Preprocessing

```
Total tokens in reject emails before removing stopwords: 4642
Total tokens in reject emails after removing stopwords: 2485
Total tokens in non-reject emails before removing stopwords: 5933
Total tokens in non-reject emails after removing stopwords: 3528
Token count for reject emails before lowercasing: 3613
Token count for non-reject emails before lowercasing: 4451
Token count for reject emails after lowercasing: 3541
Token count for non-reject emails after lowercasing: 4264
```

Figure 1 - Preprocessing

Preprocessing the data is an important step to maintain the consistency of the data. Stopwords are common words (such as “the,” “and,” “in,” “of”) that appear frequently in text but carry little meaning. These words add noise to the data without contributing much to the analysis. When I removed these words, I could focus more on the more meaningful words. This is important for frequency analysis and sentiment analysis, by reducing noise and ensuring that the data is uniform the sentiment analysis becomes more focused on the remaining words. From Figure 1, we can see that there was a token reduction of ~47% (from 4642 to 2485) in the reject emails compared to the ~40% (5933 to 3528). This difference shows that the rejected emails rely more on common and non-specific words. This also means that not_reject emails also have fewer stopwords which likely emphisises critical information such as account details. The lower casing is also an important step as it reduces redundancy and improves feature quality for further analysis. Case sensitivity can lead to redundancy as multiple words can convey the same meaning. For Example, in the data file, there are 83 instances of the word “Software” but there are also 8 instances of the word “software”, these tokens have the same meaning but appear as distinct tokens. This reduction in case sensitivity ensures consistency which is important in the sentiment analysis as they apply consistent criteria to generate more accurate analysis. As seen in Figure 1, there is a reduction in the tokens from 3613 to 3541 for reject emails and 4451 to 4264 for not reject emails.

Topic analysis:

```
Topics for reject emails:
Topic 1:
process, hiring, position, applying, selected, thank, time, application, developer, moved
Topic 2:
time, unfortunately, forward, position, application, software, thank, moving, appreciate, regards
Topic 3:
team, job, thank, best, applying, software, new, position, developer, gm
Topic 4:
solutions, forward, candidates, cox, joining, unfortunately, best, roles, community, join
Topic 5:
thank, position, time, software, opportunities, engineer, future, unfortunately, job, career

Topics for non-reject emails:
Topic 1:
resume, experience, 100, software, position, engineer, thanks, right, ve, seth
Topic 2:
software, detective, engineer, job, development, hours, experience, work, senior, project
Topic 3:
application, team, software, thank, position, engineer, talent, role, match, email
Topic 4:
account, thank, 2021, verify, link, order, 23, month, com, email
Topic 5:
password, com, change, sign, account, new, thank, help, ll, payment
```

