Project Report

Artificial Intelligence and Machine Learning

Mradul Sonkar | Vijaya Raghavendra
Instructor: Pushpak Bhattacharyya

Idea:

We aim to develop a **real-time stock market price predictor** using an online learning approach. The core goal is to build a model that, given the past few intervals of stock data, can **predict the closing price of the next interval** and continuously **update itself with new data** as it arrives.

Traditional time series models rely on batch learning and require retraining when new data is available. In contrast, our approach leverages **incremental learning** using the `deep_river` library, which allows the model to **learn and predict in a streaming fashion**, making it suitable for dynamic financial environments.

We use:

- A custom **LSTM model built in PyTorch**, trained to learn temporal dependencies in stock price movement.
- The LSTM is wrapped inside **`deep_river`'s `RollingRegressor`**, enabling online training.
- The model takes input features such as `Open`, `Close`, `High`, `Low`, and `Volume`, along with their lagged values (e.g., Open_{t-1}, Volume_{t-2}).
- For each time step, it predicts the next closing price and updates its weights using the true observed price.

The stock data is fetched dynamically using the `yfinance` API, allowing the model to operate on real-time or historical data streams.

Structure:

The source code of the project can be found **here.** Follow the following instructions to setup the project.

```
.
├── main.ipynb
├── main.py
├── README.md
├── requirements.txt
├── data/
│   ├── raw/
│   └── processed/
└── src/
    ├── dataHandler.py
    ├── lstmModule.py
    ├── model.py
    ├── processing.py
    └── tickers.json
```

- `main.ipynb`: Jupyter notebook version of main.py.
- `main.py`: Starting point of the project.
- `README.md`: Instructions to use the project.
- `src/`: Source folder containing all files.
- `tickers.json`: JSON file containing all the tickers for which we are fetching the data.

Usage:

- Start by creating a virtual environment, name it **project**, and activate it.

```
python3 -m venv project
source project/bin/activate
pip install -r requirements.txt
```

- Create the folders to fetch the data in, name it **data**.

```
mkdir data
mkdir data/raw
mkdir data/processed
```

- Run the file **main.py** or optionally **main.ipynb** for more analysis.

```
python main.py
```

Current Progress and Future PoA

So far, we have developed a model to predict future stock prices using an LSTM-based architecture wrapped in a rolling regressor. The model takes historical stock data as input and outputs a predicted price for the upcoming interval.

Future improvements include:

- Fetching current financial news from online sources and analyzing its impact on price movements.
- Implementing functionality to fetch real-time stock prices at regular intervals (e.g., every minute).
- Developing a simple user interface to visualize predicted and actual price movements on graphs. This interface could be built using Flask or a similar lightweight web framework.