Feature Analysis

1 CONTENTS

2 INTRODUCTION OF THE NEW FEATURES

I have introduced 11 new features to enhance performance. These features are:

- (13) KeywordPresence
- (12) FrequencyNormalized
- (11) KeywordPresencePlusNamedEntities
- (10) Distance
- (09) Synonym
- (08) KeywordOverlap
- (07) KeywordOverlapDistribution
- (06) NamedEntities
- (05) PartialNamedEntities
- (04) LengthPlusFrequencyNormalized
- (03) NamedEntitiesNormalized

Please note that the feature numbers correspond to their identification in the plots.

To evaluate these features, I used accuracy, best score, and buzz ratio as performance metrics. I first tested the model's performance on the provided test data by using it as a dev set locally, and then uploaded the results to Gradescope to check its performance there. The figures presented are based on the accuracy, best score and buzz ratio evaluated on the Gradescope test data. My best accuracy and buzz ratio were achieved by combining six key features, as shown in the figures. Initially, I evaluated individual features and combinations locally, then focused on the most promising combinations.

In the following sections, I will explain the purpose and function of each feature and describe the methodology used to incorporate them into the model. Additionally, I'll outline how each feature was tested to ensure it contributed positively to performance.

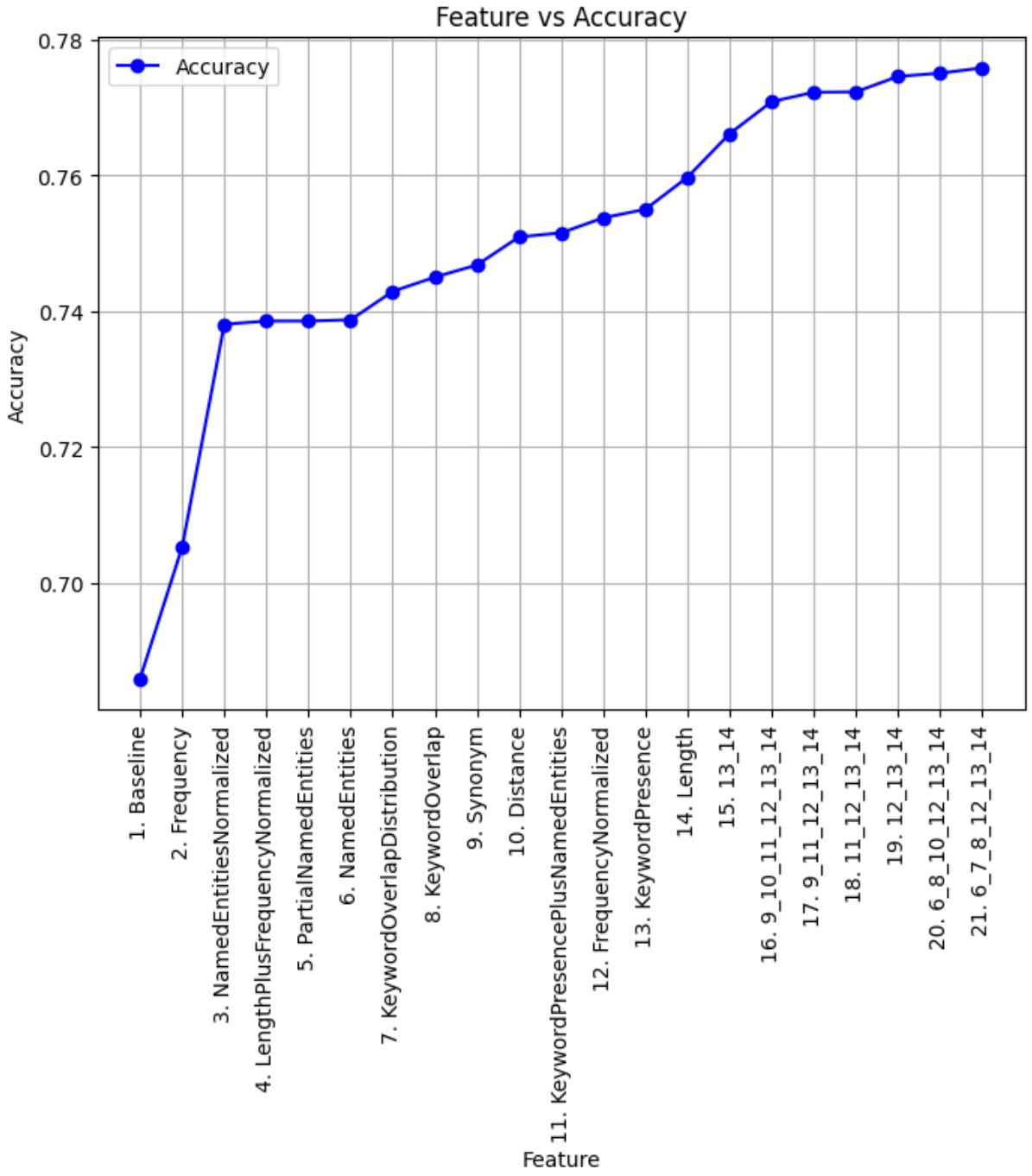
3 EVALUATION METRICS PLOTS

The table below shows the accuracy, best score, and buzz ratio for each feature and combination of features. The last three columns display the improvement from the baseline. Features numbered 3 to 13 represent the newly introduced features, while features numbered 15 to 21 are combinations of multiple features. For example, the notation "6_7_8_12_13_14" represents the combination of those numbered features.

