Machine Learning with Graphs (MLG)

Signed Link Prediction

A Case of Neural Network-based Link Prediction

Cheng-Te Li (李政德)

Institute of Data Science

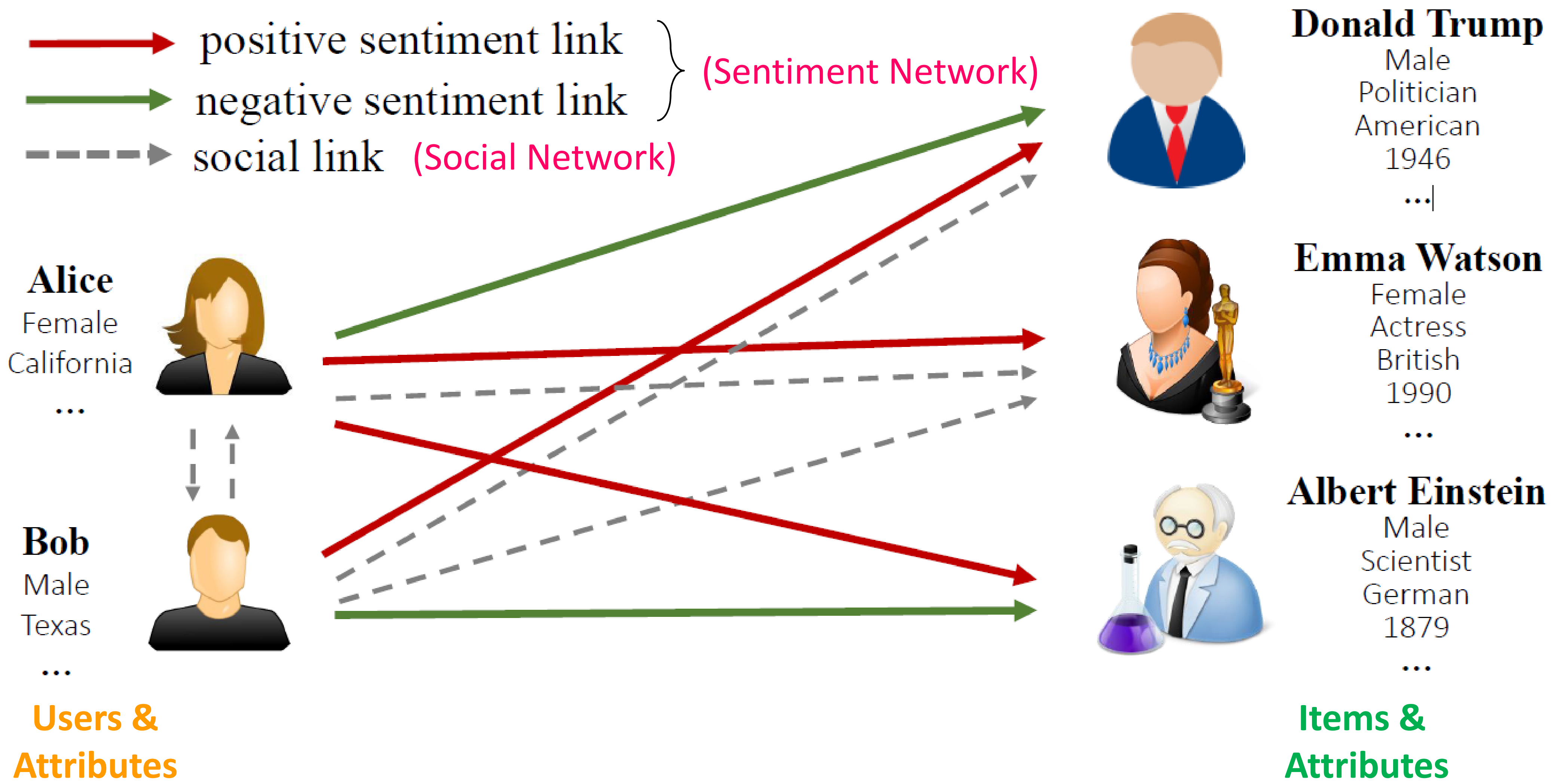
National Cheng Kung University

chengte@mail.ncku.edu.tw

Sentiment Links as Signed Links

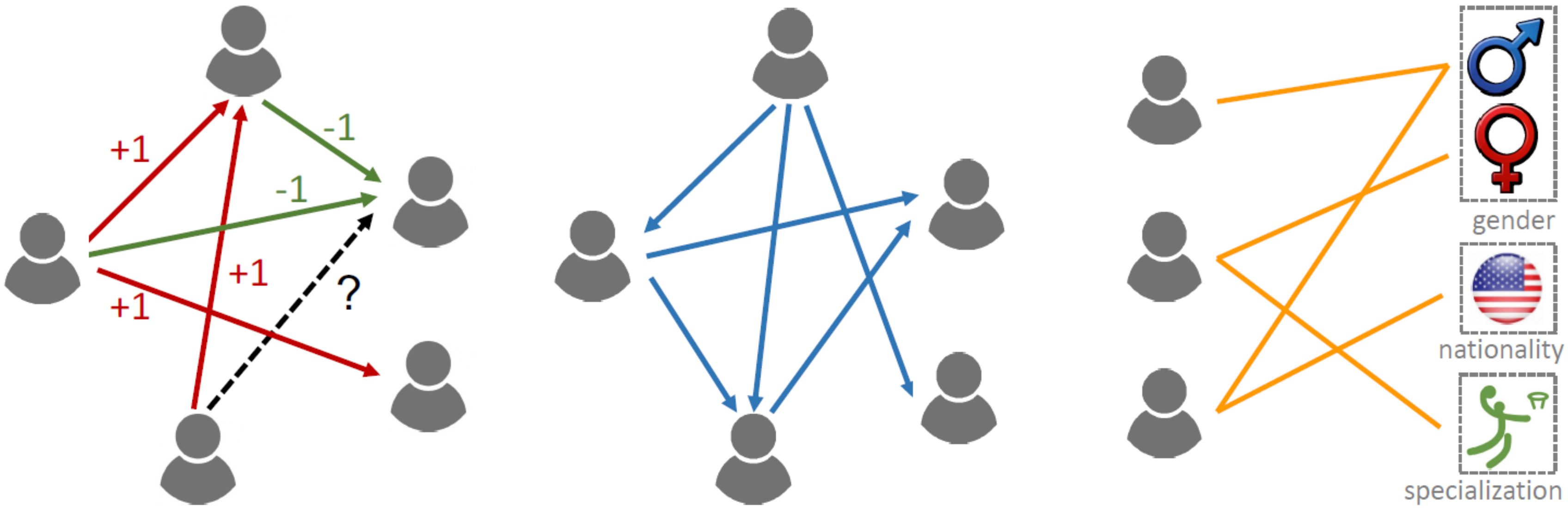
Social Trust: Users → Celebrities

Recommender Systems: Users → Items



Sentiment Link Prediction on Heterogeneous Graphs

Given a multi-relational graph, consisting of sentiment network G_s, social network G_r, and profile network G_p, the goal is to predict the sentiment of unobserved links between users in G_s



