**Attn: Dr. Sun Aixin**

NANYANG TECHNOLOGICAL UNIVERSITY

# CE/CZ4045 Natural Language Processing

We hereby declare that the attached group assignment has been researched, undertaken, completed and submitted as a collective effort by the group members listed below. We have honored the principles of academic integrity and have upheld Student Code of Academic Conduct in the completion of this work. We understand that if plagiarism is found in the assignment, then lower marks or no marks will be awarded for the assessed work.

| Name | Signature / Date |
|---|---|
| Lee Kai Shern | 31-Oct-20 |
| Tan Zarn Yao | 31-Oct-20 |
| Wang Wee Jia | 31-Oct-20 |
| Yew Wei Chee | 31-Oct-20 |
| | |
| | |

Important note:

Name must **EXACTLY MATCH** the one printed on your Matriculation Card. Any mismatch leads to **THREE (3)** marks deduction.

# Domain Specific Text Data Analysis and Processing

CE/CZ4045 Natural Language Processing

Group 12

| Lee Kai Shern | Tan Zarn Yao | Wang Wee Jia | Yew Wei Chee |
|---|---|---|---|
| U1820793J | U1820414C | U1820983E | U1820962G |

## ABSTRACT

*In this domain specific text data analysis and processing project, the use of natural language processing is demonstrated to analyse texts in several domains. In the first section, we will illustrate the results of our data analysis on three different domains (chemical, legal and sports) using tokenization, stemming, sentence segmentation and pos-tagging. We will also discuss about the possible solutions to improve the tokenizer and pos-tagger to better fit each domain. Next, we extracted 31 reviews for "Jennifer Lopez: All I Have" concert from the Yelp dataset and developed a <Noun, Adjective> pair ranker to identify the top 5 ranked pairs of noun and adjectives that appear together most frequently. Finally, we developed an application to perform data analysis on the reviews obtained from the previous section and utilized a sentiment analysis tool to predict the stars rated by the user.*

## 1 Domain Specific Dataset Analysis

The three domains and datasets selected are:

1. Research papers in chemical areas [1]: The 20 datasets are randomly selected from articles published in Journal of Chemistry.
2. Legal cases [2]: The dataset contains a textual corpus of 4000 legal cases for automatic summarization and citation analysis.
3. Sports articles [3]: The original dataset contains 1000 sports articles which were labeled using Amazon Mechanical Turk as objective or subjective. However, we have only extracted 20 articles to be used for this assignment.

### 1.1 Tokenization and Stemming

We used the *word_tokenize* function in the NLTK library [4] to perform tokenization. This function will tokenize some punctuations into separate individual tokens, such as comma. Therefore, we also filtered out the punctuations from the results of tokenization.

In the Chemistry domain, the compound chemical formulas such as *'$C_2H_5OH$'* are correctly tokenized into one single token. However, compound names such as *'Titanium dioxide'*, which are expected to be tokenized into one single token, has been tokenized into two separate tokens *'Titanium'* and *'dioxide'*. Scientific representation of numbers such as *'$5.2\times10^{-8}$'* are also incorrectly tokenized into two separate tokens *'5.2'* and *'$10^{-8}$'*.

Firstly, we can have a list of commonly used elements and compound tokens. After performing tokenization, we can loop through successive tokens to find out if there are consecutive elements or compound tokens, then group them into one single token of compound name. To identify scientific representation of numbers, we can find out if a token in the format of *'x.y'* where *x* and *y* are numerical digits, is followed by another token with the format of *'10-k'* or *'10+k'* where *k* is numerical digit(s). If this combination exists, we can combine both tokens into one single token with the format *'x.ye-k'* or *'x.ye+k'*.

In the legal domain, the tokenizer will separate the order and number of a rule e.g.: 'O 52 r 10(2)' is separated into 'O', '52', 'r', '10' and '2'. Besides, the dates are also separated into meaningless integer.

Since the date in legal documents must be formatted as "22 January 2019" whereby the month must be spelled fully and year must in full form, we can loop through the text and replaced it with '-' so it won't be tokenized.

