Using Regression Models to Predict Chatbot Arena Winners and Prompt Hardness Scores

Carl Illustrisimo and Sonja Bumann

May 17, 2025

Abstract

We perform exploratory data analysis on two data sets containing information on chatbot performances for various prompt topics. We then build a logistic regression model to predict the winning models for a given prompt, achieving an accuracy score of 54.2%. We also employ linear regression to predict the hardness score of a given prompt on a scale from 1 to 10, reaching a root mean square error of 2.39. Link to Youtube video

1 Introduction

Artificial intelligence (AI) chatbots are taking on a progressively larger role in how humanity automates tasks, with the most popular chatbots today being large language models (LLMs). LLMs showcase remarkable natural language processing capabilities, and perform well on a wide variety of tasks such as ”classification, question-answering, logical reasoning, information extraction,” and so on [6]. LLMs therefore offer enormous potentials for aiding in human tasks—but they also risk the potential for automating, and even amplifying human biases. There has been much documentation of the kinds of biases LLMs replicate, including gender, racial, and other systemic biases.

Because of this, direct human interventions, or reinforcement learning through human feedback (RLHF) has become the prevailing approach to ensure LLMs align with human values [11]. However, these approaches carry their flaws too, with researchers documenting how judges can introduce biases into LLMs explicitly or implicitly [3, 2]. Since judge's themselves can introduce biases, a deep understanding of how LLMs perform relative to each other and where their strengths and weaknesses lie is of significant importance. Research has shown that human thinking intentionally combined with machine learning and statistical modeling can help to remedy biases [7], and as such, we venture to gain understanding by modeling predictions of winners and difficulty in LLM comparisons.

Modeling emergent human preferences in human computer interactions is complex as there are many factors that shape an encounter between the two, such as topic knowledge, sentiment, subjectivity, and clarity—to name a few. There has already been focus on developing tools to study the performances of chatbots, including large and diverse datasets [9] to serve as benchmarks for emerging models. The data we use specifically was derived from ChatbotArena, a platform that has emerged as one of the aforementioned tools to study chatbot performance. ChatbotArena produces rich, quality data—which Chiang et al. find to accurately represent the diversity of human opinion through their study of the data's predictive capability and inter-rater agreements [4]. Already, important work in modeling LLM quality has been done by utilizing LLMs as judge [10]. Thus, in our work, we explore human-chatbot interactions with a large dataset of chatbot pairwise combinations and uncover qualitative patterns such as topics and textual similarity. Within the literature, discussions of subjective prompt difficulty and simulating human preference for one model over another are frequent research questions. Thus, we leverage the patterns we uncovered to model winning chatbots and prompt hardness scores with logistic and linear regression respectively. Our findings provide insight into the various factors that affect model performance which can help guide their improvement.

2 Description of Data

Our main data set contains details of 25,282 pairwise matches between two of twenty selected AI chatbots. This includes the prompt, models and their responses, judge information, and the determined winning model for each match. The data set has been preprocessed to remove multiple rounds of conversations, non-English conversations, and toxic or harmful content. We found and removed ten duplicate questions. Additionally, we have access to an auxiliary dataset of 256-dimensional text embeddings for the prompts and each model response.

Our second main dataset contains three topic and hardness score assignments for each prompt that were generated by GPT-3.5. We carried out initial data cleaning by removing duplicate rows and those with missing entries, of which there were 10 and 26, respectively. Of the missing entries, we were able to salvage 4 by extracting the available scores and topics from the provided raw data. We calculated the average hardness scores given the available scores for each prompt (the salvageable rows had either one or two), as we use this as our outcome variable in our linear regression model for Task B.

We analyzed the topic data and found the most frequent topics to be ”problem solving,” ”creativity,” and ”factual accuracy” (see page 11). We also analyzed the prompt and response lengths and found the distributions to be significantly right-skewed. The log transformations are shown on page 11. The average prompt length was 196 characters whereas the max was 2560. However, we chose not to remove any outliers because the test data revealed to follow a similar distribution with large outliers. We also investigated the use of punctuation in the prompts, which was similarly shown to be right-skewed. This made visualization of their distributions difficult, so we include a graph of log transformed response and prompt lengths to showcase their similarity (see page 11). The mean number of question marks used in a prompt was 0.56 with a maximum of 40. Additionally, we performed a sentiment analysis of the prompts using the ”distilbert-base-multilingual-cased-sentiments-student” model. We expect such information to be useful because we anticipate there to be a correlation between hardness scores and high levels of sentiment. We found the distribution of the sentiment score, grouping positive and negative sentiments together, to once again be right-skewed with an average of 0.51 (see page 12). This corresponds to generally neutral-leaning sentiments across the prompts. Finally, we looked into the subjectivity of prompts because, similar to sentiment, we would expect prompts with more opinionated language to be labeled as more difficult by a chatbot. The prompt distribution had median and mean subjectivity scores of 0.24 and 0.27, respectively, with a significant peak near 0.0 which indicates a higher representation of factual/objective prompts (see page 12).

3 Methodology

3.1 Task A: Prediction of Winning Model

For the task of predicting a winning model, we decided to use a multinomial logistic regression model with a bias term, using L1 regularization and a C parameter of ∼0.237. An earlier iteration of our model intended on using L2 regularization, which was found determined through manual experimentation with various features to outperform L1 regularization. For this final instantiation of the model however, we took a more rigorous approach and utilized GridSearchCV to evaluate every possible combination of L1 and L2 for evenly spaced C values on a log interval scale. As a result, we found that L1 with our aforementioned C parameter of ∼0.237 gave us our highest model performance.

