Quantified Self

D-Matrix

Jared Chung

2016

# Introduction

Collaboration is a way of pooling the skills of individuals together to accomplish a task or project. It has become increasingly more important in the modern world, as we have become more connected around the globe.

The purpose of this project is to gain insight into how we as individuals play a role in a collaborating on a data science project. As communication can be seen as a critical success measure, we wanted to analyse how we participated with each other.

To measure our participation, we collected data from chat conversations in Slack and JIRA issue tickets to get an understanding of how we collaborated with each other. We applied text mining techniques to transform and evaluate our data as well as using tools such as python and power BI to visualize and explain the insights that we uncovered.

Text mining is being used increasingly as organizations recognize the untapped information contained in unstructured text data (Janani and Vijayarani 2016). Social media, such as Twitter and Facebook, have been used effectively by organizations to uncover positive and negative trends that, when identified through text mining, can be used to leverage the positive trends and provide corrective action to counteract any negative comments.

# Sources of Data

#### Slack

Slack is an instant message and team collaboration tool. Some of the features of slack is being able to create channels with different topics. For example, in the case of our assignment we had channels based on the CRISP-DM data mining framework as well as channels which were completely unrelated to data science and more free flowing discussions.

Some of the additional features of slack is the ability to incorporate customized add-ons. One of the add-ons we used was a messaging bot called Howdy. The purpose of howdy is to be an additional member of the team and allow people to reflect on what they had done as well as highlight any problems or issues occurring in the project.

**Jira**

Jira is an issue tracking and project management tool that allows you to track any kind of unit of work (issue, bug, story, project task, etc.) through a predefined workflow.

The item of work and the workflow can be highly customized so we decided to break these tasks down into the CRISP-DM Framework which coincided with slack messaging data.

# Part 2 Data Collection and Preprocessing

The data preparation stage is a critical stage of the data science project. Ensuring the quality of the data is essential before any analysis can be done. Accessing the data was simple as Slack and Jira provided ways extract the data into JSON and CSV files. As the majority of our data came from messages, we spent the most amount of time formatting and transforming the text. To address these issues, we applied a number of Natural Language Processing techniques.

In this section, we will go through steps involved to transform the data as well as a background and evaluation of the techniques used in Natural Language Processing and our final decision of techniques that we used in this project. The main tools we used is the Python Programming language with emphasis on the package NLTK (Natural Language Tool Kit).

### Stop Words / Remove Words

In every language, some words are particularly common. While their use in the language is crucial, they don’t usually convey a particular meaning, especially if taken out of context. This is the case of articles (e.g., a, an, the), prepositions (e.g., at, by, in, to, from, with) and conjunctions (e.g., and, but, as, because) which are commonly called stop-words (Zaman, Matsakis and Brown 2011). Stop word removal is one important step that should be considered during the pre-processing stage.

NLTK package provided a simple list of English Stop words which we used. However, we noticed that certain messages needed to be manually removed as they were automatically added during the setup of Slack. As such, we created our own custom list of words and messages to removed.

In addition to removing words we also needed to consider punctuation. Punctuation indicate structure and the organisation of the text but for simplification of document information retrieval techniques, punctuations has to be deleted from the document (Mahmud 2013). We made the decision to remove punctuation as it interfered with our goal of analysing the words in the text.

### Privacy / Anonymization of Names

### Privacy is an issue which arises when dealing with personal data. There can be serious consequences for releasing sensitive data without a person’s consent, as was the case when the company AOL released a dataset 2016 where individuals were identifiable resulted in the withdrawal of the data and dismissal of two employees (Fu and Wong 2010).

The data we collected also made us consider the privacy policies in software. Slack and Jira collects contact, profile, billing and log information on each individual. Under the privacy policy of both Slack and Jira they state the information we provide them is used to improve their product. In saying this, there is a risk in the pursuit for a customized experience between business and individual might reveal breaches in privacy (Huang, Sankar and Sarwate 2015)

We considered these privacy issues as our data contained the names of each member, it was important to anonymize everyone’s name. We decided to replace our names with the aliases Business Analyst, Data Scientist, Project Manager and Data Analyst. We also customized the chat bots name to be Bender the Scrum Master.

### Stemming and Lemmatizing

One of the issues that arises in natural language processing is that words have morphological variants which will not be recognised by term matching algorithms (Hull 1996). This means you could have words like organize, organizes, and organizing all have the same meaning but won't be grouped together. Some of the ways used to tackle this problem include Stemming and Lemmatizing

Stemming is a procedure which attempts to reduce a word to its base form by cutting the ends of the words off with the most common stemming algorithm being Porter’s stemmer.

