Group 6 – Sentiment analysis tool comparison using Vader and SentiStrength

Miika Pernu (team lead)  
University of Oulu  
miika.pernu@student.oulu.fi

Mikko Roimaa  
University of Oulu  
mikko.roimaa@student.oulu.fi

Pilvi Tunturi  
University of Oulu  
pilvi.tunturi@student.oulu.fi

*Abstract*—This project aims to investigate the sentiment polarity (positive, negative, neutral) and intensity of hotel reviews and compare it to user given ratings. Correlation between the human ratings and sentiment intensity by the tool can be regarded as evidence that the analysis is correct. The interesting part is not that much sentiment itself, but rather argumentation to the polarity, ambiguity of reviews and what entails to that and e.g. the impact of named entities. This project will calculate Pearson’s correlation coefficient values with user rating, SentiStrength and NLTK Vader sentiment analysis results. This project will also analyze what kind of categories relate to each review and how do they correlate to positive or negative sentiments. It will also investigate what kind of named entities each review has and how do their presence correlate to the positive or negative sentiments. During this study, several hypotheses will be tested: Presence of positive/negative sentiment associated Empath category in the review entails a positive/negative sentiment, presence of a given type of named entity entails positive sentiment/negative sentiment, negative sentiment entails more argumentation, badly written reviews are likely to be included in ambiguous class, ambiguous reviews are shorter, ambiguous reviews have bad readability. All of the steps and the comparison of how well these tools can correlate the sentiments of user’s reviews is done by programming an application that can be ran step by step by using simple GUI. Repository can be found at <https://github.com/MipedD/521158S-NLP-project>. And the prebuild binaries for the graphical user interface can be found at:

Keywords—SentiStrength, Vader, sentiment analysis, opinion mining

# Introduction

With web, social media and digitalization the growth in user generated content has opened a huge opportunity for businesses to find new ways to generate value from data. Since the early 2000s, opinion mining or sentiment analysis has become one of the most researched area in natural language processing. [[1](#_References), [2](#_References)] As sentiment be an attitude or a thought as a response to a feeling [[1](#_References)], it is interesting in context like customer satisfaction and NPS surveys. However, the more interesting are the facts related to the sentiments [[10](#_References)]. The nature of sentiment analysis is interdisciplinary. It can be approached from computer science point of view, social science and for example management science point of views [[2](#_References)].

This study investigates opinion mining or sentiment analysis mainly from the lexicon-based tooling perspective. Similar studies have been made with these and other lexicon-based tools with similar and different datasets. [[6](#_References),[7](#_References)] However, neither of these studies investigate argumentation as such, which is important for any business to draw conclusions related to their products, services and customer feedback.

For businesses nowadays it is not difficult to start analyzing sentiments of their customers in their social media channels, customer support and service channels or customer reviews and feedback, especially for English. The challenges with tooling can be many though. When choosing the tool, it is important to understand the purpose. [[3](#_References), [4](#_References)] Using sentiment analysis for social media posts requires different tool than running a tool used for news analysis [[4](#_References)].

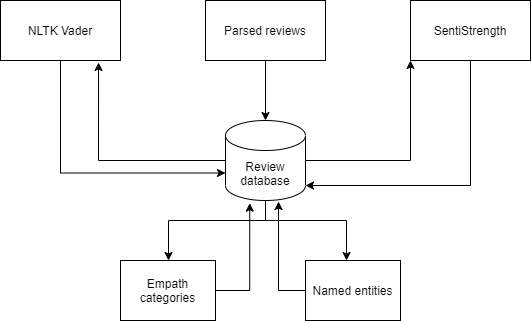
One must also understand other parts, such as developer community, programming language, portability and other aspects typically evaluated when buying similar tools. It is also important to understand and consider the usage patterns of the tooling. Is it going to be used off the shelf with standard configuration, or is it going to be trained with data.[[3](#_References)] There are similar studies, that have compared Vader and SentiStrength approach [[6](#_References),[7](#_References)]. Study by Al-Shabi [[7](#_References)] analyzed five most important lexicon-based sentiment analysis tools, including Vader and SentiStrength. The study revealed that classification with Vader was more accurate among negative and positive sentiments. The study was conducted with Twitter data. This study will be conducted using two tools, Vader and SentiStrength with data from Kaggle containing ten thousand hotel reviews in English. The data will be used as is without largely preprocessing it. Study by Zibran [[6](#_References)] analyzed four lexicon-based tools, including SentiStrength and Vader. The study used domain specific dictionaries to improve on the accuracy. The performance of the tools in dedicated or special domains such as Software Engineering was known to be less accurate. The study concluded that lexicon-based tools outperformed tools that incorporate complex techniques e.g. with subjectivity or contextuality. However, this study did not conclude a clear winner, but SentiStrength was seen performing better with negative sentiments.

