All-Hands Meeting III
2023 Winter

BertGCN: Transductive Text Classification by Combining GCN and BERT

Presenter: Minseo Yoon
(cooki0615@korea.ac.kr)
Slide Credit: Prof. Hyunwoo J. Kim

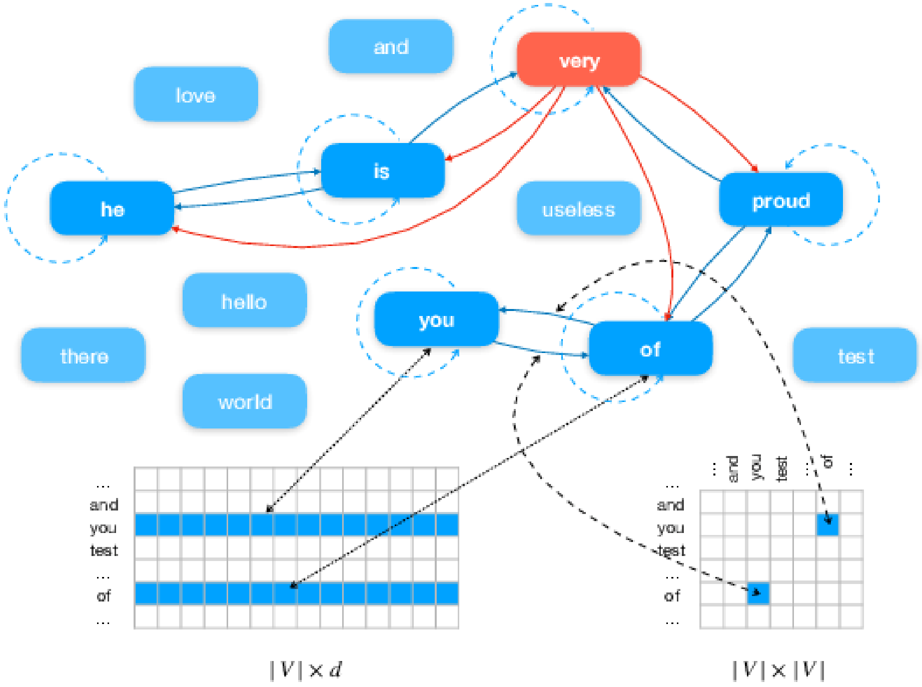
Introduction

BertGCN: Transductive Text Classification by Combining GCN and BERT

Yuxiao Lin♠, Yuxian Meng♣, Xiaofei Sun♣
Qinghong Han♣, Kun Kuang♠, Jiwei Li♠♣ and Fei Wu♠
♠College of Computer Science and Technology, Zhejiang University
♣ ShannonAI
{yuxiaolinling, kunkuang, jiwei_li, wufei}@zju.edu.cn
{yuxian_meng, xiaofei_sun, qinghong_han}@shannonai.com

Introduction

- Task
  - Natural Language Processing
  - Text Classification

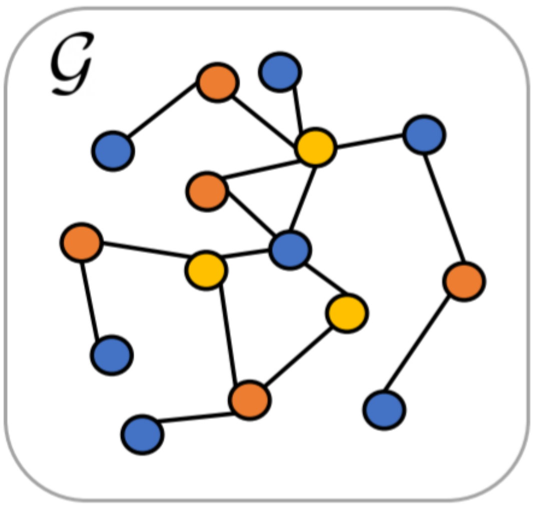


Introduction

- Idea
  - Transductive learning
    - Both labeled and unlabeled examples in the training process
  - Node: Text units (words, documents)
  - Edge: Semantic Similarity between nodes
  - Graph Neural Networks
    - Depend on its neighbors – robust to outliers
    - Supervised labels -> Unlabeled data

Introduction

- BertGCN
  - Large-scale pretrained model -> Transductive learning
  - Heterogeneous graph
  - Initialized with pretrained BERT representations



Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

Type mapping function

$f_v : \mathcal{V} \rightarrow \mathcal{T}^v$ $f_e : \mathcal{E} \rightarrow \mathcal{T}^e$

Slide credit: Jinyoung Park

Method

- BertGCN
  - Initialized with pretrained BERT representations
  - Using BERT-style Model (BERT, RoBERTa) -> Inputs
  - Iteratively updated based on GCN
  - Final Representation -> Softmax

Method

- Heterogeneous Graph
  - Word-document edges
  - Word-word edges
  - Term frequency-inverse document frequency (TF-IDF)
  - Positive point-wise mutual information (PPMI)

$$A_{i,j} = \begin{cases} \mathrm{PPMI}(i,j), & i,j \text{ are words and } i \neq j \\ \text{TF-IDF}(i,j), & i \text{ is document, } j \text{ is word} \\ 1, & i = j \\ 0, & \text{otherwise} \end{cases}$$

