Predicting US Recessions with Machine Learning Models

Clayton Brossia
Department of Computer and Information Science Fordham University
113 W 60th Street,
New York, New York 10023
United States
cbrossia@fordham.edu

Lauren Jachimczyk
Department of Computer and Information Science Fordham University
113 W 60th Street,
New York, New York 10023
United States
ljachimczyk@fordham.edu

Jason Wang
Department of Computer and Information Science Fordham University
113 W 60th Street,
New York, New York 10023
United States
jw20@fordham.ed

***Abstract*— In this paper we investigate the use of deep learning in predicting recessions in the US. We use economic indicators such as GDP and unemployment between 1976 and 2021 to predict recessions by month. Recession prediction is challenging for many reasons, one of which being the complexity of the economy. While traditional machine learning models may only view data points independently, temporal deep learning models such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) are uniquely equipped to handle time-series data and capture trends. Our results demonstrate a >95% accuracy when considering economic indicators alone. The model does not capture data outside of economic indicators, such as politics, global events, etc.**

***Keywords*—Machine Learning, Deep Learning, recession prediction, financial modeling, economic downturn forecasting, time-series analysis, Recurrent Neural Networks, Long Short-Term Memory**

I. Introduction

Background

It is important for the financial industry to be able to predict recessions, because recessions bring major drops in economic activity. This can disrupt financial markets and negatively impact everyday life for people. Governments, companies, and investors can utilize predictions from a recession model to help make better decisions that can minimize the ensuing damages associated with a recession. For example, governments can use these predictions to influence policy decisions that will help stabilize the economy. Businesses can use predictions to preempt change their hiring practices, operation scaling, and modify their allocation of capital. Investors can use these predictions to influence their investment strategy and reallocate investments to minimize risk.

Accurately predicting a recession is easier said than done. Prediction is a very complex task because economic data has many dimensions and the relationships between variables are not straightforward. Also, there is a good amount of noise and uncertainty that is inherent within economic data. Because of these reasons, traditional models have struggled with accurately learning to predict recessions.

Machine learning is changing the field by enabling a more advanced way to model economic data. Machine learning models have the ability to handle large and complex datasets, and capture the non-linear patterns hidden within them. For example, Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) models are great for working with time-series data, which is the format that economic data comes in. These models can track sequences over time and remember prior information, so when dealing with data such as historical economic trends, they are more accurately able to predict future recessions.

Objectives

Our project aims to create a working machine learning model that learns to predict US recessions. We will use a wide range of economic indicators to train the model, so that the resulting product can be robust and accurate. Our specific goals are listed below:

1. We aim to create a comprehensive recession dataset by combining multiple economic indicators together. We will standardize the time ranges and frequency of which these economic indicators are being displayed in, so our resulting model can use this comprehensive dataset
2. We aim to perform exploratory feature analysis to identify which specific economic indicators are more associated with recessions. This analysis is important because it can impact the decision making of financial institutions.
3. We aim to build multiple models and compare the performances between them. Specifically, we will

look at RNNs and LSTMs, and see how their performances differ in their ability to predict recessions. We will evaluate these models using accuracy, precision, recall, and confusion matrices.

II. Dataset

Datasets Overview

In this project, the majority of our data came from a Kaggle dataset called "Financial Indicators of US Recession". We used this Kaggle source because it provides a wealth of historic US economic data that can be used as indicators of a recession. From this source, we were able to download 26 csv files that contained separate economic data. Table 2 in the appendix shows an in-depth description of each individual csv file used in our dataset and its metadata (i.e. minimum and maximum dates).. Each file contains economic data and metrics such as GDP, consumer prices indices, and the federal funds rate. In addition, each of these files has a separate date range for the data, and a separate frequency that the data is being collected. For example, the unemployment level file tracks monthly unemployment data as far back as 1947. On the other hand, the Commercial Real Estate Prices file tracks quarterly real estate prices starting in 2005. To combine these unique datasets into our final combined table, we had to standardize the date range and frequency. This process is elaborate more in depth in the data preparation section below.

To supplement these economic indicators, we used a second dataset called "USREC.csv" that was our source for recession labels. We needed to use a second data source for this because "Financial Indicators of US Recession" did not provide a recession label for each month. "USREC.csv" contains recession label data tracked by NBER (The National Bureau of Economic Research). It has 2,040 rows of data, where each row represents a month ranging from December 1854 to November 2024. The binary value USREC indicates if the US was in a recession for a given month, with a value of 1 indicating that the country was in a recession.

Data Preparation

In order to build a model, we had to combine the valuable economic data from the 26 csv files, and our data label, into a single data frame. The first step we did was to standardize the timeframe frequencies of our data sources. Table 1 shows the distribution of frequencies that we experienced, which ranged from daily records to yearly records.

Table 1: Dataset Time Frequencies

Timeframe	Occurrences
Daily	3
Weekly	7
Monthly	11
Quarterly	5
Yearly	1

Based on these results, we decided to convert all of our data sets into monthly timeframes. Two key reasons drove this decision. 1) The largest subset of our datasets (11) are already in the monthly format and 2) our label variable is in the monthly format. Logistically, this is a good decision because it ensured that a good proportion of our data did not need to be modified. Also, a monthly time frame was the medium option of our choices. Translating a yearly value into a monthly value, or a daily value into a monthly value, will not modify the data as drastically if we chose a more extreme option. Converting daily data into a yearly value, or vice versa, poses the risk of reshaping the data to such an extent that it would not be reflective of the real world anymore.