An empirical analysis by Singha et al [[5](#_References)] showed that there is a high correlation between customer ratings and sentiments. These types of findings set a stage for our study too. The study by Thelwall et al also shows that SentiStrength is performant enough with different types of social web texts [[9](#_References)]. But the argumentation analysis is needed as a step forward from sentiment analysis to determine the impact of a certain review [[10](#_References)].

As it is clear, sentiment analysis can be ran by using different kind of tools. As a research problem, this is not unique or novel at all. This study aims to learn the pros and cons of the tools used by concentrating on two tools mainly, SentiStrength and Vader. Secondly, the aim of the study is to educate and use and expand the learned skills in practice.

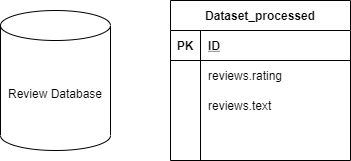
# Methodology

This project study concentrated on English hotel reviews data from Kaggle.com. The analysis flow can be illustrated with following diagram:



1. Analysis flow

The project started by downloading the data and manually inspecting it. It became clear that dataset contained hotel responses within the reviews, but a way to clean and process those ten thousand reviews could not be found. Hotel reviews in nature are such that they can be longer or shorter and there was not a common pattern to be found for the cleaning process. It was acknowledged that the impact of those responses within data to be analyzed may make the analysis less trustworthy. The study commences that the data collection process has been erroneous, but the amount of erroneous reviews is not high and the impact is not seen as critical. Three different datasets were investigated and the one with more variance in reviews text and rating was chosen, because it is easier to analyze readability and whether a review is considered badly written. Finally, the data was cleaned up by removing unnecessary columns to speed up the process. Reviews.text and reviews.ratings were taken as is and also review.id was constructed. With this step, encoding was ensured to be utf-8 throughout the project.

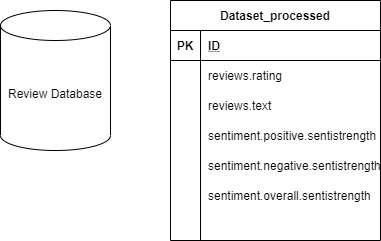


1. Data preparation

The next step in the process was to run data through SentiStrength.

The SentiStrength dictionary is constructed by combining LIWC and GI dictionaries similar to VADER, and also includes lists of emoticons, negations and intensifiers. SentiStrength is a sentiment analysis tool that performs with human level accuracy in English social media texts [8]. It is lexicon-based, designed to give a strength to a term. For example, “love” has a stronger positivity than “like” [[9](#_References)].

Data was fed into SentiStrength java client, that was seen more performant than the Windows client. In this task each review was run through SentiStrength. Data was first processed into a format which is recognized by SentiStrength (tab separated CSV) after which SentiStrength assessed the sentiment polarity for each review individually and wrote it to the file parsed for SentiStrength. After analysis was completed the input file for SentiStrength was read and combined with the main database, overall sentiment was calculated as additional step by calculating the sum of positive and negative sentiment values.

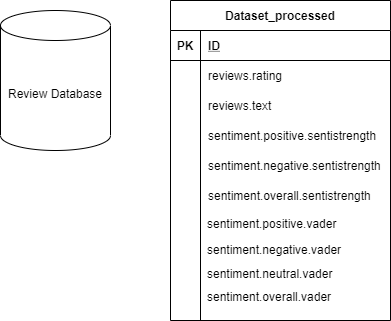


1. Running data through SentiStrength

We also ran the data through AWS Comprehend, which is a tool offered by Amazon to find insights and relationships from data using machine learning techniques. Setting up AWS comprehend to analyze the dataset was easy, but due to the cost it was decided not to be used in this analysis. While the cost seemed bearable, the unpredictability of the billing rules was a possible threat to the project.

To replace AWS Comprehend, we decided to use NLTK Vader. Vader by Gilbert et al [11] in their study present and evaluate Vader, Valence Aware Dictionary and sEntiment Reasoner. Vader is a simple lexicon and rule-based model for sentiment analysis. It is specifically attuned to polarity and intensity of sentiment expressed in social media texts. It works well on texts from other domains too [[12](#_References)]. The study by Gilbert et al [[11](#_References)] revealed that based on correlation coefficient, Vader performs and even outperforms individual human raters in classifying polarity. It is proven that Vader sentiment lexicon is gold-standard quality and has been validated by humans.

This task was almost identical with previous one, except that the results were written into the database file directly.



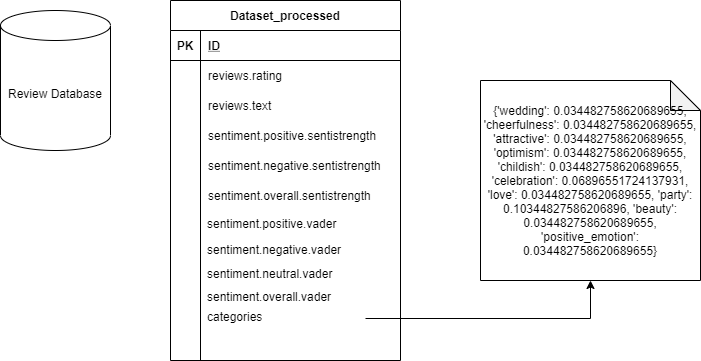
1. Running data through Vader

With this data, the Vader and SentiStrength results were plotted in the same graph along with the actual reviews. The data was arranged by the rating. As the ratings were all in different scale, the results from both analyzers were normalized along with the actual reviews to get a more meaningful graph. The graph contains all 10000 points of the dataset.

