Final Poject Report – Deep Learning

Liad Mordechai and Shoham Grunblat

Spring 2024

Predicting the Olympic Medal Table with Deep Learning

Link to our project: https://github.com/FlyingShohsho/DeepLearningProject

Table of Contents

1. Introduction

The field of sports predictions is wide and has many aspects. In the world and in Israel too, sports gambling is very popular and plays an important part in many fans' lives. In addition, sporting outcomes are a very common conversation topic among various sectors of society, sparking debate, pride, passion and connection between people.
The Olympics are the pinnacle of sporting events, drawing attention of millions globally. Along with the fact that they're the greatest display of human abilities, they introduce a very strong and significant national factor, making the results matter even more. They are also an opportunity for casual sport fans to take interest in many sport events and disciplines which they aren't normally exposed to, and as a result – they introduce a more significant degree of all the elements above: more passion, more unit-pride, more rivalry, and more gambling - often with higher stakes.

Since the Paris 2024 Olympic Games took part during our course, and more specifically, during our work on the project, we saw it as a great opportunity to utilize our newly acquired knowledge from the course to make predictions and test them in real time on the actual results.

As a highly documented and tracked event throughout history and due to the ever-growing public interest, data about Olympic results is very available and profound, making it a very natural target for data analysis and machine learning.
In our background work review, we found previous work is mainly based on classical ML models, reaching fairly accurate results – especially when keeping in mind that sports have a very significant human factor and can never be predicted with 100% certainty.

The problem with learning and prediction tasks is that they require a great amount of data with many features. While the data is vastly available, different datasets contain and document different type of features and are rarely in the exact same format, making merging them a challenging preprocessing task.
With the directions of our instructor, Lior, we put together a plan to use Transformers to solve this problem.
We figured that if we could get a transformer to automatically extract uniformly formatted features from plain text, we could skip the data-processing procedure almost entirely while obtaining a sufficiently large number of features.
In our course, we learned about Hugging Face's bert-base-uncased model, which extracts 768 different features from data and seemed to us as the best fit for our task.

2. Background Work

In a recent method review project[1], several classical methods were tested on results from historic Olympic Games along with a simple Neural Network.
The results were very good, as seen in figure 1:

Name of the Model	Accuracy
Logistic Regression	85.357825%
MultinomialNB	85.367968%
Decision Tree	84.095457%
Random Forest	88.779773%
Gradient Boosting	88.514205%
Neural Network	87.904691%

Figure 1: Results from Method Review

While the accuracies of all models were comparable, the best performing model was a Random Forest model.
When we re-tested the reviewed methods, we noticed that the pre-processing work was intensive, involving merging three different datasets, while still only obtaining 15 features. We tried engineering more features ourselves, but the task of feature engineering makes it even more complicated with diminishing returns. Since the project utilized dimensionality reduction with the “k-best features” method, adding in more features based on the current ones did not help much, while introducing new features required merging more datasets, thus involving even more pre-processing and significantly raising the computational and time complexity.

In additional review[2] we found, the Decision Tree and Random Forest, both classical methods, were compared. In this case, like the previous one, various assumptions and constraints about the data - such as number of participants per country, imbalance between female and male participants and growth of the number of female athletes with time, etc.- had to be adhered to, limiting the expressiveness of the models and increasing the complexity of the task.

---

[1] https://github.com/hrugved06/Olympics-Medal-Prediction

[2] Predicting Medal Counts in Olympics Using Machine Learning Algorithms A Comparative Analysis(2023, Singh et al): https://www.researchgate.net/profile/Nongmeikapam-Thoiba-Singh/publication/378535086_Predicting_Medal_Counts_in_Olympics_Using_Machine_Learning_Algorithms_A_Comparative_Analysis/links/65f5549d1f0aec67e29d3db3/Predicting-Medal-Counts-in-Olympics-Using-Machine-Learning-Algorithms-A-Comparative-Analysis.pdf

Here, too, the analyzed features were fixed and few, which limited the model even more.

The final work we'll present is a work done in parallel to us, trying to predict the 2024 Paris results[3]. Results for this work aren't available yet, and when we attempted to use their model and produce results, we encountered RAM usage limits in the preprocessing stage due to the large number of datasets being used, processed merged. This further emphasizes the advantage and contribution of our approach.

3. Method

3.1. Data Acquisition and Pre-Processing

As our training data we used the Kaggle dataset [Olympic Historical Dataset From Olympedia.org], which details all the medals won in the modern era.
We created a dictionary mapping every participating country's name to its NOC(National Olympic Committee) code, in order to standardize the listing.
We then formed a pipeline, prompting bert-base-uncased to search each participating country's Wikipedia page and extract features from it.
As our test set, we used the [Paris 2024 Olympic medal table,] which is available on Kaggle as well.
Our main pre-processing stage included standardizing the data between both our training and test sets, and converting the training set from an athlete-wise listing to a country-wise, year-wise listing, and excluding irrelevant data for our cause(athlete name, discipline, etc.).
We also normalized and scaled the bert features in with a Standard Scaler in order to avoid order of magnitide bias.

