Human Choice Predictions in Language-based Persuasions Games: Incorporating new hybrids architectures

Karsenty Eva and **Bijaoui Tom**
Technion – Israel Institute of Technology

evakarsenty@campus.technion.ac.il , tom.bijaoui@campus.technion.ac.il

Abstract

The rapid advancements in Large Language Models (LLMs) have driven the creation of LLM-based agents designed for interactive tasks involving both human and artificial agents. This paper advances previous research and try to introduces a novel approach to enhancing the performance of these agents in predicting human decisions in off-policy evaluation (OPE), specifically within language-based persuasion games. Our objective is to improve decision prediction accuracy through the creation of new hybrid architectures. To achieve this, we implemented and evaluated three distinct hybrid models combining LSTM and Transformer architectures: a stacked model, a concatenated model, and a fully connected hybrid model. Moreover, we also performed modifications in the Transformer model. Our experimental results demonstrate a slight performance, while in the majority of the cases, we do not note a very important improvement . These findings show the potential of hybrid architectures in advancing the predictive capabilities of LLM-based interactive agents.

1 Introduction

In recent times, considerable endeavors have been directed towards crafting adept artificial agents for gaming contexts, with emerging methodologies focusing on language-based approaches. In fact, Apel et al. were the first to focus on natural language communication, especially in persuasion games where this type of interaction is essential. They introduced a distinctive multi-stage, non-cooperative language-based persuasion game. In this setup, an expert (travel agent) and a decision-maker (DM), the customer, interact. The expert selects a hotel review from a set of scored textual reviews to persuade the DM to choose the hotel. The DM's acceptance or rejection of the recommendation results in stochastic payoffs, determined by the review scores known only to the expert. After observing the payoffs, both players move to the next stage, which involves a different hotel but follows the same structure.

Apel et al. (2022) focused on predicting DM actions, while Raifer et al. (2022) developed an artificial expert using the Monte Carlo Tree Search algorithm, leveraging deep learning models to anticipate DM actions and maximize hotel acceptance.

Shapira et al. (2024) paper extends previous work by predicting human agents' choices in games based on interactions with various artificial agents rather than determining optimal policies. They use off-policy evaluation (OPE) to test strategy profiles, pairing an untried bot expert with an unseen human DM to predict the DM's decisions. Their objective was to create predictive models that reveal how humans respond to unfamiliar partners, based on interactions with known counterparts. Building on the importance of simulation components like learning-over-time, past behavior, and review content modeling, our work has for goal to implement new language-based architectures to enhance existing predictive models for DM decision-making.

The same algorithmic strategy will persist, incorporating both human-bot and simulated DM-bot interactions. Additionally, our DM simulation model will also assumes that DMs employ strategies informed by past game behavior and review content, evolving over time irrespective of the sender's specific strategy.

2 Related Work

Actions and Decisions Predictions in ML and NLP In the fields of ML and NLP, predicting human decision-making actions has been explored in various contexts (Plonsky et al., 2019;

Rosenfeld and Kraus, 2018; Bourgin et al., 2019). While these studies did not focus on language, other research attempts to predict human decisions in language-based contexts, like Oved et al. (2020) on predicting NBA players' actions from interviews. On the other hand, Apel et al. (2022) introduced a language-based persuasion game to predict decision-makers' actions. Shapira et al. (2024) proposed a novel approach for predicting human decisions in language-based persuasion games using LLM-based agents, enhancing off-policy evaluation through a simulation technique and achieving significant gains in prediction accuracy.
We adopt their setup but focus on creating new NLP models' architectures in order to enhance the performance of the prediction accuracy.

NLP models and architectures Over the years, numerous researchers have explored the development of new NLP architectures by leveraging and combining existing models. Xiangyang et al. (2018) proposed a text classification algorithm based on a CNN-LSTM hybrid model, while Yan et al. (2019) developed a hybrid deep learning model combining an ensemble long short term memory (LSTM) neural network with the stationary wavelet transform (SWT) technique to predict energy consumption for individual households. Recent studies have also introduced hybrid NLP models in the statistical domain, such as Kumar et al. (2023), who proposed a hybrid βSARMA-LSTM model for time series forecasting. Similarly, several papers have discussed hybrid transformer models: Liu et al. (2023) introduced a Spatial-Temporal Gated Hybrid Transformer Network (STGHTN) to improve traffic flow forecasting, and Padilha et al. (2022) proposed a hybrid system combining the traditional SARIMA model with a Transformer neural network for renewable energy forecasting. Our paper aims to propose several hybrid models that combine Transformer and LSTM architectures to enhance prediction accuracy. We also note that a similar architecture was proposed by Chen et al. (2022) for a different purpose and domain (industrial users).