Additionally Pearson coefficient correlation was calculated for both analyzer results in relation to the user ratings.

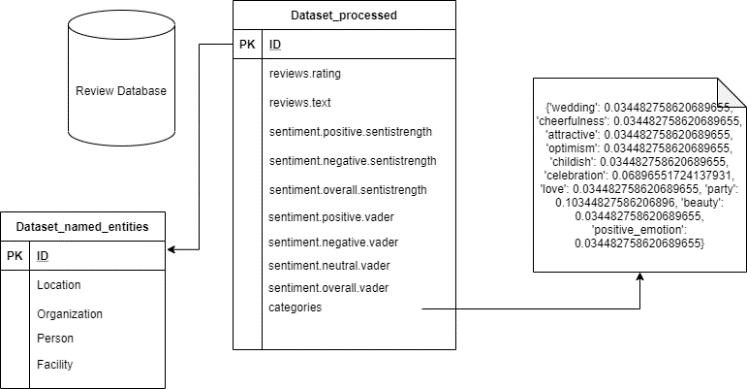
For the preprocessed, parsed and analyzed data, the process of enriching it with lexical categories was started. Empath tool was the decided tool for it. Empath is a lexicon mined from modern texts. It groups words into topics, and is human validated. Empath uses the combination of machine learning and crowd sourcing. Compared to the well-established LIWC categories, Empath is much wider. It contains hundreds of categories compared to the tens of categories of LIWC. And as our language evolves, so does Empath. [[14](#_References)]

Empath categories were constructed by Python library and written into the reviews database. Additionally, each unique category was extracted and saved into empath\_categories.txt separately and grouped into positive, negative or neutral according to VADER. Categories were stored as key-value pairs into the Review database.



1. Adding Empath categories

Next step was to include named entity categories in each reviews into the database. The problem observed with NLTK named entities was that it’s recognition capabilities are limited. It recognized capitalized nouns as “persons”. The places, like Best Western, were recognized as two different entities, one as a name (Best), other as an organization (Western). Binary format was chosen to show whether the entity belonging to an entity category was found in the review. The potential correlation between presence of given category and sentiment polarity was analyzed.



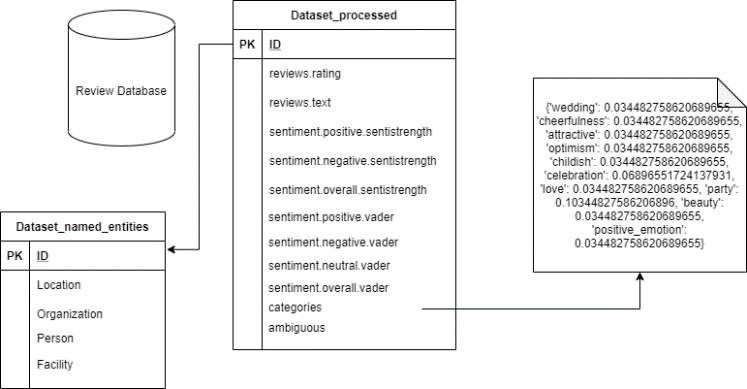
1. Adding named entities

Next step of the project was to test the hypothesis that negative reviews entail argumentation. To be able to accomplish that, a short list of explanation inducing expressions was collected and put together in a csv file. Each review in the dataset was tested for the number of these expressions found. It is notable that reviews were in the scale of 1-5, positive are >=4 and negative are <=2. Reviews with rating 3 were considered neutral.

Next steps were about the ambiguity of the review.

Resolving ambiguity is one of the biggest problems in NLP. One can see ambiguity in a sentence, if it can be interpreted in two or more ways. [[16](#_References)]

In this process, each review was split into one of two classes: ambiguous or non-ambiguous. Whether a review belonged in the ambiguous class was determined by whether sentiment analyser VADER result has significant deviation from the users own rating. Additionally in task 10 it was tested whether reviews in ambiguous class were likely to be badly written. Whether a review is badly written is determined by the percentage of known words in the review. A word is considered known if WordNet is able to find any synsets for the word. A word without synsets is considered unknown.