#	Feature	Accuracy	Best score	Buzz ratio	Improvement from the baseline		
					Accuracy	Best score	Buzz ratio
1	Baseline	0.685827	0.306568	0.242650	0.000000	0.000000	0.000000
2	Frequency	0.705227	0.292797	0.245464	0.019400	-0.013772	0.002814
3	**NamedEntitiesNormalized**	0.738040	0.425484	0.328214	0.052212	0.118915	0.085564
4	**LengthPlusFrequencyNormalized**	0.738519	0.425543	0.328483	0.052691	0.118975	0.085833
5	**PartialNamedEntities**	0.738519	0.425603	0.328513	0.052691	0.119035	0.085863
6	**NamedEntities**	0.738698	0.426322	0.328962	0.052871	0.119753	0.086312
7	**KeywordOverlapDistribution**	0.742890	0.426861	0.331327	0.057062	0.120292	0.088677
8	**KeywordOverlap**	0.744985	0.427639	0.332765	0.059158	0.121071	0.090114
9	**Synonym**	0.746782	0.427699	0.333693	0.060954	0.121130	0.091042
10	**Distance**	0.750913	0.429016	0.336417	0.065086	0.122448	0.093767
11	**KeywordPresencePlusNamedEntities**	0.751512	0.427160	0.335788	0.065685	0.120592	0.093138
12	**FrequencyNormalized**	0.753727	0.416262	0.331447	0.067900	0.109694	0.088797
13	**KeywordPresence**	0.754985	0.428537	0.338213	0.069158	0.121969	0.095563
14	Length	0.759715	0.429256	0.340938	0.073888	0.122687	0.098288
15	13_14	0.766122	0.430274	0.344650	0.080295	0.123705	0.102000
16	9_10_11_12_13_14	0.770852	0.416262	0.340010	0.085025	0.109694	0.097359
17	9_11_12_13_14	0.772229	0.417819	0.341477	0.086402	0.111251	0.098826
18	11_12_13_14	0.772289	0.417939	0.341566	0.086462	0.111371	0.098916
19	12_13_14	0.774564	0.420154	0.343812	0.088737	0.113586	0.101162
20	6_8_10_12_13_14	0.775043	0.415604	0.341776	0.089216	0.109035	0.099126
21	6_7_8_12_13_14	0.777379	0.413867	0.342075	0.091551	0.107299	0.099425

