**Tennis Prediction Analysis Using GPT**

Albert Tan, Ronak Pai, Akhilesh Bellathur, Matthew Wilson, Justin Wilson

**Introduction (~1 page)**

Predictive analytics has had a profound impact on the dynamic landscape of sports and technology. The sheer magnitude of data collected to forecast future outcomes in sports has never been greater, encompassing a wide range of player statistics, historical performance data, and real-time game metrics that provide unprecedented insights and prediction accuracy. This exponential growth in sports data collection has been coupled with a surge in sports betting. In recent years, the repeal of the Professional and Amateur Sports Protection Act which prohibits sports gambling, has created a multibillion-dollar market in 38 states that allow some form of sports wagering. In addition, sportsbook revenue reached $7.5 billion in 2022, signaling a 75% year-on-year increase. Forecasts indicate continued market expansion, highlighting the enduring relevance and growth potential of the industry [add source].

<https://www.americangaming.org/new/2022-commercial-gaming-revenue-tops-60b-breaking-annual-record-for-second-consecutive-year/>

Advancements in artificial intelligence and machine learning are increasingly being leveraged to gain advantages in sports betting. Models tailored to specific domains are trained on vast quantities of relevant data and promise advanced predictive capabilities. Even beyond sports betting, these models can provide a strategic advantage to players and coaches. Players can analyze opponents’ strengths and weaknesses, aiding in the formulation of deliberate plans based on predicted match outcomes. Players can also gain insight to their own performances, highlighting areas of improvement and fostering self-awareness.

The confluence of sports data and technological advancements forms the backdrop against which this project gains significance. Professional tennis in particular, is one sport where data collection is paramount, with statistics spanning from court favorability and head-to-head odds to groundstroke spin and velocity rates. These metrics underpin the nuanced analysis of player performance.

This report examines the development of a model designed to not only forecast the outcome of professional tennis matches, but also to provide comprehensible reasoning behind its predictions. By leveraging pretrained Large Language Models (LLMs) and fine-tuning them with datasets containing statistics from the Association of Tennis Professionals (ATP) and International Tennis Federation (ITP), the goal is to explore the potential of cutting-edge technology in transforming sports forecasting.

**Related Works(~0.75 page) → Need at least 10 references**

Early attempts to predict the outcome of tennis matches consisted of running logistic regression models based on players’ ATP ranking [1]. These models were found to be successful in improving the accuracy of predictions, and since then, there has been a proliferation in the development of models of various types and complexities to gain even marginal upgrades in forecasting accuracy. Barnett [2] and O’Malley [3] developed mathematical approaches that leveraged the hierarchical scoring system in tennis to calculate probabilities of players winning their service points. Barnett used serving statistics to deduce match outcomes and O’Malley made the assumption that points are independently and identically distributed to derive expressions for the probabilities of game, set, and match winners. Knottenbelt [4] improved on these methods in 2012 by only comparing opponents that both players have faced in the past to predict match outcomes, reducing noise associated with disparate opponent performances.

While these mathematical models produced successful results, they failed to account for the more understated details that impact the outcome of a match. Factors such as a player’s age, susceptibility to certain playing styles, and the amount of time since their last injury would have no influence on the match prediction. Other considerations such as the temperature during the match, location, and court conditions were also neglected. Somboonphokkaphan defines shortcomings in data usage with phrases such as the “out of date data problem” or the “without environmental data problem” which describe research using outdated data or data that does not fully encompass match conditions [5].

Researchers in recent years have employed the vast amount of historical tennis data to produce more informed predictions. Sipko used abstract attributes such as player fatigue and injury data to train a logistic regression model and artificial neural network [6]. He found that the betting decisions exercised by his machine learning model outperformed the best stochastic models at the time by 75%.

Vaswani et al. first suggested the Transformer neural network architecture in 2017 as a general language model to encode phrases and decode a response [7]. Since then, Large Language Models have become the most widely used deep learning model across the world. The most popular of these pretrained models is GPT, created by Open AI.

These LLMs can be finetuned for specific tasks by feeding in question-answer pairs and through reinforcement learning with human feedback. For example, Ding et al. found that LLMs could be finetuned and used to pick out headlines for stock predictions, resulting in significant gains over traditional methods [8].

