NODE2VEC: SCALABLE FEATURE LEARNING FOR NETWORKS

Srikruth Varma
srikruthvarma@gmail.com

INTRODUCTION TO NODE2VEC

Problem Statement: Traditional methods fail to capture both local and global structural patterns in networks.

Objective: node2vec learns low dimensional embeddings for nodes that represent their roles and communities.

BIASED RANDOM WALKS

->Explanation: node2vec generates random walks from nodes to learn node embeddings.

->Parameters: The parameters p and q control the type of random walk:

->BFS-like walks: Stay local (p).

->DFS-like walks: Explore distant nodes (q).

The probability of transitioning from node v to node x during a random walk:

$$P(c_i = x \mid c_{i-1} = v) = \frac{\pi_{vx} w_{vx}}{Z}$$

The bias factor π_{vx}:

$$\pi_{vx} = \begin{cases} \frac{1}{p}, & \text{if } d(t,x) = 0 \text{ (return to previous node)} \\ 1, & \text{if } d(t,x) = 1 \text{ (neighboring node)} \\ \frac{1}{q}, & \text{if } d(t,x) = 2 \text{ (further away)} \end{cases}$$

EMBEDDING LEARNING

->Word2Vec Analogy: Random walks are treated as sentences, and nodes are treated like words.

->Goal: The Skip-Gram model predicts nearby nodes in a random walk.

The objective function for learning node embeddings:

$$\max_f \sum_{u \in V} \sum_{n \in N_S(u)} \log P(n \mid u; f)$$

Probability using the Softmax function:

$$P(n \mid u) = \frac{e^{f(n) \cdot f(u)}}{\sum_{v \in V} e^{f(v) \cdot f(u)}}$$

KEY FEATURES OF NODE2VEC

->Flexible Sampling Strategy: Balances between local and global node neighborhoods.

->Scalable: node2vec can handle large graphs efficiently.

->Supports Multiple Tasks: Embeddings can be used for classification, link prediction, and clustering.

-

STRENGTHS AND WEAKNESSES

->Strengths:

-Flexibility in exploring the graph (local/global)

-High performance in tasks like classification and link prediction.

->Weaknesses:

-Hyperparameter tuning for p and q is required.

-Performance depends on the structure of the graph.

Modularity for Community Detection

The modularity function used to evaluate the quality of detected communities:

$$Q = \frac{1}{2m} \sum_{i,j} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j)$$

APPLICATIONS OF NODE2VEC

->Node Classification: Predicting labels of nodes (e.g.,social networks).

->Link Prediction: Predicting future connections between nodes.

->Community Detection: Identifying clusters of nodes (e.g., users with similar interests).

CONCLUSION

->Summary: node2vec offers a flexible and scalable way to learn node embeddings.

->Applications: Useful in various tasks such as node classification, link prediction, and clustering.

->Final Thought: Potential for future research in dynamic networks and deep learning.

Thank You