DS 7995 – Project Report

Sentiment Analysis on Trace Evaluation Data

Name: Vivek Kandregula
NUID: 001344196

Advisor: Martin Schedlbauer
Term: Summer 2 (2021)

Contents

INTRODUCTION:

TRACE (Teacher Rating and Course Evaluation System) is a portal that is used by Northeastern University for performance evaluation. This portal is used to take the inputs from the undergraduate and graduate students to provide reviews regarding a professor's style of teaching and the course work. The evaluation system prompts students to provide a numeric rating (1-5) and a textual review regarding their opinion of a course and its professor. The numeric rating and textual review questions focus on questions to assess student's online experience, course work, learning experience, instructor teaching-related, and overall effectiveness.

TRACE website provides a lot of vital data that could be used to better understand the opinions of students about their experience at Northeastern University and this project explores one aspect of understanding and analyzing this data through sentiment analysis. Sentiment analysis is a natural language processing technique which is usually used to understand whether the data is positive, neutral or negative. As a part of this project, I would develop a model that is capable of categorizing the evaluations provided by the students regarding a professor or coursework into positive, neutral, and negative evaluations using lexicon/rule-based models and also supervised machine learning algorithms such as Logistic Regression and Decision Trees. Upon completing sentiment analysis, I would like to create a rule-based system that marks an evaluation as concerning if the evaluation consists of information regarding a possible violation of university policy.

PROJECT OBJECTIVES:

1. **Testing if the rating system reflects the genuine experience of the students by performing a comparative analysis of the ratings and the sentiments of the reviews left by the students.**
   Reason: Even though the quantitative rating system allows us to quickly get an impression about a course or a professor's style of teaching, it may fail to identify the actual feelings and emotions contained in the comments

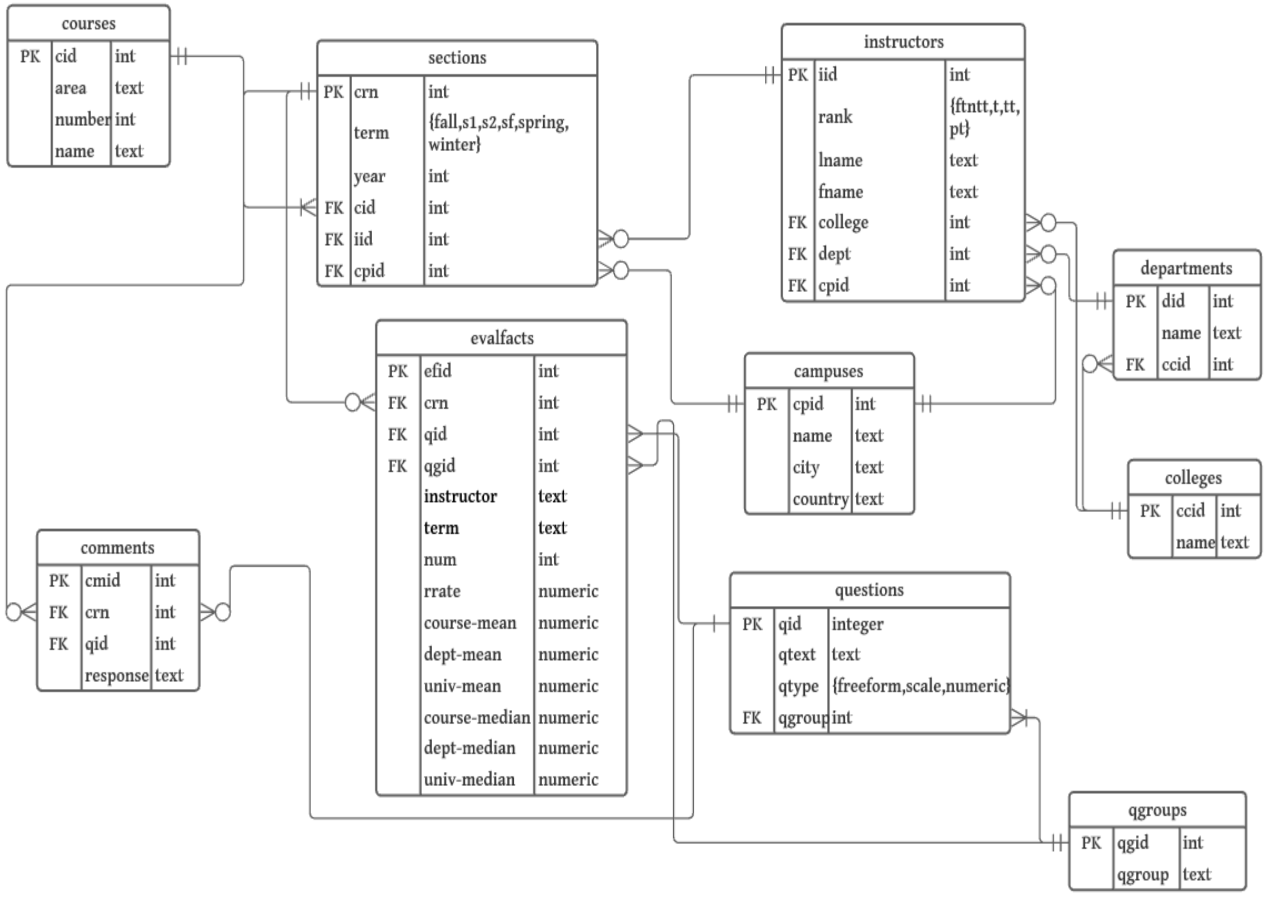
2. **Identify any mention of academic violation through the TRACE comments.**
   Reason: Many students might be hesitant to reach out to the appropriate authorities regarding any instance of misconduct by the professor and may choose to mention about it in the TRACE evaluations as the student's identity remains anonymous. Having a system in place to identify such concerning instances will provide an opportunity to take the necessary actions quicker.