However, while LLMs have been found to be good at language related tasks, they have some limitations. Xu et al. found that LLMs do not do well with tasks that require logical thinking and require large amounts of data to complete these [9]. Lappin also found that these models are prone to hallucinations and cannot separate fact from fiction [10]. Both of these factors limit the potential success of LLMs in tennis predictions.

[1] [Boulier & Stekler - “Are sports seedings good predictors?: an evaluation”](https://pdf.sciencedirectassets.com/271676/1-s2.0-S0169207000X00179/1-s2.0-S0169207098000673/main.pdf?X-Amz-Security-Token=IQoJb3JpZ2luX2VjEAEaCXVzLWVhc3QtMSJIMEYCIQDtWt21Ub6hMuch3znk10gMZ2r5TmcLkKP7a7Ii7Nw%2FgQIhANaOmOVBgZEmH2b5doYd6ETpC1xNn9%2BROz4UYBNXPm%2FUKrMFCGkQBRoMMDU5MDAzNTQ2ODY1IgyQtaX%2BpbiEmMPqX4wqkAVSCU%2BJU%2FKfTQT9pkzzHDMYdoHgprxnfSJ4m%2F7gkUJ3eMvdBgyrRKIAcDcsoME%2BkktrwqgIIcA%2F7vIwJlUYqIwWaOFXqCUlgfGXyvEmN9sznZCga58djaHnX3cM%2BipRjO9e%2FK6DeyZ1ngKdRE1sgmguqOqYmGLVtXPtYc9e5VbfK1HZK6DBVS9ZaKlLbTXHRue2rLfxfL1OUUQckh7UA3NyrRNTX9LBXWyp0Z%2FwSOCwP8ELfbOMFvqQQnbLHoKK7Hclhbv327086HukyHghim97GKlYq%2BGqx6hXwSIfjCfOpUwv38YvvV8iDCvnCXsFEJqPfhHXCY9Z9JEbX5GhcXnyDTItfAEPoGszICCG8ZwM5p7l%2B1ejJCJ9aAedNcM4B9PzarOhaSrAsMUuA%2FsABPGucXKepGP2iM02ThlacRB0ei4Psof%2BiaoXkLj3P9t8xNvwMYvADuHXUVTS%2Fzxh9NQOdC%2BrZC06e9DQVljy5pBuxmEIXOCPYBVkZukPxNZR77cGo8IeMjslQtoB5qXhpuFEqYqmjLGZ7WcSBXpGix4%2Fnxbs58%2F5M0GB0SrozHm71yDsm6QqOR2%2FTPURHgHHRL1dn6N1K8zynSTh0Ua1q2QIHTNaGZgDMdqJUIo9piYbcUjkssRj1cYBSWMUggaddRZLqblSmxoejqLnS%2FtBc4OP6V7y9%2Fm7Kg3zh3BrWKjFCJB2lEMib7gA38rquK8Hao%2F3zRmN77JsOxLF7b2eIQ%2B8a7nrsfj301pYvaORpYQVKFqGAzpQ9urYHJnHkpx1AHWcN09CGQ%2B6ADihtH5rtWigpsJothbmzh99TafZu%2FF6FfEJhRxcR16VNEymEfRo9WuFETJG3bp98yv9FjYPjnIAfzDr2bmrBjqwASwfYofavGT5rQ4F2R0bef5rL2A7Y5BsqCE5edvjM5p23p5bhUwH2W55Jbl7Vwbl91%2FVOBcd1nvEmuqafNLa7BfzgihgFbRnLogj5FC6uqOK%2FH9B1hgC%2F78MFCYuwiCzbioG%2BkvOhjnfX1BxyOF3og1%2FYQG6gkSeu2QyBu37kWONC%2B86YAoGim4qEPySpJKw1551WSO9ZTazu3Q4oVzFLAbEpK%2BiyH%2FyHcp5HWxTPIwQ&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20231205T003717Z&X-Amz-SignedHeaders=host&X-Amz-Expires=300&X-Amz-Credential=ASIAQ3PHCVTYRGTJ3U6D%2F20231205%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Signature=e00a5e8319938661be3ef3d036ed71d6ea5012050fa3865fd43745802d9eaad8&hash=9e5d5f13c357e3bb7f3ff34df4f27bf6b374148d37a5d4e25d0cd28feb5a8cba&host=68042c943591013ac2b2430a89b270f6af2c76d8dfd086a07176afe7c76c2c61&pii=S0169207098000673&tid=spdf-2fe0ac84-a956-42c4-8afb-f879143064e4&sid=f890d10e18da1145911a6958cf9facccf4d6gxrqa&type=client&tsoh=d3d3LnNjaWVuY2VkaXJlY3QuY29t&ua=131c585c515c505f57&rr=8308384149f22b94&cc=us)