Figure 2 - Topic Analysis

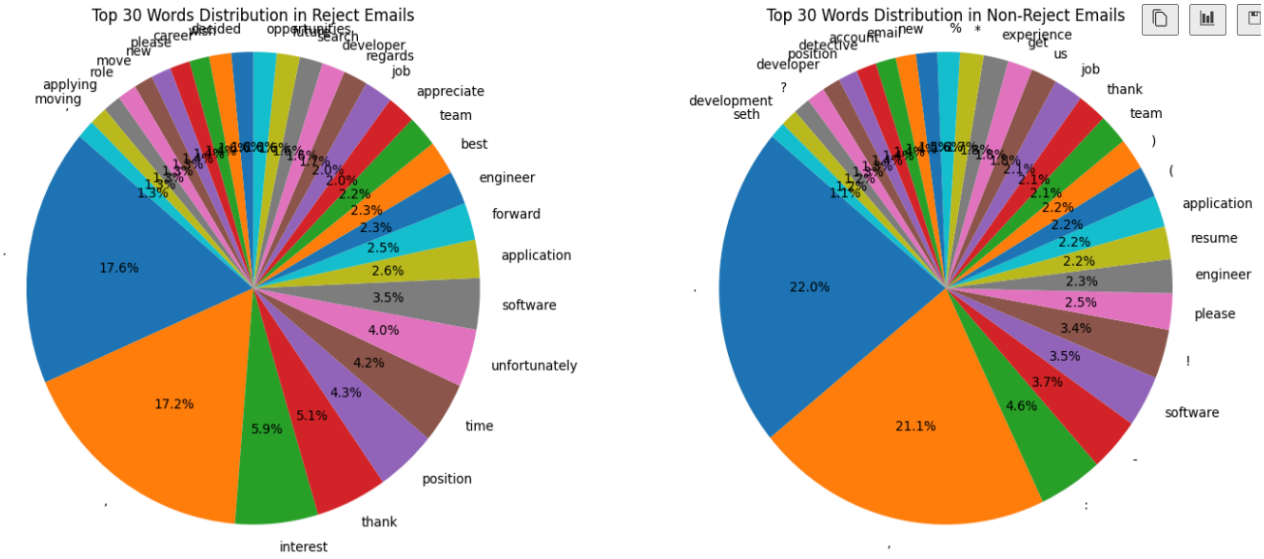


Figure 3- Frequency analysis.

This topic analysis was done using Latent Dirichlet Allocation (LDA) which works by identifying the recurring topics in the reject and not_reject emails and picking out the top 10 words. From Figure 2 we can see that the topics in the reject emails have to do with the hiring process. This is due to the inclusion of the words “hiring”, “roles” and “candidates”, these words focus on the application status and the potential opportunities as there are sentences like “applicants to consider for this role”. Whereas, not_reject emails focus more on the positions, account management and team dynamics. The word “unfortunately” is also found frequently with the reject topics compared to the not_reject words. This is to be expected as in professional communication, phrases like “unfortunately” are common as they acknowledge the recipient’s disappointment while delivering unlucky news. In Figure 3, we can also see that in the

not_reject emails, there is a greater variation in punctuation. In the not_reject, there is a presence of “?”, ”!” and ”%”. These could be clarifying questions, Important information and the scoring on a mock test. This is compared to the reject emails as they mostly consist of “.” and “,”. The use of punctuation such as the “?” and “!” suggests more interactive and expressive tones which can potentially lead to a more favourable perception of the content, whereas the “.” and “,” suggest a more formal and straightforward tone which can potentially lead to a less favourable reception of the message. Doing LDA and frequency analysis allowed me to focus on the tonality between the reject and not_reject emails.

Sentiment Analysis:

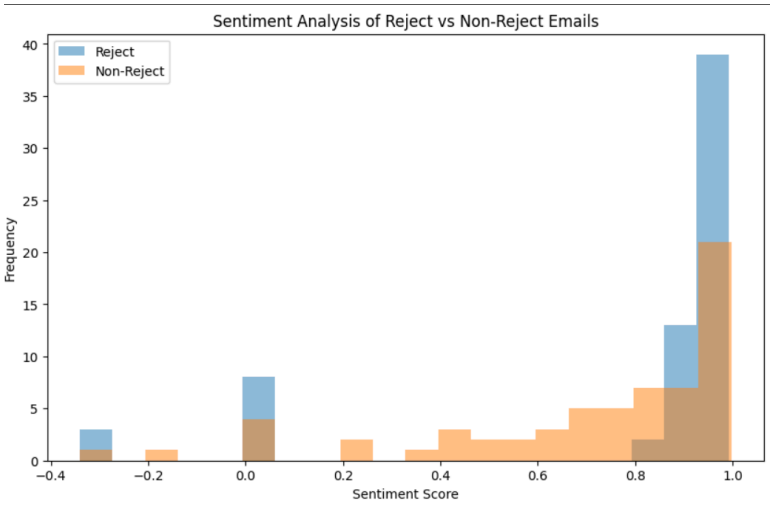


Figure 4- Vader Sentiment Analysis

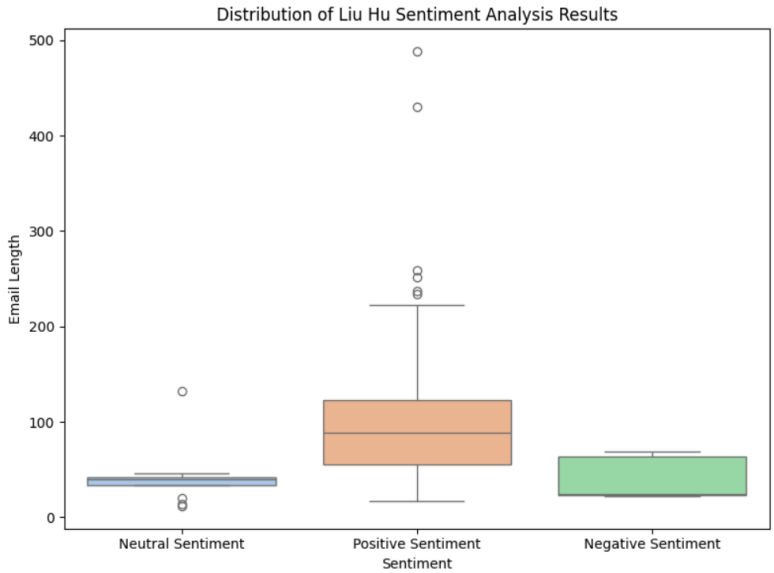


Figure 5 - Liu Hu Sentiment analysis

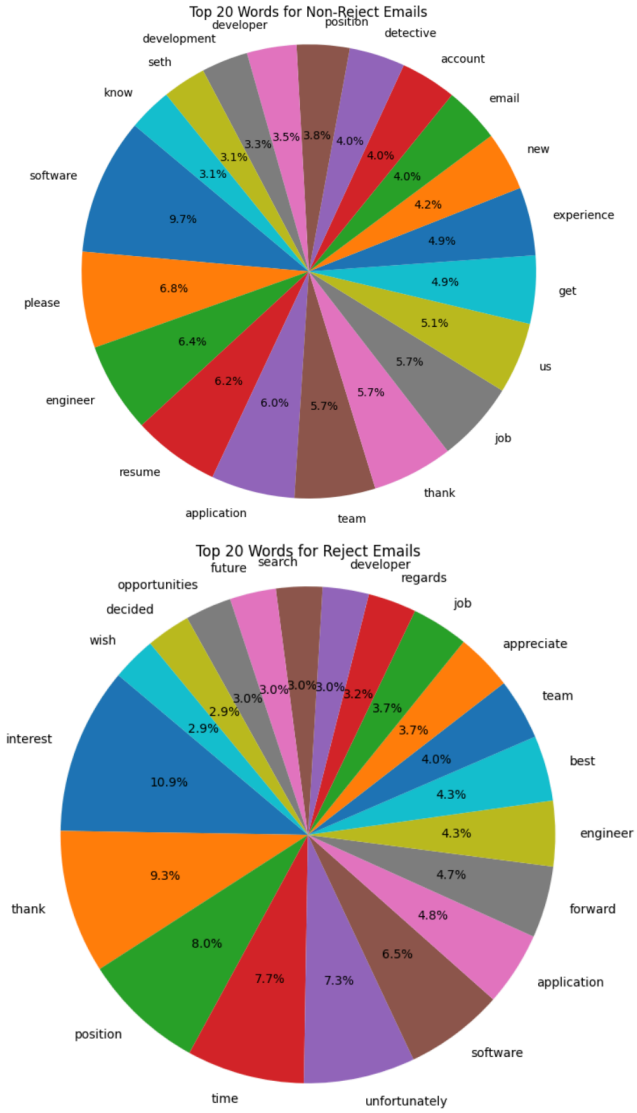
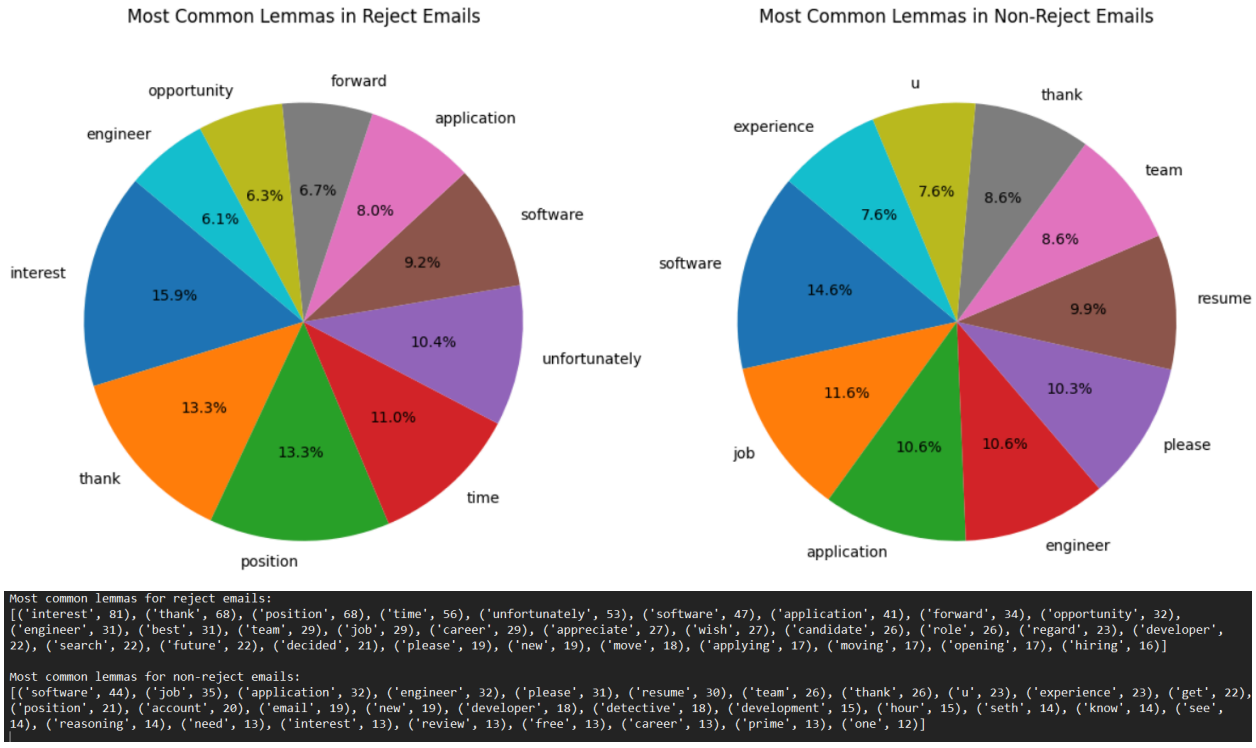


Figure 6 - Word Frequency

In Figure 4, I used VADER analysis which is a lexicon and rule-based sentiment analysis tool specifically designed for analysing text sentiment.VADER provides an automated and efficient way to analyse the sentiment of large volumes of text data, in this case, emails, without the need for manual annotation or labelling. When doing this it was surprising to see that there was a large amount of reject tokens on the positive side of the sentiment score around 0.9 and 1.0. But when taking a second look it may be explained by the use of combinations such as “Thank you for your recent application”.This combination would lead to a more positive score and is found within the word frequency in Figure 6. However, in Figure 4 we can see that on average the not_reject emails have a positive sentiment. For example, one of the most frequent words is “please”, this can be found within sentences like “please feel free to reach out” which is a sign of positive interaction. Whereas, for the reject emails a popular word is “unfortunately”, This has negative connotations as it includes words phrases like “Unfortunately, Test Innovators has moved to the next step” which delivers negative information to the receiver politely and professionally. In Figure 5, I used Liu Hu's sentiment analysis which is a purely lexicon-based

sentiment analysis method. I particularly used it to compare email lengths to the sentiment. Through this, I learnt that longer emails have a lot more positive sentiment whereas the smaller email sizes have a more negative sentiment. This may be because longer emails may contain detailed explanations, positive news, or expressions of gratitude. Positive sentiments often require more words to convey meaning. Whereas, shorter emails might be concise rejections, complaints, or negative feedback. Negative sentiments can be expressed in fewer words.