Lemmatization on the other hand, uses vocabulary and linguistic analysis of words, with the aim to remove impure endings only and to return the base or dictionary form of a word, which is known as the lemma. (Manning et al. 2008).

Balakrishnan and Lloyed-Yemoh (2014) did study on comparing the performance of stemming and lemmatization for information retrieval. In their results they found that lemmatization outperformed stemming but the differences in accuracy were insignificant.

We tried out both techniques from NLTK package for this project and concluded that lemmatization was more suitable. This was due to stemming creating words that were unrecognisable and as such could not be analysed.

### Tokenizing

Tokenization is the procedure to segregate all the words, numbers and characters in a given document and these identified words, numbers, and other characters are called tokens. In addition to the token generation process, it also evaluates the frequency value each token present (Singh and Saini 2014).

There are number of different types of tokenizers

* Treebank Word Tokenizer - This tokenizer uses regular expressions to tokenize text as in Treebank.
* Word Punct Tokenizer - This tokenizer divides a string into substrings by splitting on the specified string, which it is defined in subclasses.
* Punct Word Tokenizer- This tokenizer divides a text into a list of sentences; by using unsupervised algorithms.
* Whitespace Tokenize - This tokenizer divides text at whitespace.

In the context of the pre-processing methods we used on the data, the whitespace Tokenizer ended up being the simplest and most effective way to tokenize our data set.

# Analysis

The remaining part of this paper will outline the analysis completed and explore some of the questions I was trying to answer as well as interesting insights gathered. The results are based on data extracted from slack for the period 30th March till the 15th May.

**Text analysis**

The initial analysis I conducted on the text was to extract meaningful terms frequencies. Using tokenization, I could aggregate each word and generate a list of the most frequent words used. Table 1 Indicates the top twenty words I used.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | Word | Count |  | Word | Count | | word | 39 |  | Bender the Scrum Master | 17 | | yeah | 39 |  | okay | 15 | | think | 29 |  | text | 15 | | like | 27 |  | well | 15 | | data | 22 |  | code | 15 | | hey | 21 |  | quite | 14 | | work | 19 |  | feel | 14 | | need | 19 |  | one | 14 | | going | 19 |  | Business Analyst | 14 | | might | 18 |  | Project Manager | 14 | |  |  |  |  |  | |

Table 1: Top 20 used words (self)

The table drew an interesting link to the members I engaged most with during the course of the project. The results showed the majority of my communication was directed at the Business Analyst and Project Manager. Measuring the direct communication between group members cleared showed better collaboration whereas the opposite can be said for the communication to the other members.

Another insight, was looking at the language I used in different time segments of the project. To achieve this the data was segmented into two parts, the first part was everything before the April the 28th and the second part was everything after that date. This could then be visualized by generating a word cloud for both periods as shown in Figure 1 and Figure 2.

