Project Oracle: Autoregressive Future Event Prediction with Sequential Modeling and Transformers

Stanford CS224N Custom Project

Brian Wu
Department of Computer Science
Stanford University
brianwu@stanford.edu

Katherine Wang
Department of Computer Science
Stanford University
kyw1923@stanford.edu

Ismail Mardin
Department of Computer Science
Stanford University
ikmardin@stanford.edu

Abstract

We introduce an event prediction problem reformulated as an NLP next-token-prediction problem. Leveraging Transformers from common NLP models, we discretize events from GDELT, a large database covering global events since 1979, into tokens that mark the type of event as well as the actors involved. Initial baseline evaluations to this task were inspired by statistical methods (ARIMA) and simple neural methods (feedforward neural network and LSTM); their accuracy levels are incredibly low, however, around 2%, 1% and 0.2%. In this work, we implement the Trajectory Transformer, which can be used to solve RL problems by treating states, actions, and rewards as a sequence, while autoregressively decoding the next action to be taken. Our model vastly outperforms all baselines, achieving an accuracy of around 70% when trained and tested on USA-specific events. Interestingly, this was a significantly better performance compared to global, multi-country data (60-62% accuracy), likely due to reduced complexity and ambiguity in the prediction task. Such a forecasting model for socio-political events may be extremely useful in crisis prevention, policymaking, and resource allocation.

1 Key Information to include

- Mentor: Kaylee Burns; no external collaborators and no project sharing.

2 Introduction

Accurately forecasting future socio-political events is a challenge of significance: applications using these forecasts span crisis prevention, policy formulation, and resource allocation. This task is made difficult by the complexity and inherent uncertainty of real-world dynamics; as such, models that can understand intricate dependencies and interactions between features of events are required. Traditional approaches rely on constructing complex situations and simulations of the world with multiple agents interacting to predict long-term outcomes. However, these methods are complex to design and computationally intensive. In this work, we explore a novel approach to future event prediction using a Transformer-based sequence model, thus representing the event prediction problem as a Natural Language Processing (NLP) task and utilizing an NLP-specific model to capture and predict real-world event sequences.

The GDELT (Global Database of Events, Language, and Tone) dataset monitors print, broadcast, and web news media globally in over 100 languages in order to provide a comprehensive, multilingual repository of socio-political events. In text tokens organized by event feature, GDELT captures the who, what, where, when, and why of global occurrences, encompassing over a quarter-billion event records dating back to 1979 with continuous updates every 15 minutes. Each GDELT event is represented by detailed attributes, such as event type, actors involved, location, and timestamp – making it a rich source of structured information encoded within text. The textual nature of GDELT aligns closely with common NLP tasks involving understanding and generating token sequences.

We seek to apply Transformer-based models to GDELT to predict future events based on historical sequences. Transformers are ideal for this task because they model long-range dependencies and complex relationships in sequential data; as such, they have revolutionized NLP tasks such as language modeling and machine translation and provided state-of-the-art performance in these areas. Inspired by the success of the Trajectory Transformer in reinforcement learning tasks re-formulated as sequence modeling problems, we hypothesize that a similar architecture can effectively model and predict future event sequences in GDELT by capturing the underlying dynamics and dependencies crucial for accurate future predictions.

GDELT leverages the structure of common NLP problems to best represent event sequences: GDELT events are represented as sequences of text tokens, similar to sentences in a language. This allows us to leverage NLP techniques for sequence modeling, treating the prediction of future events akin to next-token prediction in language modeling. Furthermore, the diverse linguistic and thematic coverage of GDELT aligns with NLP's strength in handling varied and multilingual text data, enabling the model to learn from a broad spectrum of global events. The structured representation of events in GDELT, therefore, will allow us to apply dynamic attention mechanisms to enable the model to focus on different aspects of the input sequence based on their relevance, thereby enhancing prediction accuracy.

Current methods for future event prediction often fall short in several areas. Traditional time-series models such as ARIMA lack the ability to capture the complex interdependence between events. Existing event prediction models may not effectively utilize the rich textual information available in datasets like GDELT. Using Transformers, we seek to integrate NLP and sequence modeling within the event prediction problem to address these shortcomings, and use dynamic attention to prioritize relevant events and actions during the future event prediction process. In addition to ARIMA, we further evaluate our model against two simple neural baselines: one involving an Multi-Layer Perceptron (MLP) and another involving a Long-Short Term Memory Network (LSTM), both of which are tuned to predict future event token classes. These comparisons allow us to assess the impact of data diversity and the efficacy of our Transformer-based approach in capturing complex event dynamics.

This work applies NLP techniques to a rich, multilingual dataset of global events in order to reformulate future event prediction as a sequence modeling problem, inspired by similar efforts in NLP. By leveraging the Transformer architecture and the detailed, text-based representation of events in GDELT, we aspire to develop a model that can make accurate and nuanced predictions of future socio-political events. Our work not only highlights the synergy between NLP and event prediction, but also helps set the stage for future research in leveraging large-scale textual datasets for predictive event modeling.

3 Related Work

Reinforcement Learning (RL) has traditionally been approached by decomposing long-horizon problems into smaller subproblems using techniques like dynamic programming and single-step predictive models. Furthermore, these approaches involved setting up complex models of environments to approximate the task being solved, which adds significant design and computational complexity to solving problems in RL. There has been significant work on reformulating RL problems so that other neural architectures can be used to solve these problems. Janner et al. (2021) proposes the Trajectory Transformer, which we base our architecture on in our work. The Trajectory Transformer reformulates RL as a sequence modeling problem – akin to those in NLP – by modeling distributions over entire trajectories using a transformer architecture and repurpose beam search, commonly used in NLP for sequence generation, as a planning algorithm to find high-reward action sequences. Chen et al. (2021)

presents the Decision Transformer, a similar architecture in which the primary major difference compared to the Trajectory Transformer is that the Decision Transformer utilizes a reward-to-go formulation by summing rewards up until a certain state instead of the explicit reward only.