This choice meant that we had to convert the following to a monthly time frame: 3 daily datasets, 7 weekly datasets, 5 quarterly datasets, and 1 monthly datasets. Here is logic we used to convert each format:

- **Daily Data:** Converted to monthly by taking the last value of each month. This represents an end-of-month snapshot, which is common practice in financial reporting.
- **Weekly Data:** Converted to monthly by using the last weekly value in each month. Similar reasoning as daily
- **Quarterly:** Converted to monthly by forward-filling each value to apply it across the previous three months. This means quarterly data is available on a rolling basis.
- **Yearly Data:** Forward-fill each yearly value across all months within the year.

When we combined all of the data into a singular table, each row represented a month, and each column an economic metric. The next issue we had to address was the difference in date ranges between the original datasets. There is a high frequency of NaN values at the start because most of our datasets do not start until the later 1900s. To build a strong model, we wanted to balance a wide range of dates to use, while minimizing the amount of NaN values at that time. As the image below shows, there is a large drop in NaN values in the 1980s. We opted to use January 1, 1983 as the beginning date of our final dataset.

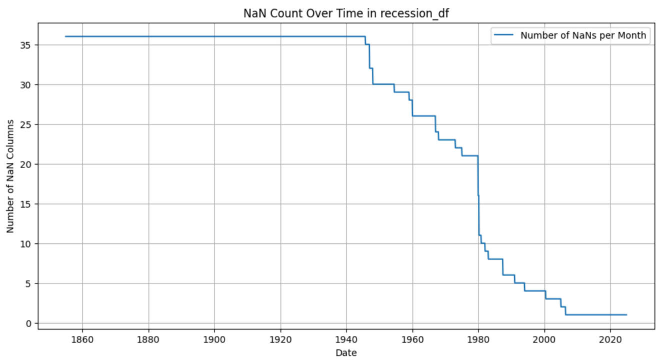

Figure 1

Now that we verified that we have a complete dataset with no null values, all that was left was some final preprocessing tuneups. After examining the data, some of the columns were being incorrectly stored as string data types instead of numeric values. We verified that these were switched to the float data types so the model could interpret them properly. We also had some columns with repeat names. For example, the NASDAQ and S&P 500 tables both had columns called Price, Open, High, Low and % Change. We changed these names to distinguish which came from which dataset.

Our dataset at this point consists of 456 rows, and 30 columns, where each row represents values for various financial indicators in a given month. This ranges from January 1983 to December 2020.

Exploratory Data Analysis (EDA)

After cleaning and filtering the dataset by date, we analyzed the distribution of each feature using a distplot which is similar to a histogram. Using this method allows us to review each feature for skewed data and anomalies. We checked each feature and found that none of them demonstrated very skewed data or significant anomalies.

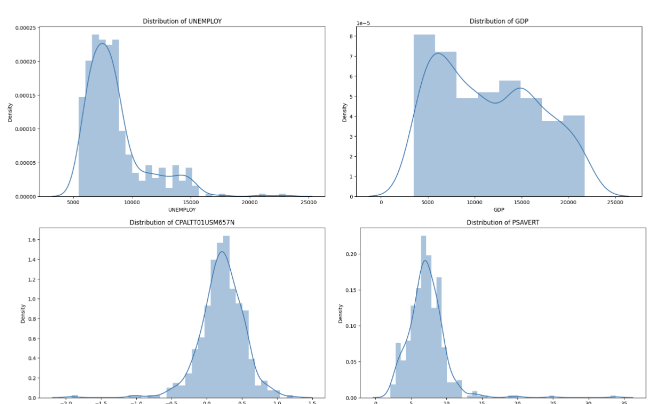

Figure 2

Some of them were normally distributed, however most of them were not, which is what we would expect with real-world data. Figure 2 shows the distribution of the unemployment rate, GDP, CPI (All Items) and Personal Savings Rate.

We also checked the distribution of our target variable, USREC, which was quite unbalanced.

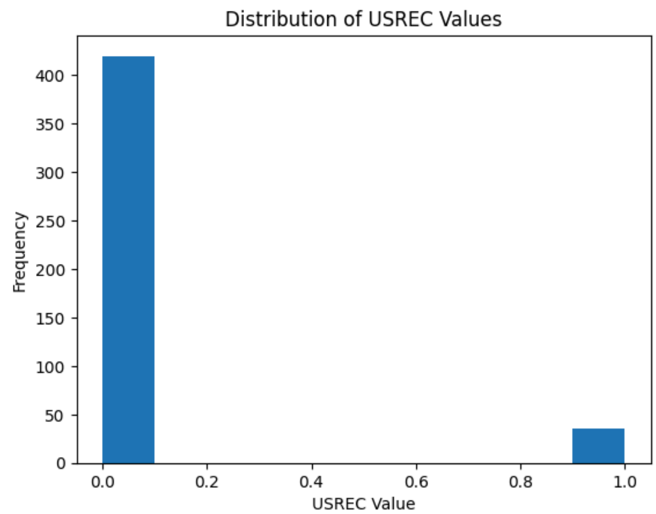

Figure 3

92.1% (n=420) of the values indicate times when the economy was not in a recession, and the remaining 7.9% (n=36) of rows are for times when the economy was in a recession. Our dataset is heavily unbalanced due to the sparsity of recessions. The economy is generally not in a recession so there are significantly more months that have a 0 label than there are months with a 1 label. By only going back to 1983, we are only capturing 4 recessions, as seen in Figure 4. This is a problem we tackle and will discuss further when we display our results.

