Calculating sentiment scores by VADER

Tetsuya Yamada

3rd year Ph.D. student in Pitt School of Education

Why sentiment analysis?

- Social scientists are surrounded by a large amount of unstructured data, including texts.
- Natural Language Processing techniques are helpful for the computer to understand text data.
- **Quantifying emotions behind the texts** can provide further insights for your social science research.

- Example
    - Interviewees’ emotions during the interview
    - Subjective feelings on political issues in social media(e.g., vaccine, the performance of the incumbent)

Sentiment analysis

- Definition

The process to detect **positive or negative** sentiments in text

- Two categories

1. Dictionary-based approach

Use the human-crafted dictionary to assess the sentiment of phrases and sentences. No need to train a model using labeled data

2. Machine Learning approach

Train a model using labeled data to predict the sentiment of new input text

Simple dictionary-based sentiment analysis

- Word count method
  - Count the negative and positive words and take the ratio of the difference of positive and negative word counts and total word counts.

  - Limitation:

-cannot capture the combination of words (e.g., not good)

-Ignore sentiment-bearing lexical items (e.g., Emoji, slang, acronyms)

-does not capture the sentiment *intensity* of words (e.g., exceptional>good)

VADER (Valence Aware Dictionary and sEntiment Reasoner)

- A human-validated dictionary for sentiment analysis developed by Hutto and Gilbert (2014)
- VADER includes a list of lexical features (e.g., words and phrases) labeled as positive/negative according to their semantic orientation.
- Fast and relatively accurate for short texts, such as texts in social media
- Detect **polarity** (positive, neutral, and negative) and **intensity of emotions**
- Compound scores: the summing up the valence scores of *each sentiment-bearing word* in the lexicon, adjusted according to the rules (-1 to 1)
  - Positive: compound score>=0.05
  - Neutral: compound score>-0.05 and compound score <0.05
  - Negative: compound score<=-0.05

Advantages of VADER (Hutto & Gilbert, 2014)

- Sentiment-bearing lexical items: Emojis (:D, :P), acronyms (LOL and ROFL), and slangs (nah and meh)
- Capitalization (sad vs SAD) and extended punctuation (? vs ???)
- Degree modifiers (e.g., the food is *extremely* good.)
- Polarity shift (e.g., the food is great, *but* the service is horrible).
- Polarity negation (e.g., the food **isn't really good**).

No need for tokenization, stemming/lemmatization, and removing stop words.

The application is simple and fast!!!

```
In [1]: import nltk

In [2]: from nltk.sentiment.vader import SentimentIntensityAnalyzer

In [3]: analyzer = SentimentIntensityAnalyzer()
```

Are these sentences positive or negative?
"I love Dr. Colaresi's scientific computation class"
"I love Dr. Colaresi's scientific computation class!!!"