With regards to our main task, Shi et al. (2023) presents LAMP, a framework that utilizes Large Language Models (LLMs) for event prediction tasks. Importantly, they rigorously define how event prediction problems can be decomposed into sequential modeling problems, using alliances and war declarations to predict mobilizations, for instance. Hua et al. (2023) builds upon this and introduces WarAgent, a framework for using LLMs for multi-agent simulations of global conflicts, which introduces a set of guidelines for accuracy analysis within these tasks. Finally, alternatives to GDELT are also emerging, though GDELT remains by far the most comprehensive dataset for the tasks we intend to evaluate our models on. One example of such an alternative is Autocast, a dataset consisting of forecasting questions and accompanying news corpus, which is introduced by Zou et al. (2022).

We will make two major contributions to the existing body of work: (1) We designed a future events prediction task specifically for the Trajectory Transformer implementation, and (2) demonstrate that Transformer-based models leveraging insights from NLP are just as competitive in solving this task compared to traditional neural baselines and more complex RL-based methods.

4 Approach

We began with implementing the Trajectory Transformer architecture as according to Janner et al. (2021). The Trajectory Transformer architecture is similar to the standard Transformer architecture; however, the main difference is that instead of having arbitrary tokens in an input/output sequence, the Trajectory Transformer formulates its sequences in a specific format as follows:

$$(s_1, a_1, r_1, s_2, a_2, r_2, ...)$$

These are sequential tuples of (states, actions, rewards). Note that in contrast to the Decision Transformer, we omit the reward-to-go formulation, meaning we utilize explicit reward tokens instead of a running sum of reward tokens at any given point in the sequence. Furthermore, the objective function used in our Trajectory Transformer model is

$$L(\tau) = sum_{t=1}^{T}(\sum_{i=1}^{N} logP_\theta(s_t^i|s_t^{<i}, \tau_{<t}) + \sum_{j=1}^{M} logP_\theta(a_t^j|a_t^{<j}, s_t, \tau_{<t}) + logP_\theta(r_t|a_t, s_t, \tau_{<t}))$$

which, building off of the sampling process of Transformer-based LLMs, places an emphasis on the decoded log probabilities of the subsequent state, action and reward in the output sequence. As such, many standard RL components – policies, value functions, and dynamic – can all be subsumed by the sequence model, which can both be a predictor and a behavioral policy. In our task, we aim to evaluate the effectiveness of NLP-inspired Transformer models on these tasks to show that NLP-inspired architectures can be used to solve problems in other domains of AI research. Within our input sequences, we define tuples of state, action, and reward tokens as follows:

$$\begin{cases} s & \text{A tokenized sequence of attributes that describe a unique event (full list in section 5.1)} \\ a & \text{The prediction of the next } \textbf{EventCode}\text{, } \textbf{Actor1Code}\text{, and } \textbf{Actor2Code} \text{ tokens} \\ r & \text{Whether or not the predictions match up with the actual codes from the subsequent event} \end{cases}$$

Next, we obtained a subset of data from GDELT that were relevant to our task and applied basic preprocessing techniques. The features that are most relevant to our future event prediction task are described in detail in section 5.1. This was necessary because GDELT is an extremely large dataset given that it is updated every 15 minutes, and guarantees could not be placed on how clean/accurate a certain subset would be. In particular, we noticed instances in which news headlines that provided no concrete information about the state of event that occurred, as well as other headlines that were misclassified. For this reason, we chose to base some of our data cleaning and preprocessing methods based on the work described in Shi et al. (2023) and Hua et al. (2023). This being the case, we examined GDELT in small slices of data, prioritizing recent events when possible due to the fact that we could verify the correctness of the occurred events, and filtered out sections that were not immediately relevant to our prediction task due to irrelevant features, misclassified headlines, etc.

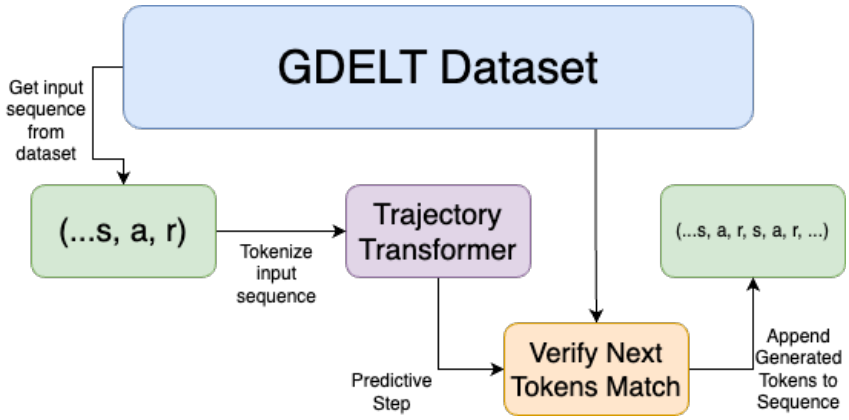

Figure 1: An overview of the future event prediction pipeline. Sequences are sampled from the GDELT dataset and subsequently tokenized before being fed into the trajectory transformer. The trajectory transformer predicts several tokens that describe the subsequently occurring event. These are compared with the actual next event from GDELT, and appended to the sequence.

To evaluate the performance of the Trajectory Transformer model, we utilize three baselines comprising both traditional statistical methods and machine learning-based neural methods. ARIMA (Autoregressive Integrated Moving Average) is a commonly-used statistical model that leverages time series data to forecast future trends, allowing it to predict the most likely future datapoints within a particular series given the existing data. Additionally, we construct a Multi-Layered Perceptron with two linear layers and a hidden dimension of 128; given an input batch of data from GDELT, the model is tasked with predicting only the class in which the next **EventCode**, **Actor1Code**, and **Actor2Code** were most likely to fall into. Similarly, we implemented an LSTM with 3 layers and a hidden dimension of 128: the LSTM was fed the input sequence preprocessed from GDELT, and was tasked with predicting the subsequent tokens falling into the categories of **EventCode**, **Actor1Code**, and **Actor2Code**. The amount and frequency of correct predictions could be directly evaluated against the Trajectory Transformer in this manner.

5 Experiments

5.1 Data

All of the data used in this research comes from GDELT, which monitors global socio-political events from over 100 languages and updates every 15 minutes. Each event record includes attributes such as event type, actors, location, and timestamp, forming a structured sequence of text tokens, thus making this dataset an appropriate choice for NLP tasks and models. In our event prediction task, the input comprises past event sequences, while the output is a set of tokens that predict the attributes of future events.

