Paper Review

Responding to News Sensitively in Stock Attention Networks via Prompt-Adaptive Trimodal Model

Haotian Liu, Bowen Hu , Yadong Zhou and Yuxun Zhou

IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, VOL 36, NO. 6, JUNE 2025

Outline

- Introduction
- Related Work
- Problem Statement
- PA_TMM Architecture
- Model Optimization
- Experiments
- Conclusion

Introduction

Backgrounds and Challenges

Rapid Growth of Multimedia Platforms	*Limitations of Existing Models*	*Two Main Challenges*
• Financial news & social media provide **crucial investment signals** • Stock prices contain randomness (random walk) • But, there is **deterministic components** driven by news	• Time-series Forecasting Model - Stocks are mutually exclusive ➡️ **Ignore inter-stock dynamics** like Momentum spillover • GNN Model: - Hard-coded microstructure ➡️ **Capture limited interactions** • GATs Model - Dynamically assign weights ➡️ **Biased attention effect** ← Massive price-related features	• Real-world: a fraction of stocks have the news 1. **Long Tail Effect in Feature Distribution** - Distracted by abundant time-series data - Breaking news receive insufficient attention ➡️ Biased attention toward price related information 2. **Data Scarcity Problem** ➡️ Poor generalization

Introduction

Backgrounds and Challenges

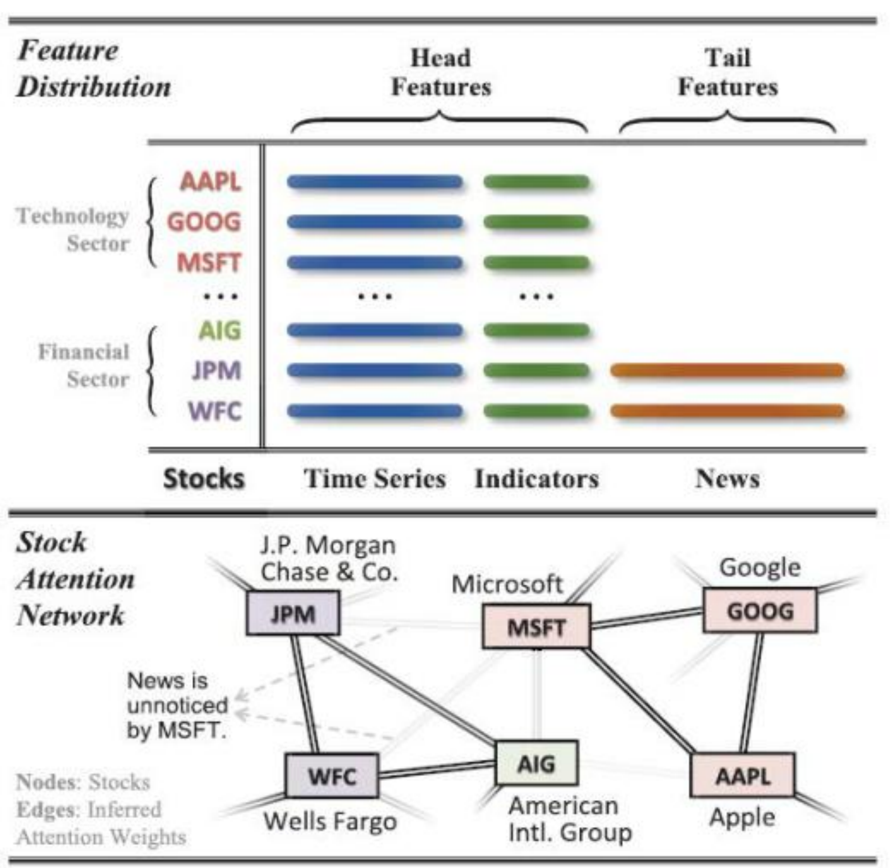


Fig. 1. Example of the long-tailed feature distribution of stocks causing biased attention for MSFT on a certain day, where breaking news from financial institutions (JPM and WFC) should have been crucial for any stock.

p. 2

Top

- Time-series features and Technical indicators
  ➡️ All stocks
- News
  ➡️ Few stocks: JPM, WFC

Bottom

- JPM, WFC: Breaking news that **impact the overall stock market**
- Other sectors: **News is unnoticed**

Introduction

Solution

Turning Point : Near-equivalence

- Idiosyncratic Nature of Financial News
  → Breaking news of specific stocks has **Instantaneous dominance over their movements**
- Example: 2024.01, the Federal Aviation Administration ordered airlines to ground more than 170 Boeing 737 aircraft
  → Boeing's stock: Drop 8%
  → **Airbus's stock: Slight increase**
  - Boeing's primary competitor:

Prompt-Adaptive Trimodal Model (PA-TMM)

- **Cross-Modal Fusion Module**
  : Trimodal features
  ➡️ Sentiments Prompts
- **Graph Dual-Attention Module**
  ➡️ Stock Attention Network (dynamic interaction)
- **Movement Prompt Adaptation**
  : Equivalence Resampling
  ➡️ Movement Prompt
- **Pretraining, Fine-tuning**
  : Adapt to feature Imbalance
  → Focus on real news

Contributions

1. Incorporate the dominance of news and capture news propagation dynamics by GATs
2. Pretraining with MPA: Sensitively respond to tailed news and avoid over-reliance on stocks carrying news
3. Equivalence Resampling Strategy (Data Augmentation)
   : Tackle data scarcity problem and enhance generalizability

Related Work

Time-Series Stock Prediction

- Encode the time series pattern by using RNN based models
  - PEN [40], MAN-SF [12], and MTR-C [41]
- Mingle market factors [42], investment behaviors [43], technical indicators
  - REST [44], Digger-Guider [45]
- Limitation of underlying assumption
  - **Stocks are mutually exclusive**

Graph-Based Stock Prediction

- Conceptualize the stock market as a graph
  - Capture peer influences by using Graph Neural Networks
  - Node (each stock), Edges (relations)
- THGNN [3], ESTIMATE [2], SAMBA [49]
- Limitation
  - **Relations are modeled by the static network**

News-Based Stock Prediction

- Integrate external information beyond the trading market
  - Financial news, Social media posts
- Graph Convolutional Networks
  - MAC [1], NumHTML [52], MFN [53]
- Graph Attention Networks
  - AD-GAT [15], DANSMP [6]
- Multi-Modal: MSMF [54]
- Limitation
  - **Lack of consideration for the long-tail effect**

Problem Statement

1 Classification Method for Optimization

- In the stock market, predicting the exact value of stock prices is far more challenging than predicting price movements
- **Classify stock as rising or falling: Compare current and previous day stock prices**

Given a set $V = \{1, \ldots, N\}$ of stocks,
the movement (labels) y_i^t of stock $i \in V$ on day t is defined as

$$y_i^t = \begin{cases} 1, & c_i^t > c_i^{t-1} \\ 0, & c_i^t \le c_i^{t-1} \end{cases} \quad (1)$$