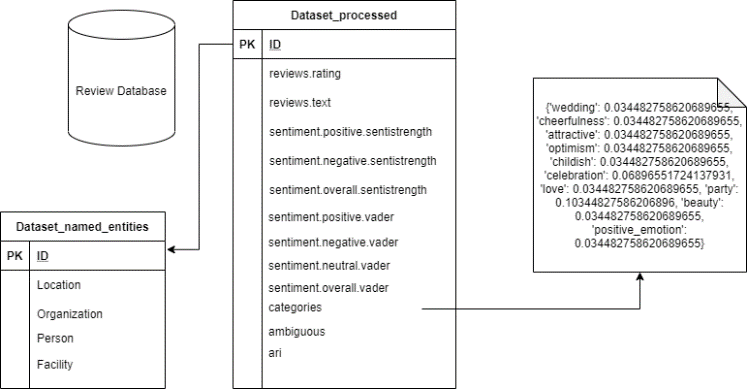
1. Adding ambiguousness value

Finally, it was checked whether ambiguous reviews were likely to be shorter than others.

After that, the next goal was to test the following hypothesis: ambiguous reviews have bad readability. This hypothesis was tested by calculating the Automated Readability Index (ARI).

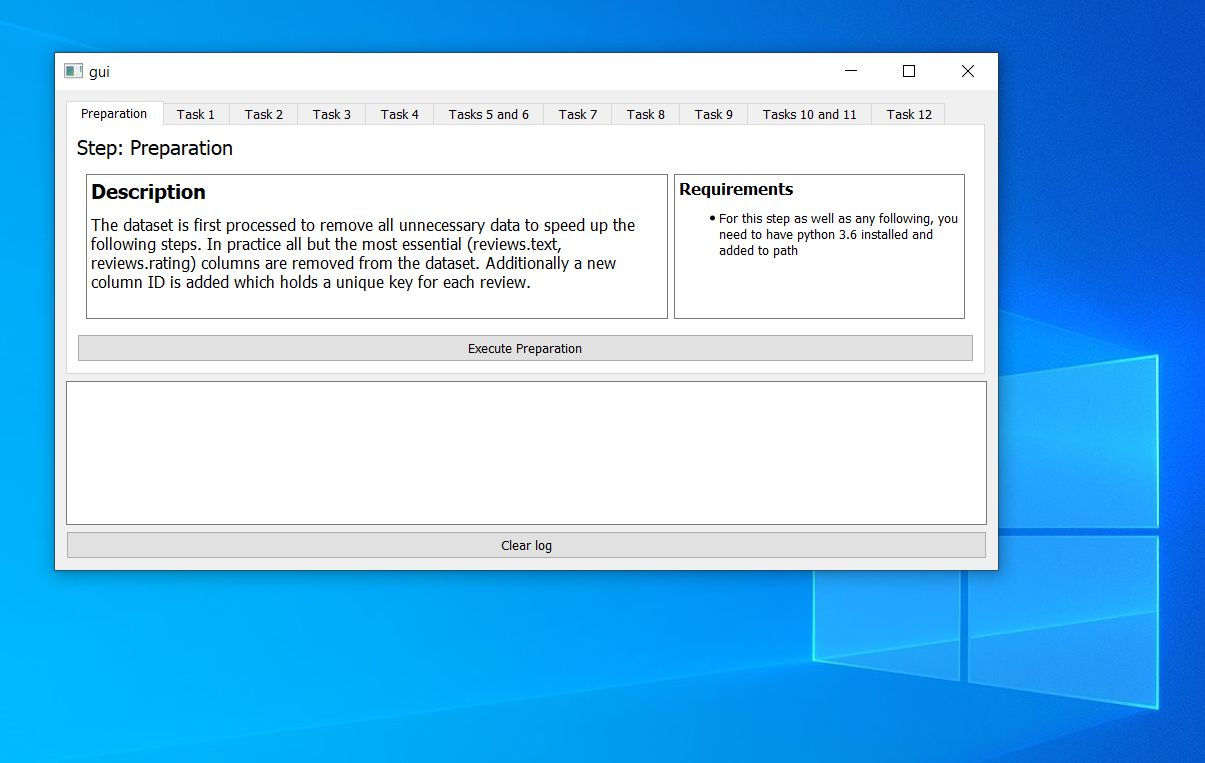
ARI is an index designed to measure understandability of English text. It represents approximately a grade level that is needed to comprehend certain text. [[15](#_References)]

ARI calculation was ran for each review to see which class (ambiguous vs non-ambiguous) has the larger value by average. The results were printed into the application output panel. Additionally, the ARI value was written into the database.



1. Adding automated readability index (ARI) value

All of the above steps were packaged into the simple graphical user interface (GUI), that was built using QT. User can run the application with simple step by step user interface, that also guides user through the process. The results were saved into review database and results were printed into the output console.