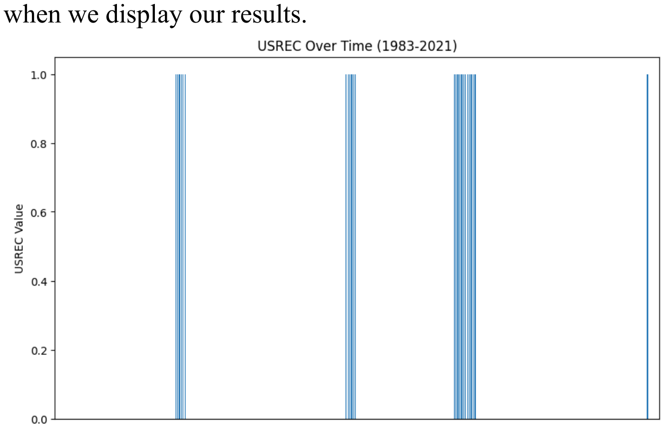

Figure 4

III. Methodology

Feature Selection

Checking the distributions of each feature did not alert us of any features that needed to be removed, however examining correlations did. We used a correlation matrix heat map to check for highly correlated features. Removing highly correlated features is important because it does not contribute anything to help the model. In fact, it just causes more noise. Table 3 in the appendix shows the correlation heat map. It is clear that the S&P 500 and the NASDAQ are very highly correlated. This makes sense because they are both stock indices meant to represent the market. We chose to prioritize the S&P 500 over the NASDAQ since it is more broad, covering all sectors while the NASDAQ is more focused on technology and growth stocks[1]. Additionally, it's clear that the other S&P indicators (Open price, high, low, etc.) are highly correlated so they will not add any value. These features are also removed. Table 4 in the appendix shows the final set of features to be run through the model.

Model Selection

As discussed, we selected RNN and LSTM as our models due to their ability to handle sequential and temporal data. Our dataset is monthly economic indicators over a period of time. A standard machine learning model might treat each data point independently, however that would not be helpful in the case of predicting recessions.

Implementation

To prepare the final dataframe for the RNN and LSTM models, we normalized the data, split the dataset into training and testing data, and then created sequences. We used sklearn's MinMaxScaler to scale the features to a common range (0 to 1). Each feature has vastly different ranges (i.e. GDP has a mean of 11440.98 while sticky price CPI has a mean of 0.248775) so the RNN and LSTM model will perform better after normalization.

Next we split the data into testing and training datasets. 80% of the data is for training and 20% for testing.

We then added a function to create sequences based on the sequence_length hyperparameter variable. It loops through the data to create overlapping sequences of the defined sequence length. The function is applied separately to training and testing sets and then .unsqueeze() is used to reshape the data to be compatible with the format Pytorch expects.

The model is built using PyTorch. We created a class for RNN and a class for LSTM, both using PyTorch's nn module. Both model's parameters are input_size, hidden_size, num_layers and batch_first=True. Input_size is determined by the number input features, hidden_size is set to 64, number of layers is set to 2 and output size is set to 1 as it is a binary classification.

The RNN model contains a layer that maps the output of the RNN's hidden state, then a forward pass which outputs a sequence of hidden states and a final hidden state. The output of the last hidden state is the input to the fully connected layer. For the training process, Binary cross-entropy (BCEWithLogitsLoss) is used as the loss function since this is a binary classification. The Adam optimizer is used and the number of epochs is set to 100.

The LSTM model is a more advanced form of RNN that handles long-term dependencies by using gates (input, forget and output), as it was developed to handle vanishing gradient problems with long sequences. The model contains a fully connected layer to map the last hidden state to the target output. The forward pass involves initializing the hidden state and cell state to zeros, forward propagating the input through the LSTM layer and taking the last hidden state as the input to the fully connected layer.

A key difference between the models is their architecture - the RNN relies only on the hidden states, while the LSTM brings in cell states and gating mechanisms. Their training process is the same, as they both use the same loss function and optimizer.

IV. Results

Model Performance Part 1

Our first attempt at the model did not output very accurate results.

Table 5: Model Part 1 Results

Metric	RNN	LSTM
Accuracy	0.4000	0.4125
Precision	0.0400	0.0408
Recall	1.0000	1.0000
F1	0.0769	0.0784

We noticed that the poor accuracy may be due to the dataset being heavily skewed toward class 0 (non-recession). We attempted the model again with class weighting. We took the weight of each class (total samples/samples in class) and directly applied it to the Binary Cross-Entropy (BCE) loss function. Unfortunately, this did not make any improvements in our model's accuracy.

Model Performance Part 2

Based on the initial performance of the RNN and LSTM model's, we were concerned about the imbalance of the dataset (only ~8% of the data points were recessions). We implemented several new strategies in an attempt to achieve higher accuracy, which were expanding the dataset, adding new features through feature engineering and finally, hyperparameter tuning.