We initially selected logistic regression as a baseline model to evaluate other classifiers against, as the task of classification with logistic regression allows for easy interpretation of our coefficients, and thus the selection of particularly predictive or weak features. Yet after taking our feature-sets into experiments with other models such as gradient-boosted decision trees and random forest classifiers, there were often little to no performance gains for much more compute. This was confirmed over 5+ K-Fold CV iterations ran with each model, respectively. We thus decided then that due to its simplicity and interpretability, logistic regression with well engineered features would be an optimal choice for the task. Comparing the two models on identical features elucidates this, where the average accuracy of logistic regression and gradient-boosted decision trees over 4 folds are ∼0.56 and ∼0.55 respectively. These findings are consistent with current research concerning machine learning methods in critical classification tasks, such as healthcare, have been found to show no performance benefits over logistic regression [5].

Deciding upon features was a matter of building from the ground up for our logistic regression model. We first began by identifying general trends within the data, such as the distributions of each outcome label (how many winners, ties, tie bothbads), and what relationships they shared with simple features such as prompt and response length (see page 13). If even a minor trend was noticed, the feature was tested first in its raw form, then tested again after a normalizing transformation. Following this logic, we chose to include the cube root of response lengths and the log of prompt lengths. Since Task A is a task concerned with classification of a victor, a subjective task, we wanted to next explore features that captured qualitative aspects of responses and prompts. A summary of the final features selected for our logistic regression thus far is listed below:

Each feature that is transformed was normalized by observing the effects of a respective transformation on the shape of a features distribution. An example of the figures considered is included (see page 13). Exploring qualitative aspects first brought us towards the creation of response similarity, which is a feature that we intend to capture the closeness of two given

Table 1: Selected features for the multinomial logistic model

Final Winner Prediction Features
Log prompt length
Cube root response A length
Cube root response B length
Response similarity cubed
Response similarity to prompt cubed
Identical response indicator
Log Bradley Terry model scores
Hardness score
50 k-means clustering labels (OHE)
10-dimensional topic embeddings

responses. From our EDA, we found that responses which tied tended to be more similar to each other than not, and captured this through calculating the cosine similarity of each response. Our identical response indicator was motivated by the same logic, as a subset of responses which were identical always tied, and we found that the inclusion of this feature improved model performance despite response similarity seemingly capturing the same kind of information. During our EDA, we wanted to figure out what kinds of responses received the label "tie (bothbad)," and found that these responses tended to be off topic or just incorrect. Thus, as a measure of being on topic, we also calculated the cosine similarity of each response to their respective prompt.

Further features explored were bradley-terry scores, which we found to add more predictive value to our model than ELO. Our model includes log bradley-terry scores to more closely match the form of a bradley-terry model, which uses log bradley terry scores and a form similar to logistic regression. Lastly, we worked to capture similarity between questions by clustering our prompt embeddings with K-Means clustering via the Sklearn library. The initial goal of this clustering was to create individual models per cluster, but we found that the one hot encoding of each cluster as its own feature proved to be a more stable and predictive use of the feature. To wit, training individual models per cluster for our classification task resulted in some low accuracies around 40% (likely underfitting), and high accuracies such as 77% (likely overfitting). Utilizing clusters as a label therefore achieved a good balance between bias and variance, whereas clusters for individual models created extremely high variance on some models and high bias on others.

To capture further complexity, we also created embeddings of the topics, motivated by the idea that some topics were redundant and overrepresented, whereas others were sparse and outliers (see page 11). Therefore, we aimed to embed the topics, so that their contextual similarities would be more apparent within the model, and topics that may be outliers as a raw string become part of a more global structure. We achieved this embedding first through the HuggingFace transformers library, particularly using the "all-MiniLM-L6-v2" model. This generated for us 360+ dimensional embeddings that we then chose to reduce down into 10 via UMAP, picking UMAP over PCA precisely due to its ability to preserve context via global and local structures [8]. Following this, we utilized each dimension of the embedding as its own feature in the model, a method similar to concatenation of word vectors onto each other, which while not ideal allows the model to parse the information of the vectors relative to each other. We experimented with, per the cited paper, different levels of neighbors to identify more local or global trends in the topics overall—and found that at 25 neighbors our classification process was greatly improved, moving from a validation accuracy of ∼0.50 to ∼*0.55* immediately.

The usage of topic embeddings was also done for our test set, which required the imputation of certain rows in the dataset which were missing their topics. Thankfully, they were all duplicates of the same prompt, and thus we imputed the missing topics from the test set with topics from the same prompt that were present in the data before embedding.

Many unsuccessful features were explored for this model, including, but not limited to: sentiment scores for each response, one hot encodings of the model names, ELO based win probabilities, Bradley-Terry based win probabilities, and so on.

3.2 Task B: Prediction of Hardness Score

For predicting the hardness scores of our prompt data, we chose to implement an ordinary least squares (OLS) model with a bias term and without regularization. We used the average hardness score data as our outcome variable. Because OLS works for continuous data, we rounded the predicted scores to the nearest integer. We chose this model because much of our potential candidates for features are numerical that have expected linear relationships with the hardness score, such as the prompt length. We did not opt for implementing regularization because 4-fold cross validation (CV) of our training data suggested that the model was stable and not prone to over-fitting.