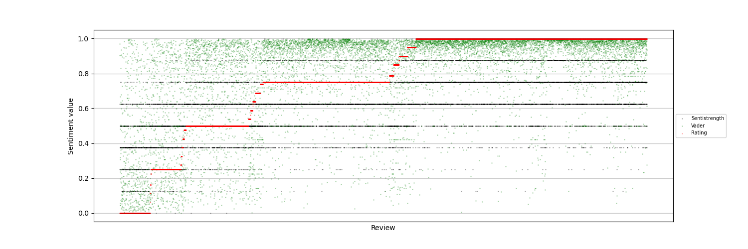
1. Application GUI

# Results

In this chapter, the hypotheses will be walked through, tested and compared with the evidence found from the literature review.

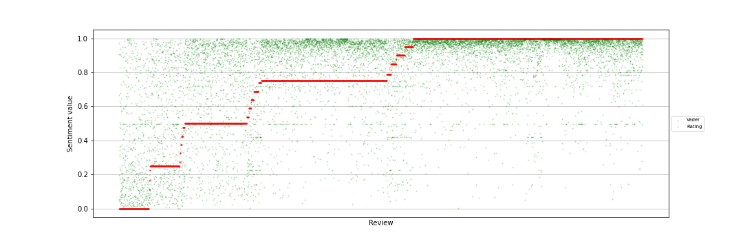
## Plotting the data from sentiment analysis tools

Finding a correlation from big chunks of data can be done using scatterplots. We plotted data from both analyzers into the same graph along with the actual user ratings. The data was sorted by the rating. As the ratings were all in different scale, the results from both analyzers were normalized along with the actual reviews to get a more meaningful graph. The graph contains all 10000 points of the dataset.

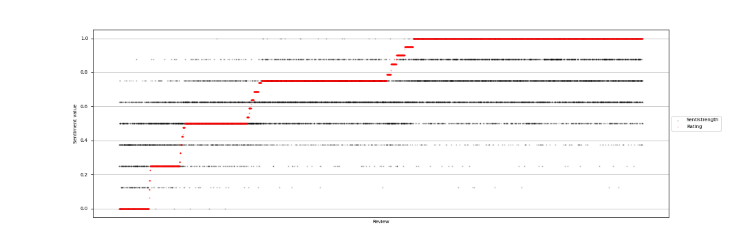


1. User, SentiStrength and Vader correlation graph

The ten thousand reviews dataset is hard to illustrate in one diagram. Thus, the data was separated per analyzer. The same order was retained. Figure below shows that with Vader, positivity and negativity are best correlated. The correlation in more neutral sentiments is more scattered and not easy to recognize. With SentiStrength, the correlation in negative sentiments is more accurate. This supports also the conclusions made by other studies [[6](#_References),[7](#_References)]. Pearson correlation coefficient was calculated for Vader (0.571814), which indicates that sentiment analyzer output correlates positively with user rating. Similar is true with SentiStrength, while it was marginally smaller (0.556305) compared to VADER. That indicates lesser correlation, but still positive correlation with user reviews.



1. User and Vader rating correlation graph



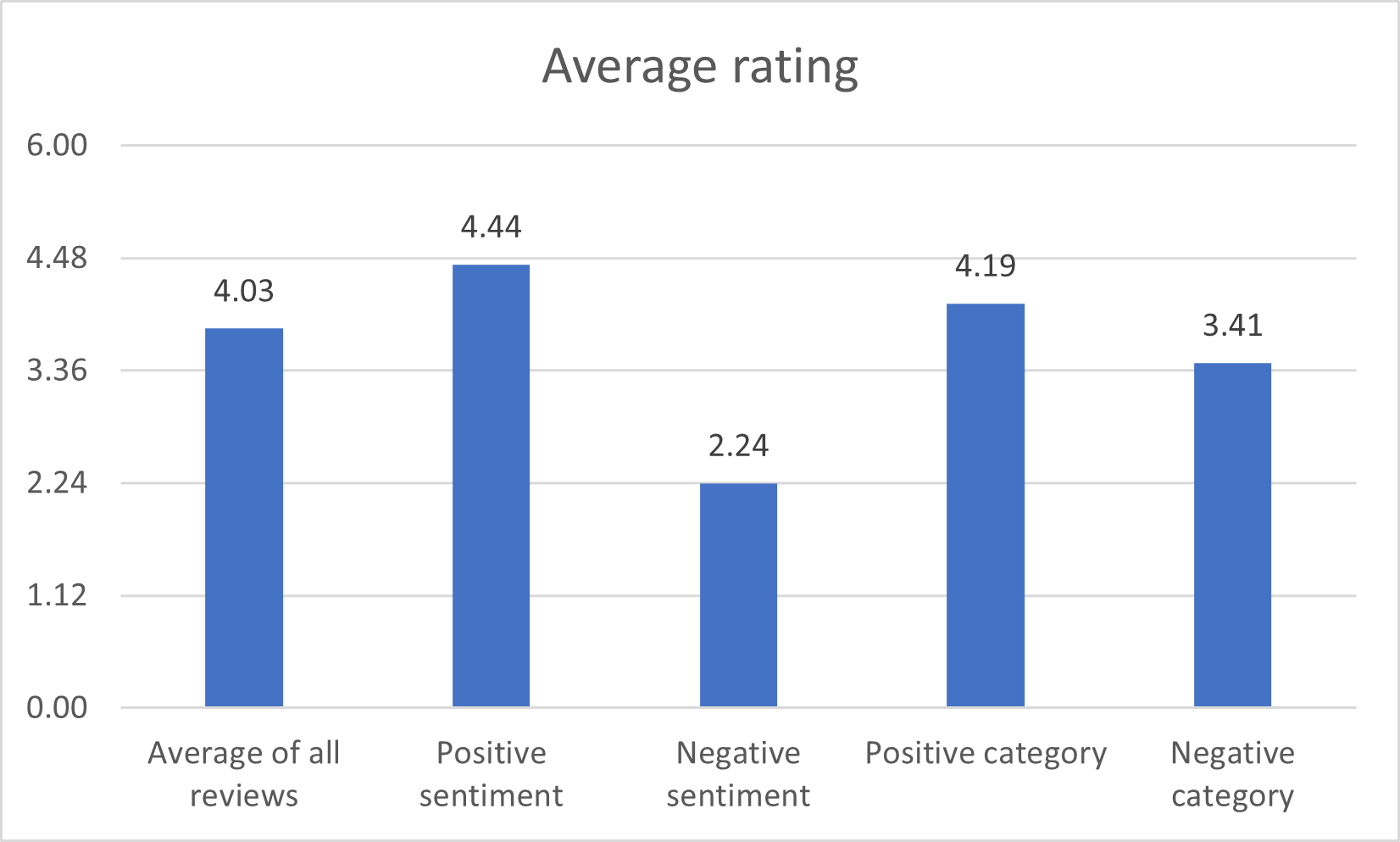
1. User and SentiStrength rating correlation graph

