RealEstateDL: Deep Learning and Reinforcement Learning for Real Estate Investment

David Nguyen
CSCI 5931: Biswas

1. Problem Statement and Motivation

The real estate market is inherently volatile, influenced by economic indicators, interest rates, and market trends. This project aims to build a deep reinforcement learning (DRL) agent that can make intelligent property investment decisions based on historical property prices and macroeconomic indicators. By combining LSTM-based forecasting with a DQN agent, we aim to simulate a decision-support system capable of adapting to dynamic market conditions and learning from prior episodes.

2. Methods

The project combines two core components:
- A deep learning model (LSTM) for forecasting property prices based on historical HPI, GDP, and dividends.
- A deep Q-network (DQN) reinforcement learning agent trained in a custom OpenAI Gym environment to make buy/hold/sell decisions.

The state space includes normalized economic indicators, account balance, and number of properties owned. The agent is rewarded for growth in portfolio value and penalized for invalid transactions (e.g., selling without owning). The LSTM model is used to bias decisions if a significant price increase is forecasted.

3. Tools and Libraries

This project is implemented using:
- Python 3.10.7
- PyTorch for deep learning (LSTM, DQN)
- Gym for environment simulation
- Pandas and NumPy for data manipulation
- scikit-learn for preprocessing
- Matplotlib for visualization

4. Results

The RealEstateDL model was trained over 1,000 episodes, combining deep Q-learning with LSTM-based price forecasting. Performance metrics such as portfolio value, reward, and ROI (Return on Investment) were tracked every 100 episodes to evaluate learning progress.

LSTM Forecasting Results

The LSTM model was trained over **50 epochs** to predict future property prices using historical economic indicators. The model's average prediction error (RMSE) showed a consistent downward trend, demonstrating effective learning:

- Epoch 1 started with an error of $112,241
- By Epoch 25, the error decreased to $92,242
- The lowest prediction error was achieved around Epoch 46–47, with a minimum of $27,883
- A slight increase occurred toward the final epochs, stabilizing near $32,059 at Epoch 50

This decline in error shows the LSTM's increasing accuracy in forecasting real estate prices over time, which was used to bias the DQN agent toward favorable actions (i.e., buying when price was predicted to rise).

Portfolio Performance

The DQN agent's ability to grow its portfolio value significantly improved as training progressed. Starting from an initial balance of $1,000,000, the agent achieved the following portfolio values:

- Episode 100: $2,784,609 (ROI: 178.46%)
- Episode 300: $3,092,154 (ROI: 209.22%)
- Episode 600: $3,108,862 (ROI: 210.89%)
- Episode 1000: $3,102,002 (ROI: 210.20%)

This demonstrates that the agent not only preserved capital but **nearly tripled** its starting investment through intelligent decision-making.

Reward and Action Distribution

The agent's total reward per evaluation period also steadily increased, peaking above $35,000 in multiple intervals:

- Highest reward observed: $35,147 at Episode 900

- Lowest reward in 100-episode checkpoints: $28,894 at Episode 500

Action breakdowns indicated learning-driven policy changes:

- **Buys** became more frequent as the agent recognized opportunities
- **Sells** decreased, likely due to fewer invalid sell attempts (penalized earlier)
- **Holds** fluctuated moderately but showed a stable trend by late episodes

These results indicate successful training convergence, where the agent consistently executed buy-heavy, growth-oriented strategies based on learned market signals. Figures below show the growth of the agent's portfolio value and return on investment.