[2] [Barnett & Clarke - “Combining player statistics to predict outcomes of tennis matches”](https://www.researchgate.net/publication/228614352_Combining_player_statistics_to_predict_outcomes_of_tennis_matches)

[3] [O’Malley - “Probability Formulas and Statistical Analysis in Tennis”](https://www.degruyter.com/document/doi/10.2202/1559-0410.1100/pdf)

[4] [Knottenbelt - “A common-opponent stochastic model for predicting the outcome of professional tennis matches”](https://www.sciencedirect.com/science/article/pii/S0898122112002106)

[5] [Somboon Phokkaphan - “Tennis Winner Prediction based on Time-SeriesHistory with Neural Modeling”](https://www.scribd.com/document/258041233/Tennis-Winner-Prediction-Based-on-Time-Series)

[6] [Sipko - Machine Learning for the Prediction of Professional Tennis Matches](https://www.doc.ic.ac.uk/teaching/distinguished-projects/2015/m.sipko.pdf)

[7] [Vaswani et al. - Attention is All You Need](https://arxiv.org/abs/1706.03762)

[8] [Ding et al. - Integrating Stock Features and Global Information via Large Language Models for Enhanced Stock Return Prediction](https://arxiv.org/pdf/2310.05627.pdf)

[9] [Xu et al. - Are Large Language Models Really Good Logical Reasoners? A Comprehensive Evaluation and Beyond](https://arxiv.org/abs/2306.09841)

[10] [Lappin - Assessing the Strengths and Weaknesses of Large Language Models](https://link.springer.com/article/10.1007/s10849-023-09409-x)

**Problem Formulation(~1 page)**

Our project aims to develop a sophisticated predictive model for professional tennis match outcomes, going beyond predictions by providing insightful explanations for the model's decisions. Our approach involved not just identifying the predicted winner but also describing the underlying rationale behind that prediction. To achieve this, we designed our model with a comprehensive set of input parameters, including player details, such as rankings, points, and historical statistics, along with match-specific factors like court surface and betting odds.

We used a dataset sourced from Kaggle, encompassing ATP match statistics spanning 2000 to 2017. The richness of this dataset, encompassing nuanced player statistics such as ace percentages, break points saved, and an array of other crucial metrics, provided a holistic view of player performance factors. Kaggle's reputation for stringent validation and data cleansing processes assured the reliability and accuracy of our dataset, mitigating errors and inconsistencies.

Beyond just the output of match winners, our model was designed to articulate reasons behind its predictions. These explanations ranged from factors like court favorability, historical match data, player-specific strengths, and other insightful features contributing to a player's perceived advantage over their opponent.

**Methodology(1~2 pages)**

Our dataset consists of ATP and ITF tennis matches spanning the past 23 years. This dataset consists of various metrics, including player rank, ATP points, court surface, tournament name, tournament round, their physical characteristics (height, weight, dominant hand), and betting odds. We split this dataset into train and test for all of our models.

To establish a benchmark for our fine tuned language models, we first developed a baseline using the dataset. The first baseline we attempted simply used parts of the dataset to choose a winner. One simple technique we used was to select whichever player was ranked higher in the ATP rankings. A second technique selected whichever player was favored by the betting odds given in our dataset.

Then, we attempted to build models using more traditional machine learning techniques. Using all of the given numerical data, we built a logistic regression model to select a winner. We also used a random forest model and a multi-layer perceptron to classify winners.

For our Large Language Models, we did three different experiments. The first used the DaVinci model provided by Open AI, which is not instruction-tuned. To finetune this model, we had to generate question-answer pairs. To do this, we developed a template phrase to convert from numbers to a phrase for the question. The answer would simply be the name of the winner of the match. Then, for testing, we just fed the question and saw if it was the same as the real winner.