Main obstacles of this project lie in developing an efficient automation script to collect all of the required data, identifying efficient data labelling approaches and identifying sentences with negation, sarcasm, terseness, language ambiguity during development of the sentiment analysis model.

DATA DESCRIPTION:

- The data consists of information about the course, section, department, college, campus, instructors, comments, evaluation ratings, questions and question groups for the spring 2021 and summer 1 2021 semesters. I decided to choose only the last two semesters to maintain consistency in the questions.

- The complete data consists of ~ 13000 reviews provided by students for 360 courses across Khoury College of Computer Science. To avoid data redundancy, we have created a normalized data model which is used as a blue print to ensure optimal data storage.



courses

PK	cid	int
	area	text
	number	int
	name	text

sections

PK	crn	int
	term	{fall,s1,s2,sf,spring, winter}
	year	int
FK	cid	int
FK	iid	int
FK	cpid	int

instructors

PK	iid	int
	rank	{ftntt,t,tt, pt}
	lname	text
	fname	text
FK	college	int
FK	dept	int
FK	cpid	int

departments

PK	did	int
	name	text
FK	ccid	int

evalfacts

PK	efid	int
FK	crn	int
FK	qid	int
FK	qgid	int
	instructor	text
	term	text
	num	int
	rrate	numeric
	course-mean	numeric
	dept-mean	numeric
	univ-mean	numeric
	course-median	numeric
	dept-median	numeric
	univ-median	numeric