(a) Sentiment network (b) Social network (c) Profile network

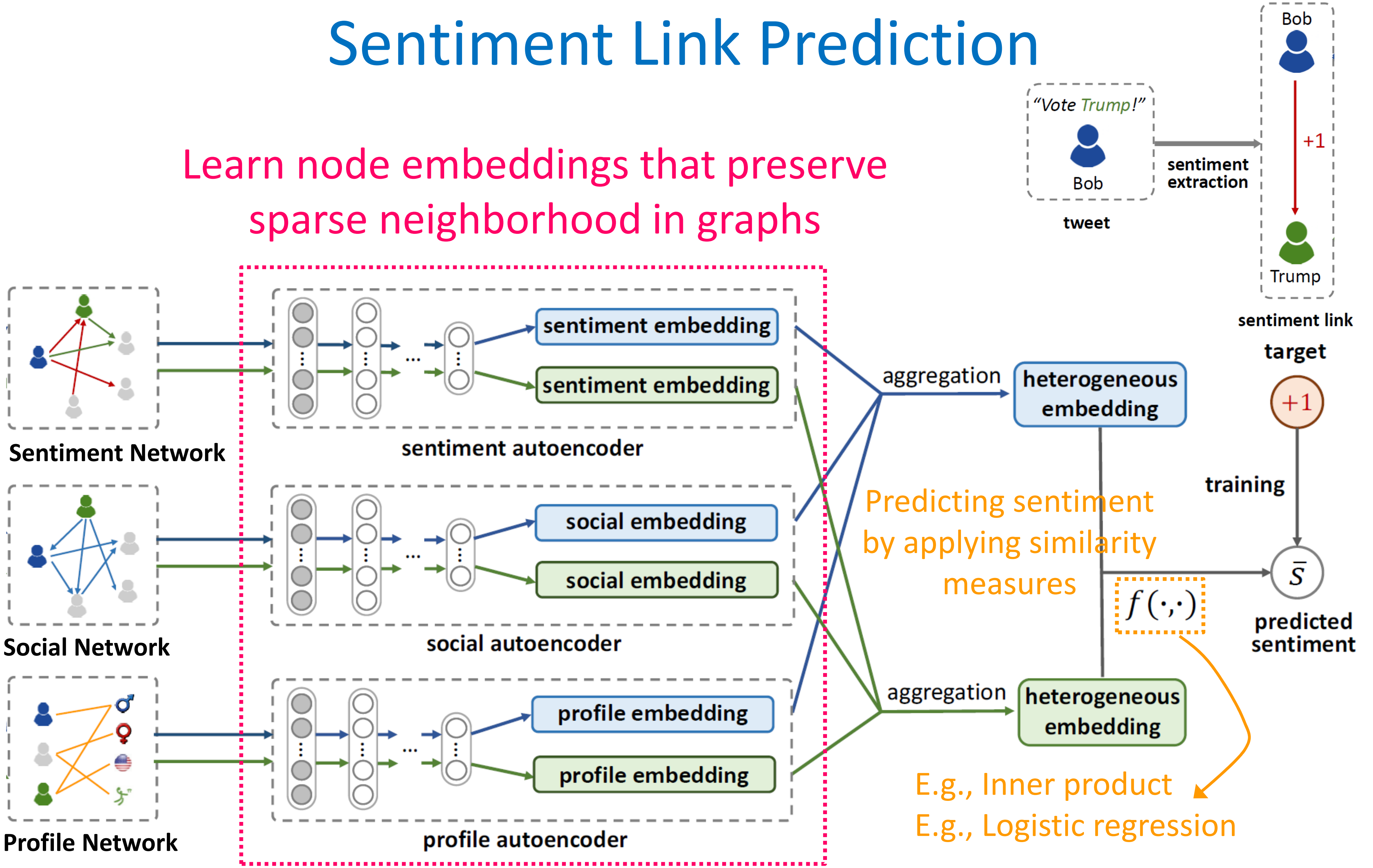
Normal Users and Celebrity Users

Normal Users and Celebrity Users

Normal Users and Celebrity Users to their Attributes

Autoencoder-based Graph Learning for Sentiment Link Prediction

Learn node embeddings that preserve sparse neighborhood in graphs



Sentiment Autoencoder

s_{ij}: user i's sentiment on j

Adjacency sentiment vector as initial feature: $\mathbf{x}_i = \{s_{ij} \mid j \in V\} \cup \{s_{ji} \mid j \in V\}$ depicting the incoming and outgoing sentiment info of user i