## Hypothesis 1a: “The presence of negative sentiment associated Empath category in the review entails a negative sentiment”

## Hypothesis 1b: “The presence of positive sentiment associated Empath category in the review entails a positive sentiment”.

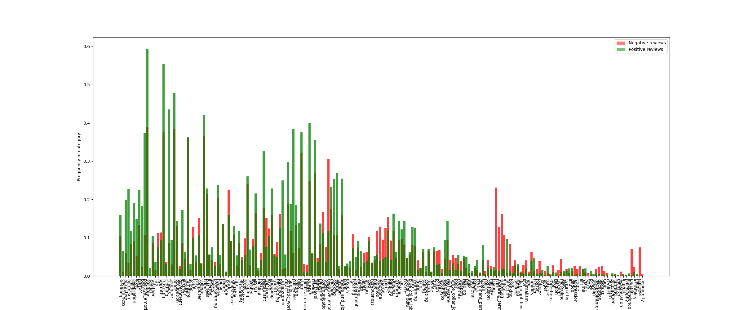
When testing this hypothesis, we compared the average user ratings of reviews with dominant negative or positive empath categories with dataset overall average user ratings. Positive sentiment is the intersection of reviews, which both SentiStrength and Vader analyzed positively. The same applies for negative.

Positive category means review with dominant positive categories. Similar is true for negative. In the below diagram, it is visible that presence of positive category entails positive sentiment, and the presence of negative category entails negative sentiment when compared to average of all reviews. In this light, hypothesis 1a and 1b are true. However, the results are not as conclusive as the results of sentiment analysis.



