Named Entity Disambiguation in Enterprise Knowledge Graph

Project Proposal

VE444 Networks - Group 7

Name	Stud. ID
Zhou Zhanpeng	*518021910594*
Tian Yuchuan	*518370910040*
Jin Haoxiang	*518370910215*

Abstract

Enterprise knowledge graph leverages collections of interlinked entities to represent enterprise basic information and the investment relationship between enterprises and investors, which could be a great weapon in investment analysis. However, it is not uncommon to see two different executives or investors with the same name in the enterprise KG. This brings great hidden troubles to the precision of investment analysis. Therefore, to tackle this **Named Entity Disambiguation (NED)** problem, we propose an automatic detection system involving technologies of **Graph Neural Network (GNN)**, **community detection** and **link prediction**. In this project, we will firstly build a national-wide enterprise KG which incorporates information about 10,000,000 enterprises in China, then based on the pre-built KG, combining the technologies of GNN and community detection, build an automatic system to detect different entities with the same name.

Project Description

Knowledge graph (KG) in the enterprise field provides an effective solution for solving financial problems. In many cases, financial institutions need to understand the various relationships between enterprises and related parties, and gain insight into enterprise risk transmission, abnormal transactions and other information. And Knowledge graph describes concepts, entities and their relationships in the objective world in a structured form, expresses the information of the Internet in a form closer to the cognitive world of human beings, and provides an ability to better organize, manage and understand massive information on the Internet. So, constructing knowledge graph which contains the basic information of the enterprise and the investment relationship information of the enterprise could help us solve many reasoning problems existing in the financial field.

However, when constructing an enterprise knowledge graph based on raw data, ambiguity is often encountered. In raw data, it is not uncommon to have two different entities with the same name. For example, we all know that Lei Jun is the actual controller of Xiaomi Technology Co., LTD, but here is another person named Lei Jun in our data who is actual controller of Shenzhen Mizuan jewelry Co. LTD. By common sense, we can easily distinguish the two people with the same name because the two companies' main businesses are quite different. But, in real-world data, we tend to have thousands of name ambiguities, and it's unrealistic to manipulate them one by one. Therefore, our project objective is to propose an automatic detection system to tackle this **Named Entity Disambiguation (NED)** problem.

Inside our automatic detection system, there are two major components corresponding to two different technologies, GNN and community detection algorithm.

First, community detection algorithm mainly leverages the information of investment relationships. In social network analysis, usually, interpersonal ties can be grouped into **strong ties** and **weak ties**, and for strong ties, it is more easily to form a strong-connected group within a group, which could be visualized as Fig. 1. Therefore, by **Strongly Connected Components Algorithm (SCCA)**, which is a kind of community detection algorithm, we can divided all the investment relationships within enterprise KG into strong ties and weak ties to find the internal strong-connected group. Intuitively, if two companies have strong investment relationships and their investors or actual controllers have the same name, then it is likely for them to be the same person. With this understanding, we can use SCCA to recognize two different entities with the same name in the same strong-connected group as the same entity.