For our second experiment, we used GPT-3.5-Turbo. This model is supposed to be more powerful than DaVinci and also is instruction-tuned. To finetune this model, we had to generate chat-completion inputs, which are meant to mimic a conversation rather than a single question. We built this also using a template, and then fed the question into the fine-tuned model for testing. In this experiment, we simply asked for which person would win the match.

For our last experiment, we also used GPT-3.5-Turbo, except this time with the goal of receiving an explanation for its selection. To do this, during testing we asked for a winner and an explanation for why that winner would win. For the training data, we used GPT-4 to come up with explanations using game data for each match. We then collected the accuracy based on which winner it selected, along with manually viewing the different explanations it gave.

Because fine tuning OpenAI’s foundation models costs money, we were unable to use most of the training set. In fact, using only 1% of the training data already cost $40, so that is the amount we ended up using for both our experiments. This was an unfortunate limiting factor for us, since we believe that many of the inaccuracies came from the fact that the models were trained on such low amounts of data. The results will be discussed in more detail in the upcoming section for the data we did manage to train on.

**Results and Discussion(1~2 pages)**

To start out with the results, we first looked at the given data that we had and tried to make some baseline assumptions. As stated previously, the data that we gathered contained the rank of each player that matched up as well as the betting odds for that game. The betting odds were determined by various sports forecasting companies that use their own statistics and human knowledge to determine matchup statistics and performance history. Because both of these stats already contained a lot of data in order for the tennis leagues to play, we can assume that the results achieved by these are the current best way to predict the outcome of a match. And as expected, the test accuracy for both choosing the higher rank and choosing the higher betting odds yielded good results. The higher ranked player won an average of 65% of the time, while the player with higher betting odds won 70% of the time.

From there, we decided to implement our own baseline prediction models using the various methods mentioned above to train. Training a logistic regression model gave us a training accuracy of 64.8% and a higher test accuracy of 65.9%. The random forest algorithm on the other hand performed marginally better, with a training accuracy of 65.6% and a test accuracy of 66.1%. Also, the multi-layer perceptron performance averaged very similarly to the other two testing methods. The training accuracy was 64.3% and a test accuracy of 66.1%, yielding no significant difference between the methods. All 3 of these methods yielded approximately the same results that also seem to match the player ranking score. This gives us an idea that the ML models are not able to use data other than rank/betting odds in an effective way to increase the prediction accuracy in the results.

Finally, we tried using another random forest model with the betting odds to see if we can improve the accuracy. The training accuracy for this was 69.3% and a test accuracy of 70.7%. This turned out to be the best model that we trained so far, but it still did not significantly beat the accuracy of just using the higher betting odds. This makes sense since the tennis insight goes into making the betting odds in the first place, so training a RF on the betting odds after they have been made wouldn’t really change the results.

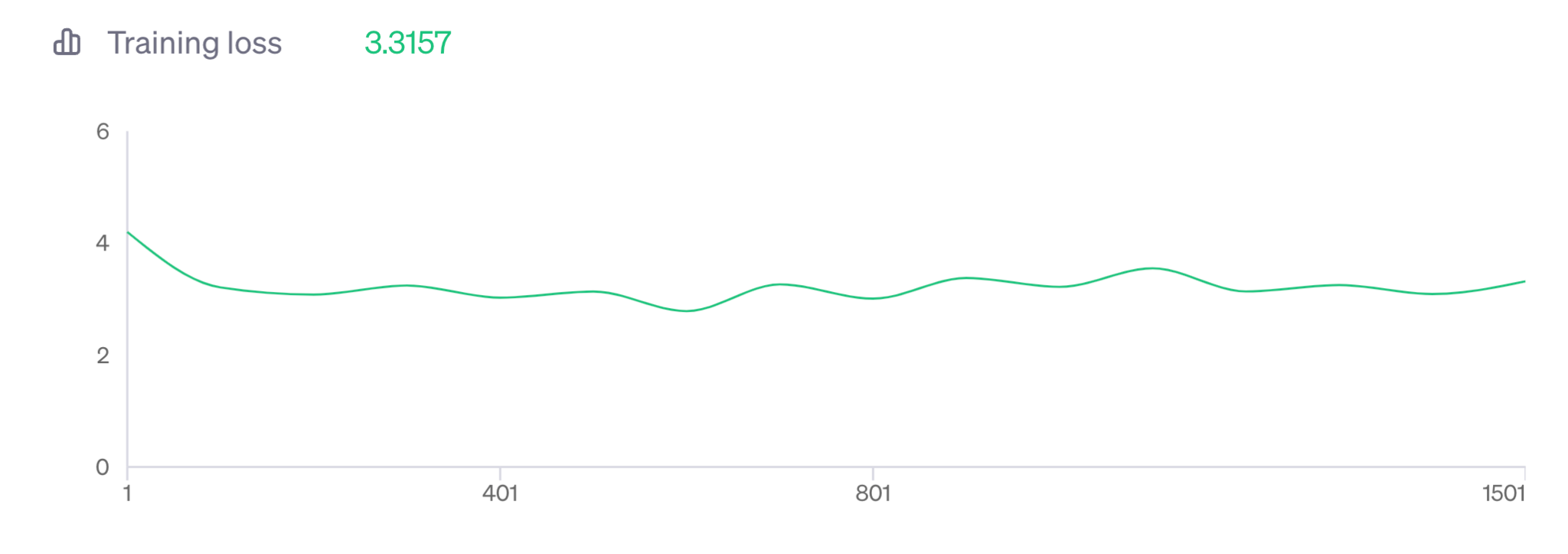
However, with the given results we came to the conclusion that a prediction accuracy of 65% - 70% is just about as good as we can get with standard models and ML predictions. This gives us a target to reach for when training our LLM. Ideally, we wanted the prediction to be higher than 70% but just matching those odds would be a good result as well, as long as there was still a little bit of explainability to go along with it.