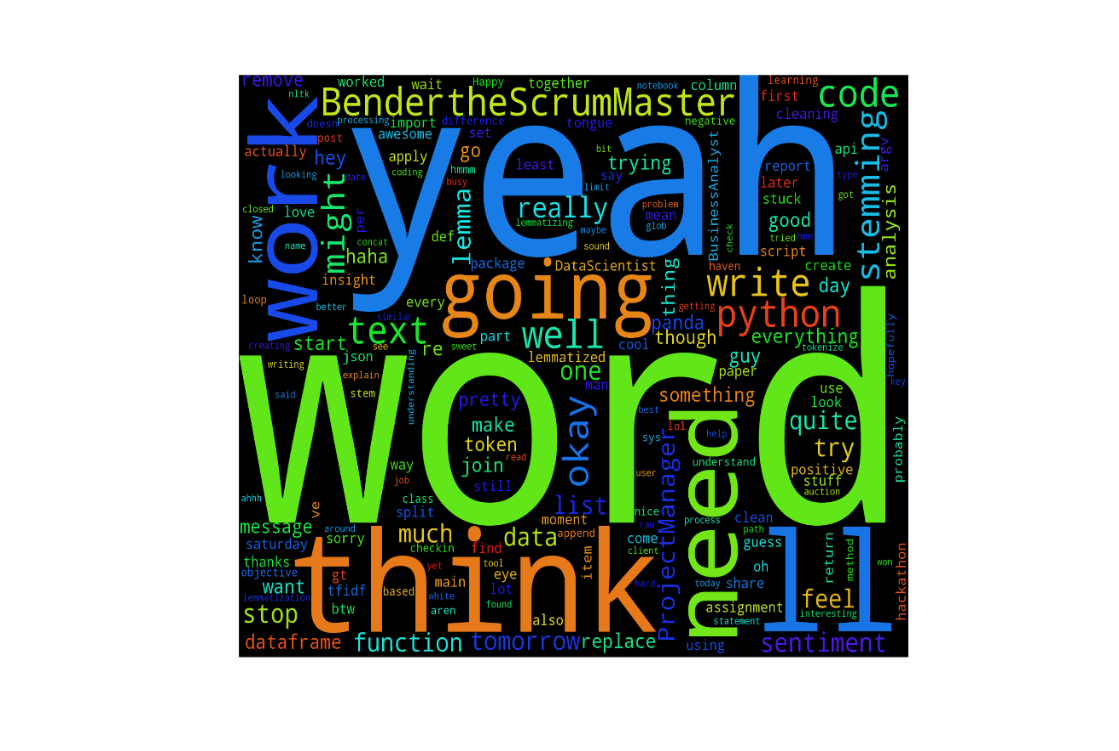


Figure 1: Words before 28th April (self) Figure 2: Words After 28th April (self)

It was interesting to see that there was a change in types of words being used in the first half (e.g think, start, might, discussion) compared with the second half (e.g work, need, yeah, write). The first half of words indicated uncertainty in the early stages of the project whereas the second showed more acceptance and urgency suggesting an emphasis on completing the work.

In addition, I wanted to compare my vocabulary against the other members of the team. Table 2 showed the total unique words used by each person. I was positioned around the middle of the group but was intriguing to see that the project manager was over double the vocabulary of everyone else.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Name | Vocabulary Count | | Jared | 903 | | | Business Analyst | 1249 | | | Data Analytics Manager | 866 | | Project Manager | 2132 | | Data Scientist | 693 | |

Table 2: Vocabulary Count

**Sentiment Analysis**

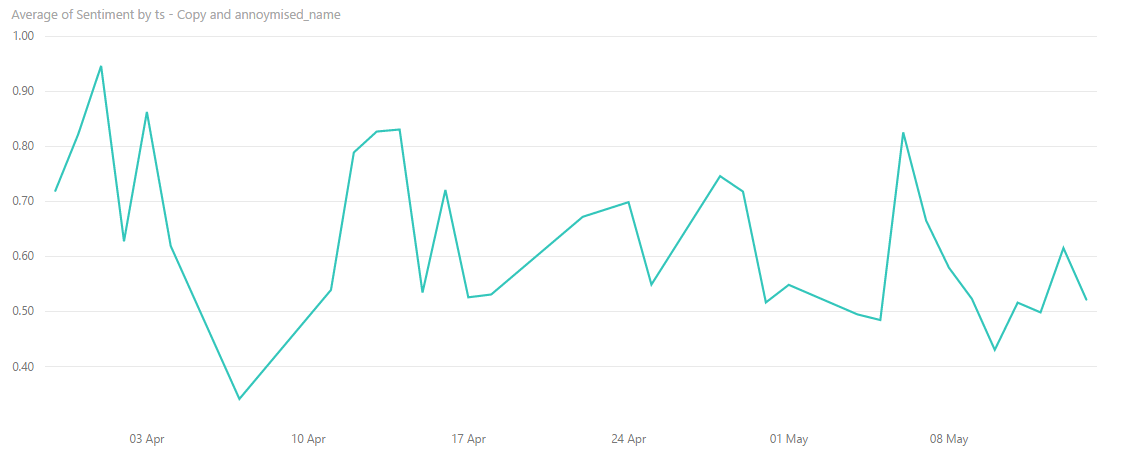
Sentiment analysis is the study of people's attitude, emotion and opinion towards a particular product, service, event or even an individual (Liu 2012). Analysing sentiment has gained a lot of attention in recent years with many companies utilizing the data from social media to understand more about their customer and using this to improve their product or service.

The wide range of applications in almost every domain has led companies providing services to meet this demand. Some of the providers of these services include Indico, Aylien and Alchemy API. These services all provide an easy way to connect to their service by using API’s (application programming interface).