Frequency Distribution:




```
Most common lemmas for reject emails:
[('interest', 81), ('thank', 68), ('position', 68), ('time', 56), ('unfortunately', 53), ('software', 47), ('application', 41), ('forward', 34), ('opportunity', 32),
('engineer', 31), ('best', 31), ('team', 29), ('job', 29), ('career', 29), ('appreciate', 27), ('wish', 27), ('candidate', 26), ('role', 26), ('regard', 23), ('developer',
22), ('search', 22), ('future', 22), ('decided', 21), ('please', 19), ('new', 19), ('move', 18), ('applying', 17), ('moving', 17), ('opening', 17), ('hiring', 16)]

Most common lemmas for non-reject emails:
[('software', 44), ('job', 35), ('application', 32), ('engineer', 32), ('please', 31), ('resume', 30), ('team', 26), ('thank', 26), ('u', 23), ('experience', 23), ('get', 22),
('position', 21), ('account', 20), ('email', 19), ('new', 19), ('developer', 18), ('detective', 18), ('development', 15), ('hour', 15), ('seth', 14), ('know', 14), ('see',
14), ('reasoning', 14), ('need', 13), ('interest', 13), ('review', 13), ('free', 13), ('career', 13), ('prime', 13), ('one', 12)]
```

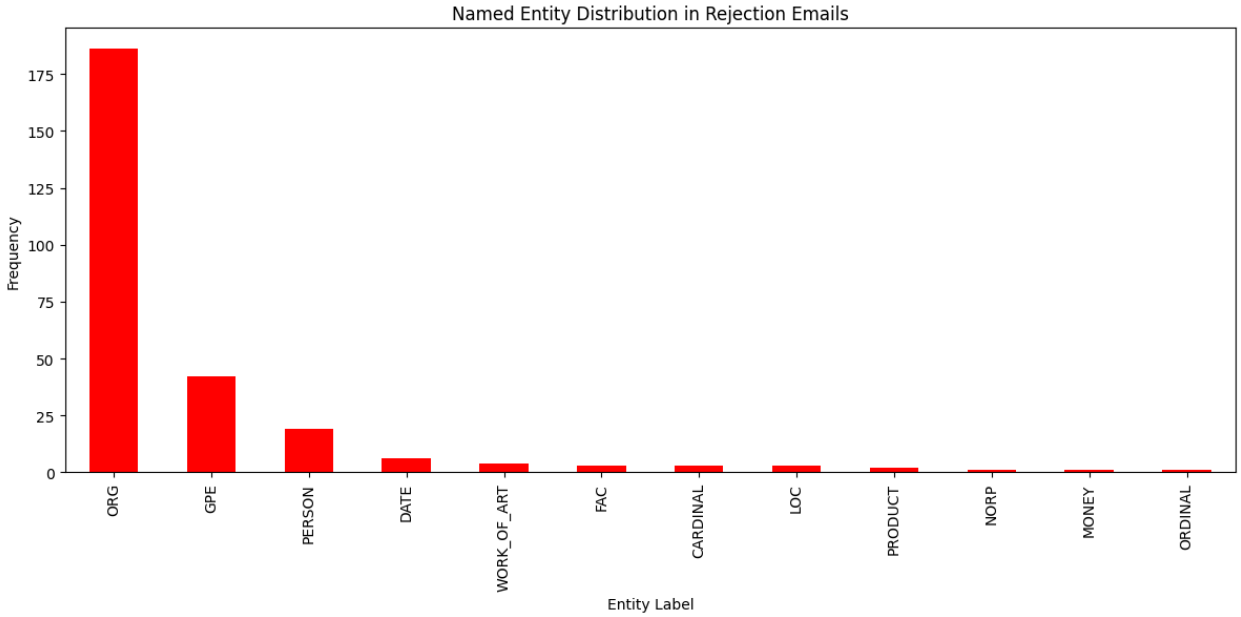
Figure 7- Frequency analysis

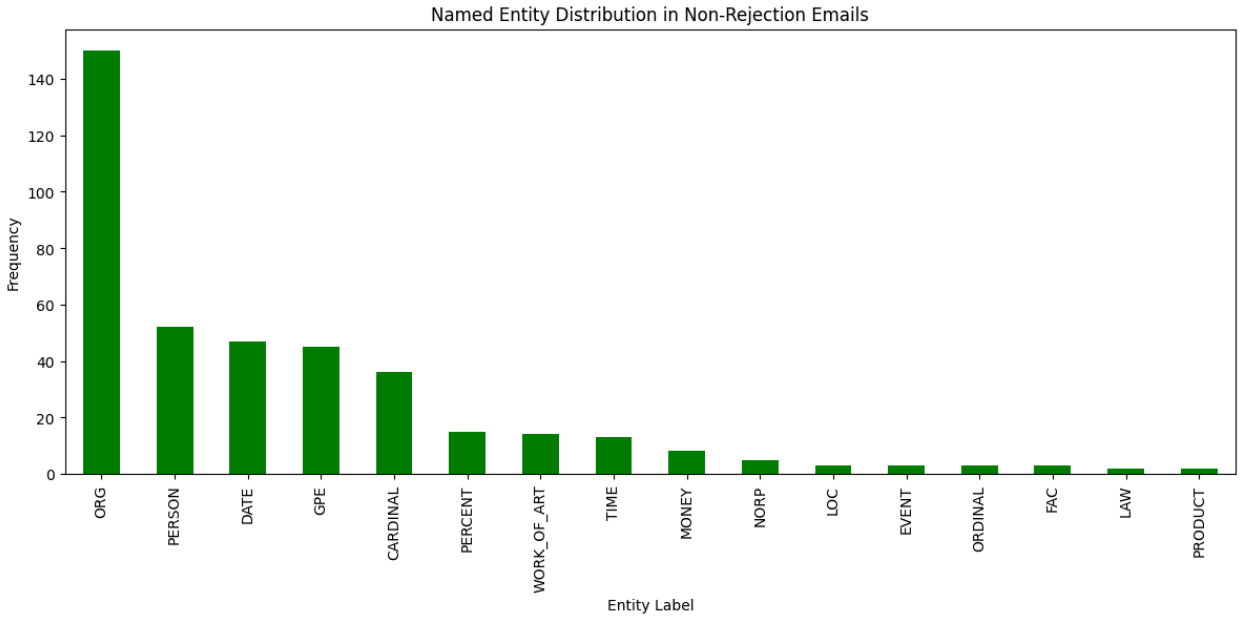
Lemmas are like the basic forms of words. Instead of listing every variation of a word, like 'interests,' 'interested,' or 'interesting,' lemmatising shortens them down to just 'interest.'

In Figure 7, the most common lemmas found in reject and non-reject emails show the differences in communication styles. In the reject emails, some of the most common lemmas are “interest”(81 appearances) and “thank” (68). These words often are in the same phrase “Thank you for your interest”(41) this is a professional manner that acknowledges the effort someone put into the application process. However, these emails also contain the lemmas for “unfortunately” which means that they are disinterested in the application or continuing the conversation. When companies reject people they have a formal template that can be modified for the particular person which is why many of the rejection emails have a similar tone and common words to save time.

In the not_reject emails, The frequency of lemmas is a lot more spread out, this indicates a more personalised email exchange compared to the reject emails. The words "software"(44) and "engineer"(35) suggest that the emails are often discussing specific roles or technical aspects of the job.

The application of lemmatization proved to be particularly helpful in this analysis, as it allowed the identification of common patterns across different emails. This allowed me to find the most common words in both the reject and not_reject emails. This highlighted the differences in the communication styles between the two categories but also allowed me to identify the underlying themes of the emails.