campuses

PK	cpid	int
	name	text
	city	text
	country	text

colleges

PK	ccid	int
	name	text

comments

PK	cmid	int
FK	crn	int
FK	qid	int
	response	text

questions

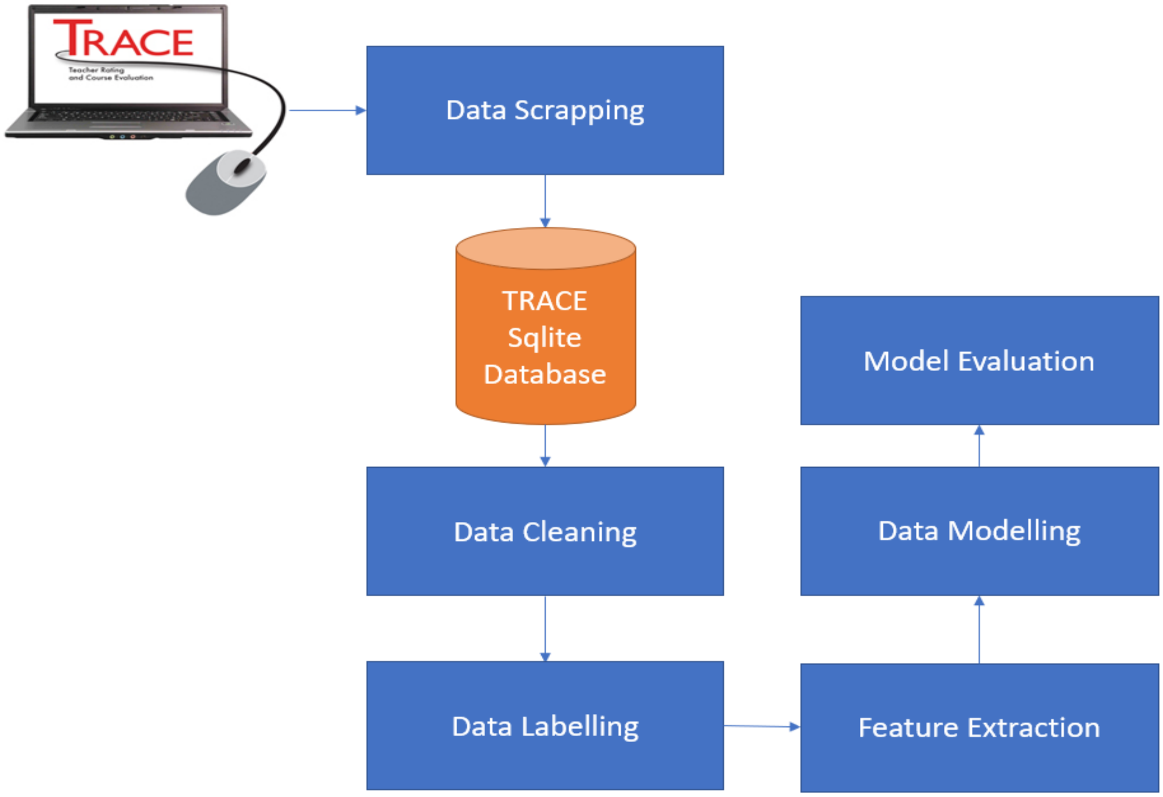
PK	qid	integer
	qtext	text
	qtype	{freeform,scale,numeric}
FK	qgroup	int

qgroups

PK	qgid	int
	qgroup	text

- Out of the 13k reviews and 24 questions, we have subset the data to obtain the evaluation ratings for the questions “What is your overall rating of this instructor's teaching effectiveness?” and “Please expand on the instructor’s strengths and areas of improvement in facilitating inclusive learning.” To ensure that the sentiment analysis model utilizes comments and ratings that solely focus on the instructors teaching effectiveness and not a combination of the student’s opinion on the course and professor. The final subset of the data consists of 2500 reviews

FLOW OF EXECUTION:



DATA COLLECTION:

- Since the TRACE portal has a multi-factor authentication security, it was not possible to perform web scraping. To get passed this security roadblock, we have utilized the option available on the TRACE portal UI to download all of the professor's overall reviews and ratings as a pdf.
- The automation script I developed using python is capable of creating necessary schema if it's not already created and automatically go through all of the pdfs available in a folder and scrape the required data from the pdf using pdfplumber.
- The scrapped information is sliced using generic python functions and stored in there respective tables present in the sqlite db.

Note:

I. Pdfplumber is a library available in python that is capable of analyzing pdf layouts and extract all of the text data available in the pdf.

II. SQLite is a self-contained, file-based SQL database. SQLite comes bundled with python and can be used in any of python applications without having to install any additional software.

DATA CLEANING:

The goal of data cleaning is to prepare the data to produce “clean text” (which is the human language just rearranged into a format that machine learning models can understand and analyze more accurately)

The reviews were cleaned following the below steps,

  - The first pre-processing step was to transform the questions data into lower case. This prevented having multiple copies of the same words
  - The next step is removal of punctuations, as it doesn’t add any extra information while treating text data. Therefore, all instances of it were removed to reduce the size of the training data
  - Removal of reviews that were mentioned as na, none, nope,n/a.
  - Removal of stop words from the text data. Stop words are commonly used words in a language that carry very little useful information. I utilized nltk.stopwords to identify the stop words and discarded them.
  - Removed non-alphabet strings that were present in the reviews.