When deciding on which features to incorporate into our model, we calculated a heatmap of Pearson pairwise correlations to test for correlation between different features (see page 14). We do this so as to mitigate multicollinearity in our model and prevent linear dependence issues in the matrix equation. For example, we opt to use the log prompt length and omit the log prompt word count due to their high correlation coefficient (c = 0.98) We also log transformed it due to the high outliers.

Table 2: Selected features for the linear regression model

Final Hardness Score Features
Log prompt length
50 k-means clustering labels (OHE)
10-dimensional topic embeddings
256-dimensional model A response embeddings

The final features we chose for our model are listed in Table 2. In addition to the log prompt length, we performed a k-means clustering (k=50) of the prompt embeddings and one-hot-encoded the cluster labels as a feature. We then assigned the test data to clusters by finding the shortest Euclidean distance with cluster centroids. Clustering the embeddings groups similar types of questions and therefore retains some topic information. Therefore, certain clusters should have a positive affiliation with different topics, which could be more or less difficult. In fact, adding these cluster labels to the model (versus just including the log prompt length) decreased the RMSE from ∼4.3 to ∼2.8.

We also calculated high-dimensional topic embeddings using HuggingFace's transformers library, particularly the "all-MiniLM-L6-v2" model, and reduced the dimensionality to 10 using UMAP. Specifically, we concatenated the three topic labels to form a single sentence and embedded those. We used each embedding dimension as its own feature. One honest practical reason for this was because we attempted to use k-means cluster again but most of the clusters did not have test data assigned to them, which conflicted with the sci-kit linear regression model. Embedding the topics themselves in this way compared to the explicit prompts captures topic categorization more directly.

We found that using the above features resulted in a model that was too simple with a test RMSE of 2.52 (just outside the full points threshold). As a result, we also added the model A response embedding elements as features, which significantly increases the size of the design matrix. We chose to omit the model B response embeddings because they should contain similar information as those for model A. Indeed, we analyzed the CV RMSE standard deviation and found that it increased greatly by the addition of the model B embeddings, which at the same time increases the number of columns in the design matrix by almost a factor of 2.

Finally, we analyzed a number of features that we eventually did not include in our model. First, and most interestingly, we studied the absolute sentiment scores of each prompt. We expected them to have a strong positive correlation with the hardness score because higher values correspond to more "emotional" responses, which we would expect to challenge chatbots compared to objective, factual questions. However, this variable had close to zero correlation with the hardness score. This is also likely due to the fact that the distribution of the sentiment scores were more neutral than polar. Similarly, we found that the prompt subjectivity scores had a very low correlation (c = 0.16) with the hardness score and therefore omitted them from our model.

4 Results

4.1 Task A: Prediction of Winning Model

Our initial L2 regularized logistic regression model using the features in Table 1 achieved a training accuracy of ∼0.56 ± 0.002, with uncertainty from a 10-fold CV range at random seed 42. Application of this model to the test data resulted in an accuracy of 0.5400, directly on top of the threshold for full points. Our new model, which is L1 regularized and running a C parameter of ∼0.237, achieves a training accuracy of ∼0.56 and test set accuracy of **0.5419**, pushing us past the final threshold into full points for our logistic regression classifier.

Table 3: Optimized accuracy for our logistic regression model using the features in Table 1

Training Accuracy	Test Accuracy
∼0.56 ± 0.002	0.5419

Accuracy ended up being the primary metric by which we evaluated this model, diverging from earlier uses of metrics such as log-loss. The main motivation for this is in the ease of interpretability accuracy offers, especially in determining overfitting (an accuracy of ∼90% while using clustering models signaled to us that something went wrong, for example). Additionally, utilizing accuracy during our runs allowed us to identify points in certain features lead to a decrease in predictive ability, e.g. increases in bias, such as the introduction of model names. Per suggestions from our peers however, we have also included a confusion matrix to evaluate the model, which elucidates where our model performs best and where it could further improve (see page 14). We explore the implications of the confusion matrix later in our discussion.

4.2 Task B: Prediction of Hardness Score

Using the features in Table 2 for our model, we achieved a training RMSE of 1.35 ± 0.01 using the uncertainty from 4-fold CV. Applying to the test data, our predictions reach a RMSE of 2.387, which is within the full points range.

Table 4: Optimized RMSE for our linear regression model using the features in Table 2

Training RMSE	Test RMSE
1.35 ± 0.01	2.387

We also investigated the resulting coefficients for each feature of the model, which is summarized in Table 5. Note that we only include the top 5/50 prompt embedding cluster labels, top 5/256 model A response dimensions, and top 5/10 topic embeddings for readability. Because we have not standardized our features, it is difficult to concretely interpret the relative importance of each feature by comparing their coefficient magnitudes. However, the signs of the coefficients should provide insight on whether the feature should reduce or increase the hardness of a prompt. For example, the prompt embedding cluster 5 has a very large negative coefficient of -2.818, indicating that a prompt assigned to that label should have a decreasing effect on the hardness score. Indeed, this appears to be the case; inspecting topic labels for prompts in that cluster show a strong theme relating to greetings (”How are you?”, ”How do you feel?,” etc.) which also have a trend of low scores of around 1-2. Similarly, prompts in cluster 31 have a relatively large positive correlation with the hardness score (coeff = 1.248) and almost exclusively have a variant of the prompt ”What is the meaning of life?” This type of philosophical question solicits a much more complex response than simple greetings due to its extremely subjective nature. As a result, these questions receive scores mostly in the range of 8-9. These observations highlight one of the significant advantages of linear regression being the high interpretability of the model.