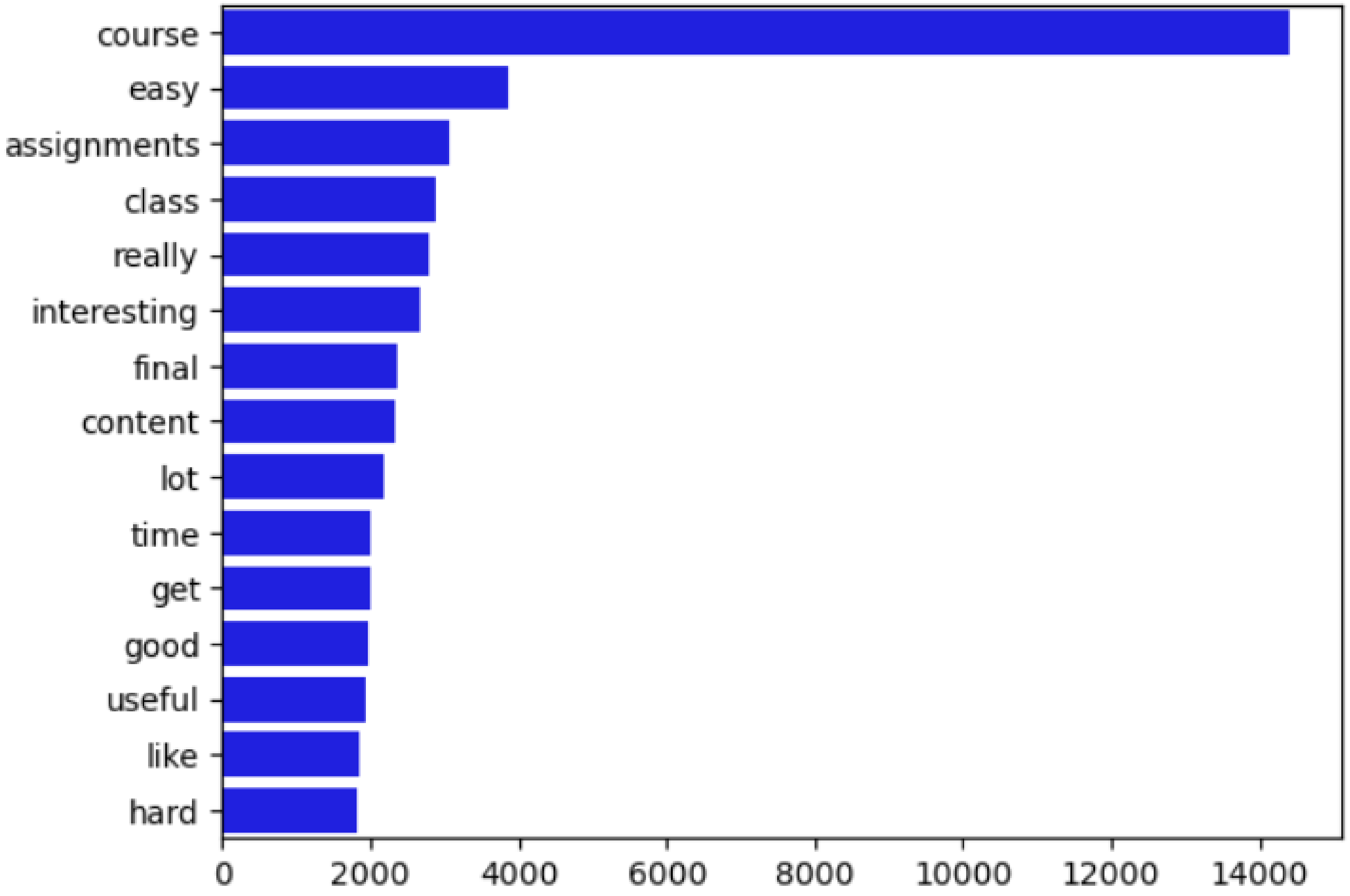
```
In [4]: analyzer.polarity_scores("I love Dr. Colaresi's scientific computation class")
Out[4]: {'neg': 0.0, 'neu': 0.543, 'pos': 0.457, 'compound': 0.6369}

In [5]: analyzer.polarity_scores("I love Dr. Colaresi's scientific computation class!!!")
Out[5]: {'neg': 0.0, 'neu': 0.496, 'pos': 0.504, 'compound': 0.7249}
```