Here is a table summarizing the results explained above:

| **Model Type** | **Train Accuracy** | **Test Accuracy** |
| --- | --- | --- |
| Higher Rank | **-** | 0.6546 |
| Betting Odds | **-** | 0.7050 |
| Logistic Regression | 0.648 | 0.659 |
| Random Forest | 0.6559 | 0.6613 |
| MLP | 0.643 | 0.6609 |
| RF w/ Betting odds | 0.6925 | 0.7067 |

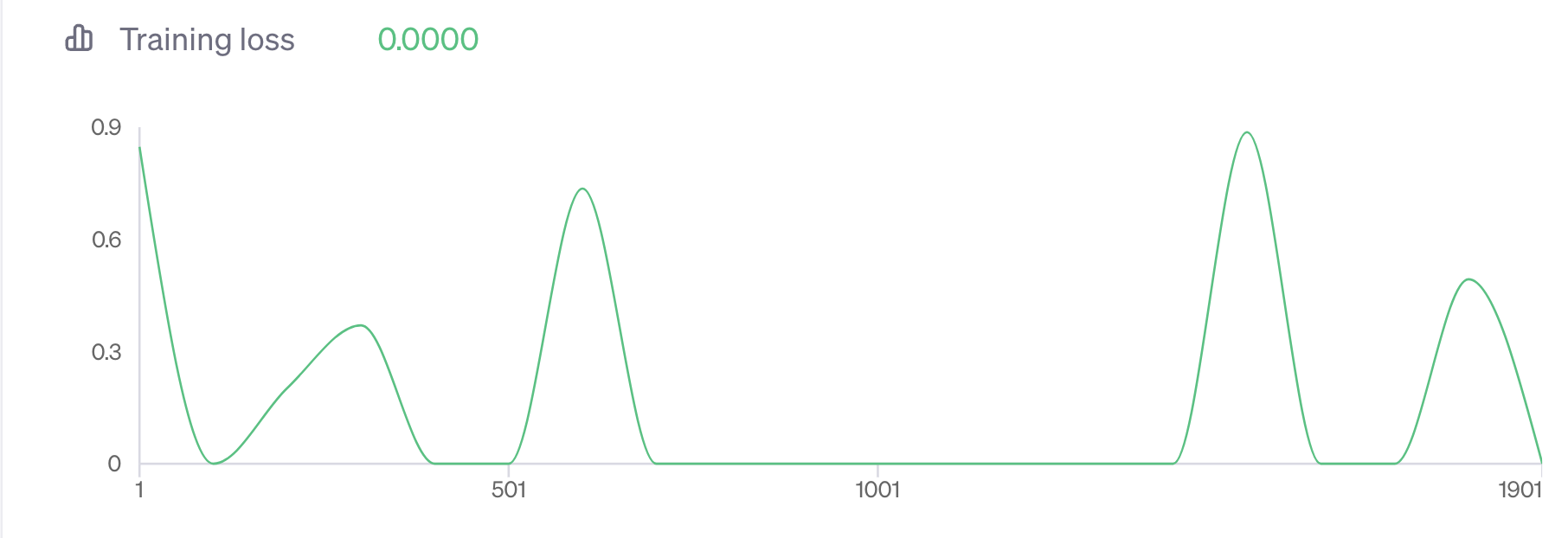
When training our LLMs to make predictions based on the same data that we trained the base models with, we first started with the DaVinci GPT-3 model. With our initial research into which model to train, we saw that online sources say that the DaVinci model was a very powerful option and well-suited for complex language tasks. Therefore, we trained the DaVinci model first, and unfortunately did not get very promising results. The model yielded an accuracy of only about 45.8% in predictions. This is because the model would often not answer the question, and instead spit out gibberish or not specify which person would be the winner.