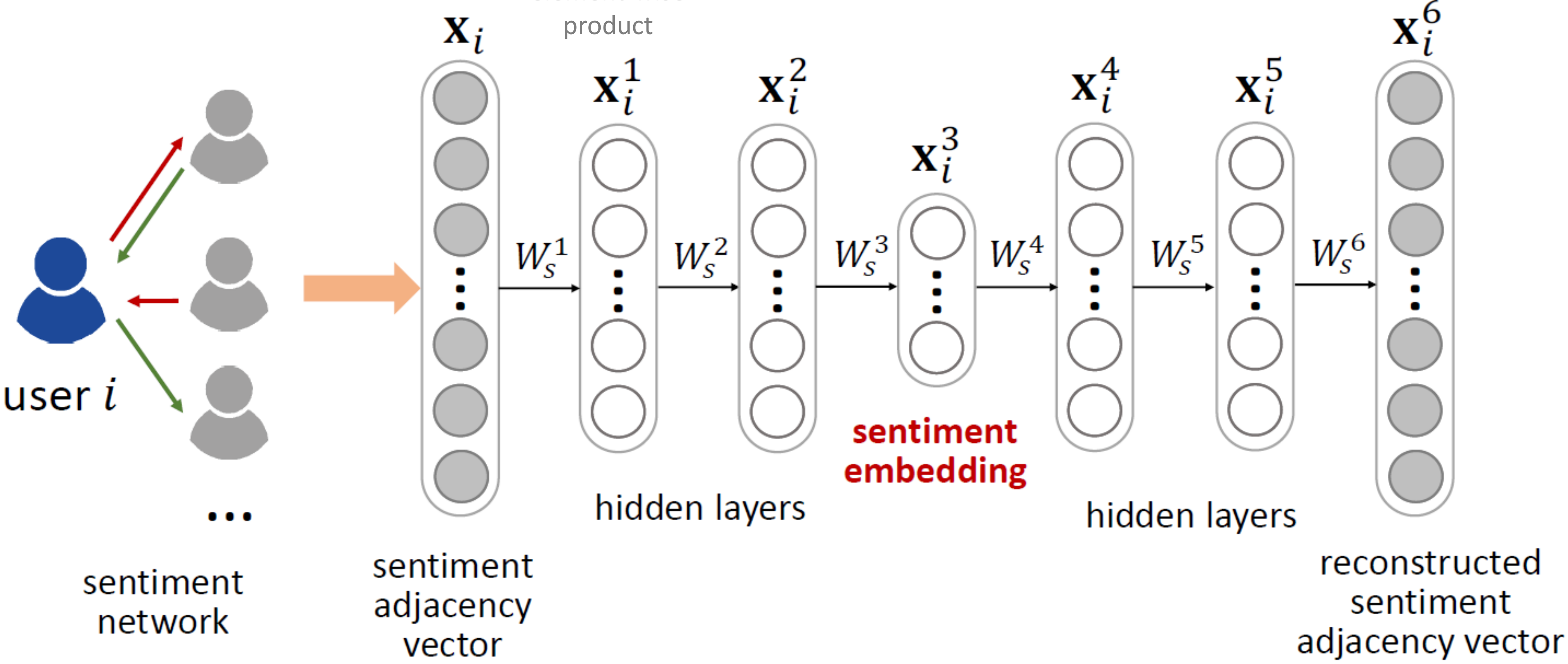
$$\mathbf{x}_i^k = \sigma\left(\mathbf{W}_s^k \mathbf{x}_i^{k-1} + \mathbf{b}_s^k\right),\ k = 1, 2, \ldots, K_s$$

$$\mathcal{L}_s = \sum\nolimits_{i \in V} \left\|(\mathbf{x}_i - \mathbf{x}_i') \odot \mathbf{l}_i\right\|_2^2$$

element-wise product

sentiment reconstruction weight vector

$$l_{i,j} = \begin{cases} \alpha > 1, & if\ s_{ij} = \pm 1 \\ 1, & if\ s_{ij} = 0. \end{cases}$$



Embedding Aggregation & Sentiment Link Prediction

- Embedding aggregation $\mathbf{e}_i = g(\cdot,\cdot,\cdot)$
  - Summation: $\mathbf{e}_i = \hat{\mathbf{x}}_i + \hat{\mathbf{y}}_i + \hat{\mathbf{z}}_i$
  - Max pooling: $\mathbf{e}_i = element.wise.max(\hat{\mathbf{x}}_i, \hat{\mathbf{y}}_i, \hat{\mathbf{z}}_i)$
  - Concatenation: $\mathbf{e}_i = \langle \hat{\mathbf{x}}_i, \hat{\mathbf{y}}_i, \hat{\mathbf{z}}_i \rangle$
- Sentiment link prediction $\bar{s}_{ij} = f(i,j)$
  - Inner product: $\bar{s}_{ij} = \mathbf{e}_i^{\mathrm{T}} \mathbf{e}_j + b$
  - Euclidean distance: $\bar{s}_{ij} = -\left\|\mathbf{e}_i - \mathbf{e}_j\right\|_2 + b$
  - Logistic regression: $\bar{s}_{ij} = \mathbf{W}^{\mathrm{T}} \langle \mathbf{e}_i, \mathbf{e}_j \rangle + b$
- Loss function $$\mathcal{L} = \sum_{i \in V} \|(\mathbf{x}_i - \mathbf{x}'_i) \odot \mathbf{l}_i\|_2^2 + \lambda_1 \sum_{i \in V} \|(\mathbf{y}_i - \mathbf{y}'_i) \odot \mathbf{m}_i\|_2^2 + \lambda_2 \sum_{i \in V} \|(\mathbf{z}_i - \mathbf{z}'_i) \odot \mathbf{n}_i\|_2^2 + \lambda_3 \sum_{s_{ij} = \pm 1} \left(f(\mathbf{e}_i, \mathbf{e}_j) - s_{ij}\right)^2 + \lambda_4 \mathcal{L}_{reg},$$