Team Project 2 (Team11)

CS 53744 Machine Learning Project - Instructor Professor: Jongmin Lee
Department of Computer Science & Engineering
20222446 Hyounsoo Kim, 20201876 Sanghyun Na, 20203009 Jaehyun Park

「A Study on LLM Response Preference Prediction and Classification 」
GitHub Repository: [github.com/MachineLearning-Project-Team11/Project02-LLM-Classification-Finetuning]

Abstract

This project addresses the complex task of predicting human preference between two competing Large Language Model (LLM) responses for a given prompt, based on the Kaggle 'LLM Classification Finetuning' competition dataset. We implemented several machine learning strategies, progressing from simple statistical baselines to a complex neural architecture. Our final model employs a **DeBERTa-v3-small** Dual Encoder fine-tuned with **LoRA**. Crucially, the model incorporates a Cross-Attention mechanism to explicitly model the relational context between the two responses, augmenting the final classifier with difference and product feature vectors. This approach achieved the lowest Validation Log Loss of 1.0144. Qualitative error analysis revealed that the model exhibits a significant verbosity bias (favoring detailed, longer responses) and shows difficulty considerably with the ambiguous **Tie** class, suggesting avenues for improvement through specialized ranking loss functions. We also had approaches with **LLaMA 3 8B** model for superior performance. Our model utilizes a dual encoder **LLaMA 3 backbone** fine-tuned with **LoRA (Low-Rank Adaptation)**. Unfortunately, this approach could not exceed the performance of our state-of-the-art model.

1. Introduction

The advent of large language models has required robust methods for evaluating and aligning model outputs with human values and preferences. This project focuses on the core problem of predicting the human-annotated winner between two competing LLM responses(**Response A** or **Response B**) or determining if the outcome is a **Tie**. This multi-class classification task utilizes a **dataset of 57,477 training samples** and is evaluated using the **Multi-class Log Loss**, a metric that rewards well-calibrated probability distributions over the three outcome classes (A Wins, B Wins, or Tie). The target class distribution in the training data is relatively balanced (A Wins: 34.91%, B Wins: 34.19%, Tie: 30.90%). The primary challenge lies in capturing the human preference, which often involves factors like conciseness, honesty, and helpfulness, rather than the correctness or length of the response. This report details the progression from baseline models to our final high-performing state-of-the-art architecture and presents a critical analysis of the model's performance and fundamental biases.

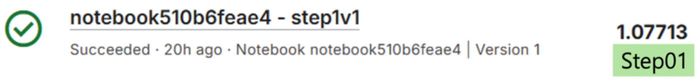
2. Methodology

The modeling strategy was designed to incrementally increase the complexity of feature representation and the sophistication of the classification head, moving from baseline level feature engineering to fine-tuned transformer models.

2.1. Baseline Modeling

Initial experimentation established two essential baselines, 1 and 2. **Baseline 1** employed a linear **Logistic Regression** model trained only on simple statistical features: the differences in character length (length_diff) and word count (word_count_diff) between the two responses. This minimal approach yielded a Validation Log Loss of **1.07713**(Baseline 1).

Recognizing the limitation of surface-level features, **Baseline 2** incorporated semantic information using a pre-trained sentence embedding model, **Sentence Transformer('all-MiniLM-L6-v2')**. The prompt and each response were combined and encoded, resulting in a **768-dimensional** concatenated feature vector for a Logistic Regression classifier. The resulting Log Loss was **1.06822**(Baseline 2), confirming that basic semantic embedding alone provided only a marginal improvement over lexical features.