Results for GPT-3.5-turbo model: 45.8%

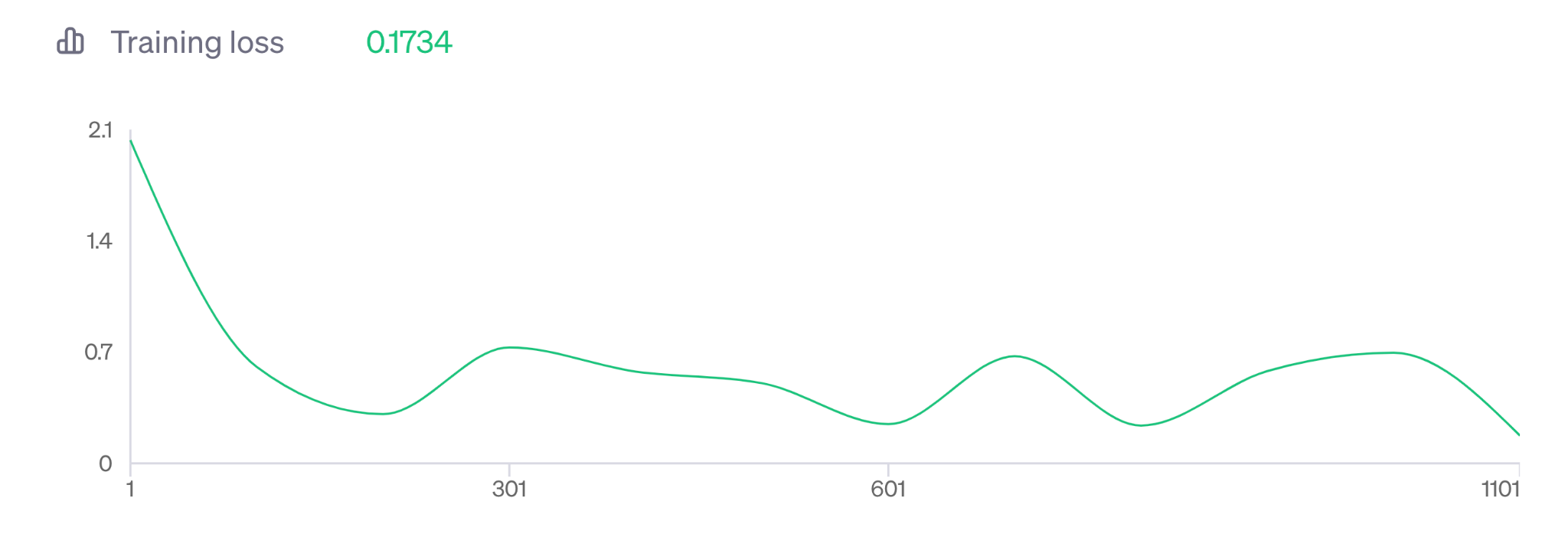


The graph above shows the training loss for the GPT-Davinci model. We observe that the loss stabilized almost immediately, which indicates that the model’s performance did not improve that much. This may explain why the accuracy was so low, as the model predicted the correct winner less than half the time. We believe that the training data may have been more aligned with the strengths of GPT-3.5-Turbo compared to GPT-DaVinci due to the simplicity of the data.