To deepen our understanding of different model performances, we chose to compare our linear regression model with the XGBoostRegressor model using 100 estimators. We chose this model because it does not assume a linear relationship between the feature and output variables and thus one might expect it to better handle our categorical features. Using the identical features as in OLS, we achieved a training RMSE of 1.23 ± 0.025 with 8-fold CV, which is a lower average than that obtained with OLS but with a higher standard deviation. This makes sense as the testing RMSE was quite worse at 3.07, indicating the

Table 5: The LR coefficients for each feature. Only the top 5 prompt embedding clusters, model A response embedding dimensions, and topic embedding dimensions are shown for readability.

Model Feature	LR coefficient
prompt embedding cluster 5	-2.818
model a dim 31	1.675
model a dim 110	1.421
model a dim 119	1.408
model a dim 152	1.274
prompt embedding cluster 31	1.248
model a dim 120	-1.180
prompt embedding cluster 12	0.888
prompt embedding cluster 21	-0.794
prompt embedding cluster 49	-0.738
Log prompt length	0.538
topic dim 5	-0.123
topic dim 2	-0.114
topic dim 7	0.056
topic dim 9	-0.056
topic dim 0	0.048

presence of over-fitting. We implemented minor hyperparameter tuning using scikit-learn's GridSearchCV and looked for optimal values of the L1/L2 regularization terms, the max tree depth, n_estimators and min_child_weight. However, despite using larger regularization terms, we were not able to meaningfully improve the test RMSE, suggesting that further parameter tuning is required. This need for hyperparameter tuning highlights that, given the nature of the task at hand, simpler models may be more appropriate. This in combination with the interpretability that OLS offers make a strong case for its use to analyze this chatbot data.

5 Discussion

5.1 Task A: Prediction of Winning Model

The final test accuracy of our model thus far at 0.5419 suggests we are performing quite well on the accuracy task as delineated by the course, and we have met what is required for full points. Reaching here was by and large thanks to the hyperparamter tuning we underwent with GridSearchCV. However, before then, we were able to sit right on the edge of the final threshold at .54000 accuracy through our engineered features. For our features, the most impactful on the logistic regression model appears to have been the embeddings of the topics. This fell in line with our predictions, as the hope behind creating topic embeddings was to capture the similarity between each possible set of topics for a given response, rather than deal individual topics which had many outliers and overrepresented labels.

However, though we have done well according to the class rubric, there is still quite the room for improvement of our model. As is visible in the confusion matrix, our model performs quite well at predicting ties, but begins to experience some confusion over predicting either model a or model b as a winner. Our logistic regression model struggles the most with but struggles with tie (bothbad). This indicates that further exploration would require uncovering not only what makes ties unique, but also how exactly to quantify that uniqueness. Additionally, we would want to explore the effects of utilizing the entirety of possible word embeddings (either through averaging or concatenation). Our peers who we reviewed experienced a good level of success with this approach, however experiments with it so far proved less fruitful for our model. It does however, logically follow that to deal with the subjective qualities of the text, representing the text in a much richer format both through embeddings and in the model would better capture signals of victory.

5.2 Task B: Prediction of Hardness Score

Our final RMSE of 2.38 suggests that, while our model performs well according to the class rubric, there is still room for improvement. Our model is likely too simple and further exploration of different features is necessary. Specifically, our model would likely benefit from more continuous data rather than one-hot-encoded features such as the prompt cluster embeddings. This is because the numerical data achieves a higher level of granularity in describing the relationship between features and the output. One avenue that we have yet to explore for this model is investigating polynomial transformations of features, which may or may not exhibit better linear relationships with the hardness score. Another area that could likely improve our model is using standardization on our numerical data to make the scaling consistent and prevent overly dominant features. This would also allow us to more directly compare the relative importance of features which could provide further insight into subsequent feature engineering.

6 Conclusion

LLMs are, as mentioned earlier, increasingly present in the public domain—and their evaluation in ChatbotArena is one such way that they are able to become more efficient, more robust, more complex, and less biased machines. As they advance, the term black box is oft repeated to denote how difficult they are to interpret. This term is both in regard to the ability to interpret their internal logics, as well as what classifies good performance. It is thus, with a level of irony, that we are able to investigate their outputs through ubiquitous and well understood methods like logistic and linear regression. Like the study of other complex systems, this is in line with a growing push from LLM researchers to know them through their observed behaviors [1]. While we feel we have made good progress on this research question per the class rubric, we also acknowledge that there is a strong sense that our models could be further refined. As we look to the refinement of modeling the quality of LLM outputs, it may be the case that even simpler models would prevail yet. The principle of parsimony suggests such simpler models, and perhaps quality has a clear measure which captures its attributes and complexity through methods that have long existed.

References

[1] Bai, X., Wang, A., Sucholutsky, I., and Griffiths, T. L. Explicitly unbiased large language models still form biased associations. *Proceedings of the National Academy of Sciences 122*, 8 (2025), e2416228122.

[2] Barnhart, L., Bafghi, R. A., Becker, S., and Raissi, M. Aligning to what? limits to rlhf based alignment, 2025.