In the sports domain, the domain specific terms are correctly identified e.g. *'home-run-to-fly-ball'* are correctly tokenized into a single token. However, proper nouns such as *'Windy City'* is separated into two tokens. Besides *'s* with possessive purposes are separated into two tokens e.g. *"Dunn's"* is separated into *"Dunn"* and *"'s"*.

To prevent splitting proper nouns into several tokens, we can skip tokenization when we reach words which are capitalized. For the possessive nouns, the word is separated because the original text file uses right single quotation marks (Unicode #8217) instead of apostrophe (Unicode #39). Proposed solution will be replacing all the left or right single quotation marks with apostrophe before tokenization.

For stemming, we used the *PorterStemmer* from the NLTK library. The number of distinct tokens before and after stemming is presented in Table 1.1. The length distribution of unstemmed and stemmed tokens of different domains are shown in Figure 1.1(a) (Chemistry), Figure 1.1(b) (Legal) and Figure 1.1(c) (Sports).

| **Domain** | **Number of Distinct Unstemmed Tokens** | **Number of Distinct Stemmed Tokens** |
|---|---|---|
| Chemistry | 10181 | 7533 |
| Legal | 10315 | 6833 |
| Sports | 8254 | 6186 |

Table 1.1: The number of distinct tokens for each domain before and after stemming

From Table 1.1, we can see that the number of distinct tokens reduced the most for legal documents, followed by Chemistry research papers and lastly sports articles. For legal documents, we can find a lot of words with the same stem such as ‘accuse’, ‘accused’, ‘accusable’, ‘accusing’, ‘accuser’, ‘accusation’, ‘accusal’, ‘accusative’ will all be stemmed to ‘accus’, which may infer our observation above.

From Figure 1.1(a), Figure 1.1(b) and Figure 1.1(c), we can see that after stemming, the number of distinct tokens that have length greater than 7 decreases significantly.

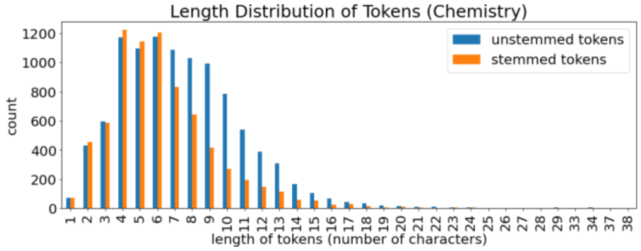


Figure 1.1(a): Distribution of token lengths for Chemistry papers in chemical areas

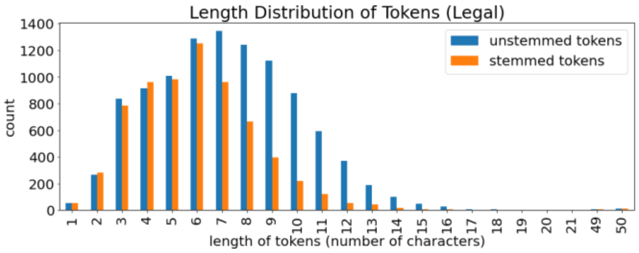


Figure 1.1(b): Distribution of token lengths for legal cases

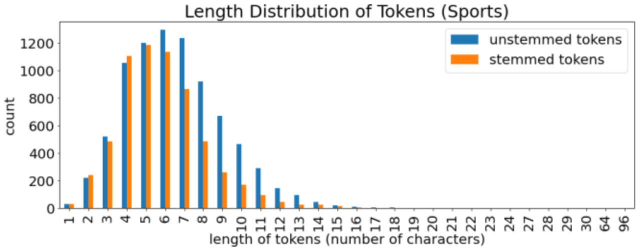


Figure 1.1(c): Distribution of token lengths for Sports Articles

## 1.2 Sentence Segmentation

We used the *tokenize* function from the *PunktSentenceTokenizer* class of the NLTK library to perform sentence segmentation on the texts from all three domains.