3 Models

In this section, we will describe the three architectures models we choose to add for the study, but also the slight modified transformer model we build. For each model, we used a consistent training function, training across three versions: one with only human-bot interaction data, one with only simulation data, and one with both interaction and simulation data. All models are designed to predict the DM's decision in a specific round, based on the previous rounds in the same bot-DM interaction. Each of our new architectures combines LSTM and Transformer models, leveraging the strengths of both. This combination enhances performance, provides better contextual understanding, and offers greater flexibility and robustness in handling sequential data.

Transformer We implemented an enhanced Transformer architecture, building on the classical Transformer model (Shapira et al. , 2024) by incorporating positional encoding and a dropout regularization parameter. We added positional encoding to help the model understand the order of input data, in order to help the Transformer model effectively handle sequential data, and dropout regularization to prevent overfitting. These changes have for objective to enhance the Transformer's ability to handle sequential information and generalize better to new data.

Stacked Model For this model, we first define an LSTM model (Shapira et al. , 2024) to process data vectors, initializing the cell state before the Decision Maker's (DM) first game with a vector estimated during training, while propagating the hidden state across games. This models the relationship between successive games against the same expert. The LSTM output is then fed into the Transformer model. By stacking LSTM and Transformer models, the combined architecture benefits from the LSTM's ability to remember long-term dependencies and the Transformer's strength in capturing global context. This synergy can lead to better performance in tasks requiring both sequential processing and understanding of broader relationships within the data.

Concatenated Model This model comprises an LSTM component that processes input sequences, followed by a Transformer component for further processing. The LSTM processes the input and outputs a transition dimension representation. The Transformer also processes the input, producing another transition dimension representation.

The outputs from both the LSTM and Transformer are then concatenated along the feature dimension. This concatenated output is passed through a fully connected layer (with dropout and activation functions) to produce the final output, which is then returned in a dictionary.

Fully Connected Hybrid Model This model utilizes three distinct architectures - LSTM, Transformer, and Stacked - to process input data, aiming to leverage the unique strengths of each for enhanced performance. Input data is separately sent to each of the three models. Subsequently, the outputs from the LSTM, Transformer, and Stacked components are then concatenated. This combined output undergoes processing through a fully connected layer, integrating dropout and activation functions, to generate the final output.

4 Data

For this study, we used the exact same dataset collected for the prior study (Shapira et al. , 2024), as our focus was on modifying the model architecture rather than altering the data. The data collection process and the dataset details are as follows: A mobile phone game app was developed, following a multi-stage language-based persuasion game. The human Decision Maker (DM) interacts with a series of 6 rule-based experts (bots, either EA or EB), each game consisting of 10 rounds. The DM earns 1 point for each good decision, with a maximum of 10 points per game. To advance, the DM must achieve a target payoff of 8-10 points, depending on the bot's difficulty. The goal is to reach the target payoff for all six experts, described as "defeating" the expert, despite the game's non-adversarial nature.

Hotels The hotel reviews used in the game were sourced from Booking.com, resulting in a dataset of 1,068 hotels, each with 7 scored reviews. Hotels were selected so that only about half were classified as good, with a median score of 8.01.

Interaction Data The game, that was available on Apple's App Store and Google Play from May 2022 to January 2023, featured group EA experts until November 2022 and group EB experts from December 2022 onwards. They have thus collected data from 87,204 decisions made by 245 players who completed the game by defeating all six experts. Incentives included lottery participation and course credit.

By leveraging this established dataset, we aim to evaluate the effectiveness of our new model architecture under consistent and reliable conditions.