Named Entity Recognition(NER) is a type of Natural Language Processing which takes information from the text and categorises it. This could include names of persons, organisations, locations, times, money, etc.
In both types of emails, ‘ORG’ (organisations) is the most common named entity. This could suggest that both types of emails frequently mention specific companies, institutions, or organisations like Nintendo.
The "PERSON" and “DATE” entities like “Seth” are more present in non-reject emails compared to reject emails. This might suggest that non-reject emails are more personalised or directly addressed more often.
The “TIME” and “MONEY” were present a lot more like “08:37” and “$ 100,000+” in the not_reject emails. This suggests that the conversation is progressing beyond the initial application stage and moving towards more solidified aspects of the job offer compared to the reject emails.
NER helped with my comprehension and the subject of the text. By identifying and categorising entities, NER allowed me to identify the nature of interactions and how they differ between reject and not_reject showing that the non_reject emails are more personalised.

Figure8 - Named Entity

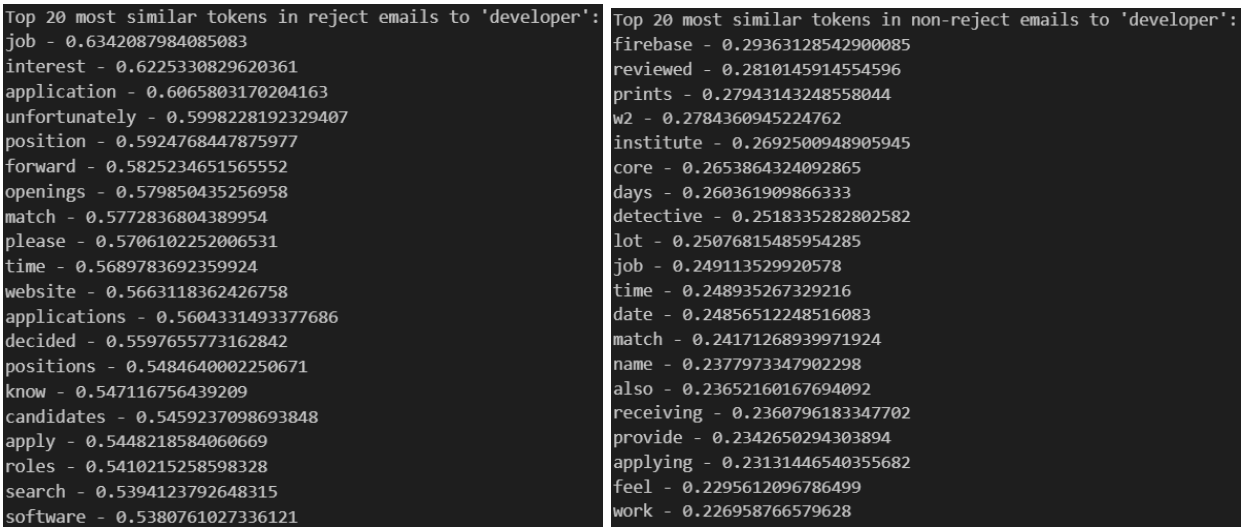


Figure 9 - Word2Vec analysis for the word developer

This is an analysis of the word developer with a modified Word2Vec model. In this model, I changed the vector_size to 150. Changing the vector size in the Word2Vec model changes how words are understood and represented. A larger vector size allows for more detailed word meanings and relationships to be captured, whereas a smaller size might simplify these representations, potentially missing subtle differences. Increasing the vector size helps the model understand words in more detail, catching a wider range of word relationships and meanings, leading to more accurate results when comparing word similarities. The changes in similarity results suggest that the model interprets the text differently after the parameter change. This solidified the idea that the reject emails were more job application focused as the most common words are “job” and “application” and that the not_reject emails are more focused on the technical aspects of the job as the most common words are “firebase” and “reviewed”.