Practical example

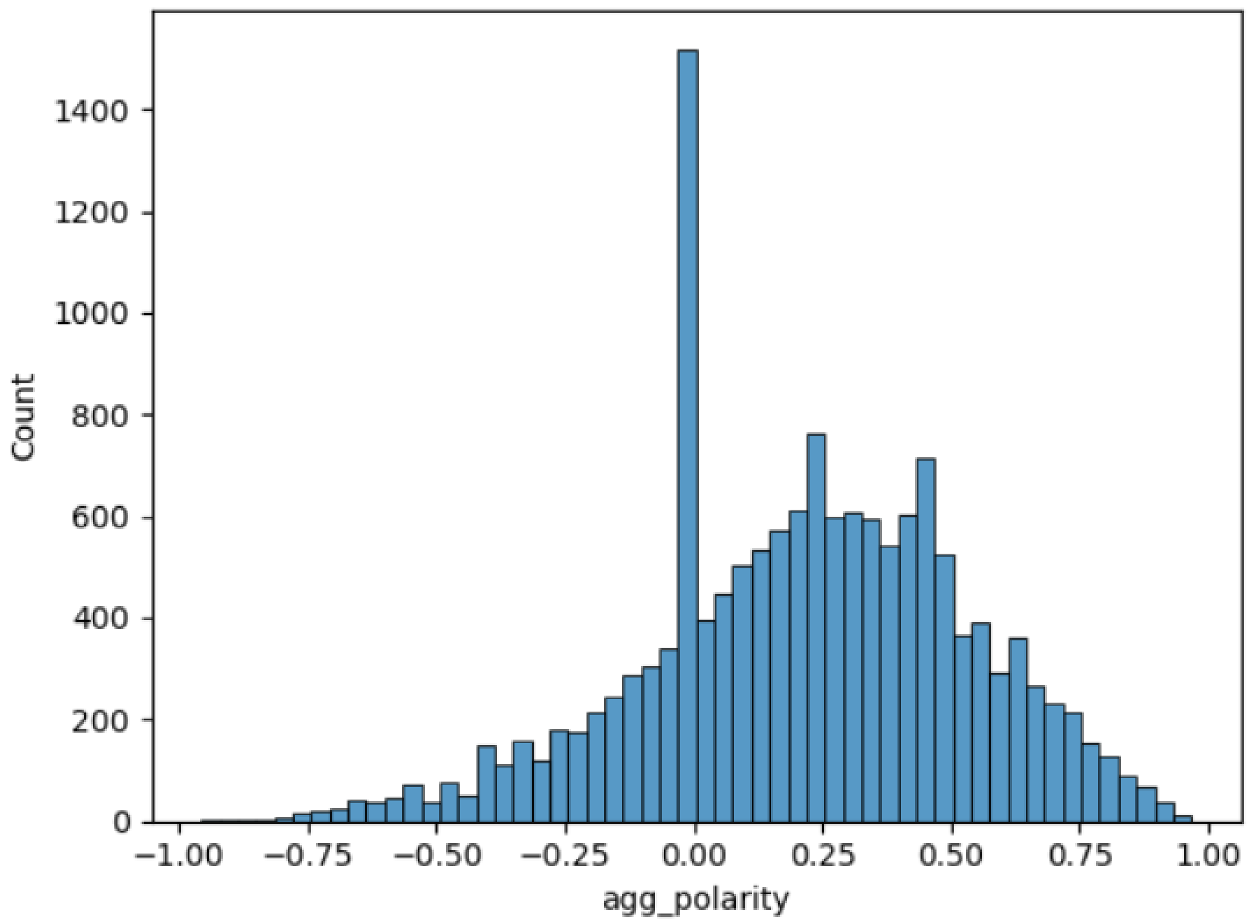
- Dataset: Course reviews at Univ. of Waterloo

- Steps to conduct sensitivity analysis
  1. Importing libraries (nltk, pandas, and seaborn) and loading dataset
  2. Preprocessing (remove missing values)
  3. Word frequency
  4. Generating sentiment scores (calculate sentiment scores for each sentence and average scores)
  5. Data wrangling