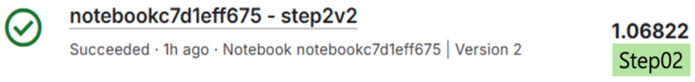


notebook510b6feae4 - step1v1

Succeeded · 20h ago · Notebook notebook510b6feae4 | Version 1

1.07713

Step01

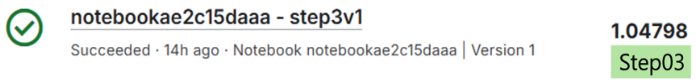


notebookc7d1eff675 - step2v2

Succeeded · 1h ago · Notebook notebookc7d1eff675 | Version 2

1.06822

Step02



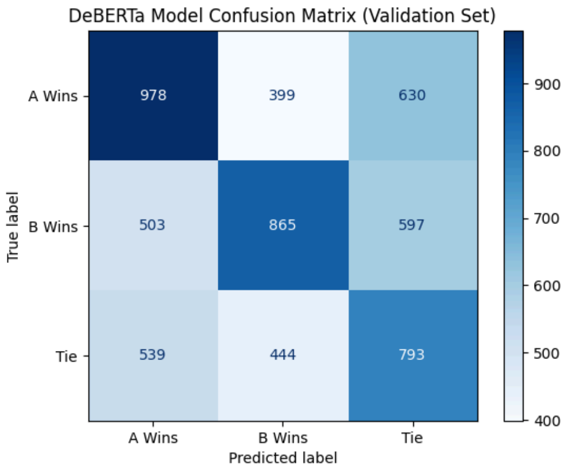
notebookae2c15daaa - step3v1

Succeeded · 14h ago · Notebook notebookae2c15daaa | Version 1

1.04798

Step03

Finally, **Baseline 3** applied **LORA** to the **DeBERTa-small** model for fine-tuning, and attached a classifier head via a three-layer MLP for end-to-end fine-tuning. LORA hyperparameters were set to r=8, LORA alpha=16, and LORA dropout=0.1. Prompts were encoded separately using Dual Encoder, and the resulting embeddings were concatenated as input to the classifier head. A validation log loss of **1.04798**(Baseline 3) showed a significant performance improvement.



The results of the **Baseline 3** model's confusion matrix are structured as the left side showing the diagonal values being largest. This confirms that the model has been trained to a reasonably significant degree. It correctly predicted approximately **50% (1002/2007)** of actual wins by A, showing the highest accuracy rate among the three classes. It correctly predicted approximately **42% (831/1965)** of actual wins by B. Furthermore, the model appears to struggle most with the Tie class. This is evident as it correctly predicted approximately **43% (766/1776)** of actual draws. Additionally, a significant number of classes predicted A as the winner despite being draws. This confirms the model struggles with distinguishing clear wins from draws.

2.2. Final Architecture: Dual Encoder with Cross-Attention

To accurately predict the winner, the model must perform a deep, contextual comparison between the two responses A, B. Our final approach utilizes a **Dual Encoder** architecture based on the powerful **DeBERTa-v3-small** transformer model.

Model Implementation

To manage training efficiency and resource constraints, we employed **LoRA (Low-Rank Adaptation)**, targeting the attention projection layers (query_proj, value_proj) with an aggressive rank of $r = 16$. This resulted in only about **0.2%** (294,912/141,599,232) of the model parameters being trained, enabling efficient GPU utilization.

Relational Modeling

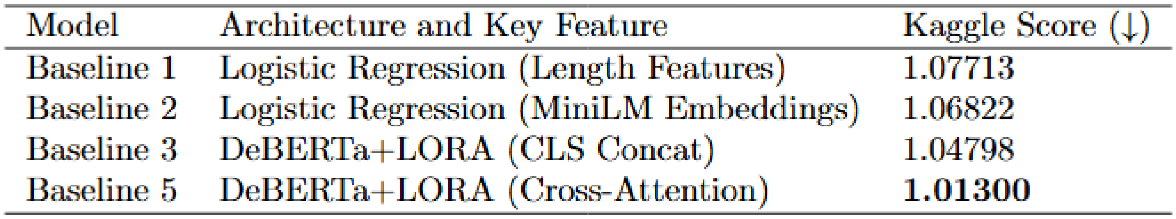
The core architectural innovation is the explicit modeling of the relationship between the responses. After encoded separately by the shared-weight DeBERTa backbone, the contextual token embeddings for Response A and Response B were passed through a dedicated **Cross-Attention** layer. This allows the tokens of one response to attend to and query the content of the competing response, generating deeply contextualized representations (P'_A and P'_B) that explicitly incorporates the competitive dynamic.

2.3. Validation Strategy

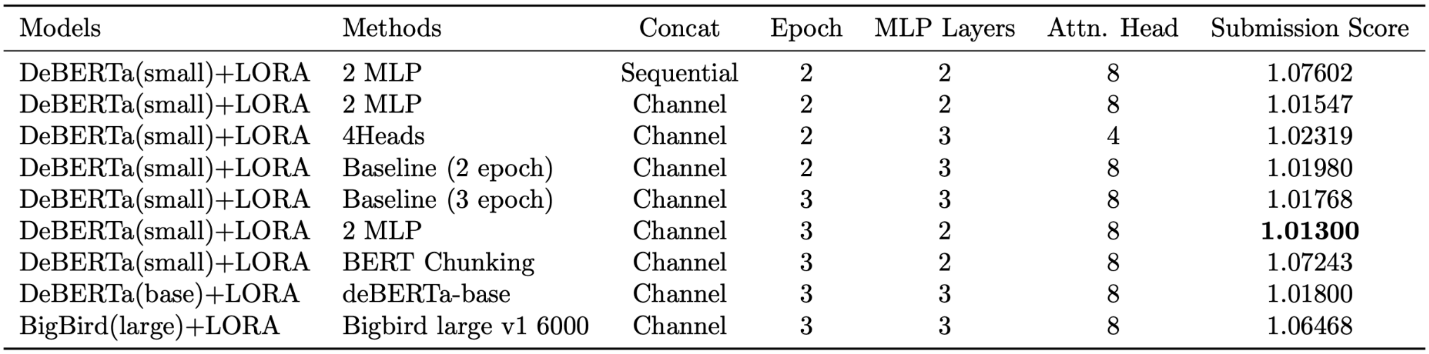
Using the train_test_split function provided by scikit-learn for validation, we extracted a validation set at a ratio of 0.1 from the training dataset. This ensured that the validation set was not used for training and could be used to assess the model's performance.