We chose the following features to include in our input event sequences, as these were most relevant to predicting a future event: **EventCode** (Encoded event type), **Actor1Code** (Encoded code for the first actor involved in the event), **Actor2Code** (Encoded code for the second actor involved in the event), **AvgTone** (Normalized average tone of the event), **GoldsteinScale** (Normalized Goldstein scale – a measure of whether a certain event is more conflictual or cooperative in nature – of the event), and **FractionDate** (Normalized fractional date of the event). From this set of inputs, we aim to predict the **EventCode**, **Actor1Code**, and **Actor2Code** of the most likely subsequent event to occur as determined by the model. We selected dataset dates for our model to test behavior over short-, medium- and long-range time frames, testing on events in 2024 using data and dependencies ranging from 1979 to 2022.

5.2 Evaluation method

We used average loss, Mean Absolute Error (MAE), and model accuracy to assess the performance of our models. We were able to use MAE on our language dataset because GDELT's non-numerical data (the identities of Actor 1 and Actor 2) could be numerically represented through their code. However, because the continuity between Actors close through their numerical code was dubious, we deferred to accuracy (the percentage of correct predictions of the total) for clearest assessment. Given the critical nature of predicting socio-political events, where even small inaccuracies can have significant consequences, we chose an evaluation metric that penalizes even the slightest errors. Accuracy, therefore, provides the most stringent and appropriate measure of our model's performance in this context.

5.3 Experimental details

We were able to run baseline evaluations and smaller-scale experiments locally, and for larger-scale evaluations involving Trajectory Transformers with more layers, we employed a remote server with an NVIDIA RTX 3090 GPU.

We trained the model first on a dataset listing events from 2024/05/29. Our accuracy levels, while beating all baselines, were not too high (30 - 45%). To refine performance, we first conducted extensive manual hyperparameter tuning. We found that 4 transformer layers with embedding, attention, and residual dropout equal to 0.5 produced the best results for short-term training over 3 epochs, ensuring model complexity while avoiding overfit. However, performance remained moderate, with a highest accuracy of 0.493.

Next, we tried to see if we could improve our accuracy by modifying the training data itself. One successful idea was to filter the training dataset for events relating only to a certain country, in this case the USA, and seeing how it performed on a USA-specific test dataset. In this case, we would only be predicting the EventCode, and one ActorCode. To ensure that the level of recency to the test dataset was not skewing our results, we tested our hypothesis on three datasets spanning different time periods: events from 1979, 2013, and 2022. This wide range of dates allowed us to assess the model's performance under varying temporal distances from the test dataset, providing a more comprehensive evaluation of its predictive capabilities. For comparison, we first trained the model on each original multi-country dataset, and tested on a common multi-country dataset from 2024/05. Then, we repeated this process, filtering each dataset for USA-specific events beforehand. The results are detailed in section 5.4.

5.4 Results

MAE and accuracy provide insights into the absolute performance and classification correctness of our models, respectively. The average loss, on the other hand, offers a normalized measure of the model's overall discrepancy between predicted and actual values across the evaluation dataset.

Our results from using multi-country datasets are shown in Table 1. These evaluations were made on the same test dataset with multi-country events from 2024/05.

Metric	1979 Dataset	2013 Dataset	2022 Dataset
Average Loss	0.5789	0.3462	**0.2113**
Mean Absolute Error	176.441	**172.469**	178.453
Accuracy (Ratio of Correct Predictions)	0.5972	0.6203	**0.6225**
Correct Predictions (out of Total Samples)	4644/7776	4823/7776	**4841/7776**

Table 1: Multi-Country Dataset Evaluation Results

Our results from using the USA-specific training datasets are shown in Table 2. These evaluations were made on the same test dataset with USA-related events from 2024/05.

The baseline performances are shown in Table 3.

We can see here that in both multi-country and USA-specific training methods, we have vastly outperformed all baseline accuracies. The USA-specific training however leads to an even more

Metric	1979 Dataset	2013 Dataset	2022 Dataset
Average Loss	0.1822	**0.1528**	0.2547
Mean Absolute Error	34.1683	34.4409	**33.2340**
Accuracy (Ratio of Correct Predictions)	0.7005	**0.7151**	0.7021
Correct Predictions (out of Total Samples)	1345/1920	**1373/1920**	1348/1920

Table 2: USA-Specific Dataset Evaluation Results (more graphs in Figures 5-10 of Appendix)

Metric	ARIMA	MLP	LSTM
Average Loss	N/A	1298.3433	150.7970
Mean Absolute Error	N/A	25.6786	7.2222
Accuracy	0.02035	0.010	0.001616

Table 3: Baseline Results: ARIMA, LSTM, and Simple NN

improved accuracy. The differences between the different training datasets (1979 vs 2013 vs 2022) are also negligible, showing that the level of recency is not a significant contributing factor.

6 Analysis

At its core, the Trajectory Transformer model works by learning to predict the next event in a sequence based on patterns in the historical event data it was trained on. It does this by attending to relevant information across the entire input sequence using the self-attention mechanism, allowing it to capture long-range dependencies and contextual cues that may inform what is likely to happen next.

We found that training the model on datasets specific to a single country, in this case the United States, led to significantly improved accuracy compared to training on a global dataset (around 70%). This makes intuitive sense, as focusing on a single country limits the scope of possible events and actors, making it easier for the model to learn meaningful patterns and correlations between actions originating from the same geopolitical entity. The model is better able to capture the nuances and dependencies in event sequences when they are contextualized within a specific national setting.

In contrast, the model's performance was notably worse when trained on datasets spanning multiple countries (around 60%). This can be attributed to several factors. Firstly, the increased diversity of actors and event types across different countries introduces much more complexity and ambiguity into the prediction task. What may be a reliable pattern in one country's event sequences may not hold in another's, due to differences in political systems, international relations, economic conditions, and so on. This makes it harder for the model to learn generalizable patterns. Secondly, the self-attention mechanism, while powerful, may struggle to consistently attend to the right cues when the input sequences are highly heterogeneous and involve many distinct actors. With a larger set of countries, there is increased potential for spurious correlations and misleading subsequences that the model could wrongly fixate on. The model may also struggle to represent the intricate web of relationships and dependencies between different international actors across varied contexts. Furthermore, the data sparsity issue is exacerbated with multi-country datasets, as there are limited historical examples of certain types of cross-country interactions to learn from. This is particularly problematic for rarer event types and actors, leading to poorer predictive performance on those cases. For instance, it might predict that the US would engage in material cooperation with the wrong country, even though material cooperation was the correct next event type, due to the rarity of that country within the training dataset.