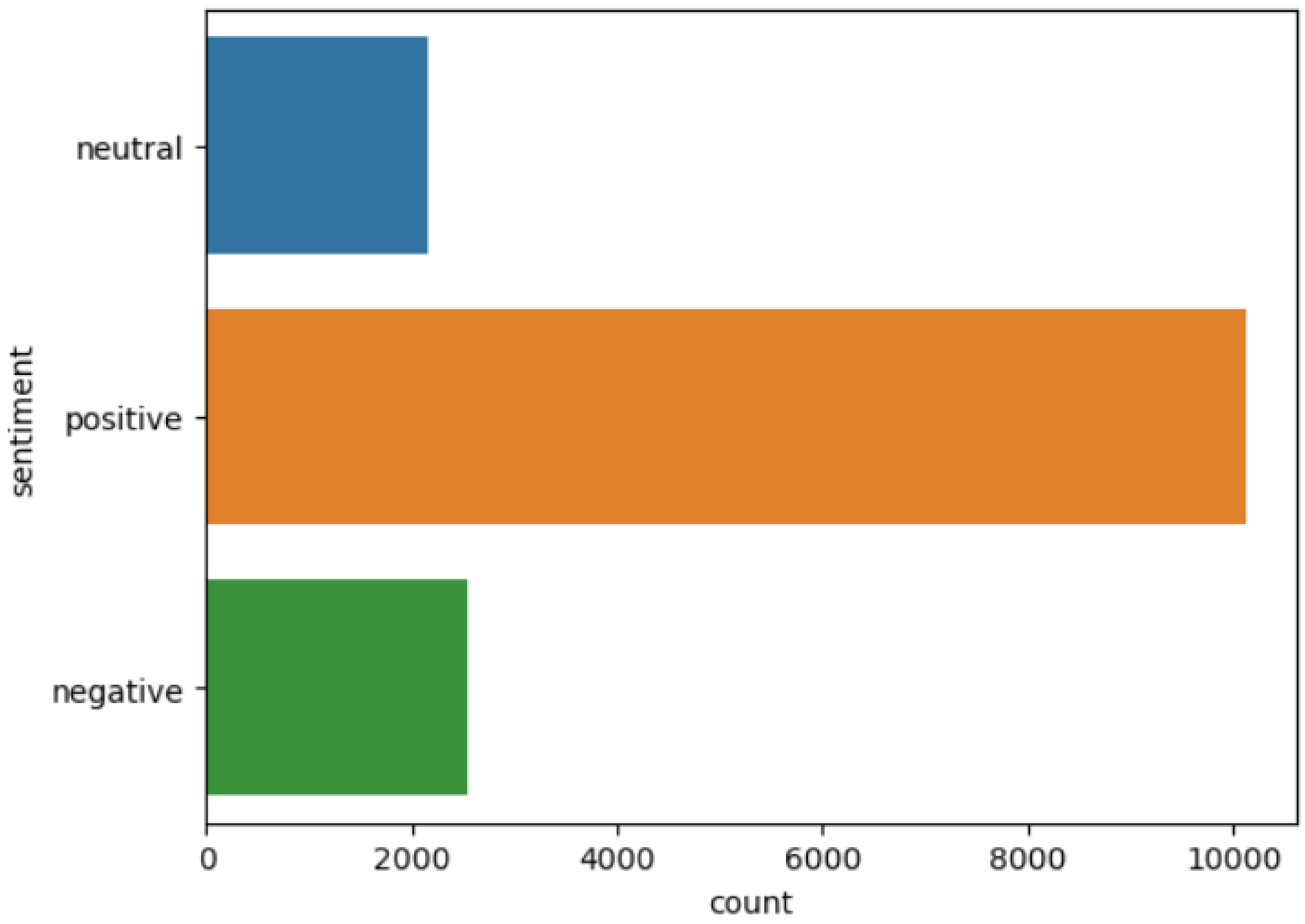
3. Word Frequency



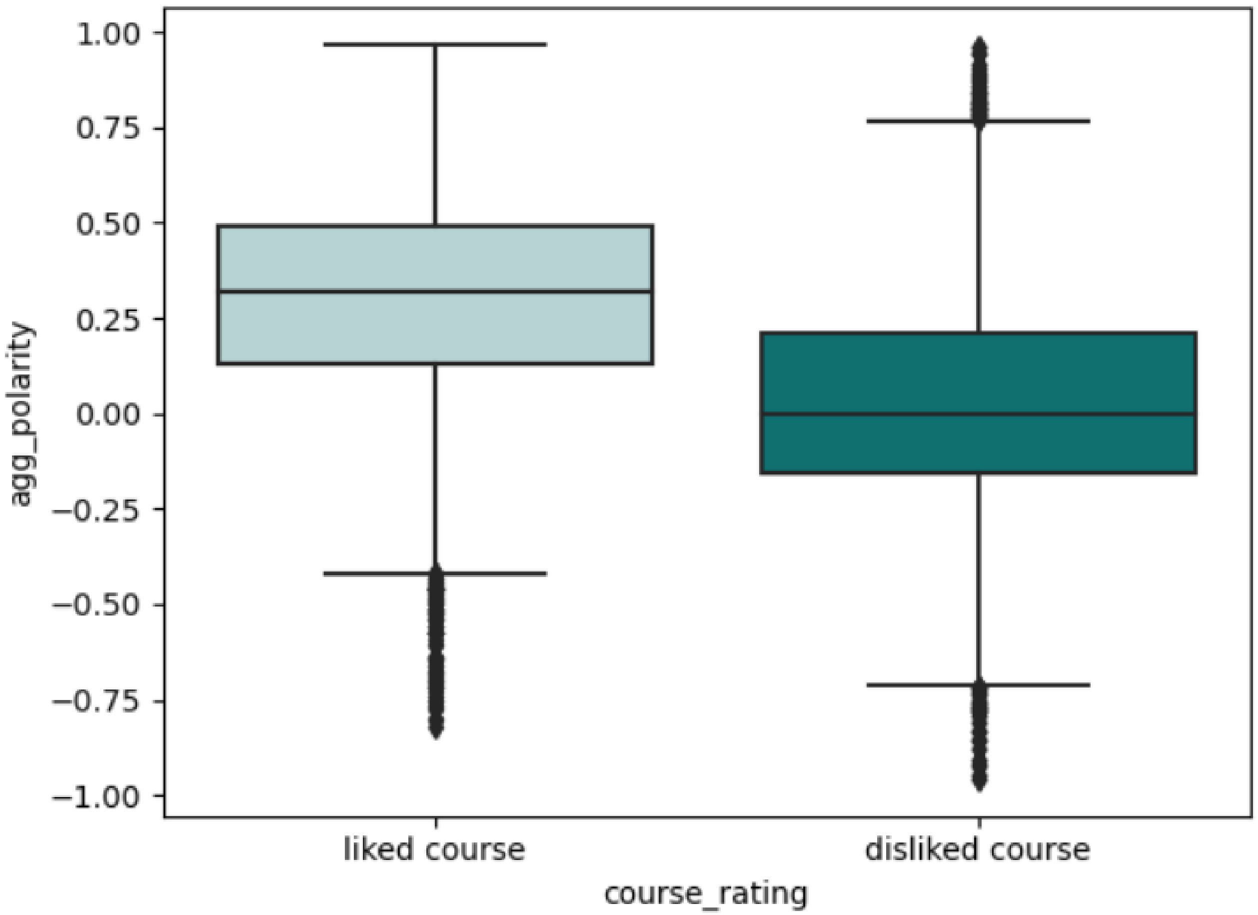
4. Sentiment scores



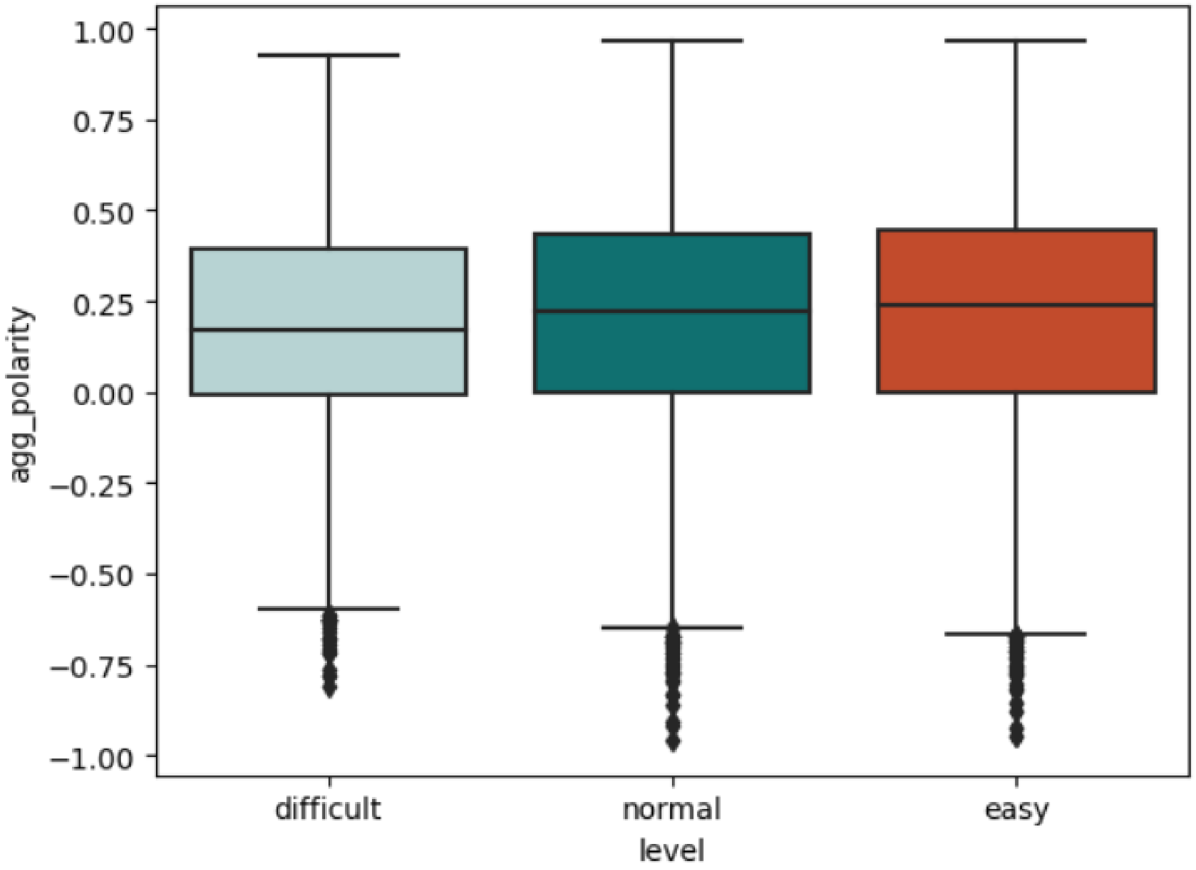
Sentiment polarity

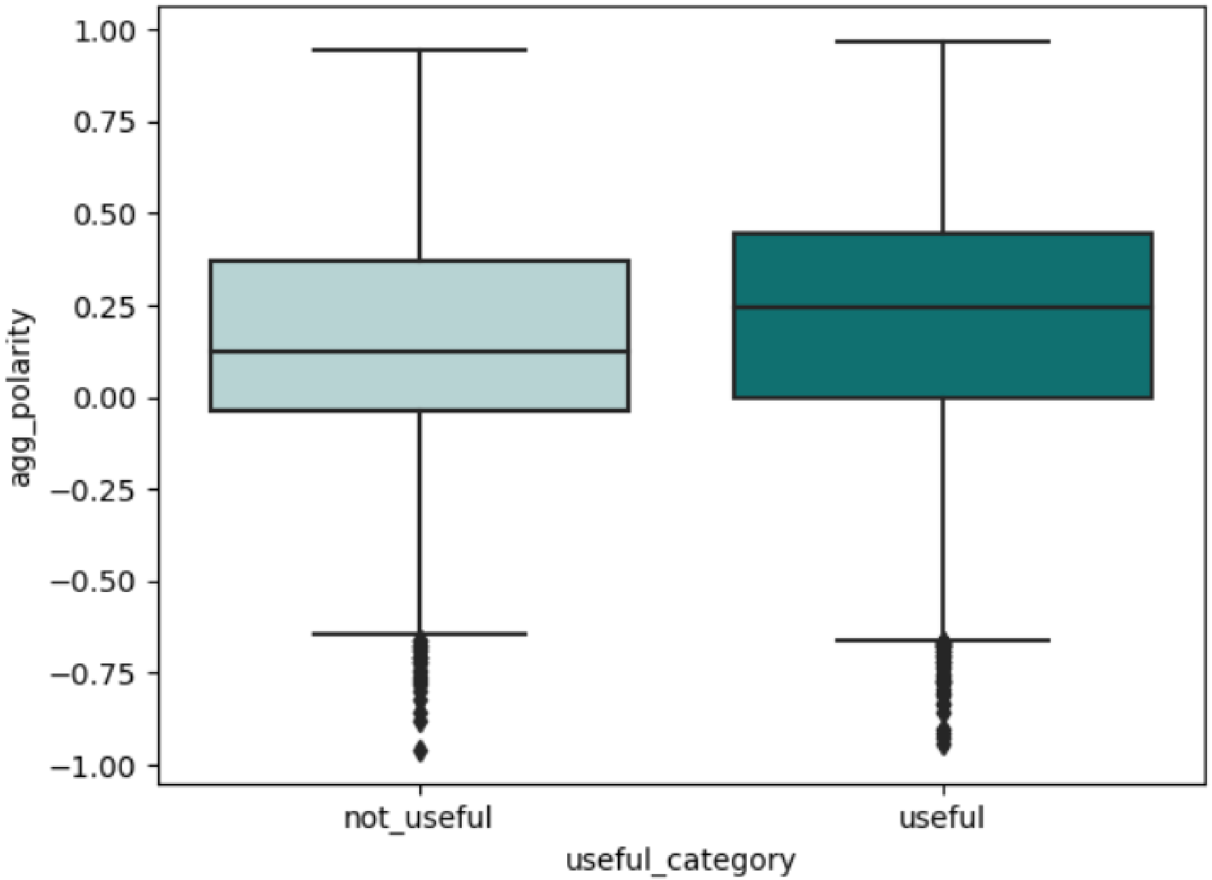


5. Data Wrangling: Sentiment scores by students' preference



Sentiment scores by the course level and usefulness





Is classification accurate?

- Sentiment scores=0.31. The student **liked** the course.

  "This course teaches proof techniques which I assume that will be used a lot in future math courses and ended with lots of