[3] Chen, G. H., Chen, S., Liu, Z., Jiang, F., and Wang, B. Humans or LLMs as the judge? a study on judgement bias. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing* (Miami, Florida, USA, Nov. 2024), Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, Eds., Association for Computational Linguistics, pp. 8301–8327.

[4] Chiang, W.-L., Zheng, L., Sheng, Y., Angelopoulos, A. N., Li, T., Li, D., Zhang, H., Zhu, B., Jordan, M., Gonzalez, J. E., and Stoica, I. Chatbot arena: An open platform for evaluating llms by human preference, 2024.

[5] Christodoulou, E., Ma, J., Collins, G. S., Steyerberg, E. W., Verbakel, J. Y., and Van Calster, B. A systematic review shows no performance benefit of machine learning over logistic regression for clinical prediction models. *Journal of clinical epidemiology 110* (2019), 12–22.

[6] Gallegos, I. O., Rossi, R. A., Barrow, J., Tanjim, M. M., Kim, S., Dernoncourt, F., Yu, T., Zhang, R., and Ahmed, N. K. Bias and fairness in large language models: A survey. *Computational Linguistics 50*, 3 (2024), 1097–1179.

[7] Kleinberg, J., Ludwig, J., and Mullainathan, S. A guide to solving social problems with machine learning. *Harvard Business Review 8*, 2 (2016).

[8] McInnes, L., Healy, J., and Melville, J. *Basic UMAP Parameters*, 2025. UMAP documentation, accessed April 10, 2025.

[9] Zheng, L., Chiang, W.-L., Sheng, Y., Li, T., Zhuang, S., Wu, Z., Zhuang, Y., Li, Z., Lin, Z., Xing, E. P., Gonzalez, J. E., Stoica, I., and Zhang, H. Lmsys-chat-1m: A large-scale real-world llm conversation dataset, 2024.

[10] Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E. P., Zhang, H., Gonzalez, J. E., and Stoica, I. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023.

[11] Zhu, Y., Sun, C., Yang, W., Wei, W., Tang, B., Zhang, T., Li, Z., Zhang, S., Xiong, F., Hu, J., et al. Proxy-rlhf: Decoupling generation and alignment in large language model with proxy. *arXiv preprint arXiv:2403.04283* (2024).