We also visualized each of our 8 attention heads using heatmaps (Figure 2). This figure is from the 1979 training dataset, though the heatmaps for the model trained on 2013 and 2022 datasets are similar (Appendix Figure 3, 4); it shows how the model attends to different parts of the input sequence when making predictions.

Each attention matrix shows a stronger diagonal pattern, indicating that tokens pay more attention to themselves and their immediate neighbors. This is expected, as neighboring events are time- and actor-related, meaning more frequent dependencies. The upper right quadrant of the attention

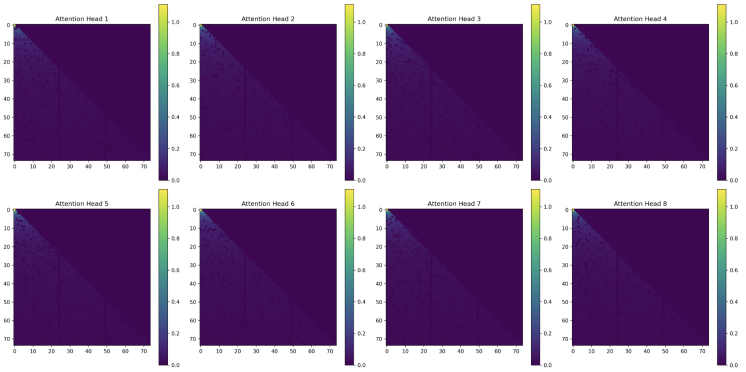

Figure 2: Attention Head heatmaps

matrices shows no attention allocation. This is due to the autoregressive nature of the model, where future events are masked to prevent the model from using information it should not have access to when making predictions. The model only attends to past and current events, ensuring predictions are made causally.

7 Conclusion

In this work, we explored a novel approach to future event prediction by reformulating it as a sequence modeling problem and leveraging Transformer-based models. By training a Trajectory Transformer model on sequences of socio-political events from the GDELT dataset, we demonstrated that NLP techniques can be effectively applied to capture complex dependencies and predict future events with improved accuracy compared to traditional baselines.

The key insight behind our approach is that event prediction can be framed as a language modeling task, where the goal is to predict the next "token" (i.e., event) in a sequence based on the historical context. This allows us to harness the power of Transformer architectures, which have revolutionized NLP by enabling models to learn rich, contextual representations of text sequences. By treating event sequences analogously to sentences in a language, we can apply similar techniques to model the complex dependencies and patterns that govern the unfolding of real-world events.

Our experiments reveal several key findings that highlight the successful transfer of NLP techniques to the event prediction domain. First, we show that Transformer models, with their ability to model long-range dependencies through self-attention, are well-suited for learning intricate patterns in event sequences. This allows them to make more accurate predictions compared to simpler models like ARIMA, LSTMs, and feedforward neural networks.

Second, we find that training on country-specific datasets leads to significantly better performance compared to training on global, multi-country data. This insight mirrors findings from NLP, where language models often perform better when trained on domain-specific corpora that capture the nuances and idiosyncrasies of particular linguistic contexts. Similarly, by contextualizing event sequences within specific geopolitical settings, the model can learn more meaningful and consistent patterns.

In future work, we aim to expand the training dataset's time scope to weeks and months rather than days, thus enabling our model to capture even sparser, longer-range connections and causalities. We are currently limited by processing power in this direction, as even one day's events may contain hundreds of thousands of rows (eg: 06/07/2024 contains 141k with a file size of 55.5MB). We would also test our country-specific approach on nations besides the USA, then conduct comparative analysis of model performance over different nations. This may help us understand where the GDELT dataset has deficiencies. It may also provide rudimentary understandings of nations' volatility. Finally, it would be relevant to try isolating our model's prediction task to solely the Event, keeping both Actor 1 and Actor 2 constant. This models a common scenario where one would want to predict the future relationship of two geopolitical entities, meaning there is great value in accurate inference performance.

In conclusion, our work demonstrates promising results in leveraging Transformer architectures for the complex task of future event prediction, and showcases the versatility of NLP-inspired approaches to predictive performance. We hope that by bridging the gap between these domains, we open up exciting possibilities for policymakers, researchers, and organizations—both governmental and non—to do their work in a more informed and cohesive manner.

8 Statement of Ethical Considerations

Constructing a system that claims to be able to predict future events presents significant ethical challenges and societal risks, which require careful consideration and mitigation on the creators/operators' end. One potential issue is the risk of biased predictions and feedback loops: if our model learns from biased historical data, it may perpetuate and amplify these biases – leading to predictions that reinforce harmful stereotypes or discriminate against certain groups. Our project involves predictive modeling based on historical data, which is often inherently biased due to existing societal disparities. Such biased predictions could adversely influence decision-making processes and exacerbate social inequalities. We will mitigate this by carefully preprocessing the training data to identify biases, using common bias detection algorithms and consulting with historical experts to ensure that our data is fair and representative. We will also emphasize the model's uncertainties and limitations when presenting results, ensuring that users are aware of these potential biases and the need for critical evaluation of the model's performance. Another significant risk is the potential for malicious use of the model: there is a concern that bad actors could exploit the model's capabilities to initiate events that would enable them to achieve a certain outcome. We believe that the powerful predictive capabilities of our model could be misused if access is not properly controlled, so we will apply stringent use case limitations. This includes authentication mechanisms to ensure only authorized users can access the model predictions. We will also monitor usage patterns to detect and prevent any attempts at misuse. Additionally, we believe that establishing clear guidelines and policies for acceptable use allows us to prevent the model from being employed for malicious purposes and ensure that its application remains ethical & beneficial to society.

9 Team Contributions

Brian primarily worked on implementing the Trajectory Transformer, baseline models, and experimental framework described in this research. He also contributed to writing the report and creating the poster.

Ismail worked on loading and preprocessing the datasets, running the experiments, implementing the Trajectory Transformer, and contributed to writing the report and creating the poster.

Katherine worked on implementing the Trajectory Transformer, loading and preprocessing the datasets, running the experiments, and contributed to writing the report and creating the poster.