3.1 ACCURACY

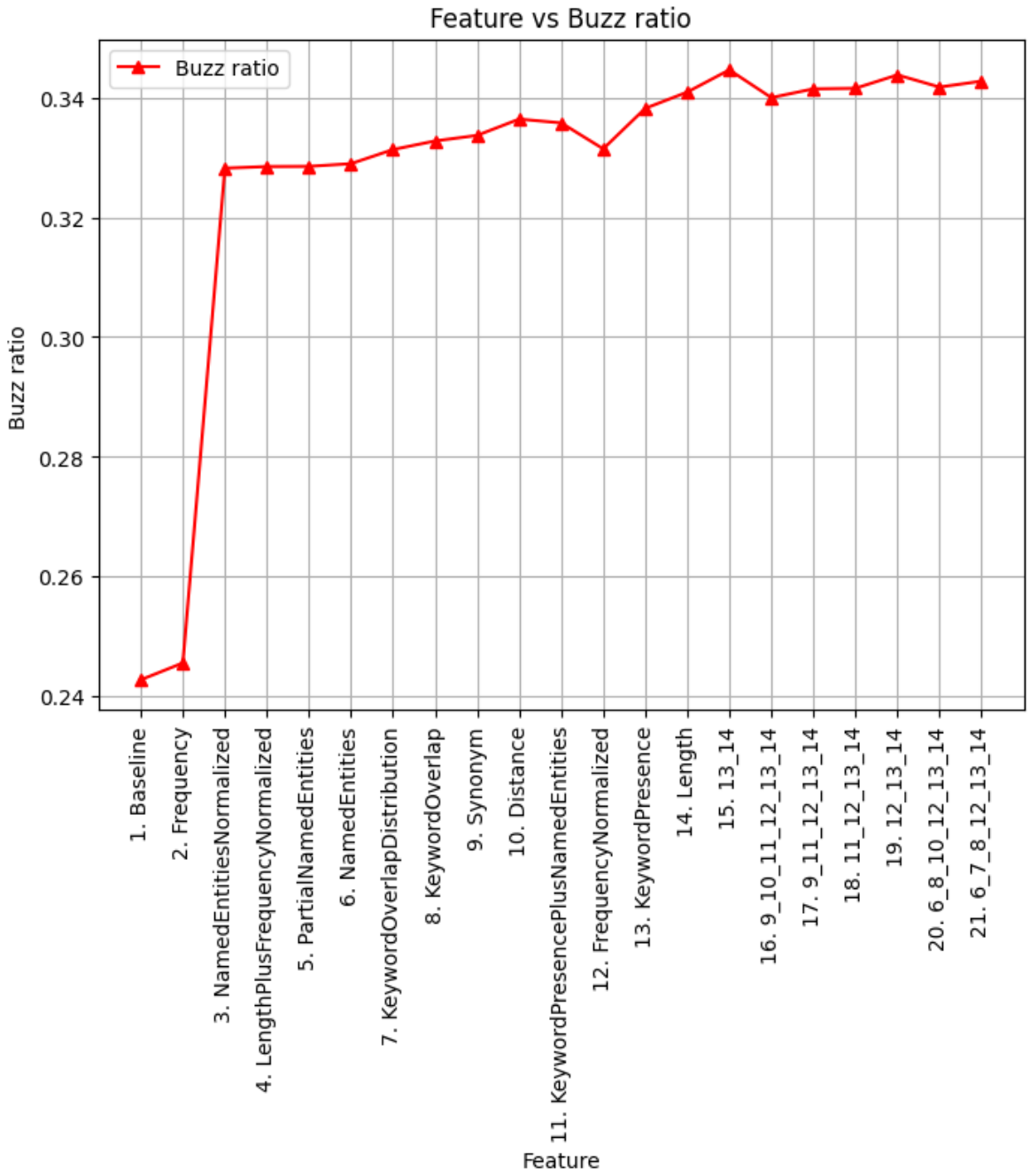
The plot below shows the accuracy for each feature and combination of features, evaluated on the Gradescope test data.



From this plot, you can see how each feature improved significantly from the baseline. It also shows the performance of each feature in increasing order, allowing me to identify the best features for my classifier. The final feature combinations were selected based on the development set (local test data) by analyzing different combinations.