- Tokenization of the text data
  - Performed Tokenization on the data, that is, dividing the text into a sequence of words or sentences
- Parts of speech tagging was performed before Lemmatizing the text data to ensure the correct root. For example, “running” and “ran” will output “run” if POS tags are assigned before lemmatization
- There are two approaches to reduce the text into inflectional forms: Stemming and Lemmatization
  - Stemming refers to the removal of suffices, like “ing”, “ly”, “s”, etc. by a simple rule-based approach
  - Lemmatization refers to converting the word into its root word, rather than just stripping the suffices
  - Lemmatization (post POS tagging) has been preferred over stemming. Because, stemming ignores context between words so it produces poor results.

Uncleaned Reviews	Cleaned Reviews
Sami is very warm and open about having difficult conversations. She was willing to share her perspective both as faculty and from her recollections of being a student.	sami warm open difficult conversation willing share perspective faculty recollection student
I could have planned out when I was doing work better. There were a lot of assignments for such a small amount of time	could plan work well lot assignment small amount time.

DATA LABELLING:

Data labelling in sentiment analysis is the process of determining whether the reviews sentiment is positive, negative or neutral. In order for supervised sentiment analysis models to work, we must initially label the text data.

There are two approaches to perform data labelling automatically using sentiment analysis tools:

- **VADER**: is a lexicon and rule-based sentiment analysis tool. It uses a list of lexical features (e.g., word) which are labeled as positive or negative according to their semantic orientation to calculate the text sentiment. Vader sentiment returns the probability of a given input sentence to be positive, negative, or neutral.
- **Textblob**: is another rule-based sentiment analysis tool. It returns two properties for a given input sentence (polarity and subjectivity). Polarity is a float that lies between [-1,1] where -1 indicates negative and 1 indicates positive and subjectivity is a float that lies in the range of [0,1]. Subjective sentences generally refer to opinion, emotion and judgement.

I chose VADER over Textblob because our reviews include a lot of slang terms. Textblob has been pretrained over formal language and VADER is frequently updated with the latest slangs since its primarily developed to work efficiently on social media data.

To ensure that all of the sentiments were labelled correctly, I manually verified the automated labels. By tuning the probability in polarities, I found that setting the reviews with a polarity greater than 0.25 as positive, less than 0 as negative and neutral if between 0 and 0.25 provided results similar to the labels obtained through manual data annotation.

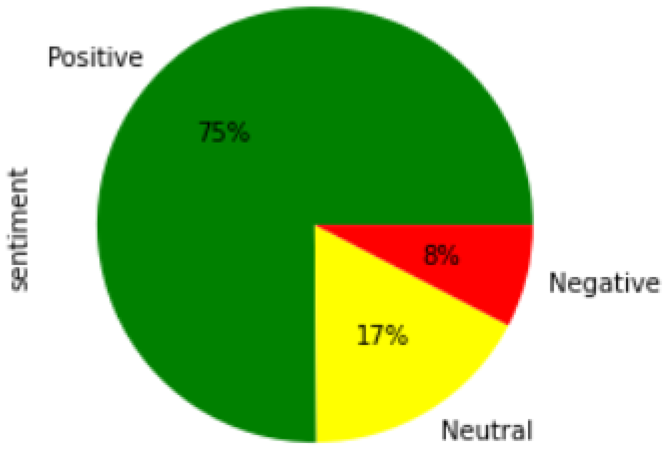


Fig. (1): Distribution of the sentiments labelled using VADER tool.

MOST FREQUENT WORDS:

Data visualizations (like charts, graphs, infographics, and more) give businesses a valuable way to communicate important information at a glance, but what for text-based dataset?

There are two approaches to find the most frequent words in the dataset:

- Counting occurrence of every word and representing them in a tabular form and arranging them in descending order
- Using word clouds

Word Cloud:

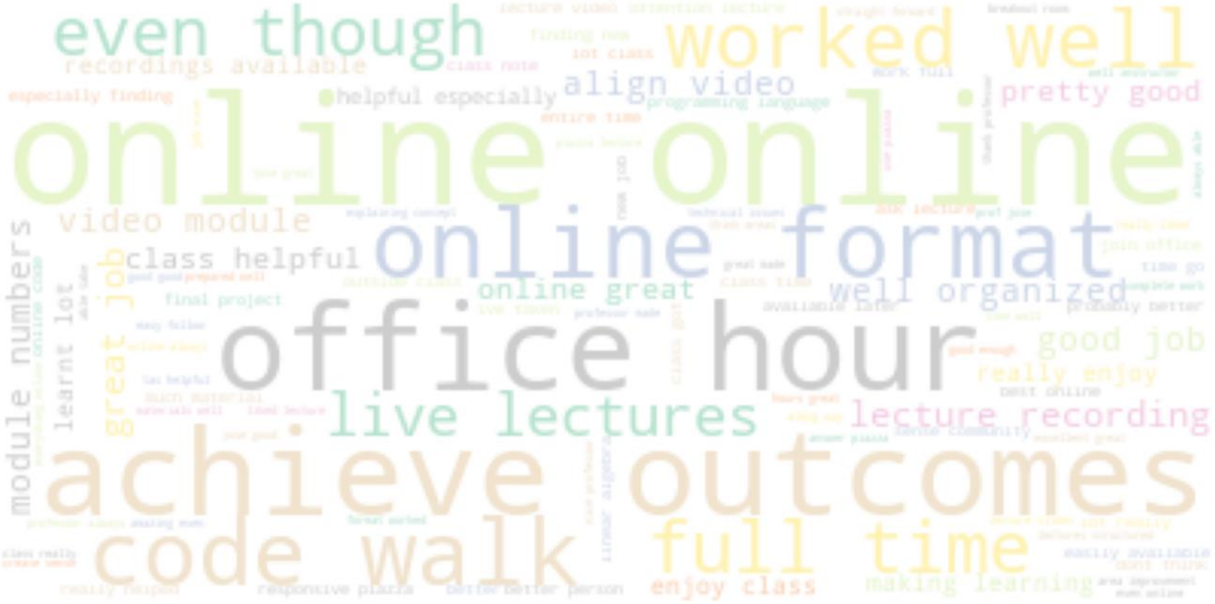
- Word clouds (also known as text clouds or tag clouds) work in a simple way: the more a specific word appears in a source of textual data, the bigger and bolder it appears in the word cloud

Word Cloud Functionality:

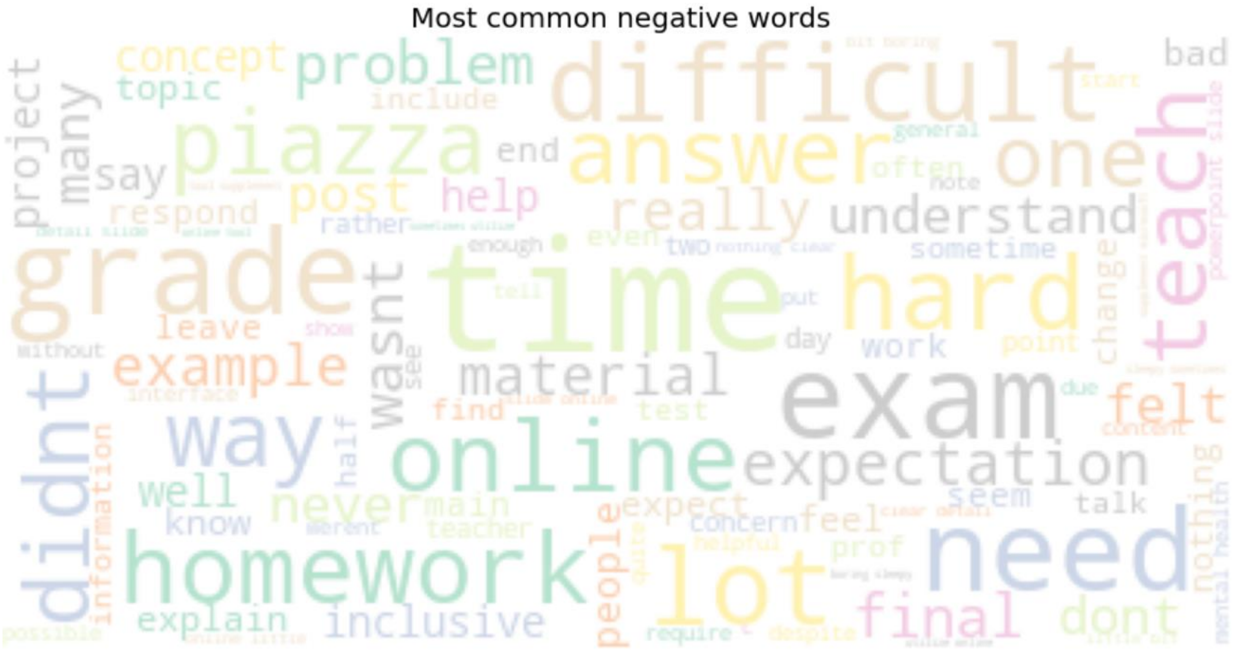
- For an intuitive visualization format that highlights important textual data points, using a word cloud can immediately convey important information about term frequencies.
- A word cloud is a collection, or cluster, of words depicted in different sizes. The bigger and bolder the word appears, the more often it’s mentioned within a given text and the more important it is.

