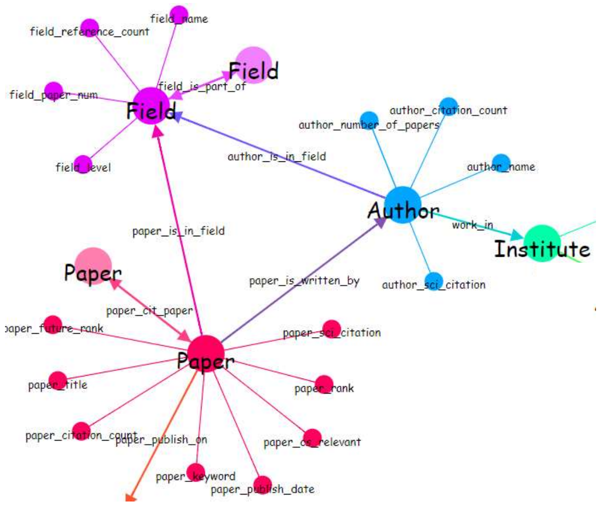
Predicting the missing links in an academic network——a Novel Approach

Chengyang Wu {wuchengyang@sjtu.edu.cn}
School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University

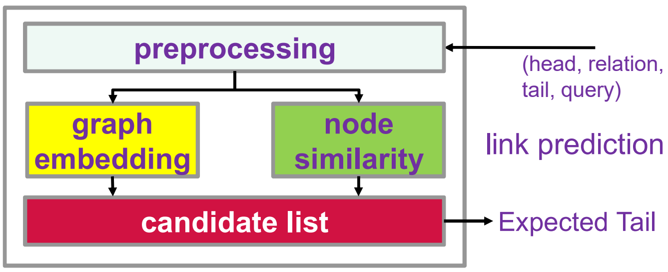
上海交通大学 SHANGHAI JIAO TONG UNIVERSITY

Background

Knowledge Graph



Link Prediction



Terminologies

Embedding Size d: Dimension of feature vectors for each entity in the graph.
Common neighbors: A common neighbor of nodes ***u*** and ***v*** has an edge connecting to ***u*** and one connecting to ***v***.

Task

Predict the missing links in an academic network.

Challenges

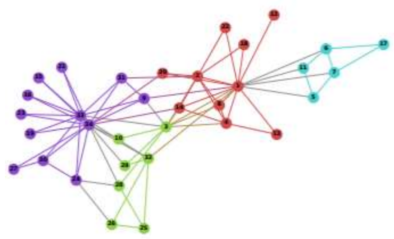
Various Relations: multiple categories of entities and queries.
Anonymity of data: impossible to employ data mining methods for sentiment analysis.
Multiple possible answers: hard to make a rank on the predicted results

paper_is_in_field	
author_is_in_field	
paper_cit_paper	
field_is_part_of	
paper_is_written_by	
work_in	5
paper_publish_on	

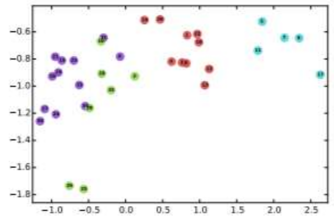
B78C8E71	8FA32D46	78D63DD8
78D63DD8	AEAC6A10	01790CC4
3617ABDD	AEAC6A10	78D63DD8
3617ABDD	E1CAE012	AEAC6A10
5587E5BC	EF6321A6	0233FA25
0FD5D361	DAD87857	AEAC6A10
3617ABDD	AEAC6A10	6F17B436
01790CC4	9057B164	B78C8E71
0233FA25	C5EB540D	8FA32D46
E1CAE012	AEAC6A10	E977E0A6

Our Approach

Deepwalk: Online Learning of Social Representations

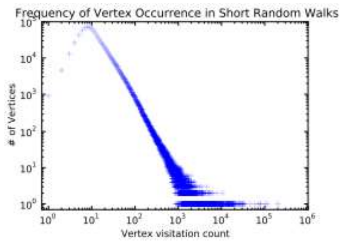


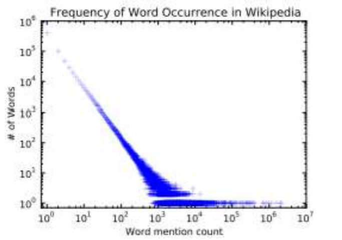
(a) Input: Karate Graph



(b) Output: Representation

Random walks: W_{v_i} denotes a stochastic process rooted at vertex v_i. Vertices are chosen at random from the neighbors of vertex v_k.
Skip Gram: Treat the generated walk as a sentence and maximize the occurrence probability among the words that appear within a window, w, in a sentence.
Optimization: Stochastic Gradient Descent is used to optimize these parameters. The derivatives are estimated using the back-propagation algorithm.





Zipf's law: The frequency that vertices appear in short random walks follows a similar distribution to the word frequency in natural language. The techniques used to model natural language can be re-purposed to model community structure in networks.
Language Modeling: Estimate the likelihood of a specific sequence of words appearing in a corpus.
Given $W_1^n = (\omega_0, \omega_1, \cdots \omega_n)$ where $\omega_i \in \gamma$ (γ is the vocabulary), maximize the $\Pr(\omega_n | \omega_0, \omega_1, \cdots, \omega_{n-1})$ over all the training corpus.
Random walks can be thought of short sentences and phrases in a special language. The direct analog is to estimate the likelihood of observing vertex v_i given all the previous vertices visited so far in the random walk. $\Pr(v_i | v_1, v_2, \cdots, v_{i-1})$

Experiments

Setup

Dataset: Knowledge Graph from Acemap (149.6K triplets)
Metrics: Mean Average Precision @3 (MAP@3):

$$MAP@3 = \frac{1}{|U|} \sum_{u=1}^{|U|} \sum_{k=1}^{\min(3,n)} P(k)$$

where |U| is the number of queries, P(k) is the precision at cutoff k, n is the number of predicted tail.
Preprocessing: Transform the unique strings into unique numbers so as to construct edge lists and adjacent matrixes. Correspondingly, change the training dataset and test queries into triplets of numbers.

Results

0.34719 MAP@3 on the entire test dataset