3.2 BUZZ RATIO

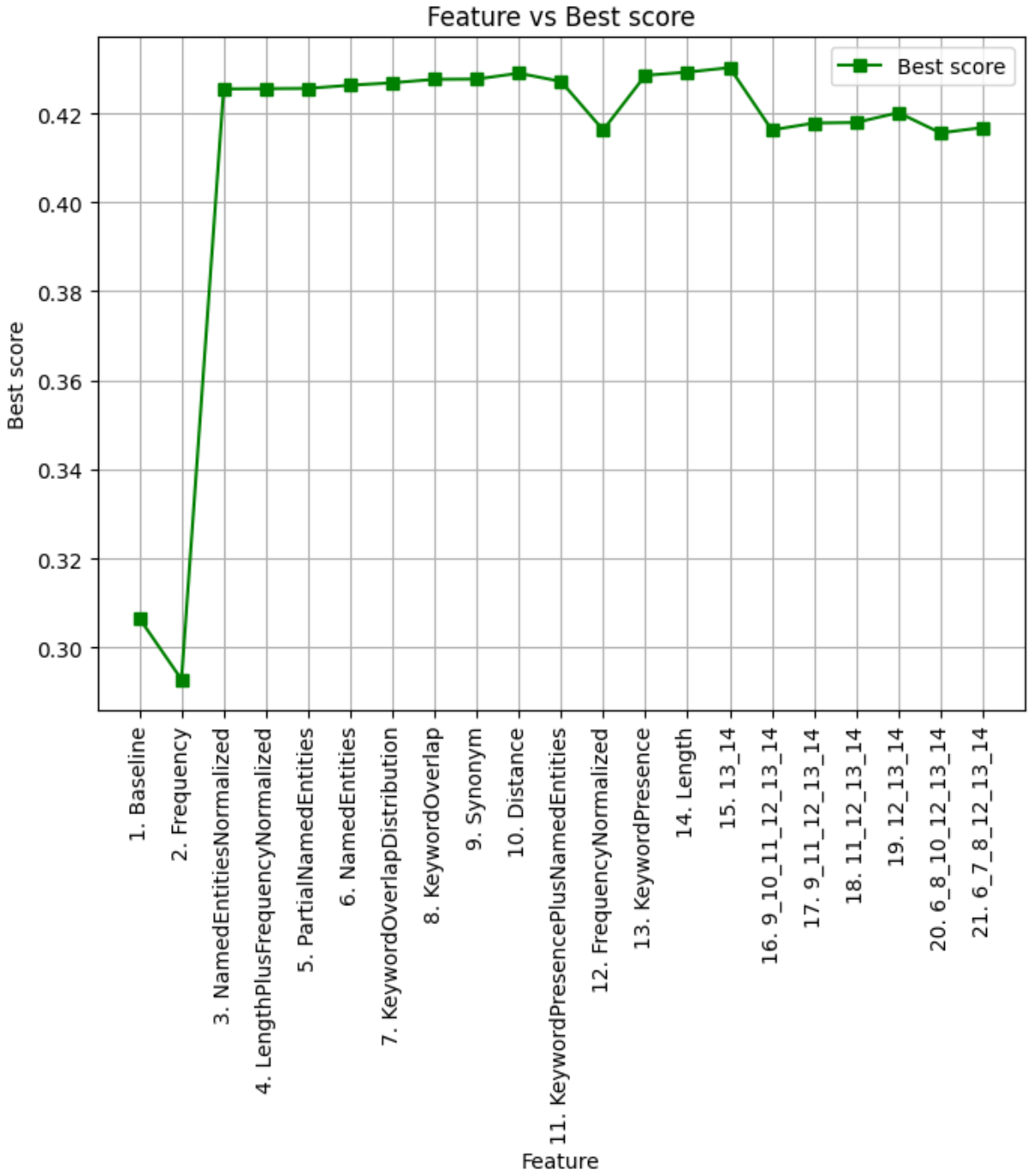
The plot below shows the buzz ratio for each feature and combination of features, evaluated on the Gradescope test data.



From the plot, you can see how the buzz ratio improved significantly for each feature. The features are ordered based on increasing accuracy. While the buzz ratio improved for the first few features up to the 10th feature, it then declined until the 11th feature before increasing again. The best buzz ratio was achieved from the combination of the Length and KeywordPresence features.