Method

- Heterogeneous Graph

$$A_{i,j} = \begin{cases} \mathrm{PPMI}(i,j), & i,j \text{ are words and } i \neq j \\ \text{TF-IDF}(i,j), & i \text{ is document, } j \text{ is word} \\ 1, & i = j \\ 0, & \text{otherwise} \end{cases}$$

$$\mathrm{PMI}(x,y) = \log_2 \frac{P(x,y)}{P(x)P(y)} = \log_2 \frac{\frac{C(x,y)}{N}}{\frac{C(x)}{N}\frac{C(y)}{N}} = \log_2 \frac{C(x,y)\cdot N}{C(x)C(y)}$$

$$TF(t,d) = \frac{number\ of\ times\ t\ appears\ in\ d}{total\ number\ of\ terms\ in\ d}$$

$$IDF(t) = log\frac{N}{1+df}$$

$$TF-IDF(t,d) = TF(t,d) * IDF(t)$$

Method

- Feature Matrix

$$X = \begin{pmatrix} X_{\text{doc}} \\ 0 \end{pmatrix}_{(n_{\text{doc}}+n_{\text{word}})\times d}$$

  - Document node embeddings $X_{\text{doc}} \in \mathbb{R}^{n_{\text{doc}}\times d}$
  - d is embedding dimensionality

Method

- GCN layer

$$L^{(i)} = \rho(\tilde{A}L^{(i-1)}W^{(i)})$$

  - Feed X into a GCN model
  - ρ : activation function
  - $\tilde{A}$: normalized adjacency matrix
  - $W^{(i)} \in \mathbb{R}^{d_{i-1} \times d_i}$: weight matrix
  - $L^{(0)} = X$
  - $Z_{\text{GCN}} = \text{softmax}(g(X, A))$ g : GCN model

Method

- Interpolating BERT and GCN Predictions
  - Optimizing BertGCN
  - With an auxiliary classifier that directly operates on BERT embeddings
  - Leads to faster convergence and better performances

$$Z_{\text{BERT}} = \text{softmax}(WX)$$

$$Z = \lambda Z_{\text{GCN}} + (1 - \lambda) Z_{\text{BERT}}$$

Method

- Memory Bank
  - Full-batch method: memory limitation
  - M : memory bank that tracks input features for all document nodes
  - Compute all document embeddings using current BERT module and store them in memory bank at each epoch
  - $B = \{b_0, b_1 ... b_n\}$: index set containing mini batch from both labeled and unlabeled document nodes
  - Compute M_B and update M
  - M is considered constant except the records in B

Method

- Memory Bank
  - Embeddings in the memory bank are computed using the BERT module at different steps in an epoch and are thus inconsistent
  - Small learning rate -> Take more time
  - Fine-tune and use it to initialize the BERT parameters

Experiments

- Dataset
  - R8: Reuters dataset with 8 categories
  - 5,485 training and 2,189 test documents

- Experiment 1
  - PPMI → Cosine Similarity > 0.9
  - PPMI → Cosine Similarity > 0.8
  - PPMI → Co-occurrence Matrix
  - TF-IDF → Jaccard Similarity
  - TF-IDF → JS-IDF

Experiments

- Experiment 2

$$Z = \lambda Z_{\text{GCN}} + (1 - \lambda) Z_{\text{BERT}}$$

$$\downarrow$$

$$Z = \alpha Z_{GCN} \; + \; \beta Z_{GAT} \; + \; (1 - \alpha - \beta) Z_{BERT}$$

Experiments

- Experiment 1

Model	Accuracy
Original (PPMI, TF-IDF)	97.9
Cosine Similarity $>$ 0.9 TF-IDF	95.7
Cosine Similarity $>$ 0.8 TF-IDF	92.8
Co-occurrence Matrix TF-IDF	53.5
PPMI Jaccard Similarity	94.7
PPMI JS-IDF	96.2

$$A_{i,j} = \begin{cases} \mathrm{PPMI}(i,j), & i,j \text{ are words and } i \neq j \\ \text{TF-IDF}(i,j), & i \text{ is document, } j \text{ is word} \\ 1, & i = j \\ 0, & \text{otherwise} \end{cases}$$

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

Experiments

- Experiment 2

Model	Accuracy
Original ($\alpha = 0.7,\ \beta = 0$)	97.9
$\alpha = 0.4,\ \beta = 0.3$	97.5
$\alpha = 0.4,\ \beta = 0.4$	95.7
$\alpha = 0.3,\ \beta = 0.4$	96.0
$\alpha = 0.5,\ \beta = 0.2$	95.5
$\alpha = 0.7,\ \beta = 0.1$	96.9
$\alpha = 0.6,\ \beta = 0.1$	97.4

$$Z = \alpha Z_{GCN} + \beta Z_{GAT} + (1 - \alpha - \beta) Z_{BERT}$$

Questions?