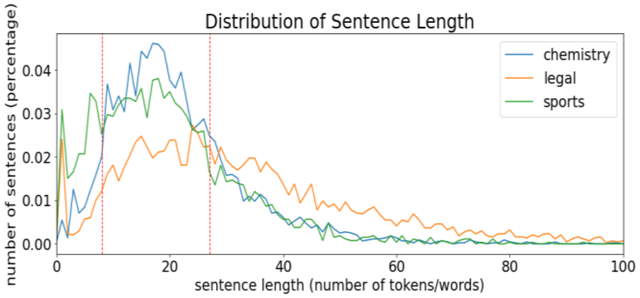


Figure 1.2(a): Distribution of normalized sentence length color coded by the specific domains.

As observed in Figure 1.2(a), sports domain has the highest frequency for sentences with length from 0 to about 8 words. The chemistry domain has the highest frequency for sentences with length in the range of 8 to about 27 words. Sentences with more than 27 words are mostly observed in the legal domain. From the aforementioned observation, we can deduce that sentences in the legal domain are usually longer in terms of number of words.

Figure 1.2(b) presented below depicts the same relationship as in Figure 1.2(a) with better visualization.

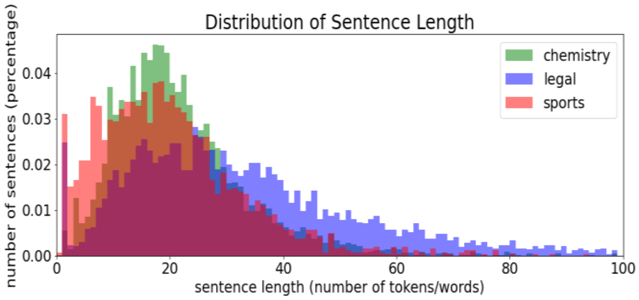


Figure 1.2(b): Distribution of normalized sentence length color coded by the specific domains.

## 1.3 POS Tagging

We selected 3 sentences randomly from each domain and perform POS tagging using the *pos_tag* function in NLTK Library. The results are shown in Table 1.3.