5 Experiments

Features Representations We adopt feature representations from Shapira et al. (2024), characterizing each DM-bot interaction round with features related to the hotel review provided by the expert and the strategic context of the decision. Reviews are represented using binary Engineered Features (EFs), initially proposed by Apel et al. (2022), which describe the topics and structural properties of the review. These features are labeled using OpenAI's Davinci model, which assesses whether each feature appears in the review. Additionally, we include binary features that capture the DM's previous decisions and outcomes, current payoff, and frequency of choosing hotels in past rounds.

Evaluation metrics We evaluate our models using the accuracy score for predictive performance, and by comparing total loss values and the probabilities of choosing the right action during the testing phases to understand learning and generalization.

Optimization and Regularization We optimize our models using Adam (Kingma and Ba, 2015), when our learning rate is initialized to 1e-3. Additionally, we kept the default hyperparameter values, as these settings produced the best results. Furthermore, we maintained the same number of epochs to assess the stability and consistency of our models.

6 Results

In this section, we present the results of each model based on our evaluation metrics and compare them with the LSTM results reported by Shapira et al.

New Transformer Model Using our new transformer architecture model, we achieved an accuracy of 78.12%, compared to the 77.98% accuracy of the original transformer. This slight improvement can be attributed to our enhanced positional encoding and dropout regularization parameters.

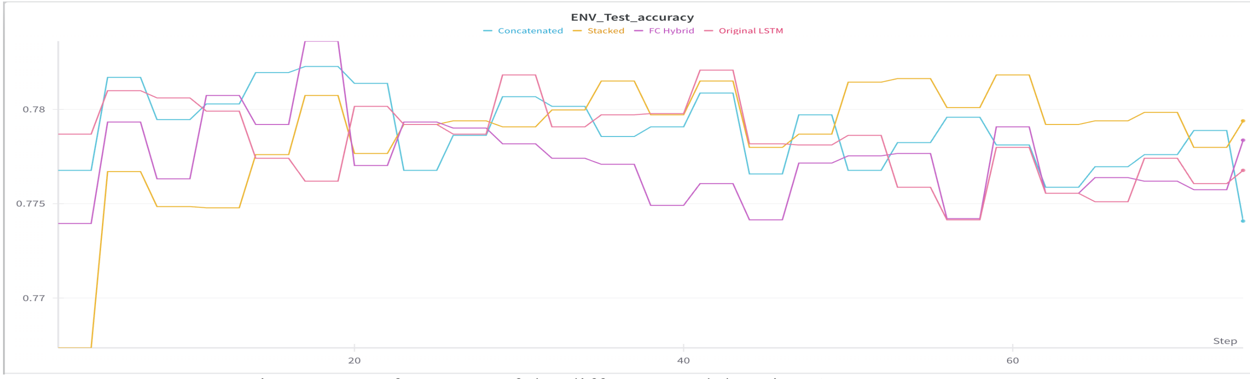

Figure 1 : Performance of the different models using Accuracy measure

Stacked Model With this architecture, we achieved an accuracy of 78.18%, slightly lower than the original LSTM's 78.21%. However, while the LSTM's performance drops after a certain number of epochs, our stacked model maintains consistent and stable results over time, without significant declines. Additionally, the test total loss for the stacked model is very satisfying, often lower than that of the original LSTM after a certain number of steps. Despite this, the LSTM model slightly outperforms our stacked model in the online simulation probability of selecting the correct action (0.5957 vs. 0.594).

Concatenated Model This model yielded an accuracy of 78.23%, marginally better than the LSTM's accuracy. While the performance decreases slightly after several epochs, it is less pronounced compared to the LSTM, indicating a slight improvement. In terms of test total loss, the curves for both models are intertwined, suggesting no significant improvement with this architecture. However, the concatenated model performs better in the online simulation probability of selecting the correct action (0.5957 compared to 0.5945 for the LSTM).

Fully Connected hybrid model This model achieved an initial accuracy of 78.23%, which decreased slightly to stabilize around 77.7%. This pattern is similar to the LSTM model but occurs slightly earlier. Regarding test total loss, the results are very similar to those of the LSTM, with slightly higher values at certain points. However, the fully connected hybrid model slightly outperforms the LSTM in the online simulation probability of selecting the correct action (0.596 vs. 0.5957 for the LSTM).