Expanding the dataset

We extended our dataset back to 1976 so we could capture more data. Ideally, we would like to go back even further but we are limited by availability of data going back too far so we decided on 1976. By extending back from 1983 to 1976, we were able to capture two more recessions periods in our dataset. In our previous dataset going back to 1983, we capture 4 recessions. By extending back 7 more years to 1976, we are able to capture a total of 6 periods of recessions, as seen in Figure 5.

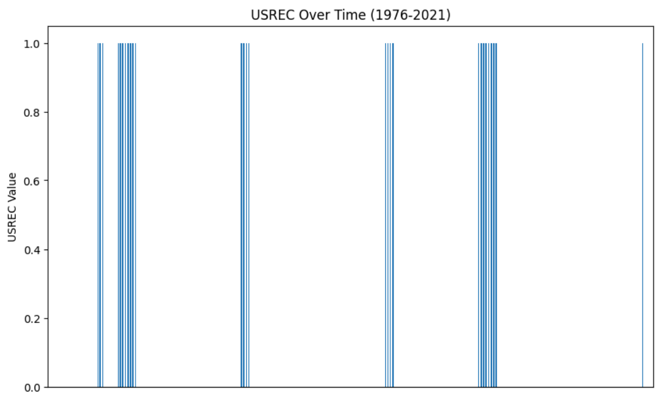

Figure 5

New Features

Additionally, we explored feature engineering. We added a new feature for the inverted yield curve, called is_inverted. The inverted yield curve is an unusual phenomenon that happens when short term interest rates are higher than long term interest rates, often an indicator of economic downturn[2]. We chose to use the 10 year treasury yield against the 2 year treasury yield as a way of determining an inverted yield curve. We took a 10 year treasury yield dataset and a 2 year treasury yield dataset, both from the Federal Reserve Bank of St. Louis's Federal Reserve Economic Data (FRED). Both data points are daily, so we converted them to monthly following the same process as the rest of our datasets that were daily. From the 10 year treasury yield column and the 2 year treasury yield column, we use logic to create a third column. If the 2 year treasury yield is higher than the 10 year treasury yield for a given month, a value of 1 is assigned, otherwise it is 0. The 10 year and 2 year treasury yield columns are dropped and we only keep the binary inverted yield curve column.

We further enhanced our dataset by adding new columns that transformed our current features into rolling averages. This was done over both a 3 month and a 12 month interval for all variables. We calculated rolling averages by taking the mean of that feature over the previous window (ie: previous 3 months). This step enabled our dataset to better capture temporal trends. Table 6 in the appendix shows the final dataset for part 2.

Hyperparameter Tuning

Based on this enhanced dataset, we reattempted the models to see if there is a higher accuracy, while also using hyperparameter tuning. We specifically focused on batch size and sequence length, running each model through several different scenarios to find the right pairing of batch size and sequence length. The pairs tested were [6 (batch size),16 (sequence length)], [6, 32], [6,64], [12, 16], [12,32], [12,64], [18, 16], [18,32], [18,64].

We ran both datasets through the RNN and LSTM models to see how the hyperparameter tuning would impact the results. The results of the models after hyperparameter tuning are as follows. These are the results of one of our most accurate runs with sequence length 18 and batch size 16.

Table 7: Model Part 2 Results

Metric	RNN 1	LSTM 1	RNN 2	LSTM 2
Accuracy	0.95	0.99	0.97	0.97
Precision	1.00	0.92	0.91	0.91
Recall	0.66	1.00	0.83	0.83
F1	0.80	0.96	0.87	0.87

Feature Importance

To assess which features contributed the most to the model's performance in predicting recessions we used permutation. We checked both datasets for both RNN and LSTM, using a batch size of 16 and a sequence length of 18 as our hyperparameters since that was one of our highest performing models. We looped through each feature in each dataset, shuffling one feature at a time to test how the model's performance degrades (higher drop in performance equates to higher importance of the feature). We used F1 score as the main metric, since it is particularly useful for binary classification problems and class imbalances. The results of our analysis are displayed below.

Table 8: Dataset 1 (1983-2021) Features Ranked in Order of Importance

```
RNN Feature Importances (Descending):
STICKCPIM157SFRBATL: 0.6667
REAINTRATREARAT10Y: 0.6667
WM2NS: 0.1961
OEHRENWBSHNO: 0.0784
CCSA: 0.0784
GDPC1: 0.0784
TOTBKCR: 0.0000
DFF: 0.0000
CHANGE_SP500: 0.0000
CORESTICKM159SFRBATL: -0.0702
PSAVERT: -0.0702
MEDCPIM158SFRBCLE: -0.0702
CPALTT01USM657N: -0.0952
GDP: -0.1333
UNRATE: -0.1515
FPCPITOTLZGUSA: -0.1905
UNEMPLOY: -0.2899
PRICE_SP500: -0.2899
CPIAUCSL: -0.2899
WM1NS: -0.2933
```

```
LSTM Feature Importances (Descending):
CPALTT01USM657N: 0.0351
FPCPITOTLZGUSA: 0.0351
CHANGE_SP500: 0.0000
MEDCPIM158SFRBCLE: 0.0000
PSAVERT: -0.0702
STICKCPIM157SFRBATL: -0.0702
GDPC1: -0.1333
CPIAUCSL: -0.1333
REAINTRATREARAT10Y: -0.1481
OEHRENWBSHNO: -0.1905
UNEMPLOY: -0.1905
UNRATE: -0.2029
TOTBKCR: -0.2564
CORESTICKM159SFRBATL: -0.2899
GDP: -0.2899
PRICE_SP500: -0.2899
CCSA: -0.2933
DFF: -0.3333
WM1NS: -0.3333
WM2NS: -0.3333
```

There is no clear pattern amongst RNN and LSTM which features were more important in the first dataset.