In addition, we tried applying some constraints and filtering to the training set, to account for the fact that not all historical participating entities are currently participating (countries which have since been deformed, banned countries, etc.), and not all currently participating entities have sufficient data (new IOC members, Neutral International Athletes, countries which have never won medals, etc.). A shallow analysis suggested that having won three medals in aggregate over the past six editions (since Sydney 2000) would be an appropriate requirement for a country or entity to be considered for prediction.
As we will later show, applying those constrains harmed our results, and so we proceeded without them and included all 204 participating entities in our prediction.
This further emphasizes the simplicity of our approach – the less preprocessing the better.

[3] https://www.kaggle.com/code/asfefdgrg/olympics-2024-predictions

3.2. Model

Our model was a Neural Network with 3 linear layers, ReLU activation and Dropout after each layer for regularization.
We used Adam as our optimizer.
Our main hyperparameters were the learning rate, layer widths, dropout rate, which we optimized with an Optuna study case on a validation set.

In addition, we tried two loss criteria- MSE and BCE, with MSE being the selected criterion.
The input of our model was a table of medals with entries per country per year, merged with our bert-base-uncased extracted features. In total, the input shape was a tensor with 774 columns – 768 bert features, country code, gold, silver, bronze, total medals and year.

We extracted a 20% large validation set from our training data and trained for 20 epochs in order to tune our hyperparameters.
We then trained the model for 40 epochs on the full training dataset with the learned hyperparameters.

Our learned hyperparameters ended up being:

- `Test Case 1:`
  - `Layer Widths: [961, 361, 256]`
  - `Dropout Rate: 0.3235705115906761`
  - `Learning Rate': 0.000996028437089951`
- `Test Case 2:`
  - `Layer Widths: [529, 464, 254]`
  - `Dropout Rate: 0.1634695845887815`
  - `Learning Rate: 0.000995697498278778`

The output of our model was a medal table, consisting of four predictions for each country: gold medals, silver medals, bronze medals and total medals. We made a total medal prediction in addition to the separate medals because it allows more margin of error and doesn't account for the internal composition of medals a country would win.
To calculate accuracy, we considered a tolerance factor, which was another hyperparameter of the model. The tolerance we used was $\max\{20\%, 2\}$ for each type of medal, and $\max\{20\%, 6\}$ for the total.
Since exact results vary highly and are a matter of chance, we feel confident with this degree of tolerance.

4. Results

In both test cases, our model successfully predicted the winners of the medal table – USA, as can be seen in this table:

1	Country	Predicted Gold	Predicted Silver	Predicted Bronz	Predicted Total	Total Predicted
2	USA	38	33	30	101	101
3	CHN	28	22	19	66	69
4	GER	16	16	20	53	52
5	AUS	14	14	16	44	44
6	GBR	14	13	16	40	43
7	FRA	13	13	14	38	40
8	JPN	12	10	11	31	33
9	KOR	11	9	9	28	29
10	ITA	10	9	10	29	29
11	NED	8	7	8	23	23
12	CUB	8	7	8	22	23
13	ESP	7	6	7	20	20
14	PUR	7	6	6	19	19
15	CAN	6	6	7	19	19
16	UKR	6	6	7	19	19
17	HUN	5	5	6	17	16
18	BRA	5	5	6	16	16
19	ROU	4	4	5	12	13
20	POL	4	4	5	12	13
21	MDA	4	4	5	13	13
22	SWE	3	4	4	11	11
23	NZL	3	3	4	11	10
24	CZE	3	3	3	9	9
25	DEN	3	3	3	10	9

Table 2: Predicted Medal Table, Case 1

1	Country	Predicted Gold	Predicted Silver	Predicted Bronz	Predicted Total	Total Predicted
2	USA	47	38	38	122	123
3	CHN	28	23	21	73	72
4	GER	19	19	20	59	58
5	GBR	14	13	16	43	43
6	AUS	13	13	15	43	41
7	FRA	13	11	12	35	36
8	JPN	12	8	11	31	31
9	KOR	11	11	10	31	32
10	ITA	10	9	10	31	29
11	UKR	8	7	9	24	24
12	NED	8	6	8	23	22
13	CAN	7	7	8	23	22
14	CUB	7	7	8	21	22
15	HUN	7	6	8	22	21
16	ESP	6	6	6	19	18
17	ISV	6	6	6	18	18
18	BRA	6	5	5	17	16
19	POL	5	4	5	15	14
20	ROU	4	4	6	14	14
21	NZL	4	4	5	13	13
22	FSM	4	4	4	12	12
23	HAI	4	4	4	12	12
24	JAM	4	3	4	12	11
25	KEN	3	5	4	13	12