Problem Statement

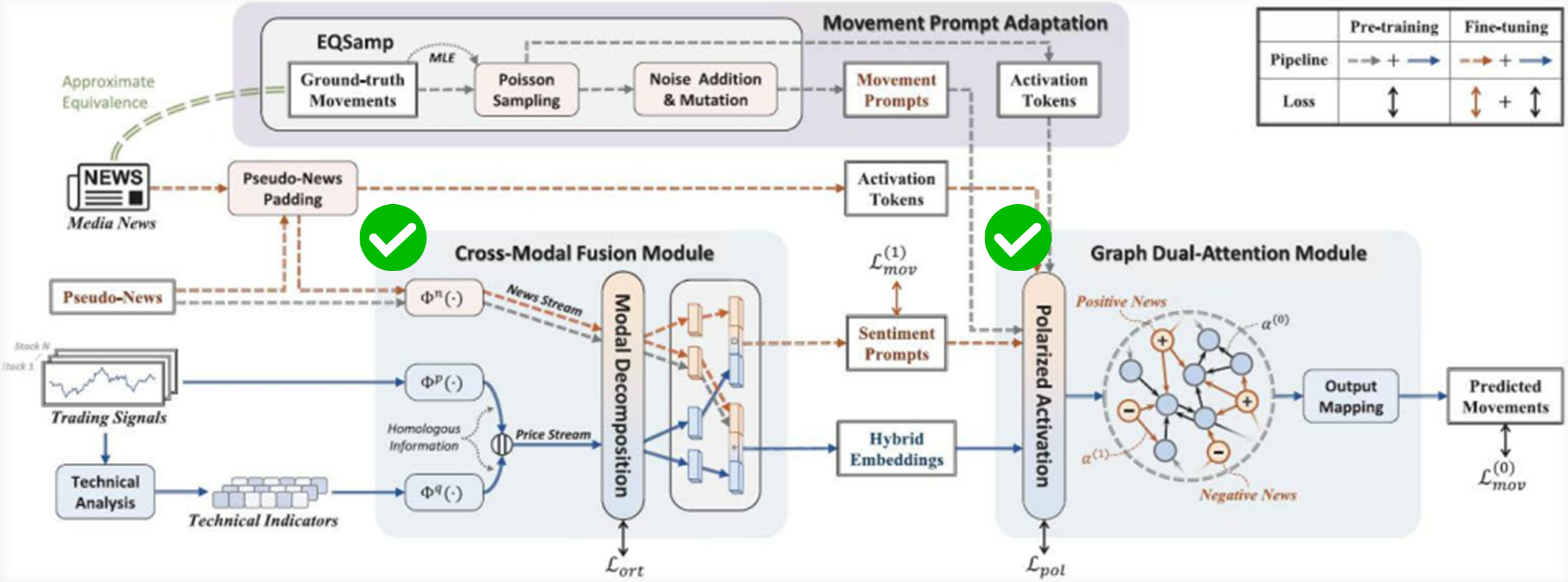
2 Three Feature Modalities

- Input features on the $(T-1)$th day ➔ Predict the movements on the Tth day

Textual News Copora T	• Labeled the relevant stocks impacted by each news item
Historical Time-Series Trading Signals	• $X_i^{[t-T;t]} = \left[x_i^{[t-T]}, \ldots, x_i^{t-1}\right] \in \mathbb{R}^{T \times d_x}$ for each stock $i \in V$ • Transaction features of stock i on the Tth day • Price (highest, lowest, Opening and Closing), Trade volume, and Rankings • Normalize prices to handle scale differences
Tabular Technical Indicators	• $I_i^{t-1} \in \mathbb{R}^{d_f}$ for each stock $i \in V$ • Computed through the technical analysis of historical trading signals • Moving Average Indicators, Momentum Indicators, Volatility Indicators, Volume Indicators

Prompt-Adaptive Trimodal Model Architecture

- Two Subnetworks: **Cross-Modal Fusion Module, Graph Dual-Attention Module**
- **Movement Prompt Adaptation → Pretraining ➡️ Fine-tuning**



PA-TMM Architecture

Cross-Modal Fusion Module

1 Pseudo-News Padding and Activation State

1. **Fill the news position with pseudo-news** (i.e., a space character; “ “)

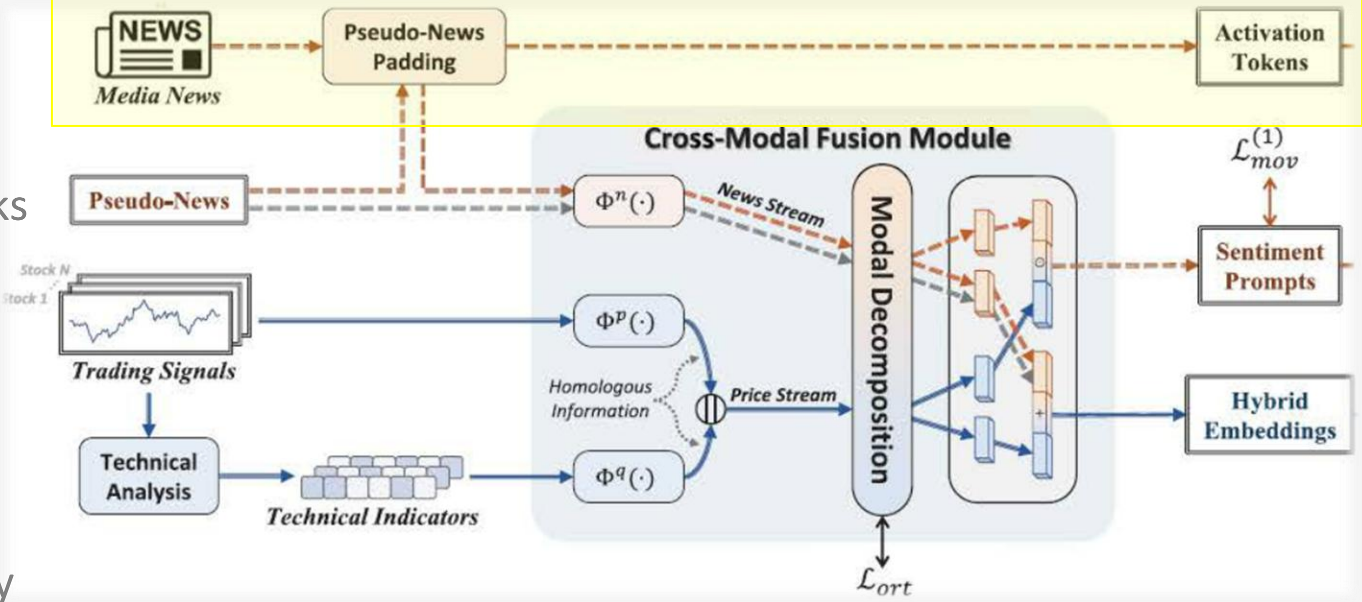
   - News may be absent for certain stocks

   - Address the issue of modality incompleteness with flexibility

2. Two mutually exclusive subsets on the day t

   - Nonactivation Subset $V^{(0)}$: Price-only

   - Activation Subset $V^{(1)}$: Price, News



PA-TMM Architecture

Cross-Modal Fusion Module

2 Representation Learning

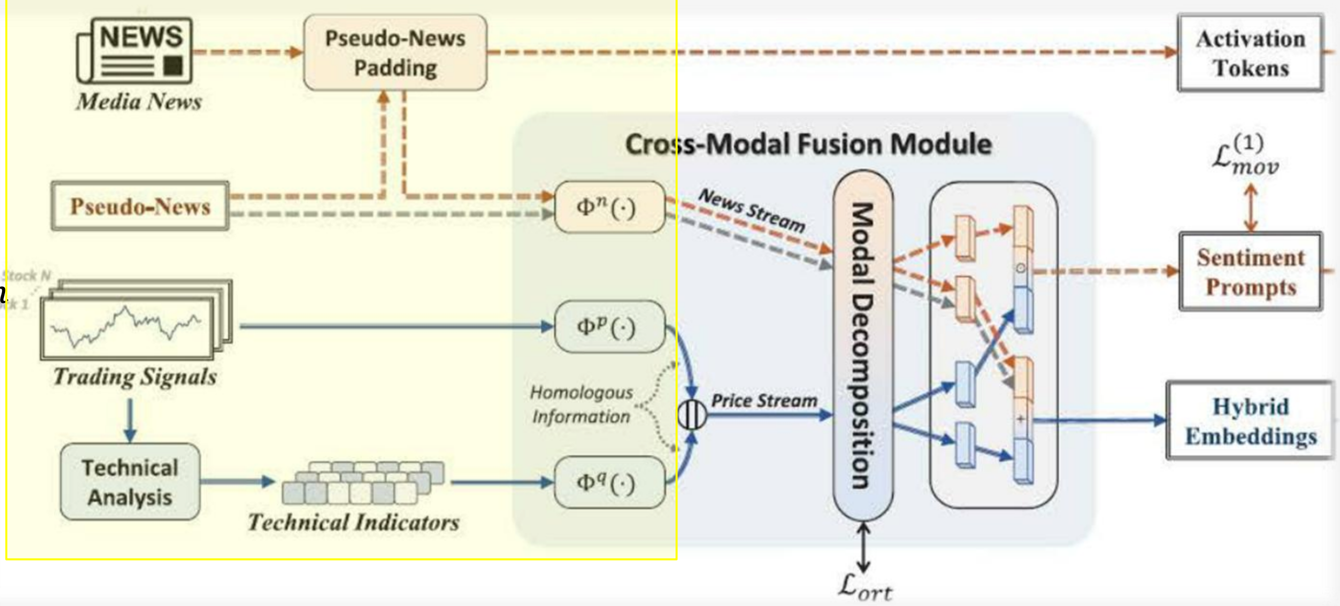
- News-Related Information

  - $m_i = \left(\frac{1}{L}\right) \sum_{l=1}^{L} BERT\,(s_i^{t,l}), \in \mathbb{R}^{d_m}$

- Price-Related Information

  - $p_i = Bi\text{ - }LSTM(X_i^{[t-T;t]})$

  - $q_i = TabNet(I_i^{t-1})$



- Where

  - stock $i \in V$ *on the* $(t-1)th$ *day*

  - $s_i^t \in News\ T$, L = # of stock-specific news (target day), $I_i^{t-1} \in \mathbb{R}^{d_f}$, $X_i^{[t-T;t]} \in \mathbb{R}^{T \times d_x}$

  - $\mathbf{m_i} \in \mathbb{R}^{d_m}, \mathbf{p}_i \in \mathbb{R}^{d_p}, \mathbf{q}_i \in \mathbb{R}^{d_q}$

  - TabNet: Technical Analysis Library(https://ta-lib.org/)