Table 9: Dataset 2 (1976-2021 with enhancements) Features Ranked in Order of Importance

```
RNN Feature Importances (Descending):
DFF: 0.3294
is_inverted: 0.2444
OEHRENWBSHNO: 0.1846
GDP: 0.1636
FPCPITOTLZGUSA: 0.1043
CPIAUCSL_Rolling12: 0.1000
CORESTICKM159SFRBATL: 0.0727
UNEMPLOY: 0.0727
WM1NS: 0.0727
STICKCPIM157SFRBATL_Rolling12: 0.0727
CORESTICKM159SFRBATL_Rolling3: 0.0500
FPCPITOTLZGUSA_Rolling3: 0.0500
DFF_Rolling12: 0.0500
WM1NS_Rolling12: 0.0500
UNRATE: 0.0174
CPIAUCSL_Rolling3: 0.0174
STICKCPIM157SFRBATL_Rolling3: 0.0174
DFF_Rolling3: 0.0174
WM1NS_Rolling3: 0.0174
GDPC1_Rolling12: 0.0174
CPALTT01USM657N_Rolling12: 0.0174
CPALTT01USM657N: 0.0000
GDPC1: 0.0000
PSAVERT_Rolling3: 0.0000
STICKCPIM157SFRBATL: -0.0182
CPALTT01USM657N_Rolling3: -0.0182
```

```
LSTM Feature Importances (Descending):
is_inverted: 0.2778
GDPC1_Rolling3: 0.1667
DFF_Rolling12: 0.1377
CCSA: 0.1061
TOTBKCR_Rolling3: 0.0833
FPCPITOTLZGUSA_Rolling12: 0.0833
CORESTICKM159SFRBATL_Rolling12: 0.0714
CPIAUCSL: 0.0507
STICKCPIM157SFRBATL_Rolling3: 0.0507
OEHRENWBSHNO_Rolling3: 0.0507
WM1NS_Rolling3: 0.0507
CCSA_Rolling12: 0.0507
CPIAUCSL_Rolling12: 0.0507
STICKCPIM157SFRBATL_Rolling12: 0.0507
OEHRENWBSHNO_Rolling12: 0.0507
TOTBKCR: 0.0152
DFF: 0.0152
PSAVERT: 0.0152
FPCPITOTLZGUSA: 0.0152
GDPC1: 0.0000
CCSA_Rolling3: 0.0000
GDP_Rolling3: 0.0000
UNEMPLOY_Rolling3: 0.0000
CPALTT01USM657N_Rolling3: 0.0000
TOTBKCR_Rolling12: 0.0000
CPALTT01USM657N_Rolling12: 0.0000
STICKCPIM157SFRBATL: -0.0238
UNRATE: -0.0238
PSAVERT_Rolling3: -0.0238
CORESTICKM159SFRBATL_Rolling3: -0.0238
```

The second dataset makes it more clear that the inverted yield curve feature is one of the most important, as it's the highest importance for LSTM and the second highest for RNN.

V. Discussion

Comparison with Related Work

We examined two studies that attempted a similar task of predicting recessions through machine learning.

The first study we compared is called "Credit growth, the yield curve and financial crisis prediction: Evidence from a machine learning approach" published by the Journal of International Economics. It uses ensemble methods to create early warning models for financial crises, focusing on macroeconomic data from 17 different countries[3]. Their dataset is significantly more broad than ours, as it not only covers 17 countries while ours focuses on the US, it also spans from 1870-2016, while ours only spans from 1976-2021. Additionally, they use ensemble methods, which is a combination of multiple machine learning algorithms. In this study, many models were used, such as Random Forest, Support Vector Machines and Neural Networks. Our project used only RNN and LSTM and each was done independently. The results were not combined or averaged in any way. One of the key differences between this study and ours is the labels. Our project defines a recession in the month that it happens as stated by the NBER. The article from the Journal of International Economics took a unique approach of focusing on the 1-2 years leading up to a financial crisis by offsetting the labels by 1-2 years (the labels are annual in this dataset). While our project used recession labels in the month they occurred, it is possible that our project could have captured some of this due to the nature of RNN/LSTM if the sequence lengths are long enough.

The second article is called "Yield curve and Recession Forecasting in a Machine Learning Framework" from the Department of Economics at Democritus University of Thrace in Greece. The study focuses on recession forecasting through SVM, with a focus on the yield curve and a dataset between 1976 and 2011[4]. It differs from our project as it uses SVM as the model, however it focuses on a similar dataset, which is US economic indicators during a similar time period as ours.

Both papers focus on the yield curve inversion as a key indicator of recessions, which is something we incorporated into our model as well. Some major difference between our project and the articles is that our labels were monthly, while theirs were either quarterly or annually. Additionally, both studies incorporated SVM into their work while we did not.

Challenges

We encountered many challenges while working on this project. In the preprocessing step, we overcame challenges of combining many different data sources with different timeframes and frequencies. We were also faced with the issue of data availability. The further back in time we went, less indicators had full sets of data. During the Exploratory Data Analysis phase, we noticed the heavy imbalance of the dataset, which had an impact during our Model Implementation phase. The models were not producing very accurate results. We implemented some techniques such as expanding the dataset, adding new features, feature engineering and finally hyperparameter tuning. Hyperparameter tuning ended up being a key part in overcoming the challenge of low accuracy.