3. Experiments and Results

The efficacy of the proposed architectural extensions is clearly visible in the performance across the models. The state-of-the-art final model achieved the lowest Log Loss, validating the effectiveness of the **Cross-Attention** strategies. The improvement from the simple concatenation approach (Step 3: 1.04798) to the relational modeling approach (**Final Model: 1.01768**) was substantial, confirming that relational context is key for this task.



Model	Architecture and Key Feature	Kaggle Score (↓)
Baseline 1	Logistic Regression (Length Features)	1.07713
Baseline 2	Logistic Regression (MiniLM Embeddings)	1.06822
Baseline 3	DeBERTa+LORA (CLS Concat)	1.04798
Baseline 5	DeBERTa+LORA (Cross-Attention)	**1.01300**



Models	Methods	Concat	Epoch	MLP Layers	Attn. Head	Submission Score
DeBERTa(small)+LORA	2 MLP	Sequential	2	2	8	1.07602
DeBERTa(small)+LORA	2 MLP	Channel	2	2	8	1.01547
DeBERTa(small)+LORA	4Heads	Channel	2	3	4	1.02319
DeBERTa(small)+LORA	Baseline (2 epoch)	Channel	2	3	8	1.01980
DeBERTa(small)+LORA	Baseline (3 epoch)	Channel	3	3	8	1.01768
DeBERTa(small)+LORA	2 MLP	Channel	3	2	8	**1.01300**
DeBERTa(small)+LORA	BERT Chunking	Channel	3	2	8	1.07243
DeBERTa(base)+LORA	deBERTa-base	Channel	3	3	8	1.01800
BigBird(large)+LORA	Bigbird large v1 6000	Channel	3	3	8	1.06468

Table 3: Performance Comparison Table

3.1. Per-Class Performance

Analysis of the Per-Class Log Loss provided critical insight into the model's strengths and weaknesses. The model achieved its best performance predicting the B Wins class, with a minimal Log Loss of **0.92560**. The A Wins class was moderately challenging at **1.01580**. However, the **Tie** class proved to be the most challenging outcome, exhibiting the highest Log Loss of **1.13130**. This quantitative finding indicates a systemic issue in accurately predicting ambiguous or neutral preference outcomes.

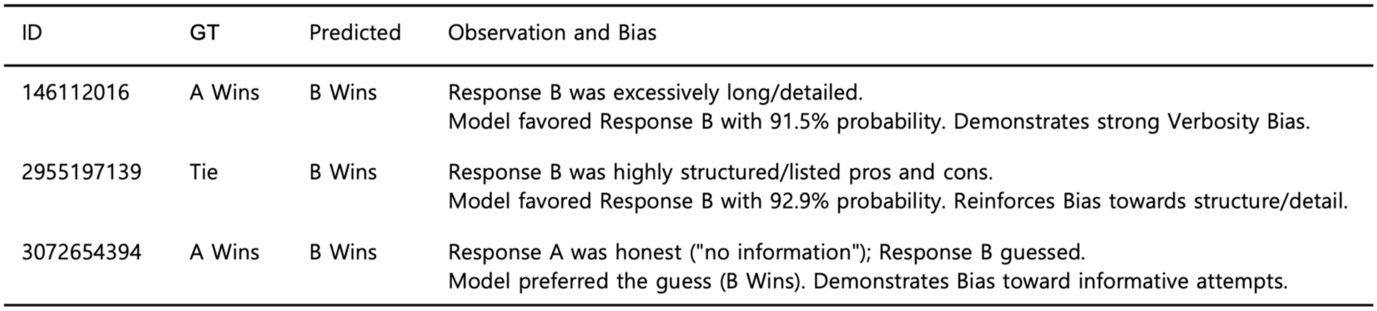
4. Analysis and Discussion

4.1. Error and Model Bias Analysis

A qualitative analysis was conducted on the top three misclassified samples (highest loss) to uncover inherent biases learned by the model. These errors directly contributed to the higher Log Loss in the Tie and A Wins category.

The analysis highlighted two major model biases:

- **Verbosity Bias:** The model consistently favored responses that were significantly longer, more structured, or provided greater detail, even when the human labeler preferred the concise, direct response. This is clearly seen in Sample ID 146112016.
- **Bias Towards Informative Attempts:** In cases where one model honestly admitted a lack of information (A Wins), the final model preferred the competing response that *attempted* to provide an answer, even if the content was generic or inaccurate. (Sample ID 3072654394).