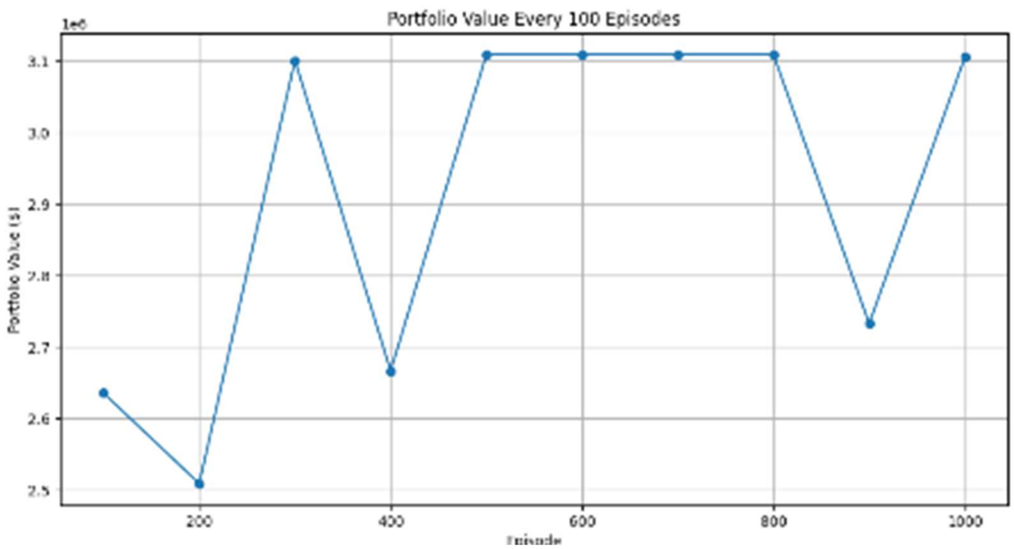

Figure 1: Portfolio Value Every 100 Episodes

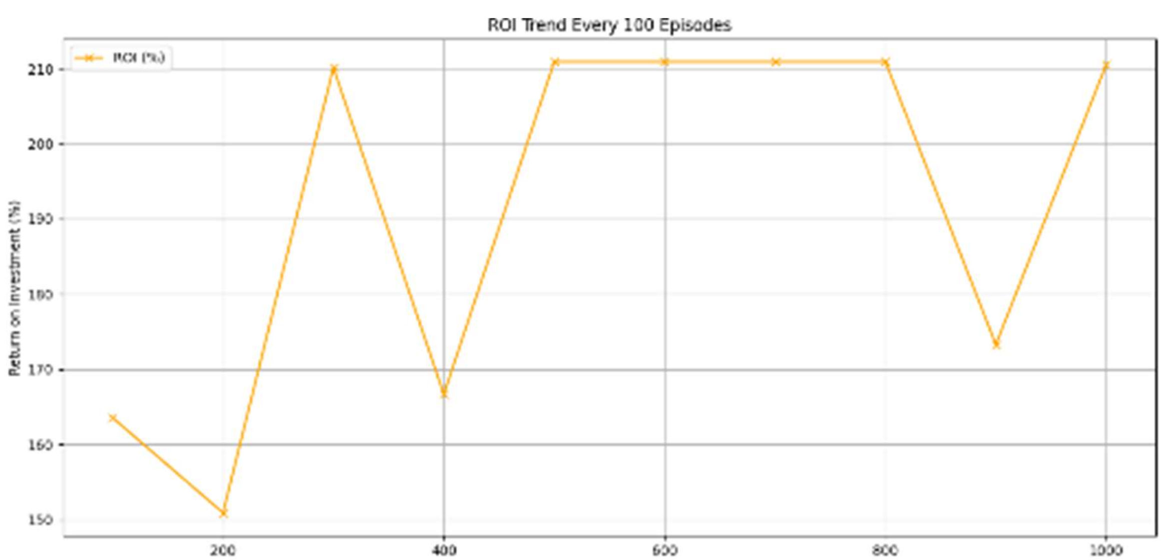

Figure 2: ROI Trend Every 100 Episodes

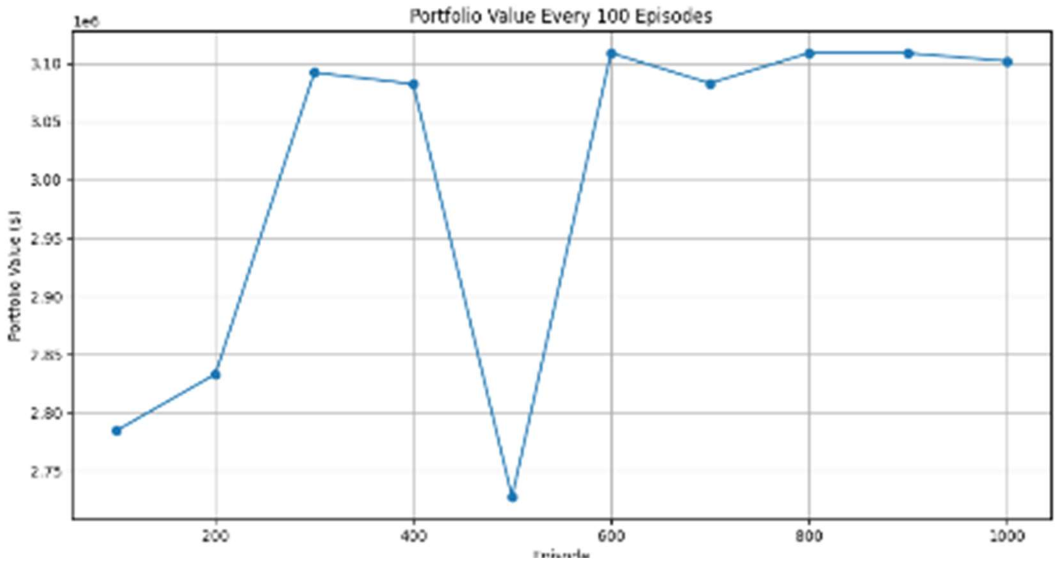

Figure 3: Example Action Distribution

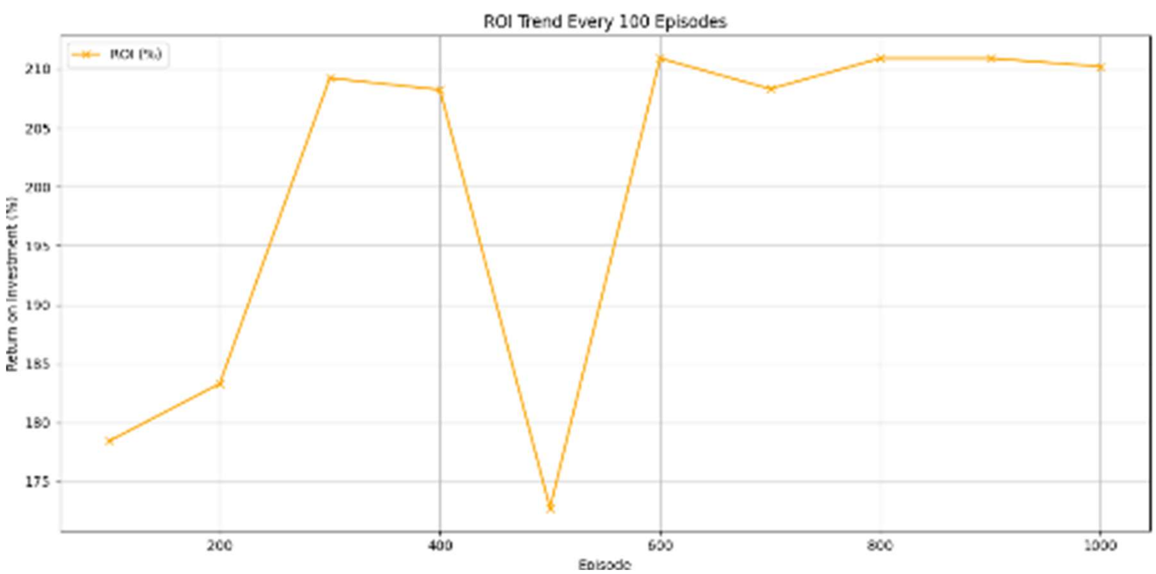

Figure 4: Reward vs Episode Progression

5. Lessons Learned and Limitations

One key challenge was tuning the reward structure to balance exploration and profitability. Another was the market's complexity—real estate is affected by numerous externalities like policy, global economy, and local supply-demand gaps. Using a live or current real estate API would allow for higher fidelity modeling but presents technical and legal limitations.

Also, training performance depends significantly on data normalization, reward shaping, and replay buffer dynamics. Further improvement could be achieved by integrating rolling forecasting and adding transaction cost modeling.

6. Individual Contributions

This project was completed individually. Tasks included data preprocessing, environment creation, LSTM model training, DQN implementation, result visualization, and documentation.

7. Conclusion

This project demonstrates the viability of combining deep learning with reinforcement learning to model a real estate investment strategy. By training a DQN agent within a custom environment informed by LSTM-based price predictions, the system was able to simulate intelligent buy, hold, and sell decisions based on historical data. The agent showed an increasing portfolio value and ROI across episodes, validating the learning framework. While real-world deployment would require more nuanced data and regulatory considerations, this project lays a strong foundation for an AI-driven real estate decision-support tool. The results highlight the importance of accurate forecasting, well-structured reward functions, and continual learning in adapting to a dynamic market environment..