PA-TMM Architecture

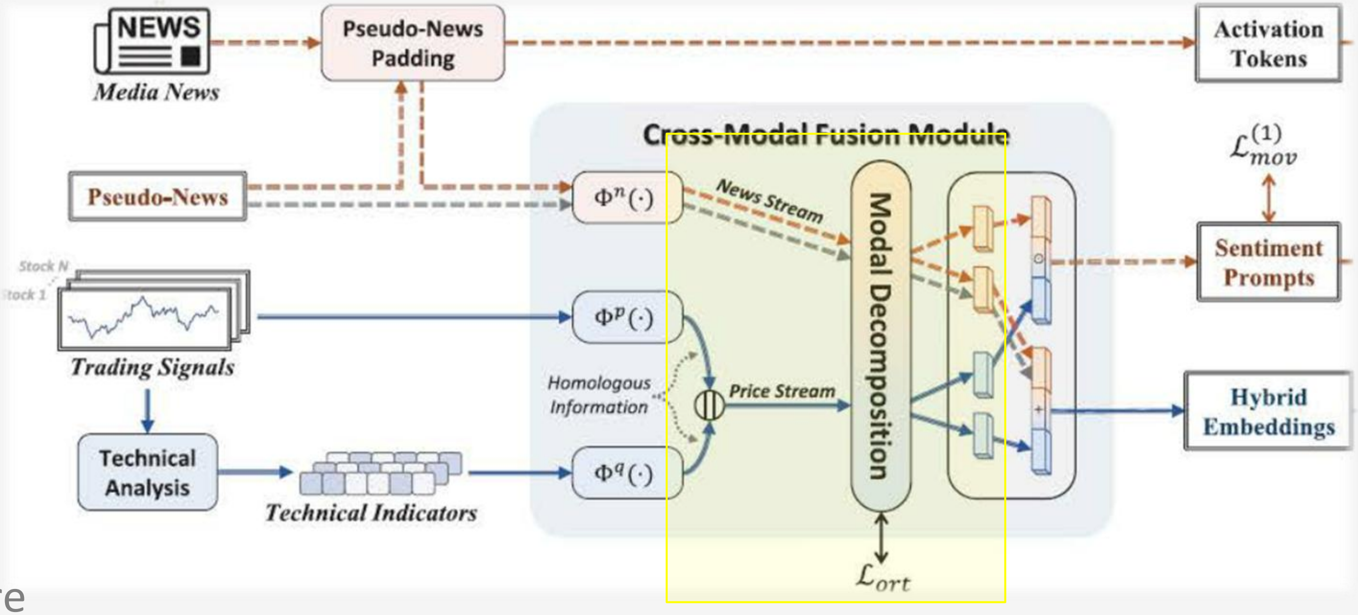
Cross-Modal Fusion Module

3 Modal Decomposition

- Project trimodal representations into **4 different spaces**

	News	Price
Specific	$\boldsymbol{u_i^m}$	$\boldsymbol{u_i^p}$
Shared	$\boldsymbol{v_i^m}$	$\boldsymbol{v_i^p}$

$$\begin{bmatrix} u_i^m \\ v_i^m \\ u_i^p \\ v_i^p \end{bmatrix} = \sigma\left(\begin{bmatrix} W_{um} m_i \\ W_{vm} m_i \\ W_{up}[p_i || q_i] \\ W_{vp}[p_i || q_i] \end{bmatrix}\right)$$



Where

- $W_{um} \in \mathbb{R}^{d_r \times d_m}$, $W_{vm} \in \mathbb{R}^{d_r \times d_m}$, $W_{up} \in \mathbb{R}^{d_r \times (d_p + d_q)}$, $W_{vp} \in \mathbb{R}^{d_r \times (d_p + d_q)}$
- $\sigma(\cdot)$: Nonlinear Activation Function

- Orthogonal Loss: **Ensure the independence** of the modal-specific spaces from the modal-shared spaces
  - $L_{ort} = \| W_{um} \cdot W_{vm}^T \|_F + \| W_{up} \cdot W_{vp}^T \|_F$, where $\| \|_F$: Frobenius Norm = $\sqrt{\sum\sum |x_{ij}|^2}$

PA-TMM Architecture

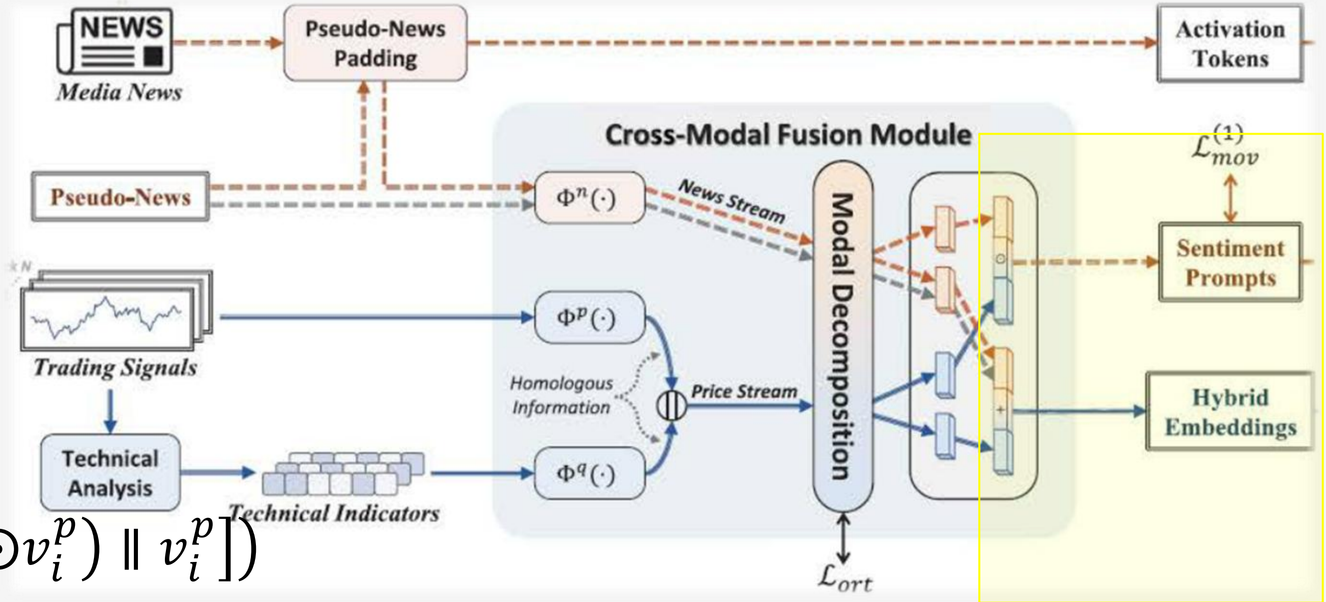
Cross-Modal Fusion Module

4 Modal Integration

- Media sentiment → Impact stock prices
- News-Stream Integration

  ➡️ **Sentiment Prompts**

$$h_i^{pmt} = [\hat{h}_i^- \parallel \hat{h}_i^+]$$
$$= softmax(W_{zr}[u_i^m \parallel (u_i^m \odot v_i^p) \parallel v_i^p])$$



- Price-Stream Integration ➡️ **Hybrid Embeddings**

$$h_i^{hyb} = \sigma(W_{hr}[u_i^p \parallel (u_i^p + v_i^m) \parallel v_i^m] + b_h)$$

Where

- $u_i^m \odot v_i^p$: News (sentiment source), Price (noise filter)
- $u_i^P + v_i^m$: Equally crucial → Addition Operation
- $h_i^{pmt} \in \mathbb{R}^2, W_{zr} \in \mathbb{R}^{2\times 3d_r}, \boldsymbol{i} \in \boldsymbol{V}^{(1)}$ (Only Activation)

  $h_i^{hyb} \in \mathbb{R}^{d_h}, \mathrm{W_{hr}} \in \mathbb{R}^{d_h \times 3d_r}, \mathrm{b_h} \in \mathbb{R}^{d_h}, i \in V$