This would also explain why there was plenty of gibberish in the output of the model, because the training loss never lowered significantly and the model did not learn what to say as an explanation.

Results for GPT-3.5-turbo model: 69.2%

The graph above shows the training loss for the GPT-3.5-Turbo model when it was fine-tuned to simply output the winner of the match. We believe that the training loss reached zero many times since the model was overfitting on the data. Specifically, we hypothesize that the model may be associating higher betting odds with winning without recognizing other important factors. The test accuracy is very similar to the accuracy when just choosing the player with the higher betting odds (70.5%), which may provide additional evidence that the model is overfitting.

Results for GPT-3.5-Turbo model with Explanations: 61.8%

The graph above shows the training loss for the GPT-3.5-Turbo model when it was fine-tuned to output the winner of the match and a short explanation for it. Here, we can see that the loss starts at a relatively high value and decreases over time, indicating that the model’s predictions are improving. We also observe that early on, the loss gradually levels out, which indicates that the model may have stopped learning useful information from the data, which may be due to the data’s simplicity in comparison with the model’s complexity. The final training loss is 0.1734, which indicates that the model has not overfitted to the training data as much as the GPT-3.5-Turbo model without explanations.

The decrease in accuracy turned out to be an unfortunate side effect of adding explainability to the models. In the first iteration of the gpt-turbo model, the output for the predicted winner followed a set format of “We predict X will win again Y”, whereas for the second attempt, we added multiple different formats of explanation in the training data so that the model can learn how to explain itself and justify the reason for picking a certain player to win the match. However, as shown above, we did see a bit of a drop in the prediction accuracy from 69% to around 61%. While this is a slightly significant drop in accuracy, the justifications for each prediction in the second model are something that we are happy to see in the results. Although both models were trained on the same data, there can be a couple of reasons that the accuracy had this drop.

**Increased Complexity of the Task**: The task of not only predicting the winner but also generating a coherent explanation is significantly more complex. Since the model has to articulate the reasoning behind the predictions, a deeper processing of various factors is needed and therefore there is a larger room for error.

**Quality of Training Data**: We chose to use GPT-4 to modify our training data to add explanations for the winning player. When doing this however, the quality and relevance of these explanations are crucial and if there was any lack in accuracy or relevance to the data and context of the match, then the learning process could have been impacted poorly in the fine-tuned model which would lead to lower accuracy.

**Overfitting**: With the added complexity of generating explanations, there's a possibility that our model might have either overfitted the training data. Since we were only able to use a small portion of our data (1%), the variance in data is much smaller and could cause the model to learn the training data too well causing bad accuracy for unseen tennis matches.

**Loss of Focus on Primary Task**: By adding the requirement to generate explanations, the model's focus might have shifted from the primary task of accurately predicting the winner. The model now has to balance between generating a correct prediction and a plausible explanation, which might reduce its effectiveness in predictions.

To improve the model, we could refine the way explanations are generated, ensure the quality and relevance of the training data, and maybe consider a different approach to balance the dual tasks of prediction and explanation. Regular evaluations and tweaking the model based on performance feedback can also be beneficial. Also, using plenty more data with larger amounts of variables than the data we had could also provide key insights that can be used to link both the prediction accuracy and the explanations.

**Conclusion and Future Work(~0.25 page)**

Our paper provides an initial exploration of using LLMs for sports predictions. Using GPT-3.5-Turbo, we were able to achieve slightly lower accuracy on predicting tennis results when compared to traditional machine learning models. However, our accuracy dropped when asking the model for an explanation as well as a result. This is unfortunate, as our main hope for using LLMs over traditional models was to get an explanation of why a player was predicted to win. The lack of logical reasoning likely makes language models unfit for tasks like this at its current stage. However, the accuracy of GPT-3.5 was not far off of traditional models, providing hope that this task may one day be possible using LLMs.

Future work should involve a wider array of data. Our data was limited in its inputs, and using more inputs may allow for more creativity for the model. Things like serve speed, return rate, and volley accuracy may allow for better predictions, although the addition of more data may also widen the gap between LLMs and traditional models. Moreover, we were unable to train on the majority of our data due to cost constraints – doing so may yield better results. Finally, as foundational models evolve, they will likely become better at tasks like logical reasoning. Using a more advanced model in the future may yield a different conclusion.