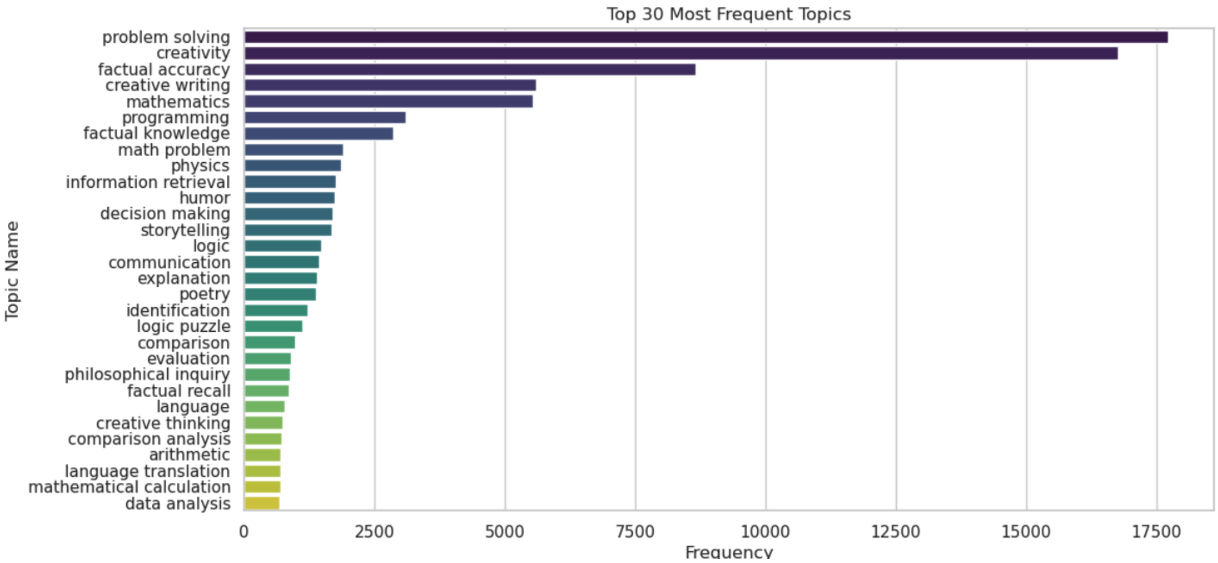

Figure 1: Top 30 Most Frequent Topics

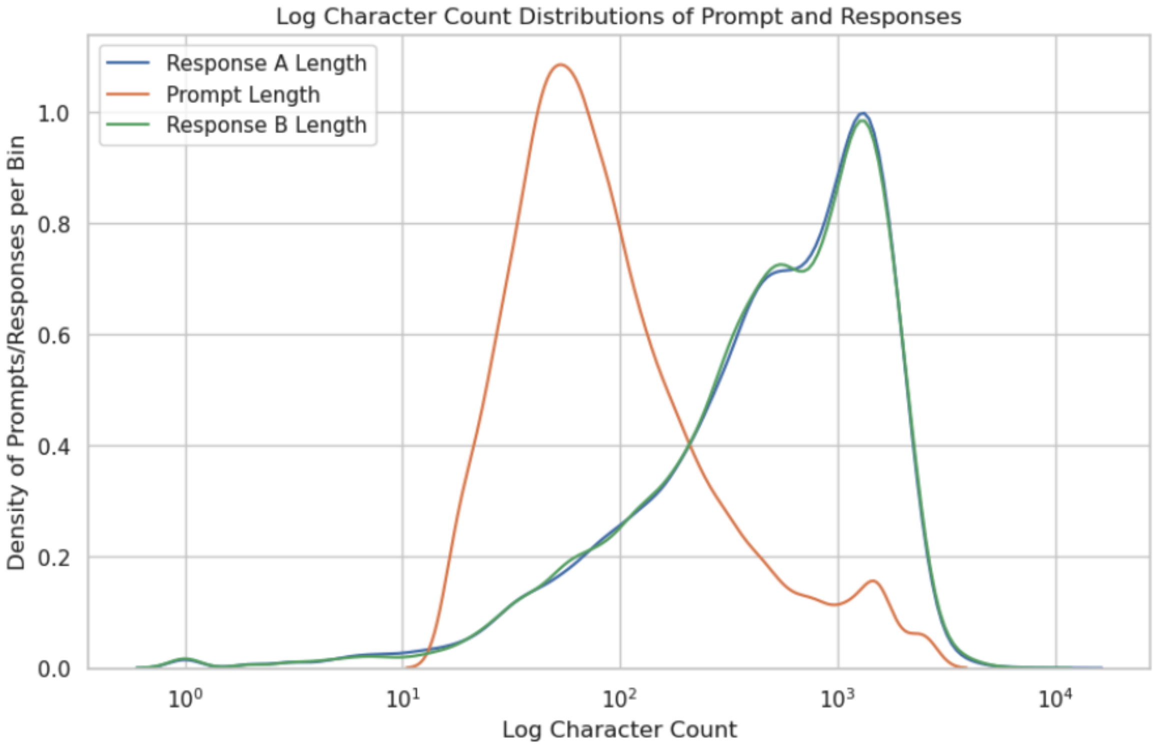

Figure 2: Log Prompt and Response Lengths

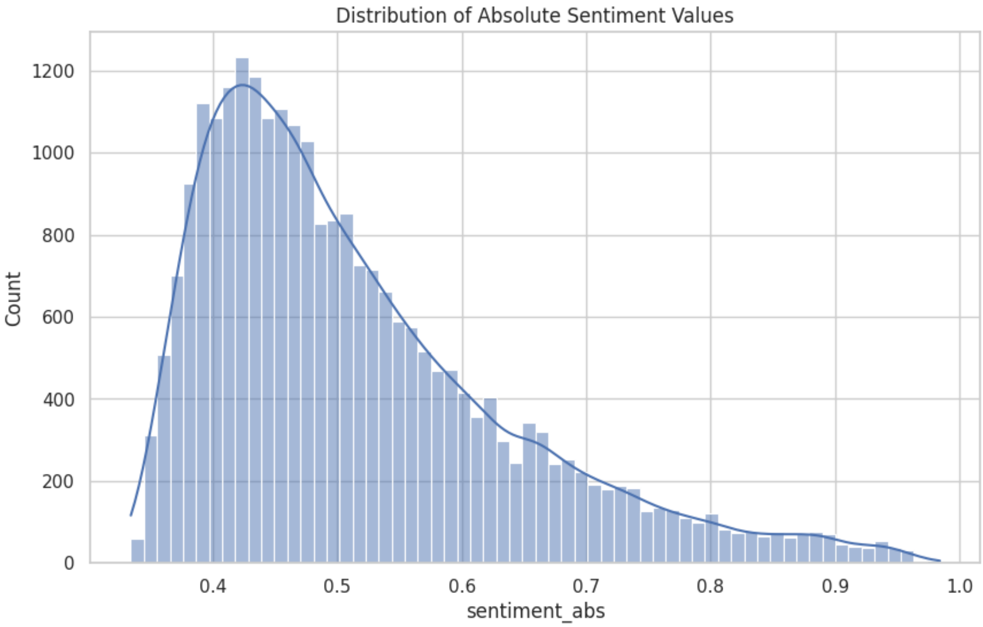

Figure 3: Sentiment values for each prompt. Higher numbers correspond to either more positive or negative sentiment.