| **Chemistry** |
|---|
| High-speed/JJ countercurrent/NN chromatography/NN HSCCC/NNP is/VBZ a/DT new/JJ type/NN of/IN liquid–liquid/JJ partition/NN chromatography/NN technology/NN |
| For/IN the/DT adsorption/NN of/IN CR/NNP and/CC SF/NNP the/DT thermodynamic/JJ parameters/NNS ΔG°/VBP ΔH°/JJ ΔS°/NNS were/VBD determined/VBN using/VBG the/DT Van/NNP t/NN Hoff/NNP equation/NN |
| The/DT nitrogen/NN atom/NN of/IN imine/JJ -C=N-/NNP Schiff/NNP base/NN group/NN is/VBZ thought/VBN to/TO involve/VB in/IN hydrogen/NN bonding/NN with/IN several/JJ cellular/JJ constituents/NNS 15/CD which/WDT can/MD modulate/VB activities/NNS and/CC processes/NNS |
| **Legal** |
| Wilcox/NNP J/NNP made/VBD orders/NNS ancillary/JJ to/TO the/DT Mareva/NNP orders/NNS on/IN 22/CD March/NNP 2005/CD requiring/VBG each/DT of/IN the/DT Sharman/NNP applicants/NNS to/TO disclose/VB on/IN affidavit/NN the/DT description/NN and/CC value/NN of/IN all/DT of/IN their/PRP$ assets/NNS wherever/RB situated/VBN and/CC to/TO specify/VB whether/IN those/DT assets/NNS were/VBD held/VBN by/IN each/DT applicant/NN either/CC beneficially/RB or/CC in/IN trust/NN for/IN any/DT other/JJ person/NN or/CC entity/NN |
| The/DT 1994/CD Award/NNP is/VBZ an/DT award/NN made/VBN by/IN the/DT Australian/JJ Industrial/NNP Relations/NNPS Commission/NNP 'the/POS Commission/NNP pursuant/NN to/TO the/DT Industrial/NNP Relations/NNP Act/NNP 1988/CD Cth/NNP which/WDT subsequently/RB became/VBD the/DT WR/NNP Act/NNP |
| The/DT first/JJ respondent/NN proposes/VBZ to/TO put/VB a/DT logo/NN on/IN the/DT façade/NN of/IN the/DT store/NN above/IN the/DT entrance/NN which/WDT displays/VBZ the/DT words/NNS Oasis/NNP Foam/NNP Rubber/NNP against/IN a/DT light/JJ blue/NN or/CC aqua/NN background/NN to/TO the/DT word/NN Oasis/NN |
| **Sports** |
| Yet/RB Howard/NNP still/RB possesses/VBZ one/CD of/IN the/DT more/RBR powerful/JJ strokes/NNS in/IN the/DT league/NN evidenced/VBN by/IN 14/CD homers/NNS in/IN 71/CD games/NNS and/CC his/PRP$ 27.5/CD home-run-to-fly-ball/JJ percentage/NN was/VBD his/PRP$ best/JJS showing/NN since/IN 2008/CD |
| For/IN her/PRP$ eighth/JJ birthday/JJ 10/CD days/NNS before/IN the/DT league/NN 's/POS first/JJ game/NN her/PRP$ uncle/NN gave/VBD her/PRP a/DT pair/NN of/IN regulation-size/JJ WNBA/NNP basketballs/VBZ one/CD outdoor/NN one/CD indoor/NN |
| Suppose/VB their/PRP$ remains/NNS suspicion/NN from/IN Dunn/NNP s/RB infamous/JJ 2011/CD output/NN .159/.292/.277/NNP in/IN 122/CD games/NNS yet/RB the/DT Windy/NNP City/NNP slugger/NN s/NN rushed/VBD return/NN from/IN an/DT early-season/JJ emergency/NN appendectomy/NN was/VBD probably/RB the/DT catalyst/NN for/IN this/DT lethargic/JJ showing/NN |

Table 1.3: POS tagging results for 3 different domains

In the Chemistry domain, the POS tagger can classify most domain specific terms correctly. However, several mis-tagging can be found, for instance:

- 'ΔG°/VBP ΔH°/JJ ΔS°/NNS' in which all should be tagged as NN or NNP.
- 'Van/NNP t/NN Hoff/NNP' in which all should be tagged as NNP
- 'hydrogen/NN bonding/NN' which is supposed to be 'hydrogen/JJ bonding/NN'.

In the legal domain, most of the mis-tagging occurs on the proper noun, which is also due to the mis-tokenization of tokenizer. Such as:

- 'Australian/JJ Industrial/NNP Relations/NNPS Commission/NNP', which should be all NNP.
- The first 'Oasis' in the phrase 'Oasis Foam Rubber' is correctly tagged as NNP, but the second 'Oasis' in the phrase 'the word Oasis' is tagged as NN

In the sports domain, the domain specific term can be tagged correctly, such as 'home-run-to-fly-ball/JJ' which is out of our expectation. However, there are several errors spotted as below:

- In the noun phrase 'regulation-size/JJ WNBA/NNP basketballs/VBZ', the word 'basketballs' is mis-tagged as VBZ instead of NNS
- '.159/.292/.277' is the score of the games, which should be tagged as CD, but is mis-tagged as NNP

Overall, the POS tagger performs better than our expectation, with just some minor mis-tagging found.

## 2 Development of a < Noun - Adjective > Pair Ranker

We extracted 31 reviews for *"Jennifer Lopez: All I Have"* [5] concert from the Yelp dataset for this section.

