Project Proposal:
Scalable Machine Learning and Deep Learning

.

Multi-Factor QQQ Prediction Using Financial and News Sentiment Data

Federico Mercurio
Margherita Santarossa

December 2025

1 Problem Description

The goal of this project is to predict the next-day return of the QQQ ETF and determine whether it will close up or down. The model will use daily signals from market prices, sector performance, volatility indices, macroeconomic variables, and news sentiment. The system will be implemented as a feature–training–inference pipeline running on Hopsworks, with XGBoost as the core model and FinBERT used to turn news text into sentiment features.

2 Dataset

The dataset will be built from three main data sources:

- Yahoo Finance (yfinance) for QQQ, XLK, and VIX. This provides OHLCV data and lets us compute QQQ technical indicators (returns, lagged returns, rolling volatility, RSI, moving-average ratios) and sector features from XLK, as well as volatility metrics from VIX (close value and daily change).
- FRED for macroeconomic variables such as the 10-year Treasury yield (DGS10) and CPI. Yield will be used as a daily time series, while CPI will be forward-filled between releases so each day has a consistent macro snapshot.
- NewsAPI + FinBERT for daily news sentiment. NewsAPI will supply titles and descriptions of relevant news articles (e.g., about QQQ, Nasdaq, big tech, and macro topics). FinBERT, a pretrained transformer model for financial text, will be applied to each article to obtain article-level sentiment scores. These scores

will then be aggregated per day into features such as average sentiment, sentiment dispersion, and article count.

After alignment by date, the final daily dataset will contain:

- QQQ technical features
- XLK sector influence features (returns and rolling correlation with QQQ)
- VIX-based volatility features
- Macro features from yields and CPI
- FinBERT-based news sentiment features (e.g., mean sentiment, number of articles)
- Targets: next-day QQQ return and a binary up/down label

3 Tools

The main tools and libraries used in the project are:

- Hopsworks for data ingestion, feature groups, feature views, model registry, and pipeline orchestration.
- XGBoost for: a regression model predicting next-day return and a classification model predicting the probability that QQQ closes higher.
- FinBERT (via Hugging Face) for computing sentiment on news text from NewsAPI, specialized for financial language.
- HuggingFace for a simple dashboard to visualize predictions, actual returns, and key features.
- APIs: yfinance for market and volatility data, FRED API for macro data, NewsAPI for news articles (titles/descriptions) used as input to FinBERT

4 Methodology

The pipeline in Hopsworks will run in several stages. First, daily ingestion jobs will fetch raw data from Yahoo Finance (QQQ, XLK, VIX), FRED (10-year yield and CPI), and NewsAPI (relevant news articles). These raw datasets will be stored in Hopsworks and then processed by feature engineering jobs. Feature engineering will create dedicated feature groups: one for QQQ and XLK price and technical features, one for volatility features from VIX, one for macro indicators, and one for news sentiment. For the news feature group, FinBERT will be applied to each article's text (e.g., title + description) to obtain sentiment scores, which will then be aggregated per day into metrics such as average sentiment, sentiment standard deviation, and article count. All feature groups

will be keyed by date, and a feature view will join them into a single, consistent daily feature table. A scheduled training job will read historical data from the feature view, construct time-series-aware training and validation splits, and train two XGBoost models: a regressor for next-day return and a classifier for up/down movement. Performance will be evaluated using appropriate metrics (e.g., MAE/RMSE and directional accuracy for regression; AUC and accuracy for classification), and the trained models will be saved in the Hopsworks Model Registry. For daily inference, a separate job will pull the latest feature row from the feature view, load the most recent XGBoost models from the registry, and generate predictions for the next trading day. These predictions, along with the corresponding input features, will be stored in a predictions dataset or feature group. Finally, a HuggingFace-based dashboard running in Hopsworks will read from the predictions and feature data to display recent predicted vs actual returns, up/down probabilities, and key input signals such as VIX, yields, XLK behavior, and FinBERT-based sentiment. The dashboard can also surface feature importance to give a quick understanding of which factors (including news sentiment) are driving the model's forecasts on recent days.