Based on the word cloud for the words present in positive sentiment review, we can understand that the students have mentioned office hours, worked well, well organized, achieve outcomes and enjoy class more frequently

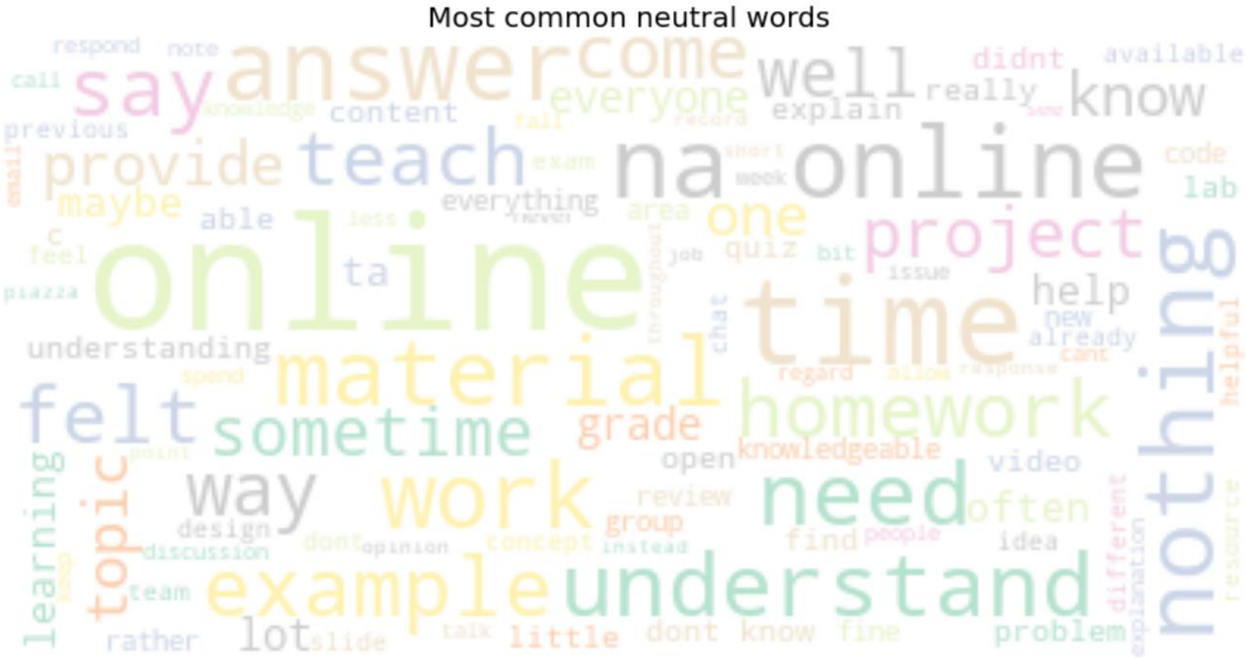
Most common positive words



Based on the word cloud present in the most common negative words. We can understand that the students have mentioned the words difficult, hard, time, homework, mental health, bad and grade more frequently.


Most common negative words

Based on the word cloud present in the most neutral words. We can understand that most of the highly frequent words do not add much value towards the sentiment.


Most common neutral words

FEATURE EXTRACTION:

Supervised machine learning algorithms use lot of mathematical computations on the data to train the data. So, it is necessary to convert the text data into a numeric form to be able to implement statistic-based models.

There are two primary approaches to perform feature extraction:

I. **Bag of Words:**
   - This is the simplest approach to convert textual data into numerical features.
   - All of the unique words are collected from all of the documents to create the vocabulary of the data.
   - Each document is converted into the size of the vocabulary and the frequency of the word in the document would replace the word in the vocabulary.