References

[1] Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Michael Laskin, Pieter Abbeel, Aravind Srinivas, Igor Mordatch. Decision Transformer: Reinforcement Learning via Sequence Modeling. https://doi.org/10.48550/arXiv.2106.01345

[2] Michael Janner, Qiyang Li, Sergey Levine. Offline Reinforcement Learning as One Big Sequence Modeling Problem. https://doi.org/10.48550/arXiv.2106.02039

[3] Andy Zou, Tristan Xiao, Ryan Jia, Joe Kwon, Mantas Mazeika, Richard Li, Dawn Song, Jacob Steinhardt, Owain Evans, Dan Hendrycks. Forecasting Future World Events with Neural Networks. https://doi.org/10.48550/arXiv.2206.15474

[4] The GDELT Project. https://www.gdeltproject.org/

[5] Wenyue Hua, Lizhou Fan, Lingyao Li, Kai Mei, Jianchao Ji, Yingqiang Ge, Libby Hemphill, Yongfeng Zhang. War and Peace (WarAgent): Large Language Model-based Multi-Agent Simulation of World Wars. https://doi.org/10.48550/arXiv.2311.17227

[6] Xiaoming Shi, Siqiao Xue, Kangrui Wang, Fan Zhou, James Y. Zhang, Jun Zhou, Chenhao Tan, Hongyuan Mei. Language Models Can Improve Event Prediction by Few-Shot Abductive Reasoning. https://doi.org/10.48550/arXiv.2305.16646

A Appendix (optional)