PA-TMM Architecture

Graph Dual-Attention Module

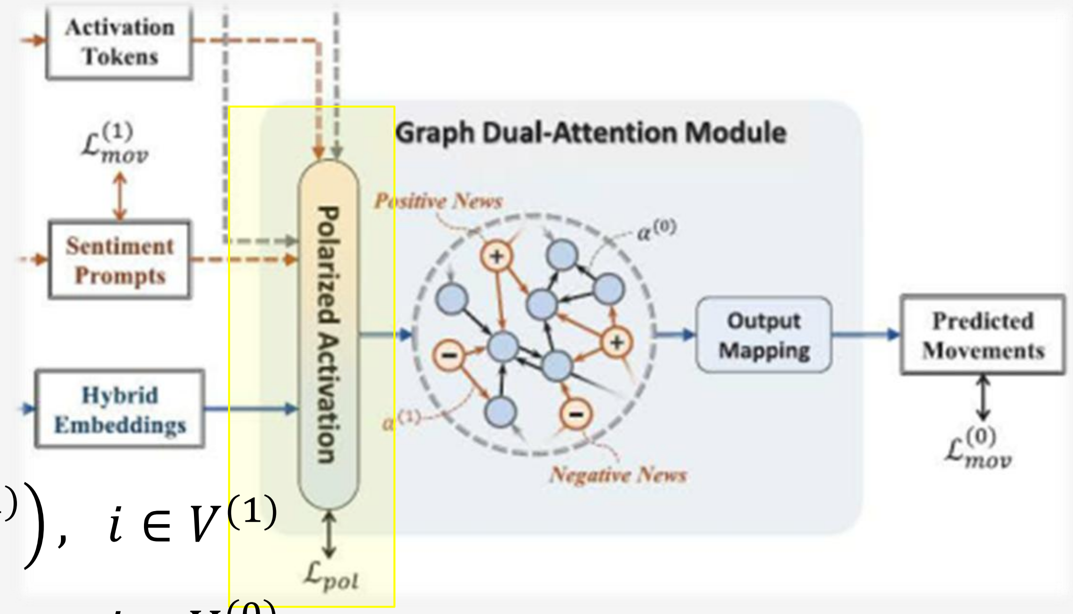
1 Stock Polarized Activation

- Activated stocks carry **more dominant information** ➡️ Asymmetric feature distribution
- Different activation states → Varying inter-stock influences ➡️ **Separate embedding**

$$\text{Node Vector } n_i = \begin{cases} \sigma\left(W_{nh}^{(1)}\left[h_i^{hyb} \parallel h_i^{pmt}\right] + b_n^{(1)}\right), & i \in V^{(1)} \\ \sigma\left(W_{nh}^{(0)}\left[h_i^{hyb}\right] + b_n^{(0)}\right), & i \in V^{(0)} \end{cases}$$

- Polarization loss (via cosine distance)
“Opposing ➡️ **Separated**, Similar ➡️ **Closer**”

$$L_{pol} = \sum_{i \in V^{(1)}} \sum_{j \in V^{(1)}} \cos(n_i, n_j)\, \mathrm{sgn}\left(\left(\hat{h}_i^{+} - \hat{h}_i^{-}\right)\left(\hat{h}_j^{+} - \hat{h}_j^{-}\right)\right)$$



Where

- $n_i \in \mathbb{R}^{d_n}$, $W_{nh}^{(1)} \in \mathbb{R}^{d_n \times (d_h+2)}$, $W_{nh}^{(0)} \in \mathbb{R}^{d_n \times (d_h)}$, $b_n^{(1)} \in \mathbb{R}^{d_n}$, $b_n^{(0)} \in \mathbb{R}^{d_n}$

PA-TMM Architecture

Graph Dual-Attention Module

2 Interaction Inference

- Dynamic Interaction ➡️ **Attnetion Network**
- Reflecting information flow[Activated → Nonactivated]

  ➡️ **Partially Bipartite** (Fig.3)

- Attention Score $\alpha_{i,j}^{(k)} = \frac{\exp\left(\varphi(n_i, n_j)\right)}{\sum_{j \in V^{(k)}} \exp\left(\varphi(n_i, n_j)\right)}$
- Message Flux

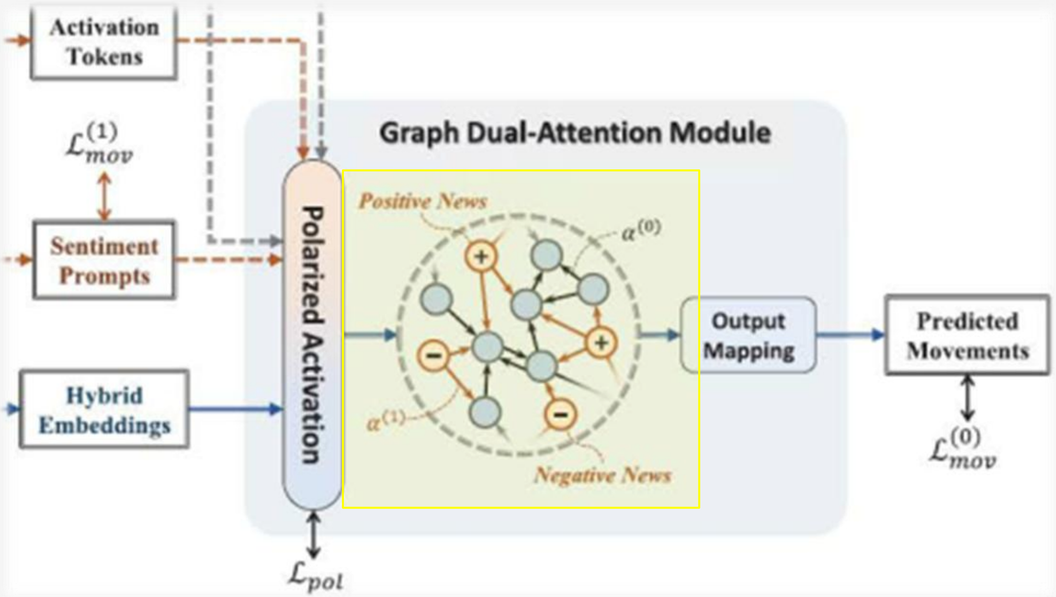
$$\varphi(n_i, n_j) = a_\varphi^T LeakyRelu\left(W_{n\varphi}\left[n_i \parallel n_j\right]\right)$$

- Where

  - $\mathbf{n}_i$: **Target Node** $\in \boldsymbol{V}^{(0)}$, $\mathbf{n}_j$: **Source Node** $\in \boldsymbol{V}$

  - $\alpha^{(1)}: V^{(1)} \to V^{(0)}, \alpha^{(0)}: V^{(0)} \to V^{(0)}$

  - $a_\varphi \in \mathbb{R}^{d_\varphi}, W_{\varphi n} \in \mathbb{R}^{d_\varphi \times 2d_n}$



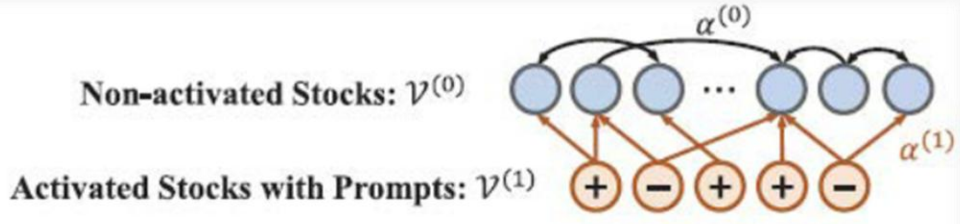


Fig. 3. Inferred partial-bipartite stock attention network.

PA-TMM Architecture

Graph Dual-Attention Module

3 Information Exchange

- Edge(information flow $N_j \rightarrow N_i$)

$$e_{i,j} = W_{eo}\left[n_i \parallel \sigma\left(W_{on}\left[n_i \parallel n_j\right]\right) \parallel n_j\right]$$

- Message Vector (weighted sum)

$$\tilde{m}_i = \parallel_{k\in\{0.1\}} \sigma\left(\sum_{j\in V^{(k)}} \alpha_{i,j}^{(k)} e_{i,j}\right)$$

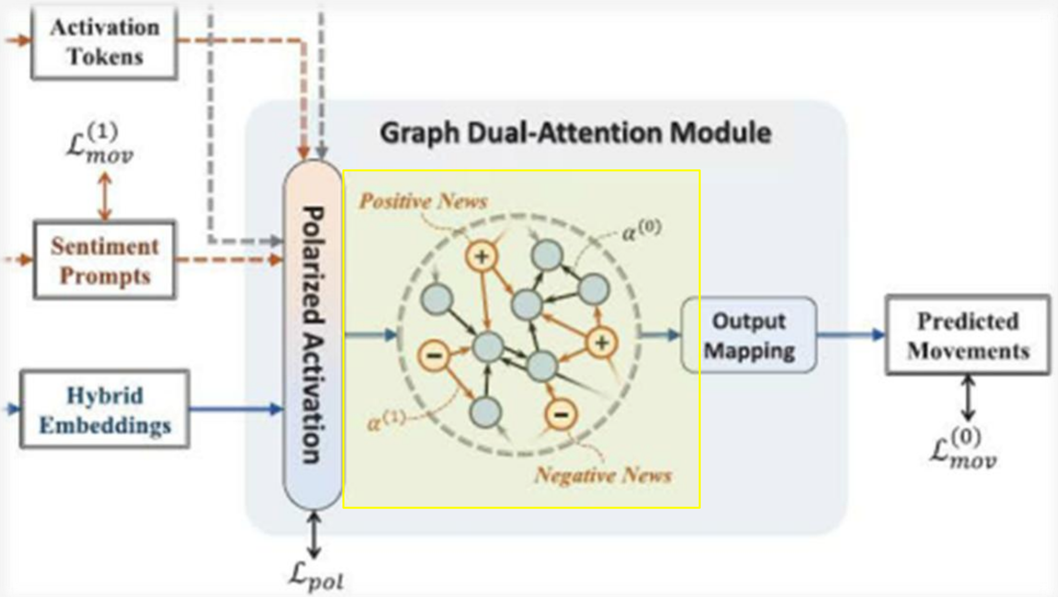
- For nonactivated stock,

summary of peer stock interactions

- Where

- $\tilde{m}_i \in \mathbb{R}^{2d_e}$, $\alpha_{i,j}^{(k)}$: *Attention Score* $(N_j \rightarrow N_i)$, $N \in \mathbb{R}^{d_n}$