### 2.1 Most Meaningful 5 Pairs of < Noun - Adjective >

After going through the reviews manually, we identified the following pairs of < Noun – Adjective > to be the most meaningful 5 pairs.

| No. | Noun | Adjective |
|---|---|---|
| 1 | show | great |
| 2 | jlo | amazing |
| 3 | fan | huge |
| 4 | performer | amazing |
| 5 | seats | worth |

Table 2.1: The most meaningful 5 pairs of < Noun – Adjective > by manual extraction

### 2.2 Development of a < Noun - Adjective > Pair Ranker

We developed a < Noun – Adjective > pair ranker with the following steps:

1. Tokenize all reviews. (*word_tokenize* function in the NLTK library)
2. Apply POS tagging on the tokens. (*pos_tag* function in NLTK Library)
3. For each review, list out all the distinct nouns and distinct adjectives from the POS tagging result.
4. For each review, generate all the possible pairs of nouns and adjectives from the list of nouns and adjectives from Step 3.
5. Combine all the < Noun – Adjective > pairs from every review into a list.
6. Calculate the frequency of every < Noun – Adjective > pair in the list.
7. Sort the < Noun – Adjective > pair according to their frequency in descending order.

After performing the abovementioned steps, the top 10 < Noun – Adjective > pairs are as follows:

| No. | Noun | Adjective | Frequency |
|---|---|---|---|
| 1 | show | great | 10 |
| 2 | jlo | great | 9 |
| 3 | jlo | good | 9 |
| 4 | seats | good | 7 |
| 5 | seats | great | 7 |
| 6 | jlo | huge | 6 |
| 7 | show | good | 6 |
| 8 | show | amazing | 6 |
| 9 | fan | huge | 6 |
| 10 | time | great | 6 |

Table 2.2(a): The top 10 < Noun – Adjective > pairs that appeared most frequently

We observed that nouns such as "show", "jlo" and "seats" are repeated a few times in Table 2.2(a). We further improve our pair ranker by extracting pairs with distinct nouns.

| No. | Noun | Adjective | Frequency |
|---|---|---|---|
| 1 | show | great | 10 |
| 2 | jlo | great | 9 |
| 3 | seats | good | 7 |
| 4 | fan | huge | 6 |
| 5 | time | great | 6 |

Table 2.2(b): The most meaningful 5 pairs of < Noun – Adjective > by pair ranker

By comparing Table 2.1 and Table 2.2(b), we observed that we managed to match < show – great > and < fan – huge > with our pair ranker. For "jlo", our pair ranker matched it with the adjective "great" which is similar in meaning with "amazing". Our pair ranker did not manage to identify the pairs < performer, amazing > and < seats, worth >.

One of the challenges we encountered in this task is that the adjectives "good" and "great" appeared in almost every review, thus affecting the ability of our pair ranker to identify other adjectives such as "worth" and "amazing". This issue can be mitigated by examining a larger dataset, allowing the pair ranker to observe more variety of words. Besides that, adjectives may appear before their respective nouns in some reviews. We approached this issue by generating all the possible pairs of noun and adjective in each review. Although this approach might result in nonsensical pair such as < fan – worth >, the frequency of such pairs will be very low thus having little effect in determining the top 5 pairs.

## 3 Application

Sentiment Analysis can help us decipher the mood and emotions of general public and gather insightful information regarding the context. It is essential for businesses and in our Yelp dataset, customers express their thoughts and feelings and by analyzing customer feedback, we are able to gain insights as to whether a customer is happy or unsatisfied with the services provided.

### 3.1 Visualization

The following figures are Word Cloud [7] that we did on the dataset's texts grouped by the stars that the customers have given. Word Cloud is an image composed of words in which the size of each word indicates its frequency or importance. The more often a specific word appears in the texts, the bigger and bolder it will appear in the word cloud.