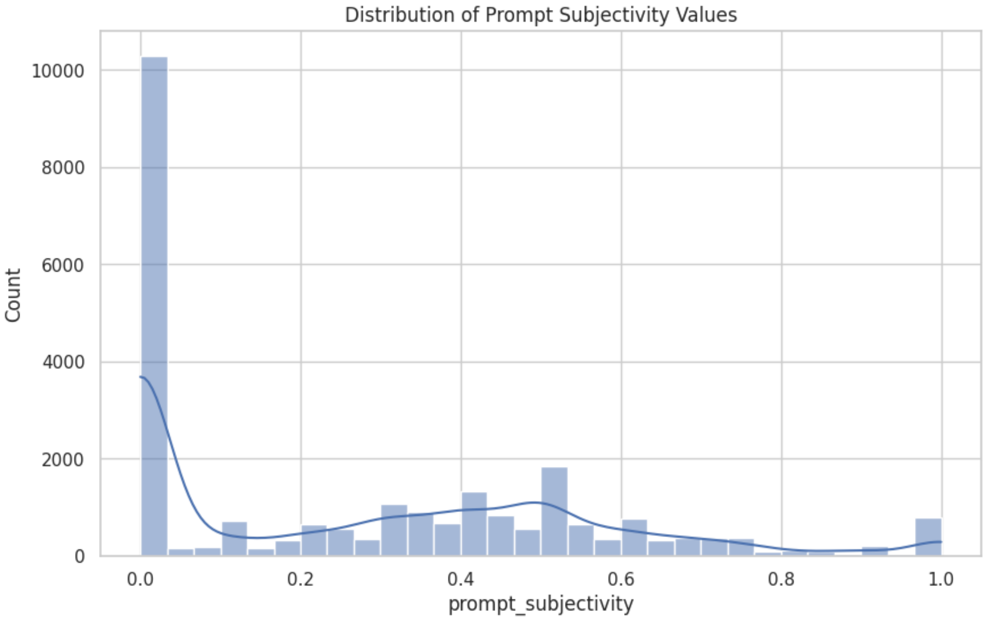

Figure 4: Subjectivity values for each prompt, with higher values corresponding to more subjective/opinionated prompt language.