- $e_{i,j} \in \mathbb{R}^{d_e}$, $W_{eo} \in \mathbb{R}^{d_e \times 3d_n}$, $W_{on} \in \mathbb{R}^{d_n \times 2d_n}$



PA-TMM Architecture

Graph Dual-Attention Module

4 Output Mapping

- For activated stock, news often dominates price movement

  ➡️ Sentiment Prompts = Movement Prediction

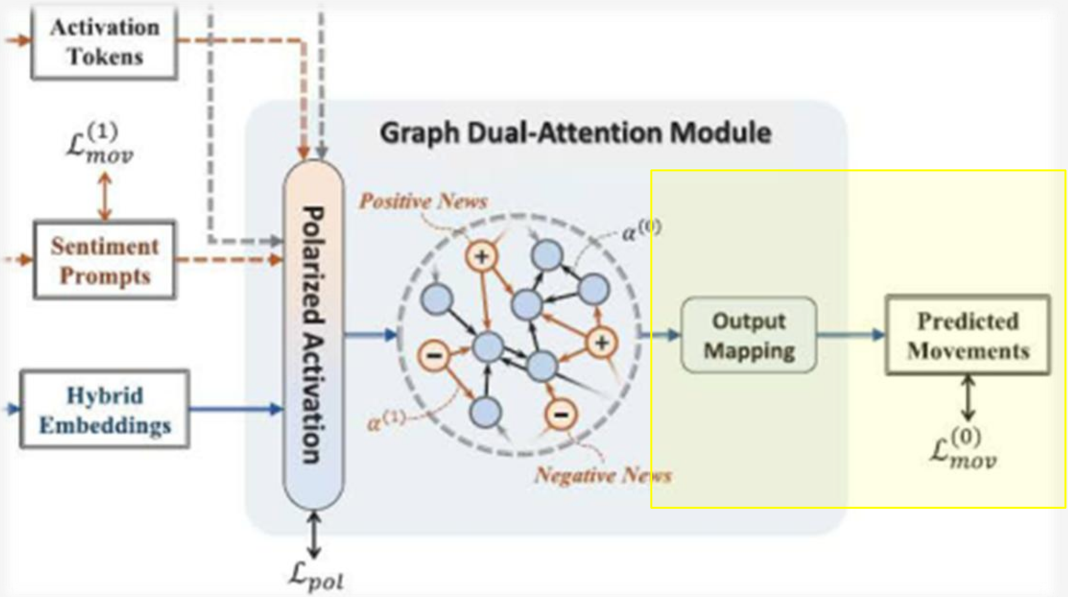
$$\hat{y}_i = h_i^{pmt} = [\hat{h}_i^- \parallel \hat{h}_i^+],\ i \in V^{(1)}$$

- Fo nonactivated stock, a feed-forward neural network is used

$$\hat{y}_i = [\hat{y}_i^- \parallel \hat{y}_i^+] = softmax(W_i[n_i \parallel \tilde{m}_i] + b_i),\ i \in V^{(0)}$$

- Where

  - $\hat{y}_i$: Movement Prediction $\in \mathbb{R}^2$, $W_i \in \mathbb{R}^{2\times(d_n+2d_e)}$, $b_i \in \mathbb{R}^2$



PA-TMM Architecture

Graph Dual-Attention Module

5 Discussion

- PA-TMM models the stock network as **a partially bipartite graph**
  - The conventional GATs: Homogeneous graphs
- **Message vectors ($\tilde{m}_i$) play a vital role** in the model's performance
  - Demonstrated by ablation experiments (TABEL V), w/o Msgs
  - Calculated by the attention scores
- Treating **activated and nonactivated nodes separately** is crucial
  - Increase computational complexity
  ➡️ However, focus is on the performance
- More lightweight attention modules could be the future work

PA-TMM Architecture

Computational Complexity

Cross-Modal Fusion Module

- Primary Cost: **Recurrent Component of Bi-LSTM**

  \- $O(N \times T \times d_p^2)$

  where, N: # of Stocks, T: Length of Time Series

  , d_p: Hidden Size of LSTM
- Other parts are negligible (linear layers)

Graph Dual-Attention Module

- Primary Cost: **Interactions Inference**

  \- $O(N^2 \times d_n)$

  where, d_n : Dimension of Node Vector
- Other parts are negligible (linear layers)

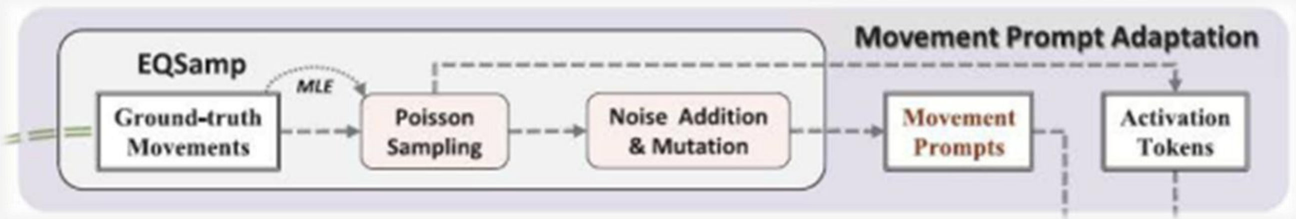
Overall Complexity

$$O(N \times T \times d_p^2) + O(N^2 \times d_n)$$

- Training Cost (hrs) : 4.7 for NASDAQ100, 7.9 for S&P500
- Test Cost (sec) : 0.11 for NASDAQ100, 0.32 for S&P500

Model Optimization

Movement Prompt Adaptation
: Equivalence Resampling



- **Data Augmentation** Strategy
  → Tackle long tail effect in feature distribution
- Wide Range of **Possible Scenarios** → Enhance generalizability

- **Generate prompts** (equivalent to market sentiments) from past stock movements

1. **Randomly activating** a stock subset $V^{(1)} \subset V$
   - Number of stocks is varying with a Poisson distribution ➔ Emulate daily variation in news-carrying stocks
2. **Ground-Truth Movements** ➡️ **Movement Prompts**

$$h_i^{pmt} = [\hat{h}_i^- \parallel \hat{h}_i^+] = \begin{cases} [(1-\epsilon_i) \parallel \epsilon_i], \ when\ y_i^t = 0\ (Down) \\ [\epsilon_i \parallel (1-\epsilon_i)], \ when\ y_i^t = 1\ (Up) \end{cases}$$