II. **TF-IDF (Term Frequency – Inverse Document Frequency):**
   - TF-IDF is a more complex approach with respect to the Bag of Words approach.
   - In TF-IDF all of the words that frequently occur in individual documents and less across all of the documents are given higher weightage.
   - The following formula shows how to calculate both the terms –

$$TFIDF\ score\ for\ term\ i\ in\ document\ j = TF(i,j) * IDF(i)$$

$$where$$

$$IDF = Inverse\ Document\ Frequency$$

$$TF = Term\ Frequency$$

$$TF(i,j) = \frac{\text{Term i frequency in document j}}{\text{Total words in document j}}$$

$$IDF(i) = \log_2\left(\frac{\text{Total documents}}{\text{documents with term i}}\right)$$

$$and$$

$$t = Term$$

$$j = Document$$

In this project, I have chosen to use TF-IDF over Bag of Words. Because in the Bag of Words approach all of the words with same frequency are given equal priority and it only provides information of the word frequencies. Whereas in the TF-IDF model we have information of the most important words in a document with respect to the complete corpus. Even though bag of words is easier to interpret, since we are directly using the word embeddings as an input for a machine learning algorithm interpretation is not the highest priority.

HANDLING CLASS IMBALANCE

From the pie chart Fig. (1), it is pretty clear that there is an imbalance in the number of reviews present in positive, neutral and negative classes. This inconsistency in label distribution is known as class imbalance. If class imbalance is not handled appropriately, most of the predictions will correspond to the majority class and the minority class may be treated as noise in the data. To avoid this, we use SMOTE which is a synthetic minority over sampling approach, this would increase the number of minority samples present in the data, bringing balance to the classes.

DATA MODELLING:

Even though we have already used a Lexicon based sentiment analysis approach to identify sentiments. I wanted to explore few supervised machine learning approaches. Rule-based approaches generally tend to have poor generalization and can only perform well in a narrow domain. So, as we continue to collect data and label them accurately. We would be able to achieve a classifier that is capable of avoiding overfitting due to generalization which is highly likely in rule-based approaches.

Since we now have the textual data in numeric form and have handled the class imbalance, we can start training machine learning models. In this project, I have chosen to implement two supervised machine learning models (logistic regression and decision trees) and perform a comparative study on the results based on their evaluation metrics.

Model 1: Logistic Regression takes in a list of features as input and outputs the Sigmoid of a linear combination of features weighted by learned parameters.

Model 2: Decision tree is a flowchart-like structure in which each internal node represents a test on a feature. It one of the most popular machine learning algorithms used. Decision tress are relatively simple to understand and make some good interpretations.

EVALUATION METRICS

Both of the machine learning models have performed similarly with logistic regression showing slightly better results across all of the evaluation metrics.

Machine Learning Model	Accuracy (in %)	Error (in %)	Precision (in %)	Recall (in %)	F1 Score (in %)
Logistic Regression	80.9	19.1	79.43	80.90	78.9
Decision Tree	78.65	21	78.77	78.65	78.33

DO RATINGS DO A GOOD JOB AT REPRESENTING STUDENTS' TRUE OPINION?

To check if the ratings are a good representative of the student reviews. We must check if the majority sentiment provided to a professor for a particular course is relevant to the rating provided to the professor.

In an ideal scenario, I would join each student's instructor rating with their reviews. But, in our case, since we only have access to the overall course mean and not the individual ratings provided by each student to a professor. I calculated the majority sentiment based on all reviews provided by the students to a professor and assigned it to its respective CRN.

- The average course mean for an instructor that has a majority positive sentiment is 4.44
- The average course mean for an instructor that has a majority negative sentiment is 3.91.

The graphs below represent the distribution of ratings for professor with a majority of positive sentiments (Fig. 2. a) and the distribution of ratings for professor with a majority of negative sentiments (Fig. 2. b).