Limitations

As discussed, two of our major challenges were data availability and model accuracy. The project can be improved in the future by spending even more time on the preprocessing step. Finding stronger datasets that extend further back in time, as well as adding additional economic indicators. The US economy is extremely complex and has a long history. Going back to only 1976 doesn't capture this rich history. Additionally, there's many other economic indicators that measure the economy so only using a small set of them may not capture the full picture.

The other challenge we faced was with model accuracy. After hyperparameter tuning, our model's accuracy significantly improved, however due to the unbalanced dataset and complexity of the economy it is possible there may be overfitting. Our model works well on our dataset, however a future enhancement would be to examine if some overfitting is happening.

VI. Conclusion

This project aimed to experiment with RNN and LSTM models to accurately predict US recessions. By creating a dataset that utilized various economic metrics from 1976 to 2021, we created a model that was remarkably accurate at doing so. This was valuable, because it enabled us to analyze the importance of our economic indicators. As a result, we can see the inverted yield curve is one of the most important features in enabling the model to understand the patterns that make up recession prediction.

This has many practical implications going forward since an accurate recession prediction model can be a vital tool in the economic industry. Investors, businesses, and policy makers can not only proactively make decisions to mitigate the negative impact of potential future recessions, but can work to make sure important indicators like the inverted yield curve phenomenon continue to be used as an alert of potential signs of economic downturn.

While these results are promising, our work highlighted several limitations that future work can be addressed. Data availability was a large issue in our project since there is a limited amount of data for some of our features that only became widely available in recent years. This impacted our ability to extend the dataset back in time, since there were not many available metrics for most early on recession periods. This absence of data meant we either would be building a model that used limited features, and likely would not be accurate, or a model with a limited amount of years, that might struggle to find a pattern. Going forward, future work should look to see if we can retrospectively calculate data for some of these economic metrics, to expand the time range longer. We could also work to include new metrics such as global economic indicators to help make the model more robust. Future work may also extend outside of economic indicators, as major unforeseen events such as election outcomes and pandemics play a role in the economy. A stronger recession predictor model will help the economic industry make valuable decisions, which can help protect the livelihood of everyone during economic downturns.

VII. References

[1] Thompson, C. (2024). Nasdaq Composite vs. S&P 500 vs. Dow: What's the Difference? Investopedia.

[2] Kovalski, M., Wessel, D. (2018). The Hutchins Center Explains: The yield curve – what it is, and why it matters. Brookings.

[3] Bluwstein, K. Buckmann, M., Joseph, A., Kapadia, S., Simsek, O. (2023). Credit growth, the yield curve and financial crisis prediction: Evidence from a machine learning approach. Journal of International Economics.

[4] Papadimitriou, T., Gogas, P., Matthaiou, M., Chrysanthidou, E. Yield curve and Recession Forecasting in a Machine Learning Framework. Department of Economics Democritus University of Thrace.

VIII. Appendix

Table 2: Economic Indicator CSV files

File Name	Description	Timeframe	Dates
NASDAQ	Daily NASDAQ stock market data, including price, volume, and percentage change.	Daily	1980-03-18 to 2023-05-03
Unemployment Level	Monthly U.S. unemployment levels.	Monthly	1948-01-01 to 2023-03-01
Total Unemployed Plus All Persons Marginally Attached	Monthly U6 unemployment rate.	Monthly	1994-01-01 to 2023-03-01
Federal Funds Effective Rate	Daily federal funds effective interest rate.	Daily	1954-07-01 to 2023-05-02
Real Estate Loans Commercial Real Estate Loans	Weekly data on commercial real estate loans.	Weekly	1987-06-17 to 2023-04-19
Unemployment Rate	Monthly U.S. unemployment rate.	Monthly	1948-01-01 to 2023-04-01
Consumer Price Index Total All Items	Monthly U.S. CPI for all items.	Monthly	1960-01-01 to 2023-02-01
SPX500	Daily S&P 500 stock market data, including price, volume, and percentage change.	Daily	1979-12-26 to 2023-05-03