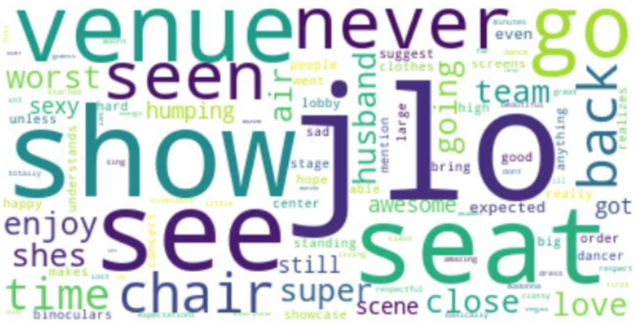


Figure 3.1(a): Word Cloud of 1 star

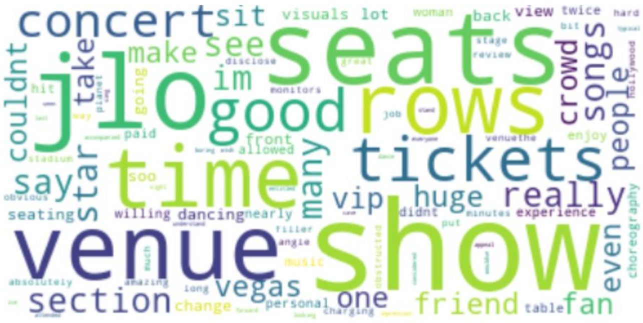


Figure 3.1(b): Word Cloud of 2 star

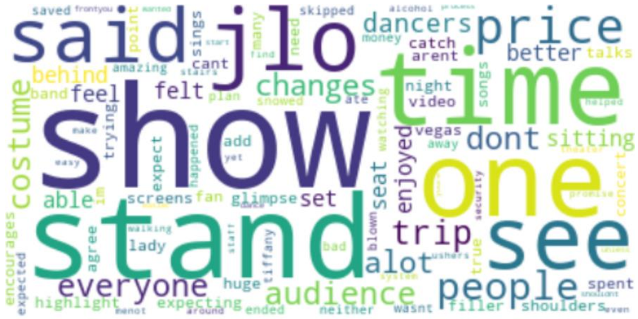


Figure 3.1(c) Word Cloud of 3 star

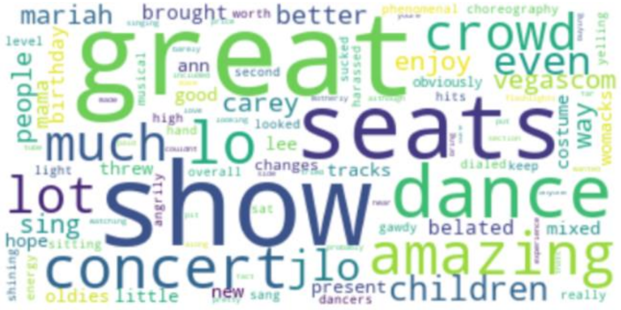


Figure 3.1(d): Word Cloud of 4 star

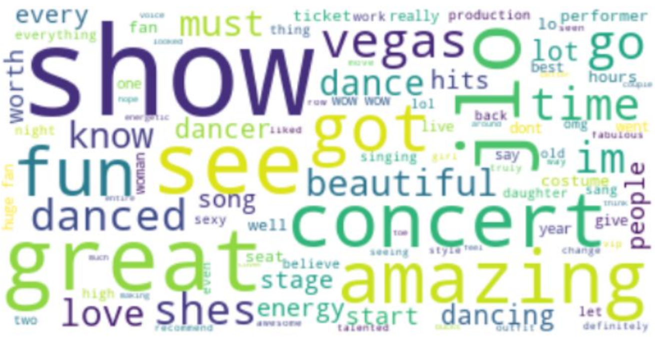
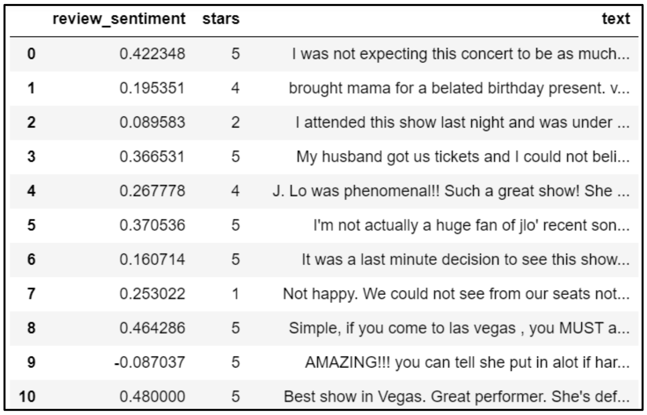