calculation. It was neither hard nor hard but I learned how to manage time wisely (took some time to get used to first-year university math tho)."

- Sentiment scores=0.28, The student **disliked** the course.

  "One of my least favourite courses. Although things were nicely organized, Racket was such an annoying language to use. The one tangible benefit I felt after using Racket was feeling more comfortable with recursion."

Challenges and Solutions

- More accurate classification
  - If you have extensive training data in your domain and computational powers, ML-approach would be an option for improving the accuracy.
- Detect emotions
  - Emotion detection sentiment analysis (e.g., happiness, anger, sadness)

References

- Original Paper

Hutto, C., & Gilbert, E. (2014). VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. *Proceedings of the International AAAI Conference on Web and Social Media*, *8*(1), 216-225. https://doi.org/10.1609/icwsm.v8i1.14550

- Open-sourced VADER codes/files by the developer

https://github.com/cjhutto/vaderSentiment

- Walk-through examples of application of VADER and other sentiment-analysis methods

https://www.red-gate.com/simple-talk/development/data-science-development/sentiment-analysis-python/

https://realpython.com/python-nltk-sentiment-analysis/

https://blog.quantinsti.com/vader-sentiment/

https://www.analyticsvidhya.com/blog/2021/12/different-methods-for-calculating-sentiment-score-of-text/

- Overall picture of the sentiment analysis

Aggarwal, C. C. (2018). Opinion Mining and Sentiment Analysis. In C. C. Aggarwal (Ed.), *Machine Learning for Text* (pp. 413–434). Springer International Publishing. https://doi.org/10.1007/978-3-319-73531-3_1