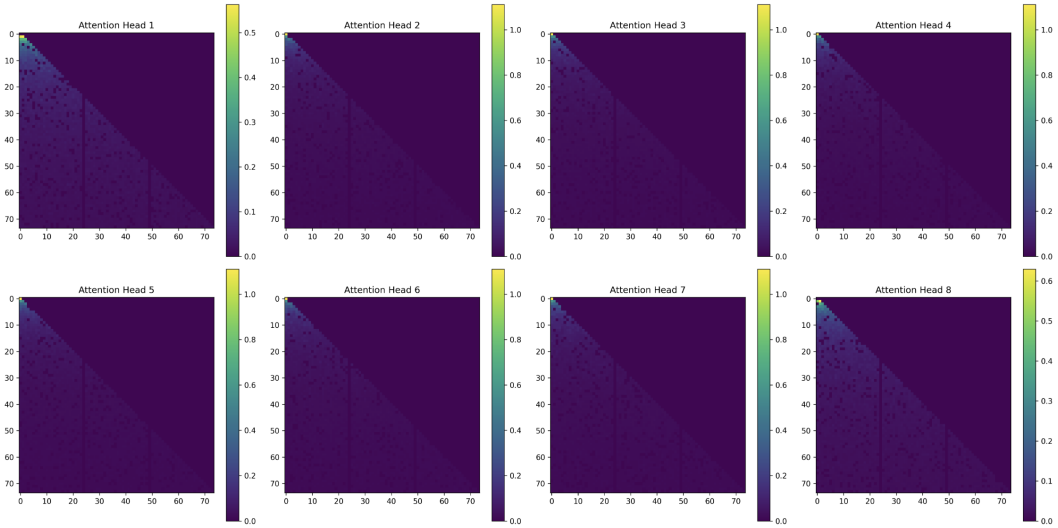

Figure 3: Attention Heads from 2013 dataset

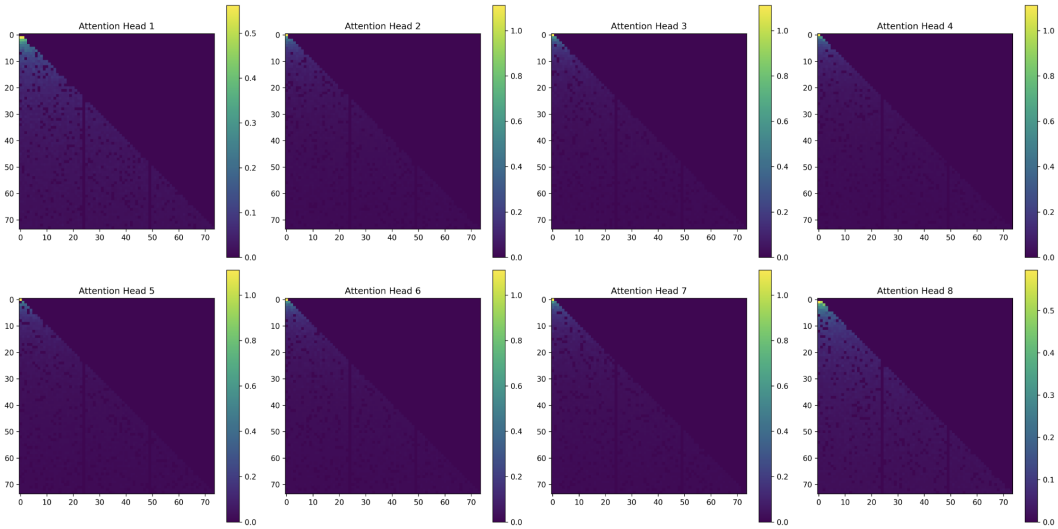

Figure 4: Attention Heads from 2022 dataset

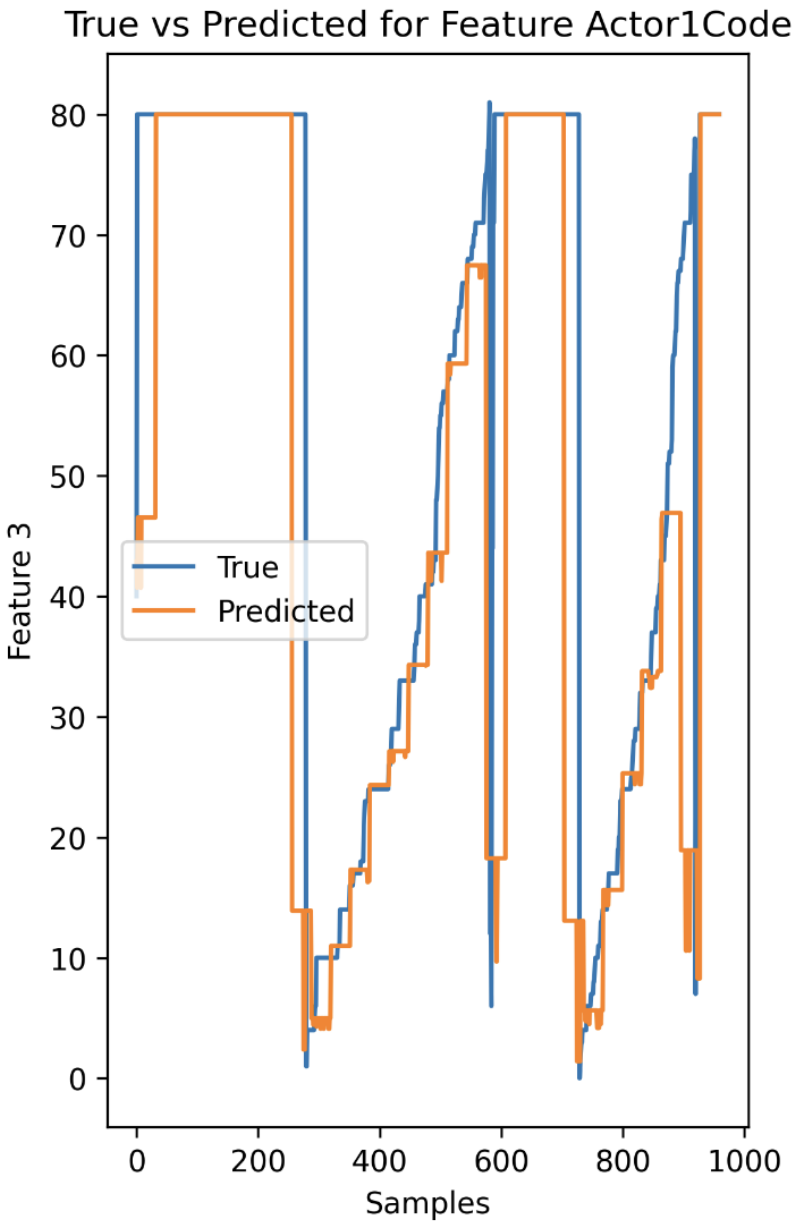

Figure 5: Predicting actor codes after training on 1979 dataset