ID	GT	Predicted	Observation and Bias
146112016	A Wins	B Wins	Response B was excessively long/detailed. Model favored Response B with 91.5% probability. Demonstrates strong Verbosity Bias.
2955197139	Tie	B Wins	Response B was highly structured/listed pros and cons. Model favored Response B with 92.9% probability. Reinforces Bias towards structure/detail.
3072654394	A Wins	B Wins	Response A was honest ("no information"); Response B guessed. Model preferred the guess (B Wins). Demonstrates Bias toward informative attempts.

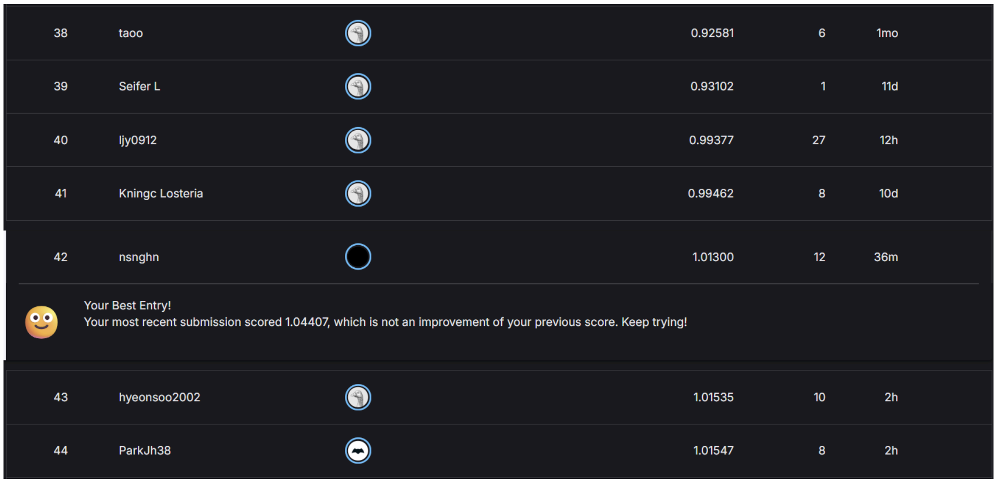
4.2. Reproducibility Notes

All modeling stages are designed for complete reproducibility. The entire process was conducted on a GPU environment (cuda) using Python 3.13.5 and the specified libraries. The random seed was fixed at random_state=42 for all stratified data splits and model initialization steps. Furthermore, all external model assets were pre-downloaded and loaded locally, ensuring compliance with the competition's No-Internet Constraint. The final model training run required approximately 1.5 hours (5400 seconds) on the T4 GPU provided.

5. Conclusion and Future Work

5.1. Conclusion

This project successfully implemented a complex, high-performing model for predicting LLM response preference, achieving a competitive Validation Log Loss of **1.01300**. The critical technical success lies in utilizing LoRA for efficient fine-tuning of DeBERTa-v3-small and integrating explicit **Cross-Attention** to model the comparison of the two responses. The project demonstrated that sophisticated relational modeling is far more effective than simple concatenation of independently encoded responses.



38	taoo	0.92581	6	1mo
39	Seifer L	0.93102	1	11d
40	ljy0912	0.99377	27	12h
41	Kningc Losteria	0.99462	8	10d
42	nsnghn	1.01300	12	36m

Your Best Entry!
Your most recent submission scored 1.04407, which is not an improvement of your previous score. Keep trying!

43	hyeonsoo2002	1.01535	10	2h
44	ParkJh38	1.01547	8	2h

5.2. Limitations and Future Work

Despite achieving competitive performance, the model demonstrated clear limitations that suggest avenues for future research. The primary limitation remains the high prediction difficulty for the ‘Tie’ class. To address these limitations and the observed model biases, future work should focus on three main directions. First, adopting a specialized Loss Function like a Pairwise Ranking Loss (e.g., RankNet or a margin-based loss) is essential. This would optimize the model directly for the relative ordering of A and B, which is expected to improve calibration and accuracy for all three classes, especially the Tie class. Second, efforts should be made toward Bias Mitigation, specifically targeting the model's reliance on length and structure (Verbosity Bias). This could be achieved by using adversarial training techniques to decorrelate the learned preference signal from simple length features, or by directly integrating normalized length and structural metadata into the final classification layer as explicit bias-aware features. Finally, the next logical step to achieve a final performance boost on the Kaggle leaderboard involves an Ensembling Strategy. This would entail creating a powerful Soft Voting Ensemble that combines the prediction probabilities from the top-performing models, including the Embedding-based Logistic Regression, the CLS Concatenation DeBERTa model, and the Final Cross-Attention model.