3.3 BEST SCORE

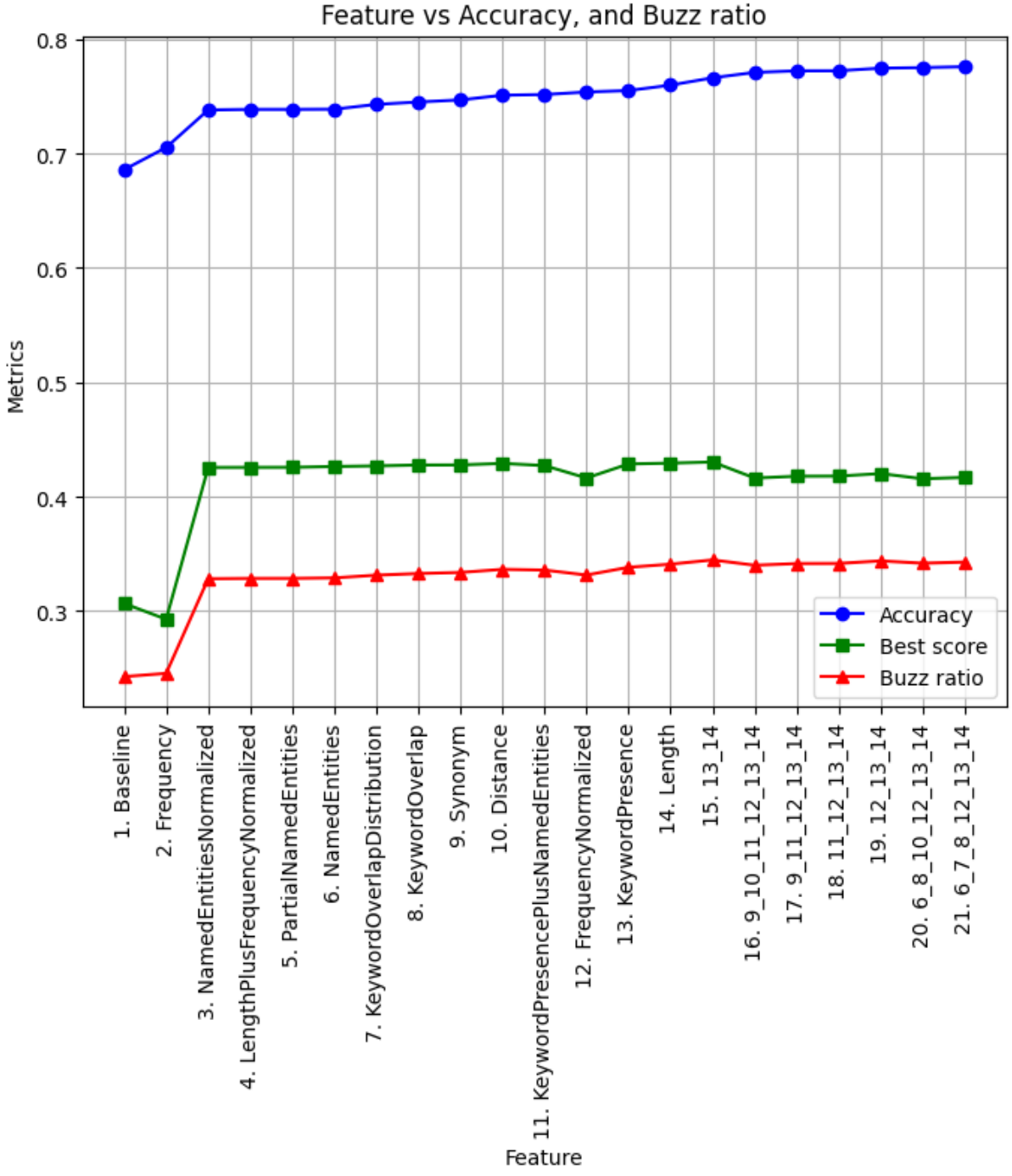
Here, I have evaluated the best score which means true positives.



This follows a similar pattern to the buzz ratio plot.

3.4 ACCURACY, BEST SCORE AND, BUZZ RATIO

The plot below shows the combination of the above three metrics in a single view. Based on this, I chose the 21st feature, which represents the highest accuracy combination.



4 FEATURE ANALYSIS AND COMPARISON

4.1 (12) FREQUENCY NORMALIZED FEATURE

Feature Description: Z-score normalization of the frequency of each guess in the training set. Essentially, I applied z-normalization to the frequency feature instead of log normalization.

Methodology: I aimed to evaluate how different normalization techniques affect the classifier's performance. To do this, I replaced log normalization with z-normalization. After testing it locally, I observed a significant improvement in both metrics. For tracking the frequency of guesses, I utilized the same training data used in the frequency feature.

