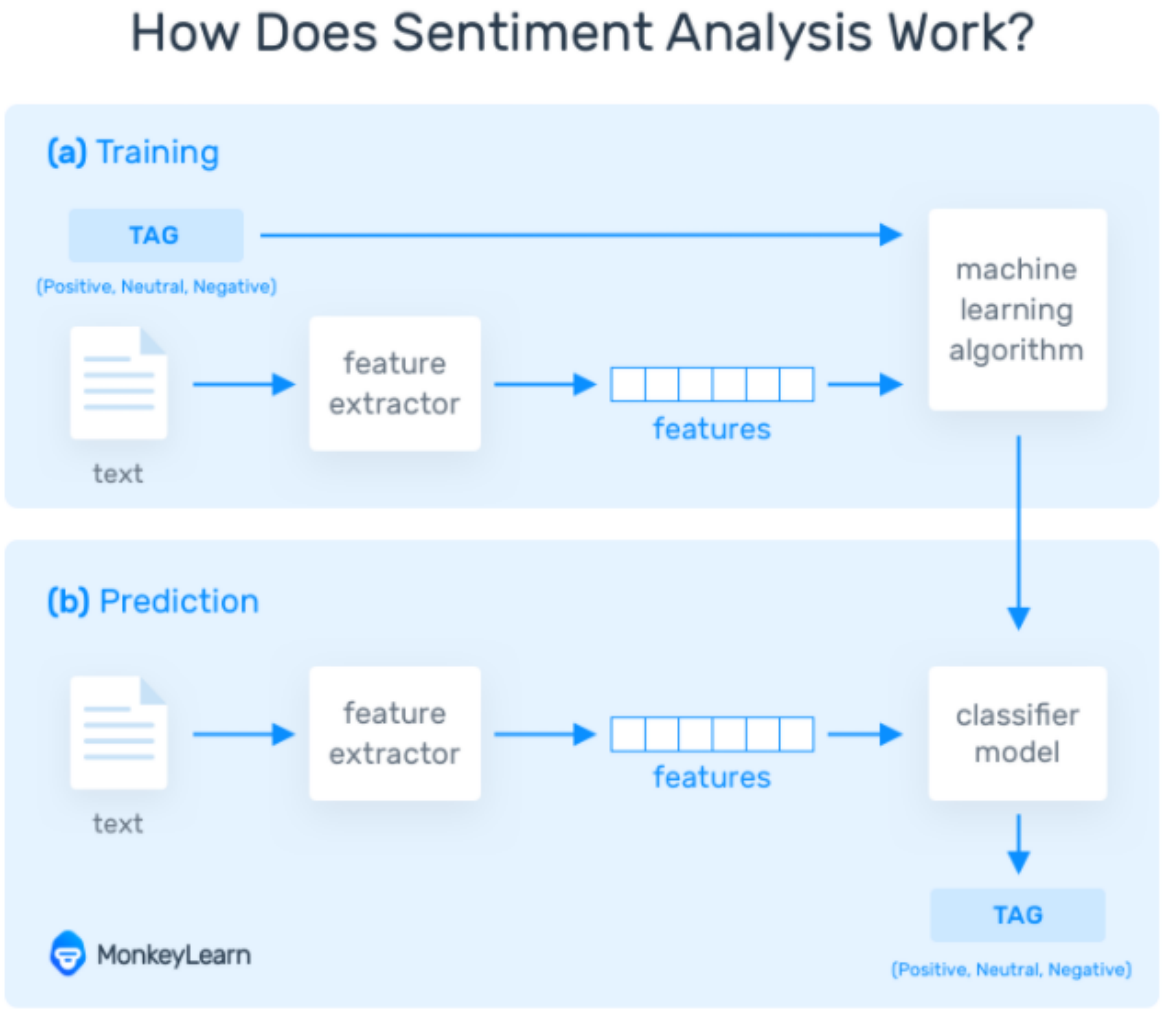
Techniques for sentiment analysis

1. Lexicon based (Rule-based) sentiment analysis
    1. define lists of polarized words.
    2. count the number of positive and negative words in a given text.
    3. Compare the appearances of both, and calculate.

2. Machine Learning (Automatic) based sentiment analysis
    1. split the data set into a training set and a test set
    2. transform text into numerical feature vectors
    3. train the model
    4. evaluate the model
    5. predict new data.

Fyi: Machine Learning (Automatic) based sentiment analysis

- feature:
  - a measurable property of the object that you are trying to analyze.
  - in datasets features appear as columns.



TextBlob library

- Sentiment analysis is basically the process of determining the attitude or the emotion of the writer, i.e., whether it is positive or negative or neutral.
- The *sentiment* function of textblob returns two properties, **polarity**, and **subjectivity**.
- Polarity returns a float which lies in the range of [-1,1] where 1 means positive statement and -1 means a negative statement.
- Subjectivity quantifies the amount of personal opinion and factual information contained in the text. The higher subjectivity means that the text contains personal opinion rather than factual information.
- Subjectivity is also a float which lies in the range of [0,1].

Python basics for today's work

lambda function

- Similar to for loops.
- "*A lambda function is a small function containing **a single expression**. Lambda functions can also act as anonymous functions where they don't require any name. These are very helpful when we have to perform small tasks with less code.*"

- keyword
- bound variable/argument
- expression

```
lambda x: x
```

```
lambda x: x + 5
```

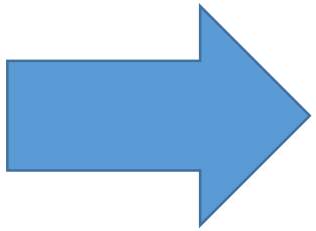
```
def add_5(x):
    return x+5
```

lambda function + apply function in pandas

- apply() function in pandas calls the lambda function and applies to every row or column of the dataframe and returns a modified copy of the dataframe.

```
>>> df['height']= df['height'].apply(lambda x: x+10)
>>> df['category'] = df['height'].apply(lambda x: 'Adult' if x>160 else 'Child')
```

name	height
A	125
B	175
C	185



name	height	category
A	135	Child
B	185	Adult
C	195	Adult

Join() Method for String

- The join() method takes all items in an iterable and joins them into one string.

syntax

```
>>> string.join(iterable)
```

Examples:

```
>>> names = [‘heidi’, ‘john’, ‘jenny’]
>>> x = “-”.join(names)
>>> print(x)
>>> heidi-john-jenny
>>> y= “ ”.join(names)
>>> print(y)
```

isalnum(), isalpha(), isdigit() methods in strings

- isalnum()
  - method returns 'TRUE' if all the characters are alphanumeric (alphabet (a-z) and numbers(0-9)
  - Not alphanumeric: (space)!(#$*etc
- isalpha()
  - method returns (a-z).
- isdigit()
  - method returns e False.
- syntax

⇒ for word cloud section, meaning this would not be included in the mid term.

```
>>> string.isalnum()
>>> x = 'hello123'
>>> x.isalnum()
>>> TRUE
```

```
>>> x.isalpha()
>>> FALSE
>>> x.isdigit()
>>> FALSE
```

Let’s go to jupyter notebook!

>>> pip install textblob

If you get module not found error try below.

>>> pip install -U textblob
>>> python –m textblob.download_corpora