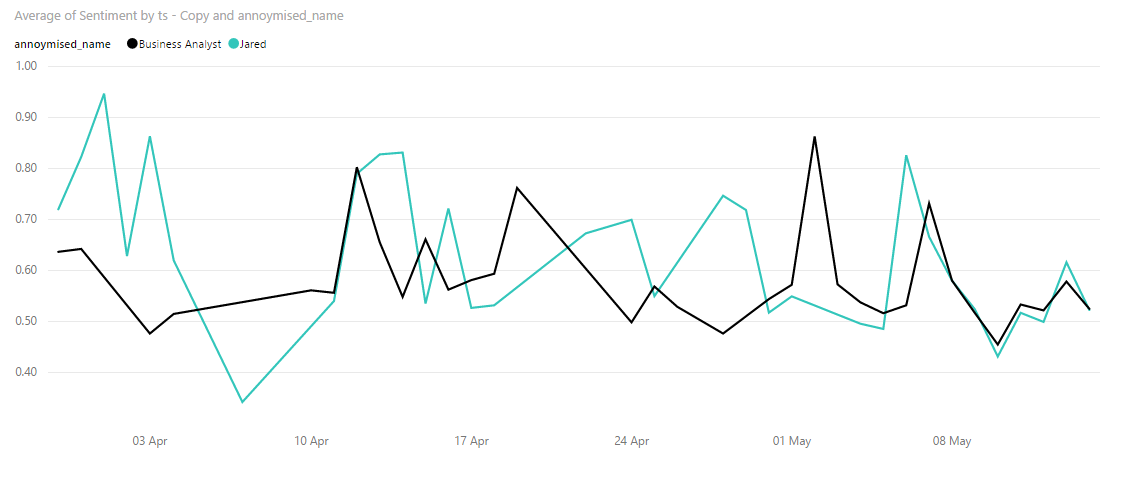
We tested out the sentiment analysis tools and decided to use Indico for our analysis as it achieved the highest accuracy standard for sentiment analysis which was outlined in the 2015 Indico Article. In addition, we faced some limitations with the other tools as we were restricted to the number of messages we could process.

The idea behind sentiment analysis is to quantify each message by a polarity measure between zero and one. Where a value close to zero would indicate a negative emotion and a value close to one would express positivity.

One way I analysed the sentiment was to plot my average sentiment over time as represented in Figure 3. The graph highlighted some interesting abnormalities in the data. One in particular is the drop in sentiment on the 7th of April. Upon further investigation, it was found the negative sentiment was based on messages in the lead up to the submission of an assignment and there was a subsequent increase in sentiment afterwards.

Figure 3: Time Series Sentiment Graph (Self)

Furthermore, I wanted to compare my sentiment with the other members of the group to find if there were any comparative trends in the data. Figure 4 shows that around the 10th of May there was a drop in sentiment between me and the business analyst. Further inspection revealed that the sentiment managed to pick up a disagreement. The disagreement was in regards to pre-processing technique we wanted to use.

Figure 4: Time Series Sentiment Graph (Self vs Business Analyst)

The final analysis I made using sentiment was to calculate the overall sentiment of each member. Figure 5 shows that generally the team was quite neutral in polarity. This might be caused by the context of the work to be formal in nature. On another note, Bender the bot showed notably differences in emotions compared to the rest of the group.

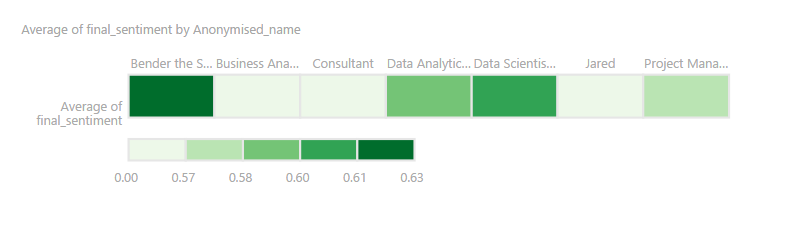


Figure 5: Heat map of Average sentiment of each member

## Jira Analysis

The purpose of Jira was to track the tasks we completed over the period of the project and place this layer over the communication we conducted in Slack. The tasks were broken down into the phases of CRISP DM. The results showed that the majority of the tasks I completed were around the data preparation and modelling. This showed the areas of the project where I was most interested in but also showed parts that needed more attention.