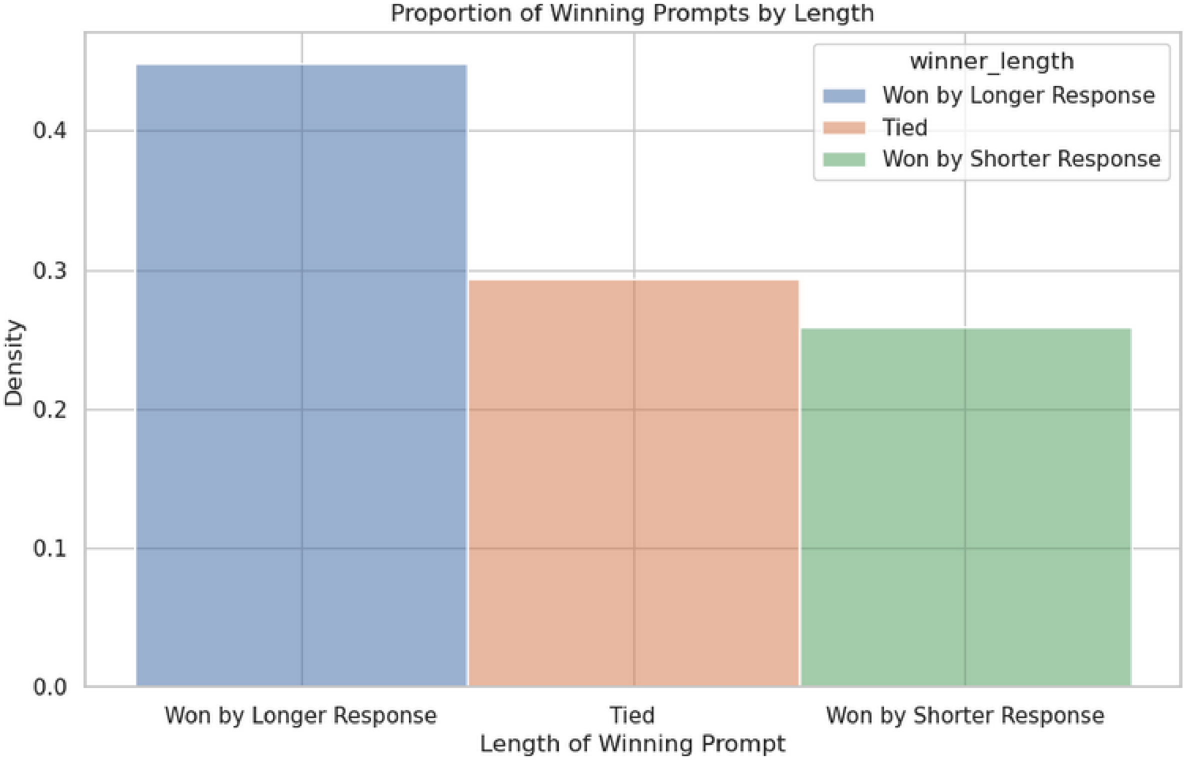

Figure 5: Distribution of Labels against Prompt Length

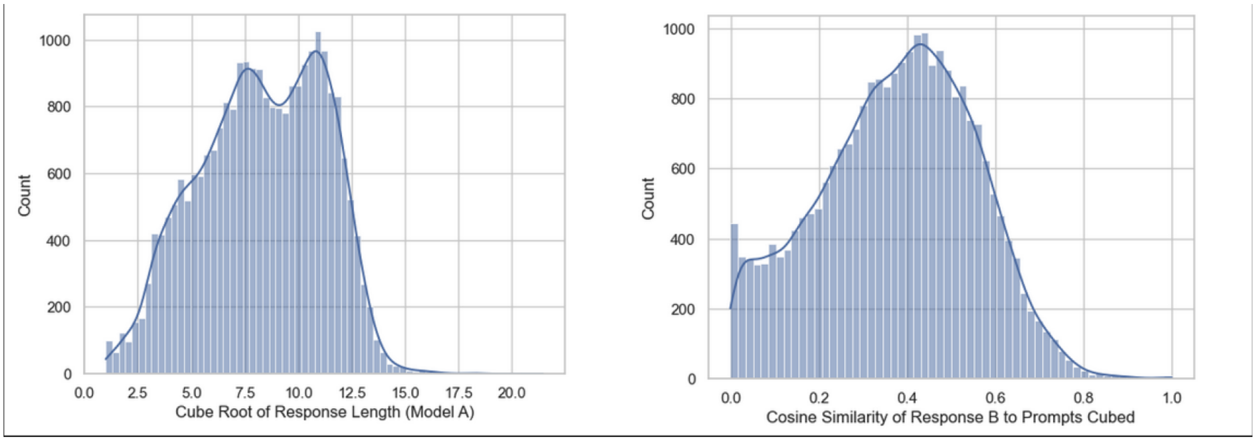

Figure 6: Side by Side of Two Transformations

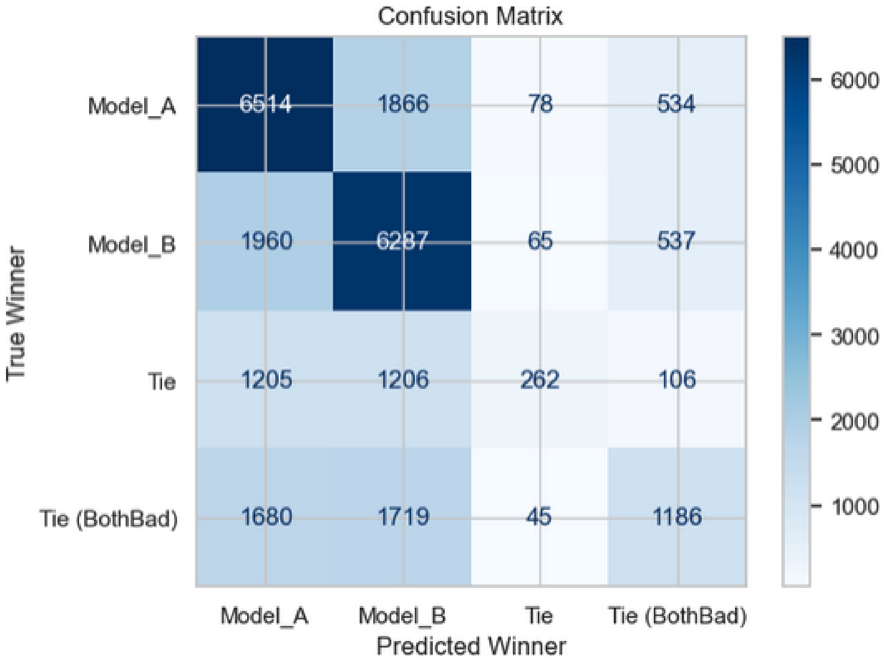

Figure 7: Confusion Matrix for Logistic Regression

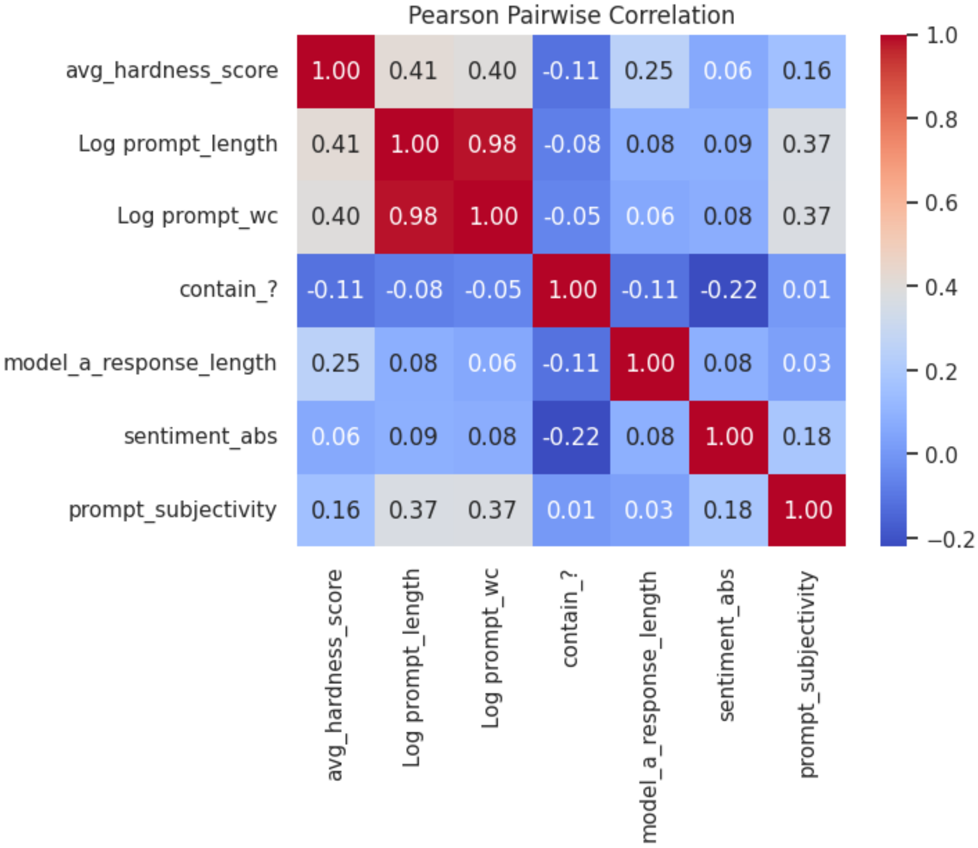

Figure 8: Pearson correlation coefficients for proposed features