where, $\epsilon_i \sim \mathrm{U}(0, 0.5)$
Randomness to Prompts for Robustness

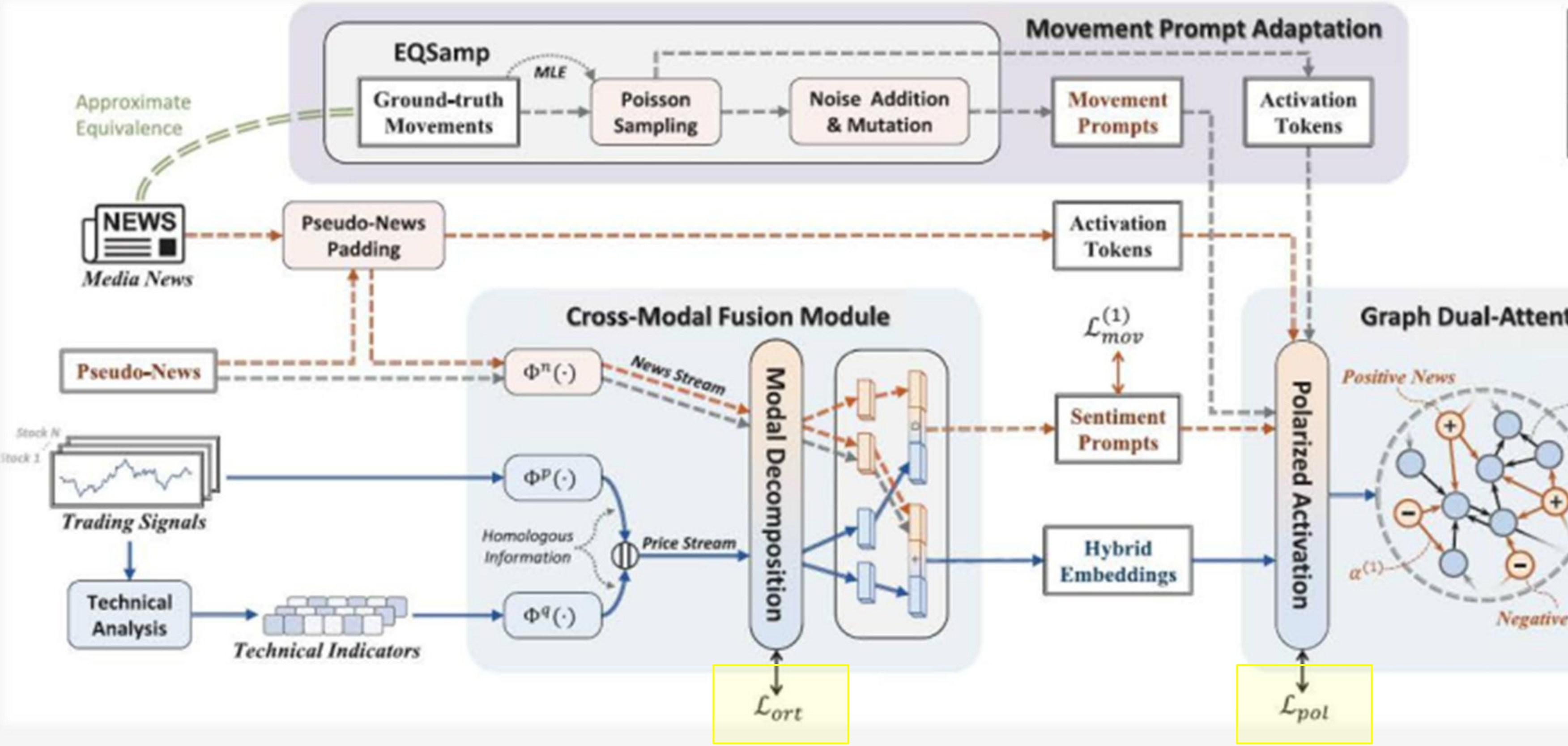
3. **Inverting** Movement Prompts: $h_i^{pmt} \leftarrow 1 - h_i^{pmt}$, with Mutation Probability θ
   ➡️ By Injecting data noise, prevent the model from over-fitting

Model Optimization

Pretraining Objectives

$$L_{mov}^{(0)} = -\frac{1}{|V|}\left(\sum_{i\in V^{(0)}}\left(\left(1-y_i^t\right)\log(\hat{y}_i^-)+y_i^t\log(\hat{y}_i^+)\right)\right) \Rightarrow L_p = \sum_t\left(L_{mov}^{(0)}+\beta L_{ort}+\gamma L_{pol}\right)$$

Where, β and γ are the weighting factors



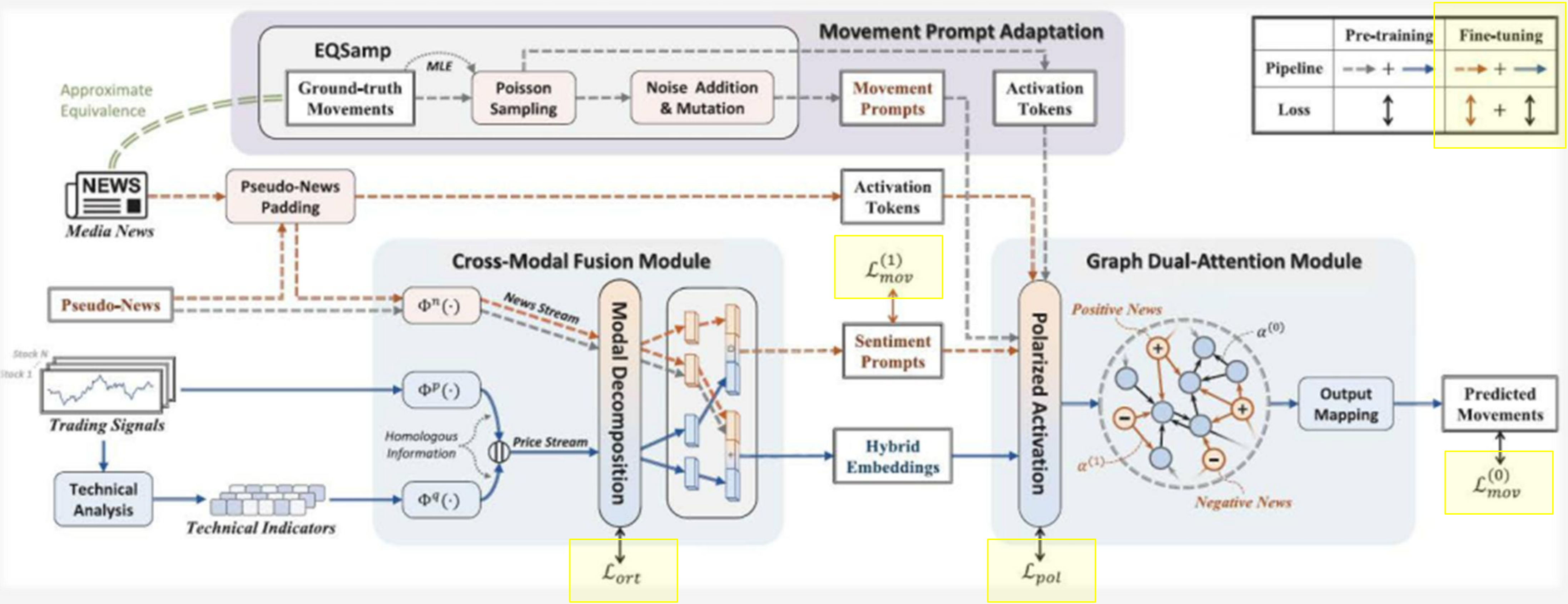
	Pre-training	Fine-tuning
Pipeline	⇢ + →	⇢ + →
Loss	↕	↕ + ↕

Model Optimization

Fine-Tuning Objectives

$$L_{mov}^{(1)} = -\frac{1}{|V|}\left(\sum_{i\in V^{(1)}}\left(\left(1-y_i^t\right)\log\left(\hat{h}_i^-\right)+y_i^t\log\left(\hat{h}_i^+\right)\right)\right) \;➡\; L_F = \sum_t\left(L_{mov}^{(1)} + L_{mov}^{(0)} + \beta L_{ort} + \gamma L_{pol}\right)$$

Where, β and γ are the weighting factors



Experiments

Evaluation Setup

1 Datasets

Historical Trading Data

Table1	NASDAQ 100	S&P 500
# Stocks (Nodes)	118	510
# Stocks with news (Average activated nodes)	11	26
Pre-training period	Jan. 2014 - Dec. 2015	Jan. 2014 - Dec. 2015
Fine-tuning and validation period	Jan. 2016 - Dec. 2018	Jan. 2016 - Dec. 2018
Test period (12 tests in total)	Jan. 2019 - Dec. 2019	Jan. 2019 - Dec. 2019

- Source: Yahoo Finance, Nasdaq Data Link
- Trading information: highest price, lowest price, opening price, closing price, and trade volume

Technical Indicators

- Using TA-lib (Technical Analysis Library)
- e.g., Moving Average Indicators, Momentum Indicators, Volatility Indicators, Volume Indicators

News Headlines

- Period: 2016.01 ~ 2019.12
- Source: Benzinga(Financial News Platform)
- Labeled the relevant stocks impacted by each news item.
- Total: 10,536 news articles

Experiments

Evaluation Setup

2 Compared Baselines

- **Comparative analysis** against nine state-of-the-art baselines

1. Sequential Models (RNN variants)

   - LSTM [37], Transformer [39], Frequency Interpolation Timeseries Analysis Baselines (FITS) [60] and Pathformer [61]

2. Graph-Based Methods (GNNs variants)

   - ESTIMATE [2], Temporal Graph Convolution (TGC) [23], Subsequence based Graph Routing Network (S-GRN) [46], and SAMBA [49]

3. Multimodal Method

   - Bimodal (time series and news): PEN [40], STHAN-SR [14], AD-GAT [15], DANSMP [6]

   - Trimodal (time series, news, and technical indicators): MCASP [62], and MSMF [54]

Experiments

Evaluation Setup

3 Evaluation Metrics

- Accuracy (ACC)

  - Ratio of correctly predicted labels (both positive and negative) to the total number of predictions

  - $ACC = \frac{TP+TN}{TP+FP+TN+FN}$, where T: True, F: False, P: Positive, N: Negative

- Mathew's Correlation Coefficient (MCC)

  - $MCC = \frac{TP\times TN-FP\times FX}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$, which handles imbalanced datasets

- Backtesting Profitability for a simulated trading

  - $Annual\ Rate\ of\ Return = \frac{Final\ Value - Principal}{Principal}$

  - $Annulized\ Sharpe\ Ratio = \frac{ARR - R_f}{\sigma_p}$

  - where R_f: ARR of Risk-Free Asset, σ_p: Annualized Standard Deviation of the Portfolio