Evidence: Refer to features 2 and 12 in the plots. The accuracy improved from 0.705 to 0.754 due to this normalization. The buzz ratio increased by 0.086, and the best score improved by 0.123 with this feature. This is the second-best feature among my features, demonstrating how normalization impacts performance. Additionally, this feature contributes to the chosen feature combination.

4.2 (13) KEYWORD PRESENCE FEATURE

Feature Description: This feature checks whether at least one keyword in the guess is present in the run text of the question. It is a binary feature that returns 1 if true and 0 if not.

Methodology: The creation of this feature was based on analyzing the training data. I observed that most keywords in the guess were present in the question for many correct guesses. Additionally, I wanted to investigate how a binary feature could affect the performance of the classifier compared to a continuous feature.

Evidence: This feature is the best among all the newly introduced features. When compared to the baseline, the improvement in accuracy was 0.069, while the buzz ratio improved by 0.096, and the best score increased by 0.122, indicating a significant enhancement. In contrast, when comparing this feature with the frequency normalized feature (which is the second-best), the improvement in accuracy was only 0.001, and the improvement in buzz ratio was 0.007. Moreover, this feature contributes to the chosen feature combination.

4.3 (10) DISTANCE FEATURE

Feature Description: This feature assesses the similarity between the guess and the run using Python's built-in package, ‘difflib’, specifically the ‘SequenceMatcher’. It measures how closely two sequences resemble each other, making it useful for string matching, finding duplicate text, and detecting subtle differences between sequences. It returns a similarity score between 0 and 1.

Methodology: Since the classifier generates guesses based on the tokens in the question, I wanted to reflect this in a feature by examining the differences between the tokens in the question/run and the tokens in the guess. While evaluating this feature, I limited the tokens to five from the run to manage training time and improve validation accuracy. This feature outputs the total difference ratios for each token in the guess compared to the run sequence.

Evidence: When comparing this feature with the baseline, the improvement in accuracy was 0.065, the buzz ratio improved by 0.094, and the best score increased by 0.122, indicating a significant enhancement.

4.4 (9) SYNONYM FEATURE

Feature Description This feature checks for synonyms of the guess tokens and determines whether they are present in the run. It outputs the count of synonyms found in the run after applying log normalization.

Methodology: This feature was developed based on an analysis of the training data, where I noticed that some guesses relied on synonyms of the question tokens. I utilized the 'nltk WordNet' package to retrieve synonyms for the guess tokens. The evidence section demonstrates how this feature improves performance.

Evidence: When comparing this feature with the baseline, the improvement in accuracy was 0.061, the buzz ratio improved by 0.091, and the best score increased by 0.121, indicating a significant enhancement. The performance of this feature is similar to that of the distance feature based on the metrics. I anticipated that this feature would enhance the KeywordPresence feature, considering it an advancement by incorporating synonyms. However, as discussed in the README, I suspect that its performance may be lower than that of the KeywordPresence feature due to the infrequent occurrence of synonyms, which could also increase false positives.

4.5 (6) NAMEDENTITIES FEATURE

Feature Description: This feature identifies named entities within the guess tokens and returns the count after applying log normalization.

Methodology: This feature was developed based on an analysis of the training data, where I observed that some guesses contained named entities such as names, places, and objects. To recognize these tokens, I utilized the spacy "en_core_web_sm" package.

Evidence: This feature alone results in a 0.053 improvement in accuracy, a 0.086 improvement in buzz ratio, and a 0.120 improvement in the best score compared to the baseline. Overall, this feature enhances the baseline performance and contributes to the final chosen feature combination.

4.6 (5) PARTIALNAMEDENTITIES FEATURE

Feature Description: This feature is a partial version of the NamedEntities feature; it specifically checks for named entity types such as "person," "organization," "nationality," "religious," and "political groups" within the guess tokens, returning the count after applying log normalization.

Methodology: The development of this feature was informed by an analysis of the training data, where I observed that these three types of named entities occurred more frequently in the guesses. To recognize these tokens, I utilized the spacy "en_core_web_sm" package.