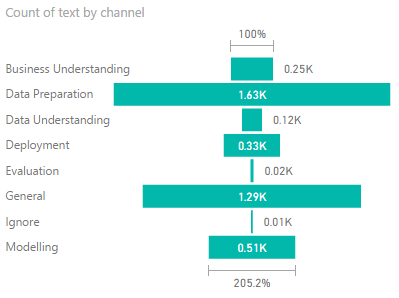
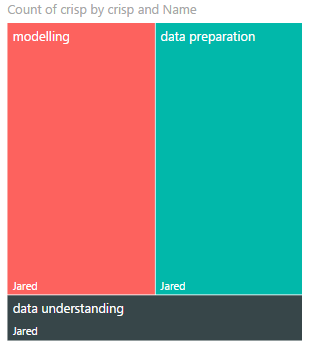


Figure 6: Breakdown of Jira tasks Figure 7: Total Number of Messages per Channel

# Conclusion

​This project was aimed at understanding the way we collaborated in a data science project by analysing Slack message data and Jira Issue tickets. We tackled some of the issues in dealing with unstructured data by applying a number Natural Language Processing techniques using the NLTK python package.

The results showed that language can change over time. In particular, the words used at the beginning of the project can change towards the end. The vocabulary can show the range of your language and variety of language between each person.

We also used sentiment analysis on the messaging data. Sentiment allowed us to measure our attitudes throughout the course of the project. We found that you could pin point moments in time where there was a peak or drop in attitude.

Collaboration is an essential part of project work. Being able to dive deep into the way we communicate can enrich our understanding and enhancing the individual and the group.

# Reflection

The project showcased a number of topics, concepts and techniques used to apply on a real data science endeavour. We applied the industry standard framework CRISP DM for data mining. Throughout the project I had mixed opinions about the framework. I valued the initial stages of the framework where it emphasized establishing the question of the problem. However, we had to jump to the modelling stages before we could actually start doing the data preparation.

The cleaning part ended being a lot more work than we anticipated. We spent the majority of time assess the different approaches to tackling our messages. This is an issue which I can see arising in future projects as the obstacles of dealing with data can be quite

Some points to add

* We originally were applying the CRISP DM framework for data mining but found that it was difficult to apply for this task
* Cleaning was a hard task as there were lots of things we need to remove/adjust
* The recursive approach of analyzing work together
* 1942 Big Brother feel?

References

Vijayarani, S. & Janani, R. 2016, 'Text Mining: Open Source Tokenization Tools - An Analysis', Advanced Computational Intelligence (ACII), vol. 3, no. 1, viewed 9 May 2016, <http://aircconline.com/acii/V3N1/3116acii04.pdf>.

Manning, C.D., Raghavan, P., Schütze, H. & others 2008, Introduction to information retrieval, vol. 1, Cambridge university press Cambridge, viewed 8 May 2016,<http://www.langtoninfo.co.uk/web_content/9780521865715_frontmatter.pdf>.

Balakrishnan and Lloyd-Yemoh 2014, Lecture Notes on Software Engineering, Vol. 2, No. 3, August 2014

D. A. Hull, “Stemming algorithms: A case study for detailed evaluation,” Journal of the American Society for Information Science, vol. 47, pp. 70-84, 1996.

### Mahmud, S.R., 2013. A Simple Information Retrieval Technique. *Int. J. on Recent Trends in Engineering and Technology*, *8*(1).Stemming and Lemmatizing

Zaman, A.N.K., Matsakis, P. and Brown, C., 2011, September. Evaluation of stop word lists in text retrieval using Latent Semantic Indexing. In *Digital Information Management (ICDIM), 2011 Sixth International Conference on*(pp. 133-136). IEEE.

Fu, AU, Wong, RC, 2010. Privacy-Preserving Data Publishing: An Overview. 1st ed. University of Waterloo: Morgan & Claypool Publishers.

Huang, C., Sankar, L. and Sarwate, A.D., 2015. Designing incentive schemes for privacy-sensitive users. *arXiv preprint arXiv:1508.01818*.

Singh, V. and Saini, B., 2014. AN EFFECTIVE PRE-PROCESSING ALGORITHM FOR INFORMATION RETRIEVAL SYSTEMS. *International Journal of Database Management Systems*, *6*(6), p.13.

Liu, B., 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, *5*(1), pp.1-167.

<https://slack.com/privacy-policy>

<https://www.atlassian.com/legal/privacy-policy>

<https://confluence.atlassian.com/doc/data-collection-policy-659783908.html>

<https://indico.io/blog/sentimenthq-new-accuracy-standard/>

<https://indico.io/>

<http://aylien.com/>

<http://www.alchemyapi.com/>

​

## Appendicies

Need to add Cleaning Summary Table