Experiments

Evaluation Setup

4 Implementation Details

- Pretraning Datasets: 2014.01 ~ 2015.12
  - **Resampled the same day 50 times** to generate 50 different activation subsets in two years
- Fine-tuning Datasets: 2016.01 ~ 2019.12
- Grid Search: Hyper Parameters (Table2) Optimization
- For learnable parameters
  - Glorot Initialization, AdamW Optimizer
  - Maximum of 200 Epochs
- Training Cost (hrs) : 4.7 for NASDAQ100, 7.9 for S&P500
- Test Cost (sec) : 0.11 for NASDAQ100, 0.32 for S&P500
- GPU: NVIDIA Titan V

Table2 **Hyper-parameters**	**NASDAQ 100**	**S&P 500**
Window size T	12	16
Dimension d_m	768	768
Dimension d_r	64	128
Dimension d_ϕ	192	192
Dimension d_e	256	320
Dimensions d_p and d_q	128	128
Dimensions d_h and d_n	256	384
Mutation probability θ	0.25	0.25
Loss weight β	0.15	0.2
Loss weight γ	0.025	0.015
Learning rate (pre-training)	5e-4	5e-5
Learning rate (fine-tuning)	2e-4	1e-5
Batch size	8	4

Experiments

Evaluation Setup

5 Trading Portfolios

- Holding **20 stocks**
- Purchasing **a maximum of 10 of the highest-ranked stocks**, daily basis
  - Using movement prediction scores
  - Not already present in the portfolio
- Selling **an equivalent quantity of the lowest-ranked stocks**

➡️ Control the turnover rate

- Initial Account Capital: **U.S. $5 million**
- Transaction Costs: Buying 0.05%, Selling 0.15%

Experiments

Stock Movement Prediction

TABLE III

PERFORMANCE EVALUATION OF STOCK MOVEMENT PREDICTION (MEAN ± STANDARD DEVIATION)

Methods	NASDAQ 100		S&P 500	
	ACC (%)	MCC	ACC (%)	MCC
LSTM	53.41±0.34	0.065±0.046	53.18±0.69	0.061±0.034
Transformer	52.24±1.32	0.043±0.028	52.40±1.07	0.043±0.032
FITS	53.83±0.80	0.073±0.022	53.07±0.61	0.059±0.037
Pathformer	54.48±1.03	0.087±0.056	53.91±0.95	0.072±0.031
ESTIMATE	55.77±0.48	0.113±0.071	53.50±0.67	0.078±0.061
TGC	56.86±0.79	0.144±0.056	54.13±1.14	0.093±0.043
S-GRN	57.29±0.55	0.143±0.049	54.88±0.72	0.106±0.037
SAMBA	58.09±0.76	0.151±0.042	55.16±0.89	0.117±0.054
PEN	57.91±0.82	0.157±0.028	55.06±0.96	0.114±0.045
STHAN-SR	58.48±0.89	0.161±0.073	55.88±0.47	0.125±0.080
AD-GAT	58.95±1.91	0.184±0.093	56.39±1.52	0.133±0.084
MCASP	58.44±1.29	0.175±0.049	57.90±1.31	0.152±0.072
DANSMP	59.25±1.14	0.186±0.042	57.14±0.63	0.140±0.058
MSMF	59.45±0.92	0.181±0.055	57.44±1.08	0.148±0.087
PA-TMM	**60.34**±0.75	**0.199**±0.059	**59.21**±1.22	**0.187**±0.074

p. 10

Smaller Advantage

TABLE IV

DIEBOLD-MARIANO TEST RESULTS BETWEEN PA-TMM AND EACH BASELINE ON TWO DATASETS

Baselines	NASDAQ 100	S&P 500
LSTM	0.0000***	0.0000***
Transformer	0.0000***	0.0000***
FITS	0.0004***	0.0031***
Pathformer	0.0000***	0.0005***
ESTIMATE	0.0004***	0.0012***
TGC	0.0035***	0.0094***
S-GRN	0.0108**	0.0364**
SAMBA	0.0329**	0.0255**
PEN	0.0278**	0.0041***
STHAN-SR	0.0793*	0.0429**
AD-GAT	0.0351**	0.0230**
MCASP	0.0394**	0.0277**
DANSMP	0.0675*	0.0223**
MSMF	0.0545*	0.0381**

Note: *, **, and *** represent statistical significance at the levels of 10%, 5%, and 1%, respectively.

p. 11

Significantly Superior

Experiments

Stock Movement Prediction-Analysis

Time-Series Stock Prediction

- LSTM, Transformer, FITS, and Pathformer
- **Ignore complex relationships** among stocks
- Perform worse than graph-based models

Graph-Based Stock Prediction

- ESTIMATE, TGC, S-GRN, and SAMBA
- Outperform time-series models
- **Ignore external news information**
- Constrained by efficient capital markets
- Unnecessary noise information

News-Based Stock Prediction

- PEN, STHAN-SR, AD-OAT, and DANSMP
  - Bimodal (time series, news)
  - Superior performance compared to nonnews methods
- MCASP and MSMF
  - Trimodal (time series, news, and technical indicators)
  - Better performance on the S&P 500 dataset
- **Overlook the long-tailed feature distribution** ➡️ Underutillize the news

PA-TMM

- **Optimal prediction performance** in terms of both ACC and MCC
- **Overcome the long tail effect** inherent in stock feature distribution

Experiments

Ablation Study

Effectiveness of the Model Architecture

- Sentiment prompts and Graph aggregation mechanism
  - Pivotal role in addressing the long tail effect
  ➡️ Ensure the feasibility of implementing MPA
- Modal decomposition and Polarized activation
  ➡️ Enhance the efficiency of utilizing news

Effectiveness of MPA

- MPA: Responsive to activated nodes
  ➡️ Promptly capture news without over-fitting
- Varying number of activated nodes
  ➡️ Preadapt to the real-world pattern
- Mutation Strategy
  ➡️ Avoid over-reliance on activated nodes

TABLE V

PERFORMANCE OF DIFFERENT PA-TMM VARIANTS

Variants	NASDAQ 100		S&P 500	
	ACC (%)	MCC	ACC (%)	MCC
w/o Pmts.	53.37	0.0741	53.82	0.0771
w/o Msgs.	54.94	0.0916	53.19	0.0734
w/o $\mathcal{L}_{ort}$	58.76	0.1740	57.45	0.1463
w/o $\mathcal{L}_{pol}$	57.98	0.1623	57.12	0.1408
w/o MPA	56.32	0.1277	53.04	0.0595
w/o MPA-P	58.36	0.1682	56.12	0.1338
w/o MPA-M	57.65	0.1543	54.81	0.0924
PA-TMM	**60.34**	**0.1992**	**59.21**	**0.1873**

p. 11

Experiments

Backtesting Profitability

- Simulate real-world investment
- **Highest ARR Score, Highest ASR Score**
- Backtesting Results + Prediction Performance

  ➡️ **The more precise, the more profitable**
- Performance on the S&P 500 is lower

  - More stocks than NASDAQ 100.

  ➡️ Increased complexity of risk management
- News-Based Multimodal Methods

  - Higher ASR scores than those of GNN based methods

  ➡️ **Better resilience to risks**
- PA-TMM without constraint of predefined relationships

  - **Effectively leverage news sentiments**

TABLE VI

PERFORMANCE EVALUATION OF BACKTESTING PROFITABILITY

Methods	NASDAQ 100		S&P 500	
	ARR (%)	ASR	ARR (%)	ASR
LSTM	12.57	0.952	11.36	0.973
Transformer	10.47	0.873	10.02	0.791
FITS	14.15	1.025	12.08	0.997
Pathformer	16.76	1.143	13.64	1.185
ESTIMATE	19.63	1.396	12.65	1.022
TGC	22.46	1.424	15.37	1.214
S-GRN	24.02	1.969	16.83	1.581
SAMBA	24.51	1.660	17.53	1.347
PEN	24.27	1.817	19.05	1.426
STHAN-SR	25.98	1.898	20.26	1.440
AD-GAT	27.52	2.067	21.83	1.557
MCASP	26.21	2.050	23.84	1.697
DANSMP	28.07	2.155	23.23	1.682
MSMF	27.43	2.129	24.21	1.732
PA-TMM	**30.44**	**2.381**	**26.90**	**1.942**

p. 12

Experiments

Stress Test During Market Crash

- Market Crash: 2018.10 ~ 2018.12
  - The Federal Reserve persistently increased interest rate
  ➡️ Stock market came under pressure
- Demonstrates **substantial resilience** in extreme market conditions
  - Early: Struggled to navigate the market's panic sentiment
  - Later: Progressively acclimated to the market conditions
- Showcase an **aptitude for risk management**