Personal Saving Rate	Monthly U.S. personal saving rate.	Monthly	1959-01-01 to 2023-03-01
Consumer Loans Credit Cards and Other Revolving Plans	Weekly data on consumer loans and credit cards.	Weekly	2000-06-28 to 2023-04-19
Bank Credit All Commercial Banks	Weekly data on total bank credit for all U.S. commercial banks.	Weekly	1973-01-03 to 2023-04-19
Continued Claims (Insured Unemployment)	Weekly continued unemployment claims data.	Weekly	1967-01-07 to 2023-04-15
Households Owners Equity in Real Estate Level	Quarterly data on household equity in real estate.	Quarterly	1945-10-01 to 2022-10-01
M1	Weekly data on M1 money supply.	Weekly	1975-01-06 to 2023-04-03
Sticky Price Consumer Price Index Less Food and Energy	Monthly sticky-price CPI excluding food and energy.	Monthly	1967-12-01 to 2023-03-01
Consumer Price Index for All Urban Consumers	Monthly CPI data for all urban consumers.	Monthly	1947-01-01 to 2023-03-01
Inflation Consumer Prices	Yearly consumer price inflation.	Yearly	1960-01-01 to 2021-01-01
Commercial Real Estate Prices	Quarterly commercial real estate price data.	Quarterly	2005-01-01 to 2022-04-01
M2	Weekly data on M2 money supply.	Weekly	1980-11-03 to 2023-04-03
10-Year Real Interest Rate	Monthly real interest rate on 10-year treasury bonds.	Monthly	1982-01-01 to 2023-04-01
Gross Domestic Product	Quarterly U.S. nominal GDP data.	Quarterly	1947-01-01 to 2023-01-01
Real Estate Loans Residential	Weekly data on residential real estate loans.	Weekly	1987-06-17 to 2023-04-19
Median Consumer Price Index	Monthly median CPI data.	Monthly	1983-01-01 to 2023-03-01
Real Gross Domestic Product	Quarterly real (inflation-adjusted) GDP data.	Quarterly	1947-01-01 to 2023-01-01
Sticky Price Consumer Price Index	Monthly sticky-price CPI for all items.	Monthly	1967-01-01 to 2023-03-01
Delinquency Rate on Credit Card Loans	Quarterly delinquency rates on credit card loans.	Quarterly	1991-01-01 to 2022-10-01

Table 3: Correlation Heatmap

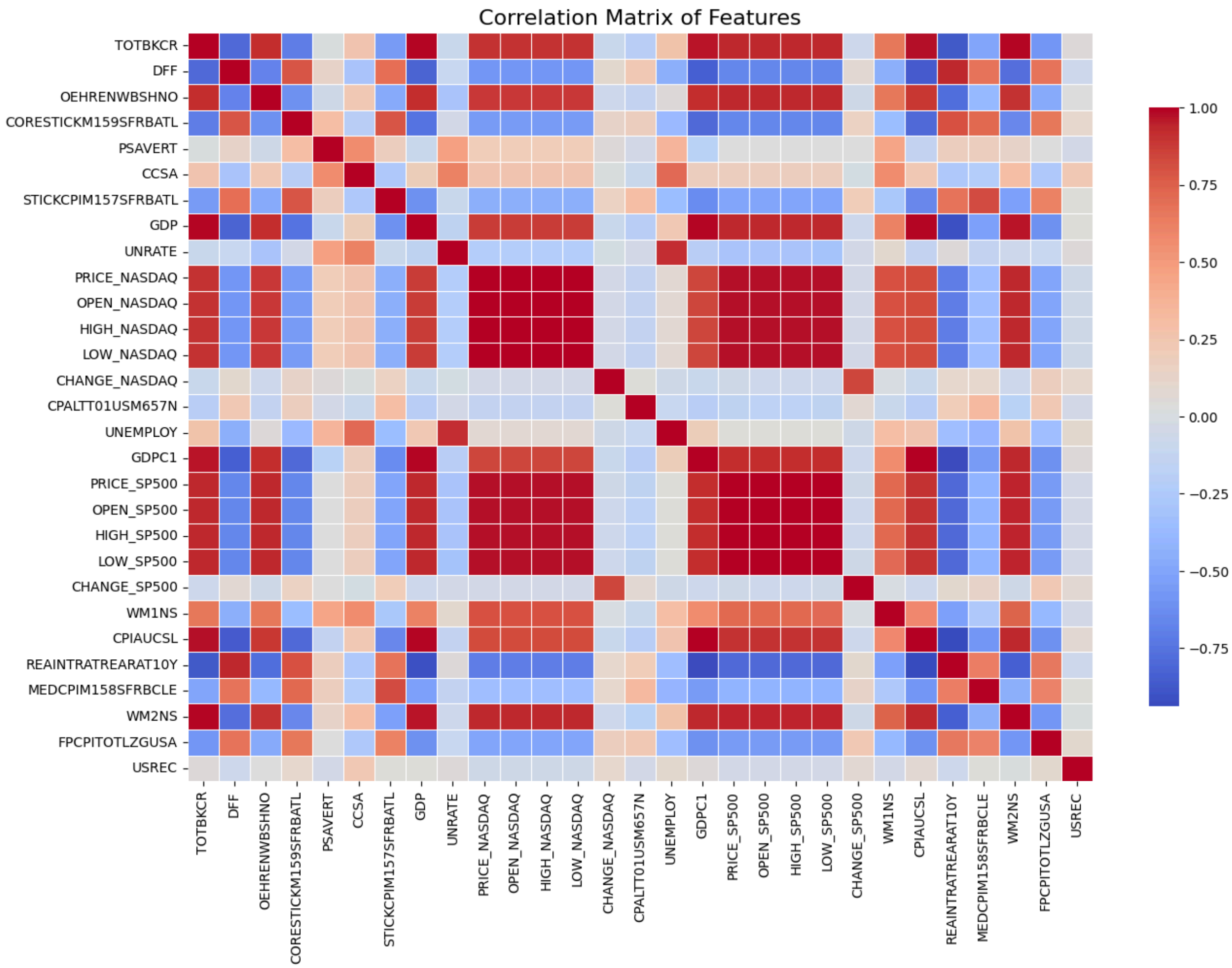

Table 4: Final Dataset Features - Part 1

Feature	Description
DATE	The date corresponding to the recorded economic indicator values (monthly)
TOTBKCR	Total bank credit extended by all U.S. commercial banks (weekly data).
DFF	Daily federal funds effective interest rate.
OEHRENWBSHNO	Household owner equity in real estate levels (quarterly data).
CORESTICKM159SFRBATL	Sticky-price Consumer Price Index (CPI) excluding food and energy.
PSAVERT	U.S. personal saving rate as a percentage of disposable income.
CCSA	Continued claims for insured unemployment (weekly data).
STICKCPIM157SFRBATL	Sticky-price Consumer Price Index (CPI) for all items
GDP	U.S. nominal Gross Domestic Product (quarterly data).
UNRATE	U.S. unemployment rate (percentage of the labor force unemployed).

