# Task 3 and Task 4

Chaithra

January 2025

## 1 Task 3: Sector Level Modeling

### 1.1 Inter Sector Graph Construction

Here, we aim to learn how different sectors influence each other by modeling their latent relations.

Input: Embedding $\tau_G^i(S_q)$ of stocks belonging to sector .

Output: Sector Embedding per sector $\tau_i(\pi_c)$

The long term embedding $\tau_i^G(S)$ of all stocks belonging to sector $\pi_c$ is max pooled to get a sector embedding $Z_{\pi c}$.

$$\mathbf{z}_{\pi_\mathbf{c}} = \text{MaxPool}(\{\tau_i^G(S_q) \mid \forall s_q \in M_{\pi c}\}) \tag{1}$$

The set of sector embeddings

$Z_\pi = Z_{\pi_1}, Z_{\pi_2}, ...Z_{\pi_c}$ where c is the number of sectors. A fully connected graph is constructed $G_\pi = (Z_\pi, E_pi)$ .The intersection embeddings $\tau_i(\pi_c)$ of sector $\pi_c$ generated by GAT is given by

$$\tau_i(\pi_c) = GAT(G_\pi, \pi_c) \tag{2}$$

The procedure is illustrated in the Figure 1

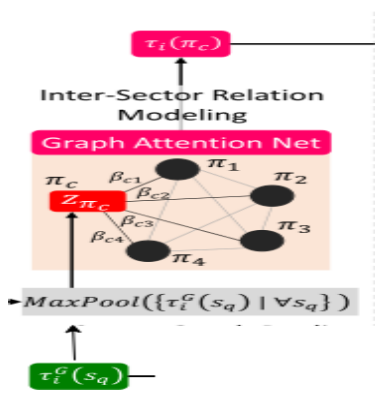


Figure 1: Sector Level Modeling

# 2 Task 2: Model Learning

### Embedding Fusion

The derived feature vectors, short-term embeddings $\tau_i^s(s_q)$, intra-sector embeddings $\tau_i^g(s_q)$, and inter-sector embeddings $\tau_i(\pi_c)$, are used. These features are combined via an embedding fusion layer to obtain the final feature vector $\tau_i^F(s_q)$, given by:

$$\tau_i^F(s_q) = ReLU\left([\tau_i^g(s_q) \| \tau_i^s(s_q) \| \tau_i(\pi_c)]W_f\right),$$

where stock $s_q$ belongs to sector $\pi_c$, $W_f$ is the learnable weight matrix, and ReLU is the activation function. In other words, the past $t$ weeks, i.e., from week $i-t$ to week $i-1$, is used to produce the final embedding vector $\tau_i^F(s_q)$ and to predict the corresponding daily return ratio.

## 2.1 Multi-Task Learning

To predict the profitability of ranking of stocks based on predicted return ratio and the movement of stocks is done. Pairwise ranking aware loss is used for ranking the stocks and cross entropy loss for stock movement.

The predictions of return ratio and movement for stock $s_q$, denoted by $\hat{y}_i^{return}(s_q)$ and $\hat{y}_i^{move}(s_q)$, can be performed by their respective task-specific layers, given by:

$$\hat{y}_i^{return}(s_q) = \mathbf{e}_1^\top \tau_i^F(s_q) + b_1, \hat{y}_i^{move}(s_q) = \phi\left(\mathbf{e}_2^\top \tau_i^F(s_q) + b_2\right) \tag{3}$$

where $\mathbf{e}_1, \mathbf{e}_2 \in R^d$ denote the hidden vectors of task-specific layers that project $\tau_i^F(s_q)$ into the prediction results of return ratio and binary movement, respectively. $\phi$ is the sigmoid function, and $b_1$ and $b_2$ are bias terms.

$$\mathcal{L}_{FinGAT} = (1-\delta)\mathcal{L}_{rank} + \delta\mathcal{L}_{move} + \lambda\|\Theta\|^2, \tag{4}$$

where $\mathcal{L}_{rank} = \sum_i \sum_{s_q} \sum_{s_k} \max\left(0, -\hat{\Delta} \times \Delta\right),$

$\hat{\Delta} = \left(\hat{y}_i^{return}(s_q) - \hat{y}_i^{return}(s_k)\right),$
$\Delta = (y_i^{return}(s_q) - y_i^{return}(s_k)),$
$\mathcal{L}_{move} = -\sum_i \sum_{s_q} y_i^{move} \log\left(\hat{y}_i^{move}(s_q)\right) + \left(1 - y_i^{move}\right) \log\left(1 - \hat{y}_i^{move}(s_q)\right).$

# 3 Ablation Study

- FinGAT (Full Model): using all components of the proposed FinGAT.
- w/o intra-sector graph attention (w/o intra): FinGAT without using embeddings derived from intra-sector graph attention network.

- w/o inter-sector graph attention (w/o inter): FinGAT without using embeddings obtained from inter-sec- tor graph attention network. mean square error loss (w/ MSE): replacing move- ment prediction loss (i.e., binary cross entropy loss) with regression loss

# 4 Hyper Parameters

No of Weeks 1,2,3,4=5,10,15,20 days
GRU and GAT embedding Dimension 8,16,32,64
Balancing Parameter delta =0,0.0001,0.001,0.01,0.1,1
Learning rate=0.0005,0.001,0.005
Batch Size=32,64,128

# 5 Evaluation Metric

Mean Reciprocal Rank (MRR @K) for K=5,10,20
Precision @K
Accuracy