

# **Deep Learning Approaches for Multi-Horizon Market Cap Growth Prediction**

Fidelfolio Investments



**Team Bitmask**



# Introduction

Fidelfolio Investments initiated this project to quantify relationships between fundamental financial indicators and future market performance using deep learning techniques. The goal was to develop a model that could predict market capitalization growth across multiple time horizons (1-year, 2-year, and 3-year) to support investment research and decision-making processes.

The slide features decorative geometric shapes in the corners. The top-right corner has a yellow-to-orange gradient triangle. The bottom-left corner has a purple-to-blue gradient triangle. The bottom-right corner has a red-to-purple gradient triangle.

# Objectives

This project aims to develop and compare different deep learning models (like LSTMs and MLPs) for forecasting multiple financial target variables across a panel of companies over time. It involves robust data preprocessing, including feature engineering and dimensionality reduction, training models using a realistic expanding window approach, and evaluating their predictive accuracy (using RMSE/MAE) to identify effective strategies for this financial forecasting task.

# Dataset overview



## Dataset Summary

- Contains data for 2,796 publicly listed Indian companies
- Time period: 1999 to 2024
- Total of 24,751 company-year observations
- Each observation includes normalized financial features
- Targets: 1-year, 2-year, and 3-year forward market capitalization growth rates



## Feature Characteristics

- Some features showed wide ranges or extreme values
  - Indicates potential outliers or varying inherent scales

Table 2.1: Representative Statistics of Key Features

Statistic	Feature 1	Feature 2	Target 1Y	Target 3Y
Count	20914	21999	22934	19606
Mean	148.094	17.857	15.418	34.888
Std Dev	1985.689	15.149	87.923	168.034
Min	0.01	-109.44	-178.49	-344.27
Median	52.000	14.830	-4.255	-13.685
Max	227428.00	444.80	988.17	995.22
Range	227427.99	554.24	1166.66	1339.49

# Preprocessing Pipeline

## DATA CLEANING & MISSING VALUES

### Data Cleaning

- Standardized column names
- Ensured all financial features were numeric

### Missing Value Imputation

- Forward fill within each company's time series
- KNN imputation ( $k = 5$ ) for remaining missing entries

## OUTLIERS, NORMALIZATION & DIMENSIONALITY

### Outlier Handling

- Applied Winsorization at the 1st and 99th percentiles

### Feature Normalization

- Used z-score normalization across all features

### Dimensionality Reduction

- Applied PCA to retain 95% variance
- Reduced feature set from 56 to 6 principal components

## SEQUENCE STRUCTURING & SPLITTING

### Sequence Preparation (for LSTMs)

- Sorted data chronologically by company-year
- Created fixed-length lookback windows
- Matched each sequence to its future growth targets

### Company Encoding

- Label encoded company IDs
- Passed through embedding layers

### Train-Validation Split

- Used an 15% validation split
- Prevented data leakage by preserving time order

# Model 1: Baseline LSTM

Captures temporal patterns in historical financial features, with static context from company ID embedding.

- Inputs:
  - Sequence Input: (batch\_size, max\_seq\_len, num\_features)
  - Company ID Input: (batch\_size, 1)
- Layers:
  - a. Company Embedding Layer
    - Embedding(input\_dim = num\_companies, output\_dim = 10)
    - Followed by Flatten()
  - b. LSTM Layer
    - LSTM(units = 64 or 128)
    - Returns final hidden state or full sequence
  - c. Concatenation Layer
    - Merges LSTM output with flattened company embedding
  - d. Dense Layer
    - Dense(32, activation='relu')
  - e. Output Layer
    - Dense(3) → Predicts 1Y, 2Y, and 3Y market cap growth

# Model 2: Improved LSTM

Enhances the baseline LSTM by improving training stability and generalization through scaling, regularization, and tuned hyperparameters.

- Target Scaling:
- Targets (1Y, 2Y, 3Y growth) scaled using StandardScaler
- Used PCA for Dimensionality Reduction
- Predictions inverse-transformed after training
- Dropout Regularization:
- Dropout(0.25) applied after LSTM and Dense layers
- Helps prevent overfitting
- Hyperparameter Tuning:
- Lower learning rate (e.g., 0.0001)
- Increased training epochs and patience for early stopping

# Model 3: MLP Model

A non-sequential approach using an MLP architecture, engineered features, and PCA to predict each growth horizon separately.

- Year-over-Year Differences:
  - Computes  $\Delta$  (delta) features from original financial metrics (e.g., FE\_ columns)
  - Captures annual rate of change
- PCA Transformation:
  - Reduces dimensionality while retaining 95% variance
  - Applied on original and/or engineered features
- Inputs:
  - Flattened PCA Sequence  $\rightarrow$  (batch\_size, max\_seq\_len  $\times$  num\_pca\_components)
  - Company ID  $\rightarrow$  Embedded (output\_dim = 16)  $\rightarrow$  Flattened
- Layers:
  - Input Concatenation
  - Dense(128, activation='relu')  $\rightarrow$  Dropout(0.3)
  - Dense(64, activation='relu')  $\rightarrow$  Dropout(0.3)
  - Output Layer: Dense(1)  $\rightarrow$  Predicts 1Y / 2Y / 3Y growth separately

# Model 4: Encoder-Decoder LSTM

Captures sequential dependencies using an encoder LSTM and predicts each horizon using separate decoder networks. Enhances context handling over time.

- Sequence Input:
  - PCA-reduced financial time series: (batch\_size, max\_seq\_len, num\_pca\_components)
  - Masking applied for padded entries
- Company ID Input:
  - Embedded → Flattened for company-specific context
- Encoder LSTM:
  - LSTM(128, return\_state=True)
  - Outputs final hidden state = context vector
- Company Embedding:
  - Embedding layer → Flattened vector
- Decoder Input Preparation:
  - Concatenate context vector with company embedding
- Dense Layer 1:
  - Dense(64, activation='relu') → Dropout(0.25)
- Output Layer:
  - Dense(1) → Predicts 1Y / 2Y / 3Y growth separately

# Comparative Analysis

Model	Prediction Type	1Y (RMSE / MAE)	2Y (RMSE / MAE)	3Y (RMSE / MAE)	Remarks
Baseline LSTM	Combined (1Y, 2Y, 3Y)	129.8627	280.0713	464.2194	Overall RMSE: 321.25 Overall MAE: 110.67
Improved LSTM	All targets simultaneously	126.54 / 57.20	276.19 / 108.42	472.04 / 181.07	Used target scaling & regularization
MLP (PCA + Feat. Diffs)	Separate model per target	1402.89 / 285.06	2674.45 / 583.33	3930.21 / 952.18	Used PCA & engineered features, no time series
Encoder-Decoder LSTM	Separate model per target horizon	126.90 / 58.11	277.80 / 111.45	478.01 / 187.80	15,044 prediction pairs per target




# Conclusions

## Summary of Findings

- Developed deep learning models for predicting 1Y, 2Y, 3Y forward market cap growth for Indian companies (1999-2024)

## Key findings:

- LSTM models outperformed non-recurrent MLP (flattened sequences, feature engineering)
  - Improved LSTM (target scaling, dropout, hyperparameter tuning) resulted in substantial performance gains
  - Encoder-Decoder LSTM showed comparable performance to Improved LSTM, highlighting the utility of LSTM encoder for contextual representation
  - Demonstrates the potential of deep learning for investment decision-making at FidelFolio Investments
- 



**Thank you**