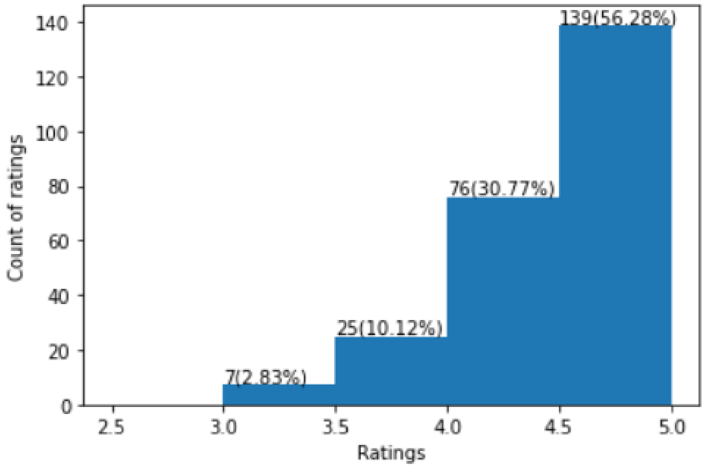


Fig. 2. a

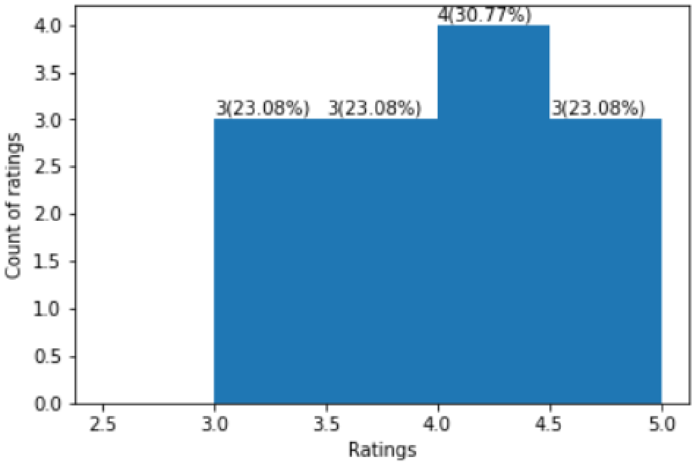


Fig. 2.b

- From the graphs above we can see that the professors with majority positive sentiment reviews have majority ratings (~57%) between and 4.5 and 5.0 and professors with majority negative sentiment reviews have majority ratings (~31%) between 4.0 and 4.5.
- Professors with majority positive sentiment reviews have only ~13% ratings less than 4.0 and professors with majority negative sentiment reviews have ~46% ratings less than 4.0.

Even though the above graphs represent that the professors with majority positive sentiment have higher ratings than professors with majority negative sentiment and show clear discrepancy between course means as well. We cannot confidently come to the conclusion that ratings do a good job at representing the student's true opinion due to the smaller number of professors with majority negative sentiment compared to the professors with majority positive sentiment.

IDENTIFYING VIOLATION OF UNIVERSITY POLICIES:

To ensure that students are getting education in a safe and nurturing environment, it is extremely important to ensure that the instructors are thoroughly following all of the university policies. The agenda of this section is to create a system to identify and report any comments that may indicate a violation of the university policy by a professor.

Since, the data that we currently have in place does not consist of any examples of comments that suggest violation of university policies. I have created a list of identifier words that could be used to identify if a comment suggests any type of violation of university policy and mark a Boolean feature as 1 if violation is present and 0 if violation is not present.

FUTURE SCOPE:

- Once we have access to more data with a consistent questionnaire or individual ratings, we could alleviate the issue of class imbalance and also produce more confident results in identifying the extent of consistency between the ratings and the reviews.
- Utilize state-of-art word embeddings that are capable of understanding the grammar and context of the reviews. This could help obtain better results in our supervised sentiment analysis models.
- Development of a dashboard, that produces graphs based on the course and professor. This type of visualization is important to understand the general trends in students' opinions.
- Since none of the data available in TRACE as of now indicates violation of university policy. Data indicating sexual harassment, discrimination and lack of professionalism in an academic environment can be collected from social media platforms or other online professor rating websites. Having this data would allow us to develop a classifier that is capable of identifying such violations.

REFERENCES:

- TRACE: https://registrar.northeastern.edu/article/faculty-class-evaluations-trace/
- VADER vs TextBlob: https://neptune.ai/blog/sentiment-analysis-python-textblob-vs-vader-vs-flair?utm_source=datacamp&utm_medium=post&utm_campaign=blog-sentiment-analysis-python-textblob-vs-vader-vs-flair
- Decision Tree: https://medium.com/deep-math-machine-learning-ai/chapter-4-decision-trees-algorithms-b93975f7a1f1
- SMOTE: https://www.datacamp.com/community/tutorials/diving-deep-imbalanced-data