Figure 3.1(e): Word Cloud of 5 star

## 3.2 Sentiment Analysis

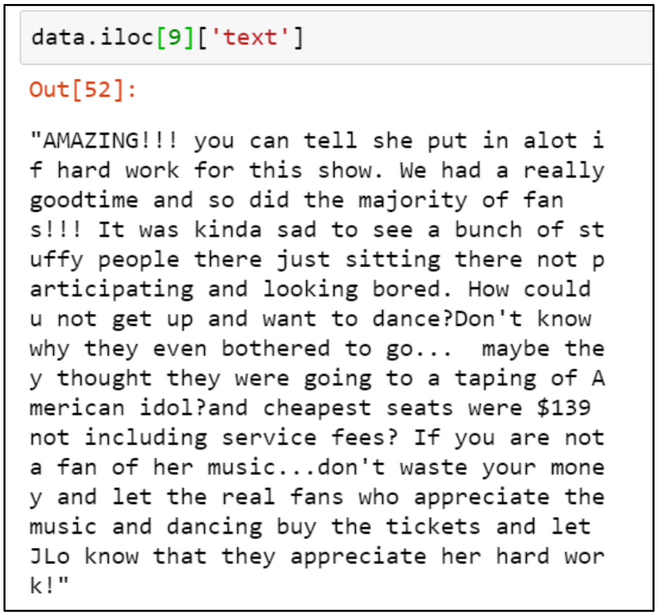
For the Yelp dataset, we utilized TextBlob [6], a python library for Natural Language Processing, which actively uses Natural Language Tool Kit (NLTK) to achieve its task. A sentiment is defined by its semantic orientation and the intensity of each word in the sentence and for this instance, we require a pre-defined dictionary classifying negative and positive words. As our dataset contains texts that are represented by a bag of words, and by assigning individual scores to all the words, final sentiment is calculated by some pooling operation by taking an average of all the sentiments.



|  | review_sentiment | stars | text |
|---|---|---|---|
| 0 | 0.422348 | 5 | I was not expecting this concert to be as much... |
| 1 | 0.195351 | 4 | brought mama for a belated birthday present. v... |
| 2 | 0.089583 | 2 | I attended this show last night and was under ... |
| 3 | 0.366531 | 5 | My husband got us tickets and I could not beli... |
| 4 | 0.267778 | 4 | J. Lo was phenomenal!! Such a great show! She ... |
| 5 | 0.370536 | 5 | I'm not actually a huge fan of jlo' recent son... |
| 6 | 0.160714 | 5 | It was a last minute decision to see this show... |
| 7 | 0.253022 | 1 | Not happy. We could not see from our seats not... |
| 8 | 0.464286 | 5 | Simple, if you come to las vegas , you MUST a... |
| 9 | -0.087037 | 5 | AMAZING!!! you can tell she put in alot if har... |
| 10 | 0.480000 | 5 | Best show in Vegas. Great performer. She's def... |

Figure 3.2(a): TextBlob polarity on text

The below example shows a perfect example as to how TextBlob might incorrectly predict the stars given by the customers if we were to assign the 'review_sentiment' to stars based on the magnitude of the polarity produced by TextBlob. A negative polarity was assigned for the text even though the user was really satisfied by the performance such that he/she has given the maximum star of 5 for the performance, but continued to elaborate about how his/her experience was somewhat diminished by the interests shown by other participants.