7 Conclusion And Discussion

We proposed several variations to the classical approach to language-based models, combining them into new hybrid structures: the stacked model, the concatenated model, and the fully connected hybrid model. Additionally, we reviewed and enhanced the Transformer architecture with positional encoding and regularization methods. Experimental results demonstrate that these models can achieve performance on par with, and sometimes superior to, the classic LSTM model. This suggests a promising new direction for exploring architectures in human choice prediction.

References

SHAPIRA, Eilam, APEL, Reut, TENNENHOLTZ, Moshe, *et al.* Human Choice Prediction in Non-Cooperative Games: Simulation-based Off-Policy Evaluation. *arXiv preprint arXiv:2305.10361*, 2023.

SHE, Xiangyang et ZHANG, Di. Text classification based on hybrid CNN-LSTM hybrid model. In : *2018 11th International symposium on computational intelligence and design (ISCID)*. IEEE, 2018. p. 185-189.

YAN, Ke, LI, Wei, JI, Zhiwei, *et al.* A hybrid LSTM neural network for energy consumption forecasting of individual households. *Ieee Access*, 2019, vol. 7, p. 157633-157642.

KUMAR, Bhupendra, YADAV, Neha, *et al.* A novel hybrid model combining βSARMA and LSTM for time series forecasting. *Applied Soft Computing*, 2023, vol. 134, p. 110019.

LIU, Jiansong, KANG, Yan, LI, Hao, *et al.* STGHTN: Spatial-temporal gated hybrid transformer network for traffic flow forecasting. *Applied Intelligence*, 2023, vol. 53, no 10, p. 12472-12488.

GALINDO PADILHA, Guilherme Afonso, KO, JeongRyun, JUNG, Jason J., *et al.* Transformer-Based hybrid forecasting model for multivariate renewable energy. *Applied Sciences*, 2022, vol. 12, no 21, p. 10985.

AL-ALI, Elham M., HAJJI, Yassine, SAID, Yahia, *et al.* Solar energy production forecasting based on a hybrid CNN-LSTM-transformer model. *Mathematics*, 2023, vol. 11, no 3, p. 676.

CHEN, Yuhao, WANG, Yuanyuan, LIU, Xiaohan, *et al.* Short-term load forecasting for industrial users based on Transformer-LSTM hybrid model. In : *2022 IEEE 5th International Electrical and Energy Conference (CIEEC)*. IEEE, 2022. p. 2470-2475.

A Additional Results Graphs

In this section we present additional graphs that show the performance of our multiple architectures, using different evaluation metrics. We note that all of our graphs (including Figure 1) came from the wandb visualizations of our experiments.

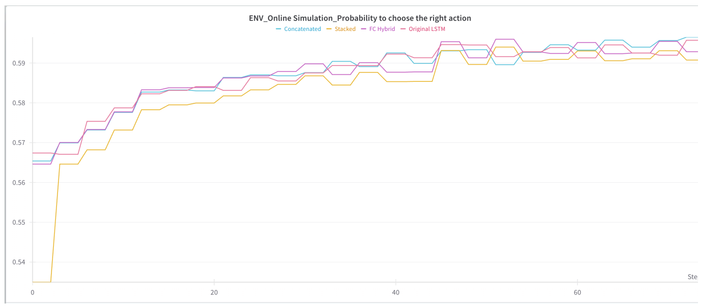

Figure 2 : Performance of the different architectures using Online Simulation Probability to choose the right action measure

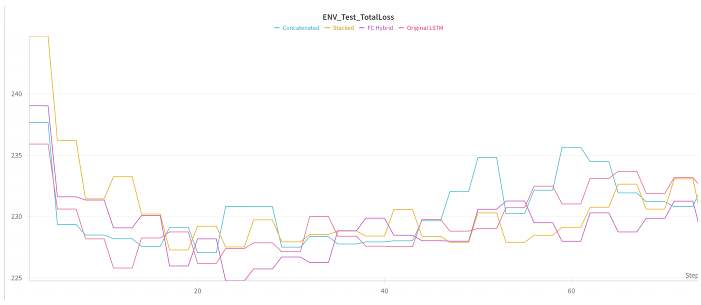

Figure 3 : Performance of the different architectures using the test total loss measure