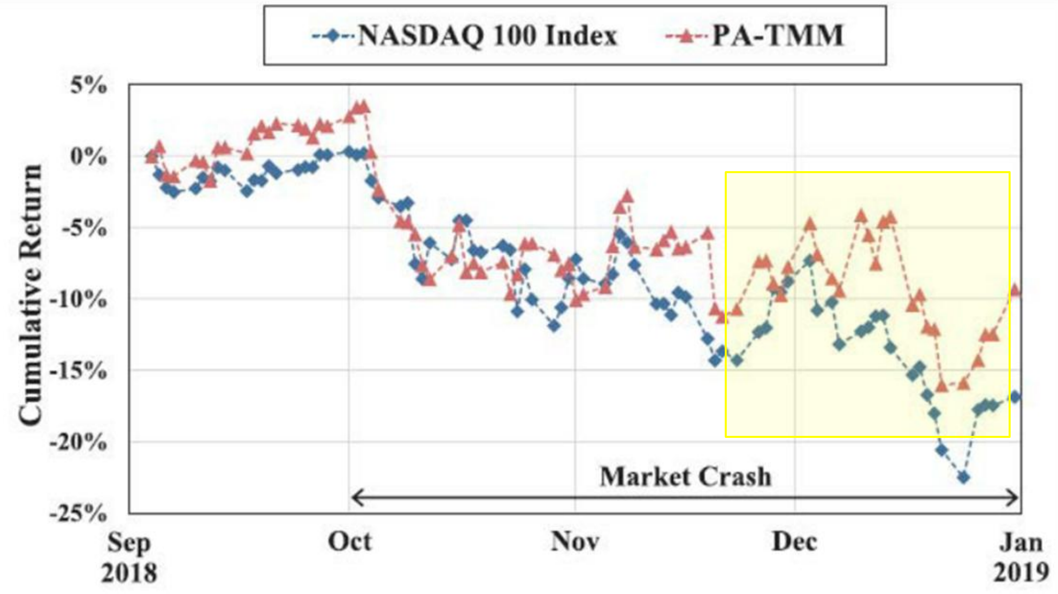


Fig. 4. Stress test during the market crash period.

p. 12

Experiments

Parameter Sensitivity Analysis

- Window Sizes T (N 12, S 16): Reduce ➡️ Overlook long-term trends, Increase ➡️ Introduce stale information
- Mutation Probability θ (N 0.25, S 0.25): Huge θ ➡️ Mistrust movement prompts
- Dimensions of d_n (N 256, S 384) and d_e (N 256, S 320)

  - n: node vector, e: edge vector

  - Excessively low or high dimensionality ➡️ Under-Fitting or Over-Fitting

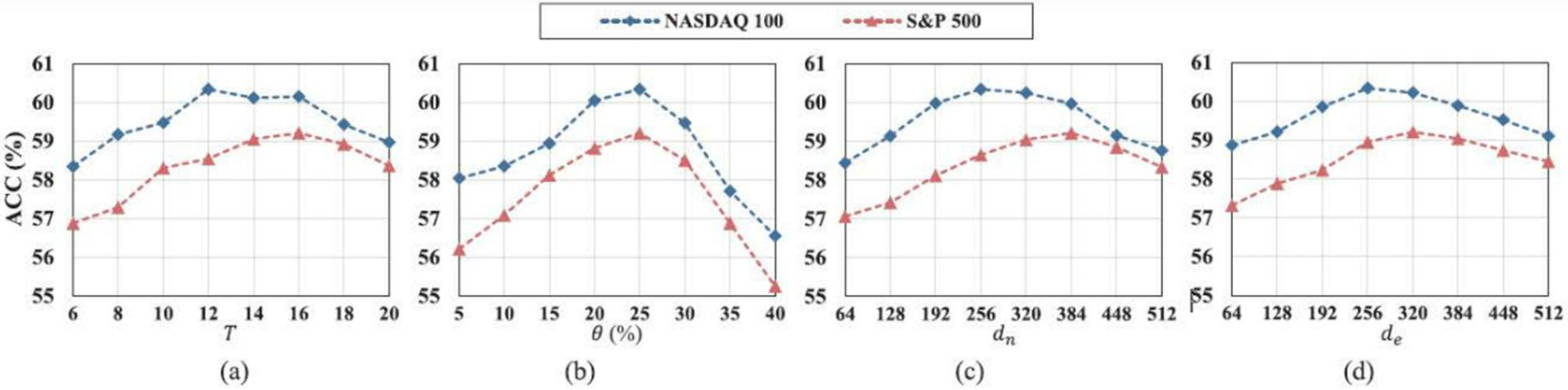


Fig. 5. Parameter sensitivity analysis with respect to the window size T, mutation probability θ, dimension d_n, and dimension d_e. (a) Window size. (b) Mutation probability. (c) Node representation. (d) Edge representation.

Experiments

Case Study on Exploring Stock Attention Networks

- On April 12, 2019, investigate the effectiveness of MPA with S&P500 dataset
  - News: Strong earnings reports from JPM and WFC
  - ➡️ Broadly positive market sentiment and Strong bullish signals
- Without MPA
  - Independently distributed attention
  - **Limited attention to the news from other stocks**
- With MPA
  - **Significant increase in cross-sector attention**

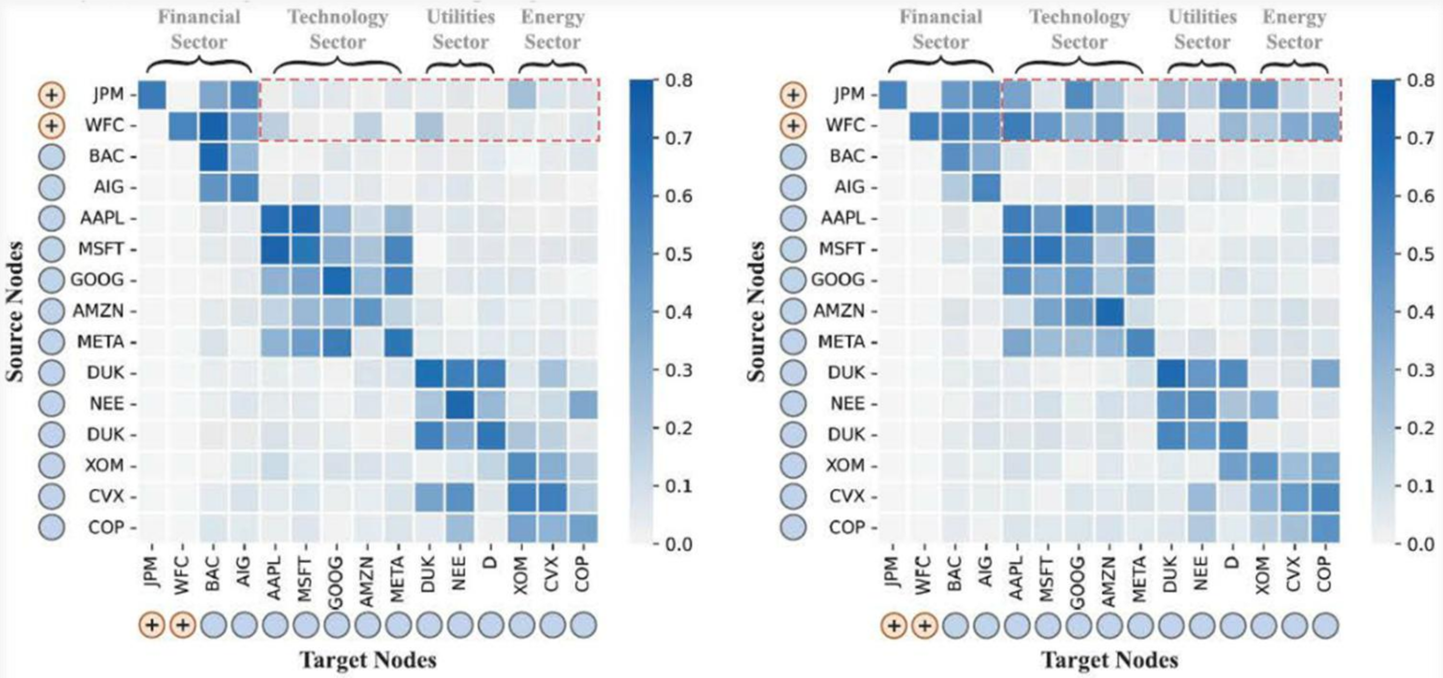


Fig. 6 (a): Without pretraining (w/o MPA) (b): After pretraining (PA-TMM)

Conclusion

- PA-TMM

  - Novel model for stock movement prediction and quantitative trading

  - **Address the long-tail distribution problem**

  - MPA (pretraining) + EQSamp (data augmentation) ➡️ **Enhance sensitivity to news**

  - Leverage news sentiment as prompts ➡️ **Capture cross-modal signals more effectively**

- Experimental results

  - Superior prediction performance in terms of ACC and MCC

  - Various comparative studies validate effectiveness, profitability, and robustness

- Future work

  - Explore lightweight attention mechanisms

  - Integrate other textual sources like financial reports, social media, geopolitical events, and regulations for to enhance the models’ understanding of stock market and prediction accuracy

End of Documents