1. Negative sentiment vs. negative category

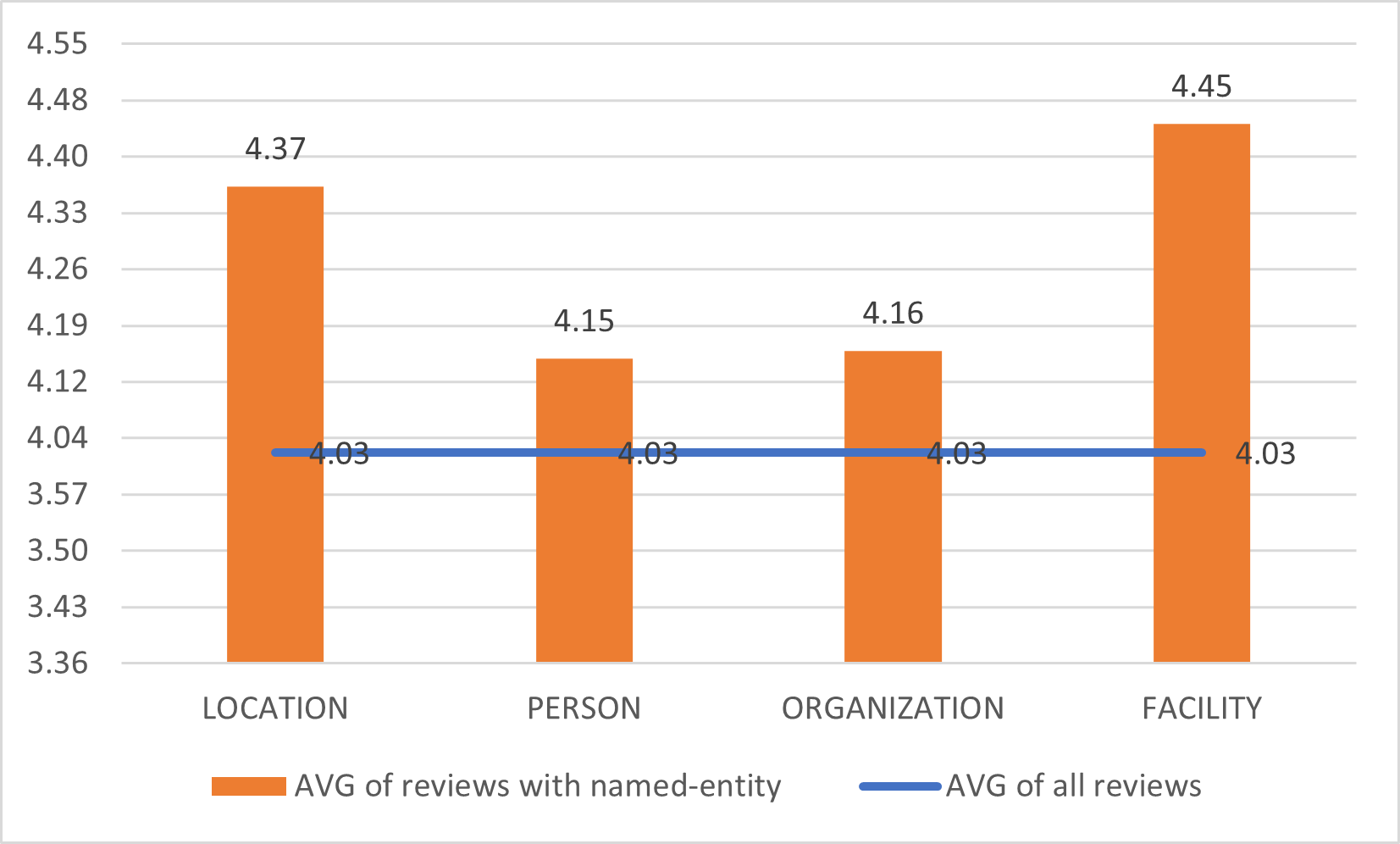
The Empath categories were finally crafted into a histogram. The list of categories, at the order of commonness is available in the application output.



1. Empath categories found from the reviews

## Hypothesis 2: “Presence of a given type of named entity would entail positive sentiment or negative sentiment.”

It was decided to look more closely into the entities found by NLTK. NLTK found 4 entities, such as “location”, “person”, “organization” and “facility”. Average review rating was 4.03. That was compared to the average of reviews with specific named entity category. Presence of category seemed to entail positive user ratings and as proven, user ratings correlate with sentiment polarity. This is proven true for positive sentiment.



1. Average rating of reviews containing entity categories compared to dataset overall average rating

## Hypothesis 3: “Negative sentiment entail more argumentation.”

Presence of argumentation words is interesting area of NLP. The hypothesis was that negative sentiment entails more argumentation. For this analysis we considered the list of words that induce explanation, such as “because”, “for the purpose”, “since”. The analysis revealed that explanatory words entail negative sentiment. This was proven true with positive reviews containing by average 0.15 explanatory expressions while the corresponding value for negative reviews was significantly higher at 0,37.

## Hypothesis 4a: “Badly written reviews are likely to be included in ambiguous class”

Ambiguity of each review was determined by VADER sentiment analysis deviation from the user rating. In the dataset, there were very few ambiguous reviews. Threshold for significant deviation was considered to be 0.25 or more when both the user rating and VADER analysis results were normalized. Badly written reviews were determined by the percentage of known words in the review. For this, WordNet was used.

Average known words per review showed that percentage for ambiguous was almost same as for unambiguous, both resulting to 65.6%. As no difference in percentage was found based on our analysis with WordNet, thus, this is false.

## Hypothesis 4b: “Ambiguous reviews are shorter”

Hypothesis 4b claims that ambiguous reviews are shorter than unambiguous. This is easily testable by counting the words in reviews. On average, the total words of the reviews for ambiguous was 50 words. For unambiguous reviews, the word count was 69.

Ambiguous reviews seem to be on average 27% shorter than the unambiguous ones. This hypothesis is proven true.

## Hypothesis 4c: “Ambiguous reviews have bad readability”

To determine if the ambiguous reviews have bad readability, project compared the ambiguousness to Automated Readability Index. Within the dataset, around 97 % of reviews were unambiguous. Out of those, the average ARI was 7.63. For the ambiguous reviews, the ARI was 7.13. The lower the ARI, the more readable text is. This hypothesis was tested to be false. However, as the size of the ambiguous class was significantly smaller than unambiguous class, the result can also be deemed as inconclusive.