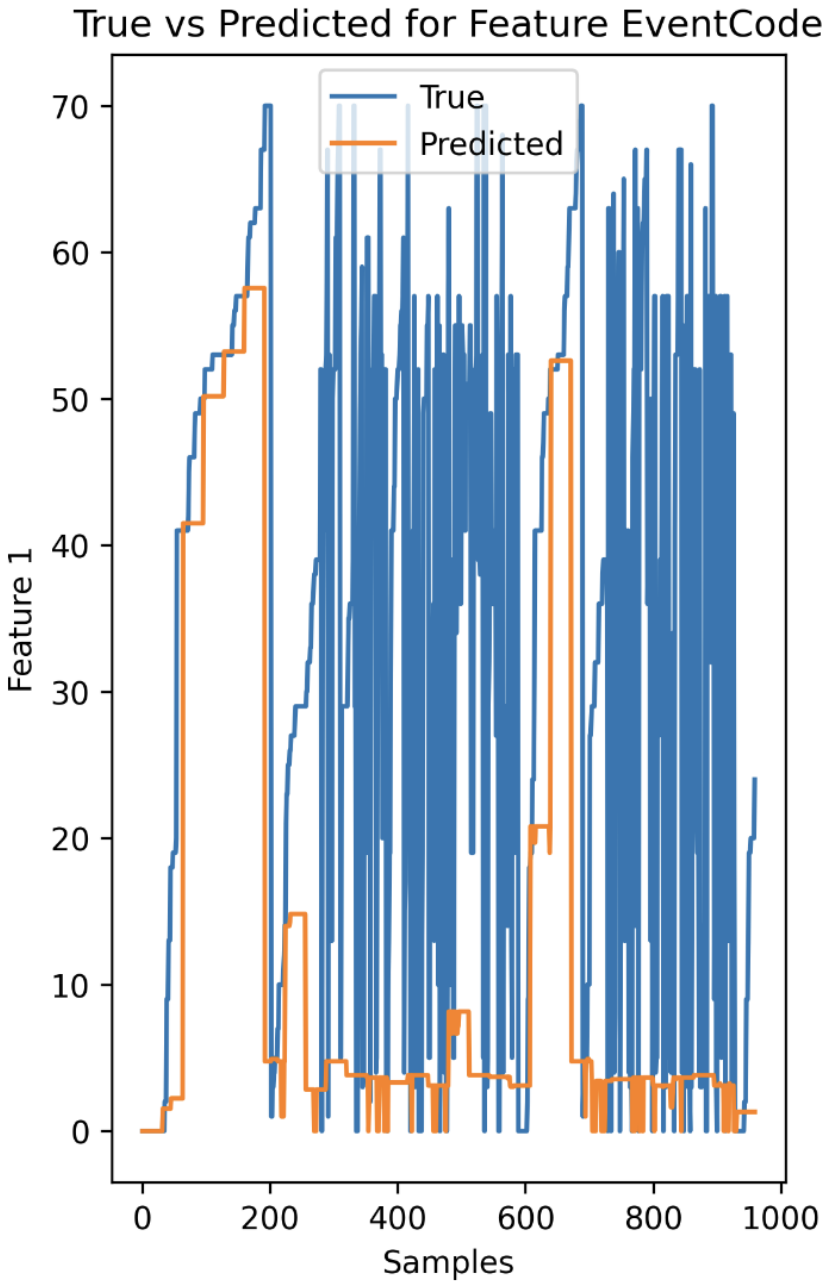

Figure 6: Predicting event codes after training on 1979 dataset

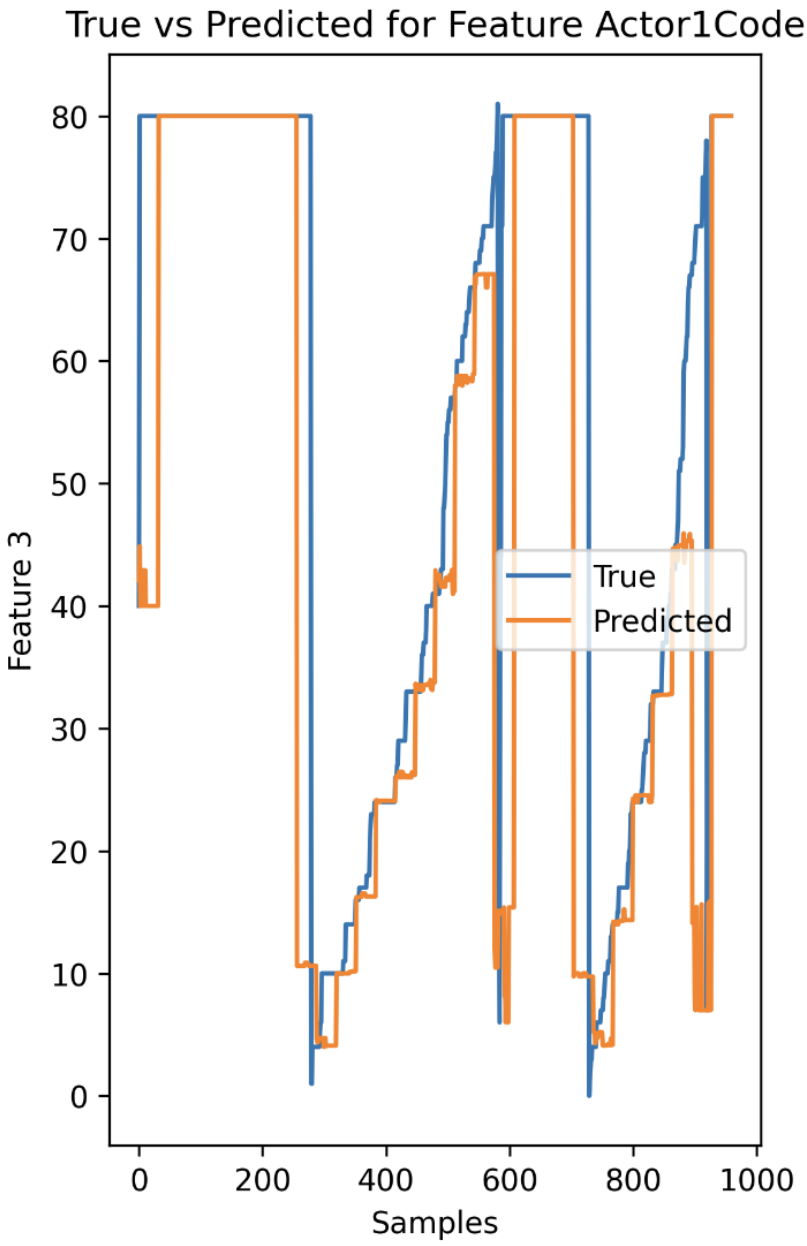

Figure 7: Predicting actor codes after training on 2013 dataset

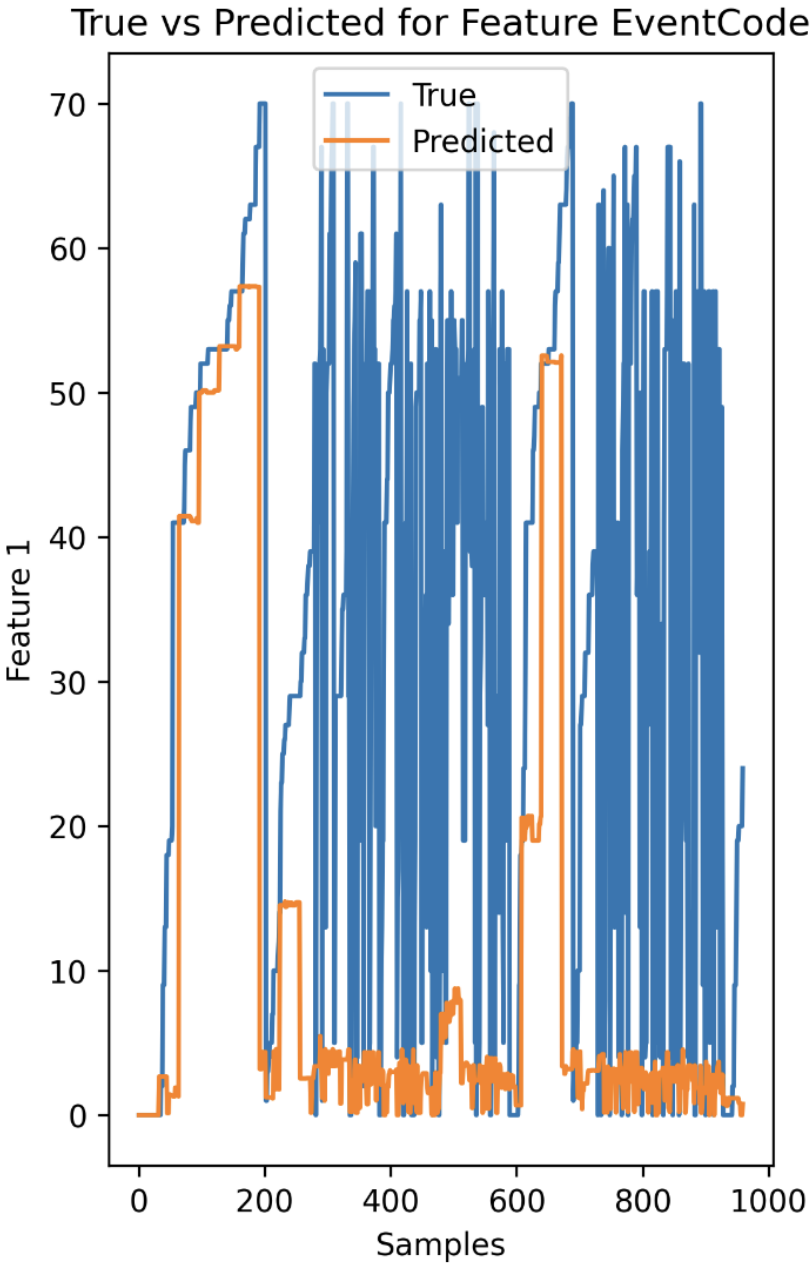