Evidence: As expected, this feature reflected lower performance than the NamedEntities feature, as it only considers specific types of named entities. This feature alone results in a 0.053 improvement in

accuracy, a 0.086 improvement in buzz ratio, and a 0.119 improvement in the best score compared to the baseline.

4.7 (3) NamedEntitiesNormalized feature

Feature Description: This feature is an advanced version of the NamedEntities feature, as it normalizes the guess before checking for named entities.

Methodology: The development of this feature was based on an analysis of the training data, where I noticed that some guesses contained tokens such as "_", "the", "an", and "a" at the beginning. These tokens can be misleading when checking for named entities, so I believed that removing them from the guess might enhance the feature's performance.

Evidence: This feature alone results in a 0.052 improvement in accuracy, a 0.086 improvement in buzz ratio, and a 0.119 improvement in the best score compared to the baseline. However, its performance was lower than that of the NamedEntities feature, as this feature may be more aggressive because of normalization approach.

4.8 (11) KeywordPresencePlusNamedEntities feature

Feature Description: This feature enhances the NamedEntities feature by incorporating the effect of the KeywordPresence feature, using a logic aimed at improving the overall performance of the NamedEntities feature.

Methodology: The development of this feature was informed by analyzing the performance metrics of both the KeywordPresence and NamedEntities features. I aimed to enhance the NamedEntities feature by outputting a binary value of 1 only when there are named entities present in the guess and at least one keyword from the guess is also found in the run.

Evidence: This feature alone results in a 0.066 improvement in accuracy, a 0.093 improvement in buzz ratio, and a 0.121 improvement in the best score compared to the baseline. It ranks as the third best feature in my new feature set, significantly enhancing the NamedEntities feature's performance.

4.9 (8) KeywordOverlap feature

Feature Description: This feature enhances the KeywordPresence feature by providing a keyword overlap count between the guess and the run, rather than a binary output. The count is returned after applying log normalization.

Methodology: The development of this feature was based on the performance metrics of the KeywordPresence feature and an analysis of the training set. The goal was to transition from a binary feature to a continuous one to capture more information.

Evidence: This feature alone results in a 0.059 improvement in accuracy, a 0.090 improvement in buzz ratio, and a 0.121 improvement in the best score compared to the baseline. However, it did not enhance the KeywordPresence feature because the number of overlaps does not significantly contribute to the accuracy of the guess. Nonetheless, this feature contributes to the final feature combination along with the KeywordPresence feature.

4.10 (7) KeywordOverlapDistribution feature

Feature Description: This feature enhances the KeywordOverlap feature by utilizing a training set to determine the overlap count, rather than relying solely on the overlap count from the given test run data.

Methodology: The objective was to train the feature using an additional dataset, aiming to improve its performance by leveraging more contextually relevant information.

Evidence: This feature alone yields a 0.057 improvement in accuracy, a 0.089 improvement in buzz ratio, and a 0.120 improvement in the best score compared to the baseline. However, it did not enhance the KeywordOverlap feature, likely because the training data is based on a different context than the test data. Therefore, this feature may be more suitable for datasets sharing the same context. Nonetheless, this feature contributes to the final feature combination along with the KeywordOverlap feature.

4.11 (4) LengthPlusFrequencyNormalized feature

Feature Description: This feature combines the effects of the length of the guess with the FrequencyNormalized feature.

Methodology: The objective was to assess the performance of the classifier by directly combining these two features. This evaluation was conducted early in the process, before developing the other new features mentioned above.

Evidence: This feature alone yields a 0.053 improvement in accuracy, a 0.086 improvement in buzz ratio, and a 0.119 improvement in the best score compared to the baseline. However, the performance of this feature was worse than that of both individual features.

5 The method of choosing feature combinations

To choose the best combination, I analyzed the performance of all the features listed above based on accuracy, best score, and buzz ratio. After evaluating different combinations using the development set, I identified the top seven combinations. Subsequently, I assessed the performance of these combinations using the Gradescope test data. The improvements in performance for these combinations can be observed in the provided plots.

6 Final Feature combination

The best combination of features included Length, KeywordPresence, FrequencyNormalized, KeywordOverlap, KeywordOverlapDistribution, and NamedEntities. I believe this combination has a stronger correlation with the output guess, which is why it improved the baseline by 0.09 in accuracy, 0.1 in buzz ratio and 0.11 in best score.