# Discussion and conclusion

During the project, NLP pipeline application was developed to analyze the data for sentiment analysis of hotel reviews. The pipeline and the performed steps were mandated in the project description, which was followed by the project team. The data, that was available for the analysis, was examined first before choosing which one to use. The decision was to use Datafiniti hotel reviews dataset. This project managed to create an NLP pipeline in a fairly short timeframe, using free non-commercial tools and free python libraries, but the usage for deeper analysis needs additional work.

The application that was developed during the project, could in principal work with different types of datasets, but due to time constraint, corners were cut, and practical implementation does not support that. That would be good improvement idea for the future projects. With little effort, the application could be made more generic and able to work with different kind of datasets. However, the most interesting part of the project is not exactly tool itself, but rather how different businesses could utilize the social web data that they already possess. Mining the opinions of customers, partners and employees from different channels, is no longer only necessary but business critical. With improved tooling and improved access to those, companies, who lack such thinking, will lose their competitive edge.

For large enterprises offering enterprise SaaS products and services, expanding sentiment analysis to their portfolio is an interesting business opportunity. Sentiment analysis in Microsoft Teams discussions, Workday Finance and HR process tool and Slack discussions could indicate the sentiments around company culture, employee performance and team dynamics. However, such development opens strong privacy concerns which must be tackled.

The project did not reveal anything new; novelty aspect was not strong part of it. It was more practical study on how to create NLP pipeline and understand the efforts and limitations of it. One of the big learnings was that there is data around us that can be utilized for solving different kind of problems in different industries.

As a main conclusion, according to our study SentiStrength and VADER work extremely well with this kind of data. Our results showed high correlation between user ratings and sentiments provided by both tools. Based on literature, these tools can be contextual though.

The project was seen challenging at times. The project team learned a lot from different NLP tools and toolkits. The main challenge of the project was time, some of the tasks were extremely broad and time consuming. Additionally, the project description was ambiguous at times, but very educational.

##### References

[1] Fan, Xing and Zhan, Justin June 2015 “Sentiment analysis using product review data.” 2015.

https://journalofbigdata.springeropen.com/articles/10.1186/s40537-015-0015-2 Web. Accessed: November 1, 2020.

[2] Liu, Bing 2015 “Sentment Analysis: Mining Opinions, Sentiments, and Emotions”. Second edition, preface xi.

[3] Alexandre Pinto, Hugo Gonçalo Oliveira and Ana Oliveira Alves 2016 “Comparing the Performance of Different NLPToolkits in Formal and Social Media Text ”https://drops.dagstuhl.de/opus/volltexte/2016/6008/pdf/OASIcs-SLATE-2016-3.pdf

[4] Aue, Anthony and Gamon, Michael January 2015 “Customizing Sentiment Classifiers to New Domains: A Case Study”.

[5] M.Geetha, Pratap, Singha, SumedhaSinha “Relationship between customer sentiment and online customer ratings for hotels – an empirical analysis. “

[6] Islam, Rakibul and F. Zibran, Minhaz 2017 “A Comparison of Dictionary Building Methods for Sentiment Analysis in Software Engineering Text”

[7] Al-Shabi, M. A. January 2020 “Evaluating the performance of the most important Lexicons used to Sentiment analysis and opinions Mining”

[8] <http://sentistrength.wlv.ac.uk/>, Accessed: November 1, 2020.

[9] Thelwall, Mike, Buckley, Kevan, Paltoglou, Georgios 2012, “Sentiment Strength Detection for the Social Web” <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.278.5294&rep=rep1&type=pdf>

[10] Engels, Gregor, Palarkarska, Tsvetomira, Stein, Benno, Trenkmann, Martin, Wachsmuth, Henning 2014 “A Review Corpus for Argumentation Analysis”

[11] Gilbert, Eric and Hutto C.J 2014 "VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text"

[12] https://pypi.org/project/vaderSentiment Web. Accessed November 1, 2020

[13] Michael S. Bernstein, Binbin Chen, Ethan Fast 2017, Lexicons on Demand: Neural Word Embeddings for Large-Scale Text Analysis https://www.ijcai.org/Proceedings/2017/0677.pdf

[14] Bernstein, Michael S., Binbin, Chen, Fast, Ethan 2017 "Lexicons on Demand: Neural Word Embeddings for Large-Scale Text Analysis" https://www.ijcai.org/Proceedings/2017/0677.pdf

[15] Automated readability Index, web. Accessed November 5th 2020. https://en.wikipedia.org/wiki/Automated\_readability\_index

[16] Automatic Ambiguity Resolution in Natural Language Processing - An Empirical Approach, 1996 https://books.google.fi/books?hl=fi&lr=&id=QuE1OV3EaQIC&oi=fnd&pg=PA1&dq=ambiguity+short+text+nlp+hypothesis&ots=Qcr6NJH6f6&sig=LhzQS5JMmSbCtrDOY2oMTLDPbWs&redir\_esc=y#v=onepage&q&f=false