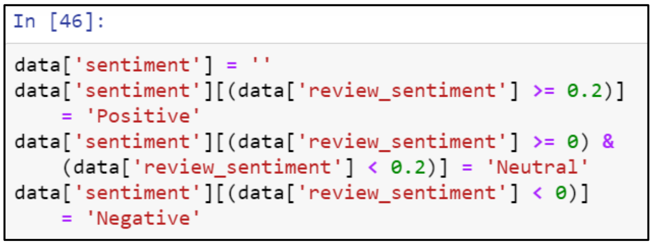
```
data.iloc[9]['text']
```

```
Out[52]:

"AMAZING!!! you can tell she put in alot i
f hard work for this show. We had a really
goodtime and so did the majority of fan
s!!! It was kinda sad to see a bunch of st
uffy people there just sitting there not p
articipating and looking bored. How could
u not get up and want to dance?Don't know
why they even bothered to go...  maybe the
y thought they were going to a taping of A
merican idol?and cheapest seats were $139
not including service fees? If you are not
a fan of her music...don't waste your mone
y and let the real fans who appreciate the
music and dancing buy the tickets and let
JLo know that they appreciate her hard wor
k!"
```

Figure 3.2(b): Example of a 5-star text with negative polarity

Assuming that we set the threshold of the texts being assigned 'Positive', 'Neutral' and 'Negative' that were shown below.

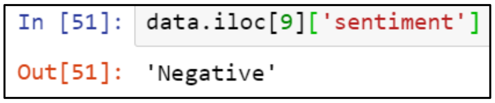


```
In [46]:

data['sentiment'] = ''
data['sentiment'][(data['review_sentiment'] >= 0.2)]
    = 'Positive'
data['sentiment'][(data['review_sentiment'] >= 0) &
    (data['review_sentiment'] < 0.2)] = 'Neutral'
data['sentiment'][(data['review_sentiment'] < 0)]
    = 'Negative'
```

Figure 3.2(c): Threshold values for polarity

The sentiment analysis being done on the text which was given 5 stars will be assigned a negative.



```
In [51]:  data.iloc[9]['sentiment']

Out[51]:  'Negative'
```

Figure 3.2(d): The prediction result of the 10th review

## REFERENCES

[1] *Hindawi. 2020. Journal Of Chemistry. [online] Available at: <https://www.hindawi.com/journals/jchem/contents/year/2020/page/2/> [Accessed 4 October 2020].*

[2] *Archive.ics.uci.edu. 2020. UCI Machine Learning Repository: Legal Case Reports Data Set. [online] Available at:<https://archive.ics.uci.edu/ml/datasets/Legal+Case+Reports> [Accessed 4 October 2020].*

[3] *Archive.ics.uci.edu. 2020. UCI Machine Learning Repository: Sports Articles For Objectivity Analysis Data Set. [online] Available at: <https://archive.ics.uci.edu/ml/datasets/Sports+articles+for+objectivity+analysis> [Accessed 4 October 2020].*

[4] *Nltk.org. 2020. Natural Language Toolkit — NLTK 3.5 Documentation. [online] Available at: <https://www.nltk.org/> [Accessed 4 October 2020].*

[5] *Yelp.com. 2020. Yelp Dataset. [online] Available at: <https://www.yelp.com/dataset> [Accessed 4 October 2020].*

[6] *Textblob.readthedocs.io. 2020. Textblob: Simplified Text Processing — Textblob 0.16.0 Documentation. [online] Available at: <https://textblob.readthedocs.io/en/dev/> [Accessed 4 October 2020].*

[7] *Amueller.github.io. 2020. Wordcloud For Python Documentation — Wordcloud 1.8.0.Post1+G5f23ed4 Documentation. [online] Available at: <http://amueller.github.io/word_cloud/> [Accessed 4 October 2020].*

## CONTRIBUTIONS

| Name \ Task | 3.1 | 3.2 | 3.3 | 4.1 | 4.2 |
|---|---|---|---|---|---|
| Lee Kai Shern | 25% | 25% | 25% | 25% | 25% |
| Tan Zarn Yao | 25% | 25% | 25% | 25% | 25% |
| Wang Wee Jia | 25% | 25% | 25% | 25% | 25% |
| Yew Wei Chee | 25% | 25% | 25% | 25% | 25% |