Table 1: Predicted Medal Table, Case 2

1	Country	Actual Gold	Actual Silver	Actual Bronze	Actual Total
2	USA	40	44	42	126
3	CHN	40	27	24	91
4	JPN	20	12	13	45
5	AUS	18	19	16	53
6	FRA	16	26	22	64
7	NED	15	7	12	34
8	GBR	14	22	29	65
9	KOR	13	9	10	32
10	ITA	12	13	15	40
11	GER	12	13	8	33
12	NZL	10	7	3	20
13	CAN	9	7	11	27
14	UZB	8	2	3	13
15	HUN	6	7	6	19
16	ESP	5	4	9	18
17	SWE	4	4	3	11
18	KEN	4	2	5	11
19	NOR	4	1	3	8
20	IRL	4	0	3	7
21	BRA	3	7	10	20
22	IRI	3	6	3	12
23	UKR	3	5	4	12
24	ROU	3	4	2	9
25	GEO	3	3	1	7

Table 3: Actual Medal Table

Our MSE loss function decreased continuously, due to optimizing the learning process and hyperparameters.

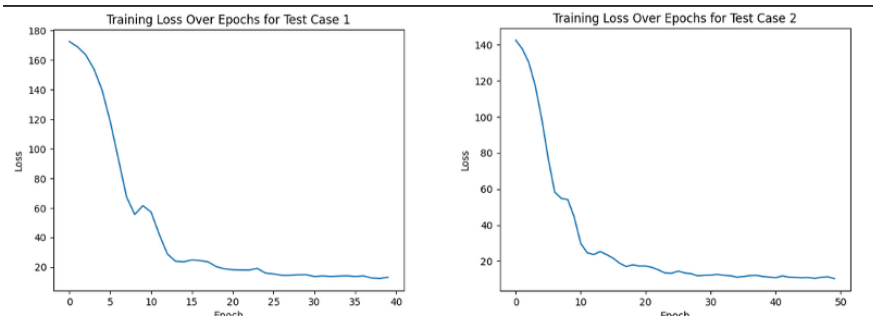

Figure 2: Loss

We kept a few different accuracy measures.
First, we measured the accuracy of prediction for each medal type and total prediction separately.
Then, we measured the accuracy of the sum of the predicted medals.
Lastly, we considered that the official medal table is ranked by Gold, then Silver, then Bronze. So, we assigned weights to the accuracies and received a weighted accuracy.
The accuracies we got can be seen in this table:

Metric	Case 1 Accuracy(%)	Case 2 Accuracy(%)
Gold	79.2208	91.6667
Silver	81.8182	92.1569
Bronze	75.3247	88.7255
Total, predicted	80.5195	91.6667
Total, manual	80.5195	90.6863
Total, weighted	79.35	91.23

Table 4: Accuracy Measurements

Comparing with previous work, we received better results! Even when allowing a lower degree of tolerance ($\max\{10\%, 5\}$ for total, $\max\{10\%, 2\}$ for medals), we received better results than previous work:

Metric	Case 1 Accuracy(%)	Case 2 Accuracy(%)
Gold	78.2712	91.1336
Silver	81.8182	91.452
Bronze	75.112	88.2231
Total, predicted	80.1175	91.0539
Total, manual	80.112	90.1516
Total, weighted	78.85	90.13

Table 5: Accuracy Measurements, Reduced Tolerance

5. Conclusions and Future Work

Predicting sporting results is a complicated task which can never be done with 100% certainty. As demonstrated in this report, our project managed to simplify the task and improve upon previous results, while taking a unique approach and utilizing what we learned in the course.

Our project can be continued and built upon in a few directions.

- Firstly, since the Olympics happen every 4 years, results may be treated as a time-sequence and input to a Recurring Neural Network for the prediction.

- Secondly, as LLM's grow and develop, it may be more beneficial to prompt and LLM and have bert-base-uncased extract features from it rather than Wikipedia, thus gaining access to a wider, dynamic pool of data – virtually the entire internet.

- By the nature of bert-base-uncased's feature extraction, we don't have access to the meaning of our features, only to their values. We are in effect predicting with an implicit model, with a meaningless latent space. As transformers evolve, we may find a way to access the meaning of the features. Dimensionality reduction techniques may then be used to narrow down the features and emphasize the most meaningful ones which are indicative of Olympic success. The data can then be analyzed for the benefit of athletes and societies and utilized to improve circumstances, conditions and opportunities for better performance.

- Our model trains on data from the Summer Olympics, while the dataset we used includes data from the Winter Olympics as well. The Winter Olympics are very different in nature, and the successful Summer Olympic countries aren't necessarily the same as successful Winter Olympic countries, as well as indicators for success in each. Our model can be modified and trained on Winter Olympic results, as it will be interesting to see if the feature extraction works for them.

- Lastly, the model can be re-tested come the next Olympics in 2028.