Figure 8: Predicting event codes after training on 2013 dataset

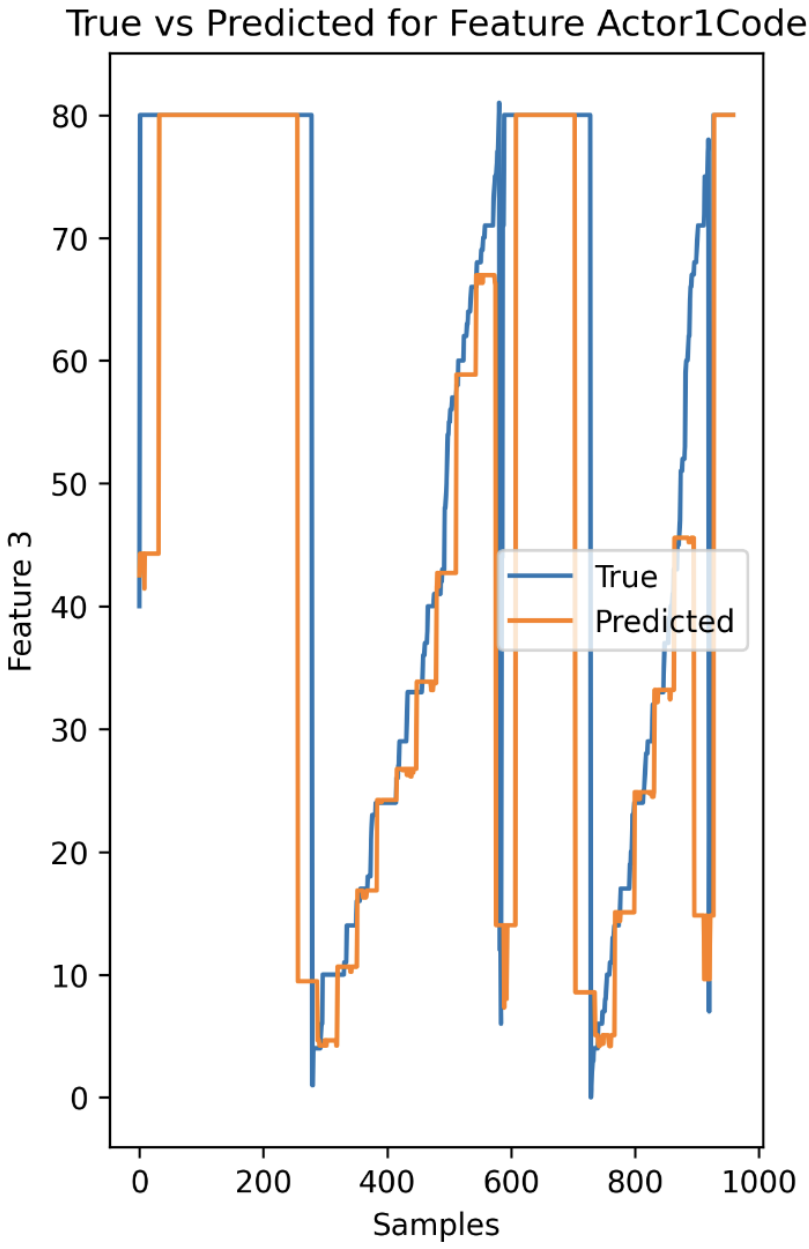

Figure 9: Predicting actor codes after training on 2022 dataset

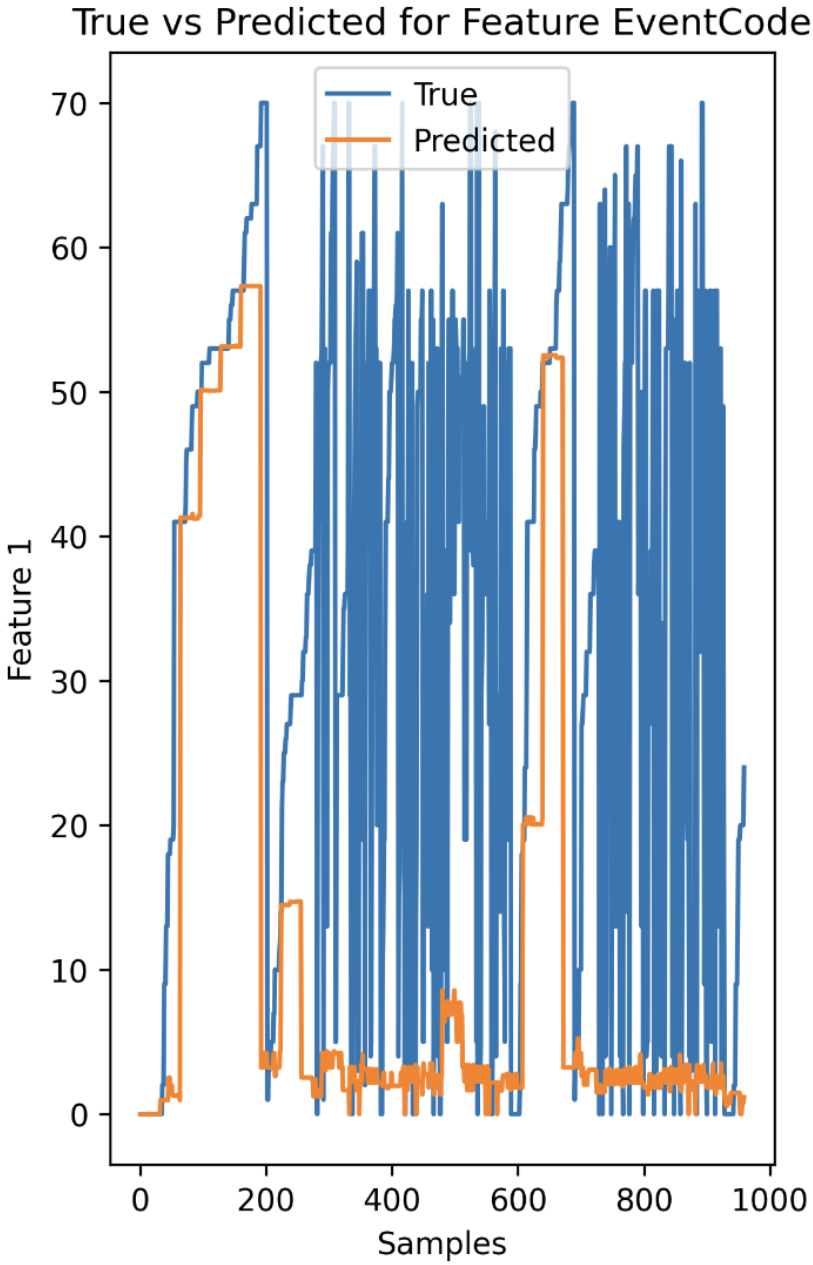

Figure 10: Predicting event codes after training on 2022 dataset