CPALTT01USM657N	Consumer Price Index (CPI) for all items, measuring inflation on a monthly basis.
UNEMPLOY	Total unemployment levels in the U.S. (number of people).
GDPC1	Real (inflation-adjusted) Gross Domestic Product (quarterly data).
PRICE_SP500	S&P 500 closing price on the given date.
CHANGE %_SP500	Percentage change in S&P 500 price from the previous day.
WM1NS	M1 money supply (weekly data).
CPIAUCSL	Consumer Price Index (CPI) for all urban consumers.
REAINTRATREARAT10Y	Real interest rate for 10-year treasury bonds (monthly data).
MEDCPIM158SFRBCLE	Median Consumer Price Index (CPI) value.
WM2NS	M2 money supply (weekly data).
FPCPITOTLZGUSA	Yearly consumer price inflation in the U.S.
USREC	Binary indicator for U.S. recessions (1 = recession, 0 = no recession).

Table 6: Final Dataset Features - Part 2

Feature	Description
DATE	The date corresponding to the recorded economic indicator values.
USREC	Binary indicator for U.S. recessions (1 = recession, 0 = no recession).
TOTBKCR	Total bank credit extended by all U.S. commercial banks (weekly data).
DFF	Daily federal funds effective interest rate.
OEHRENWBSHNO	Household owner equity in real estate levels (quarterly data).
CORESTICKM159SFRBATL	Sticky-price Consumer Price Index (CPI) excluding food and energy.
PSAVERT	U.S. personal saving rate as a percentage of disposable income.
CCSA	Continued claims for insured unemployment (weekly data).
STICKCPIM157SFRBATL	Sticky-price Consumer Price Index (CPI) for all items
GDP	U.S. nominal Gross Domestic Product (quarterly data).
UNRATE	U.S. unemployment rate (percentage of the labor force unemployed).
CPALTT01USM657N	Consumer Price Index (CPI) for all items, measuring inflation on a monthly basis.
UNEMPLOY	Total unemployment levels in the U.S. (number of people).
GDPC1	Real (inflation-adjusted) Gross Domestic Product (quarterly data).
WM1NS	M1 money supply (weekly data).
CPIAUCSL	Consumer Price Index (CPI) for all urban consumers.
FPCPITOTLZGUSA	Yearly consumer price inflation in the U.S.

is_inverted	Binary indicator for yield curve inversion (1 = inverted, 0 = not inverted)
CCSA_Rolling3	CCSA 3 month rolling average
CPIAUCSL_Rolling3	CPI (urban consumers) 3 month rolling average
GDP_Rolling3	GDP 3 month rolling average
STICKCPIM157SFRBATL_Rolling3	Sticky CPI all items 3 month rolling average
PSAVERT_Rolling3	Personal savings rate 3 month rolling average
DFF_Rolling3	Daily fed funds rate 3 month rolling average
TOTBKCR_Rolling3	Total bank credit 3 month rolling average
CORESTICKM159SFRBATL_Rolling3	Sticky CPI excl food/energy 3 month rolling average
GDPC1_Rolling3	Real GDP 3 month rolling average
UNEMPLOY_Rolling3	Unemployment level 3 month rolling average
OEHRENWBSHNO_Rolling3	Household owner equity 3 month rolling average
CPALTT01USM657N_Rolling3	CPI all items 3 month rolling average
WM1NS_Rolling3	M1 money supply 3 month rolling average
UNRATE_Rolling3	Unemployment rate 3 month rolling average
FPCPITOTLZGUSA_Rolling3	Consumer price inflation 3 month rolling average
CCSA_Rolling12	CCSA 12 month rolling average
CPIAUCSL_Rolling12	CPI (urban consumers) 12 month rolling average
GDP_Rolling12	GDP 12 month rolling average
STICKCPIM157SFRBATL_Rolling12	Sticky CPI all items 12 month rolling average
PSAVERT_Rolling12	Personal savings rate 12 month rolling average
DFF_Rolling12	Daily fed funds rate 12 month rolling average
TOTBKCR_Rolling12	Total bank credit 12 month rolling average
CORESTICKM159SFRBATL_Rolling12	Sticky CPI excl food/energy 12 month rolling average
GDPC1_Rolling12	Real GDP 12 month rolling average
UNEMPLOY_Rolling12	Unemployment level 12 month rolling average
OEHRENWBSHNO_Rolling12	Household owner equity 12 month rolling average
CPALTT01USM657N_Rolling12	CPI all items 12 month rolling average
WM1NS_Rolling12	M1 money supply 12 month rolling average
UNRATE_Rolling12	Unemployment rate 12 month rolling average
FPCPITOTLZGUSA_Rolling12	Consumer price inflation 12 month rolling average