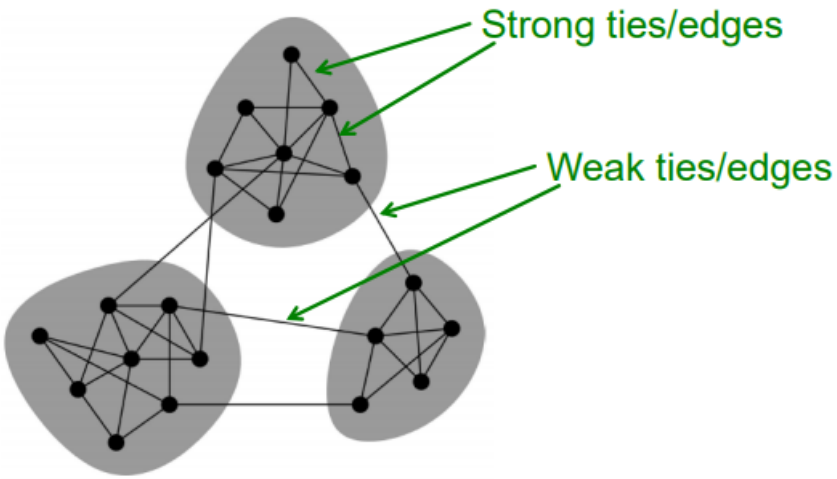

Fig.1 Strong and Weak Ties

However, considering the accuracy of the algorithm, we introduce another component, **Graph Neural Network (GNN)**, which is mainly used to learn other information in the graph, including the regional information of the enterprise and industry information. Usually, GNN is applied to entity linking in KG. If we can link two different entities with same name to indicate they are actually same entity, then we could embed the technology of GNN to our NED problem, using GNN to predict the link of two different entities with the same name.

In addtion to solving NED problem, in our project, we will also propose an efficient algorithm to create massive graph in commonly-used graph database, e.g. Neo4j and etc. Also, we draw on the concept of broad map in Amazon's product knowledge graph [1], which greatly improves the scalability of our enterprise knowledge graph. The massive graph is used to train and test our automatic detection algorithm.

Combining above two components and training with the pre-built massive enterprise knowledge graph, our automatic detection system could be able to distinguish different entities of the same name with high precisions and recalls on our test dataset. Here, we define these two quantities as

$$precision = \frac{TP}{TP + FP}$$
$$recall = \frac{TP}{TP + FN}$$

where TP refers to the number of pairs of same entities which are detected by the system as "pair of same entities", FP refers to the number of pairs of different entities with same names are detected by the system as "pair of same entities", and FN refers to the number of pairs of different entities with same names are detected by the system as "pair of different entities".

In fact, entity disambiguation is a common difficulty in the construction of knowledge graph and also one of the major research directions in the field of KG. We hope that our project is not only applicable to the knowledge graph of enterprises, but also has strong generalization ability and provide a feasible method for entity disambiguation problems of other types of KG.

Related work

Different data result in different approaches to disambiguation. Many papers look at extracting data from language as the first step. [4] and [10] prefer to sort and extract data first, performing data mining first via Bag of Words (BOW) and generative model in order to waive out noises; [8] uses a Category2Vec joint learning model to evaluate sematic similarities; [2] and [9] select a related candidate pool first by similar phrases in sentences. This step is unnecessary in the case of our research, as all data are extracted and clearly labeled. In terms of disambiguation methods, [2] applies a Support Vector Machine ranker to differentiate name entries; [9] ranks as well, but it proposes its own Fine-grained ranking function; [5] uses a Deep Neural Network; and use SCSNED to deal with disambiguation. Those approaches could be effective in addressing a large pool of irrelevant semantic information, but is wasteful of time and resources in tackling data of commercial enterprises that have explicit linkages. On the other hand, Graph Neural Networks, which we are applying in this project, could be an effective and efficient approach in addressing ambiguation.

As for the topic, most literatures ([2] [3] [4] [6] [8] [10]) focus on the entry disambiguation of the names of people on Wikipedia, and their main focus are biased towards language processing, namely, extracting valuable data from sentences and phrases. A lack of focus on commercial enterprise information manifests.

Data Set

The data for training and testing our system all comes from The National Enterprise Credit Information Publicity System (国家企业信用公示系统). The National Enterprise Credit Information Publicity System provides information reporting, publicity and inquiry services for national enterprises, farmers' professional cooperatives, individual industrial and commercial households and other market entities. According to the Regulations of the People's Republic of China on The Disclosure of Government Information and the Provisional Regulations on Enterprise Information Disclosure, the data we collect from the system are completely legal.

We divide the data collected from The National Enterprise Credit Information Publicity System into four tables, namely, the table of investment information, the table of enterprise basic information, the table of shareholder information and the table of industrial information. Both the table of investment information and the table of shareholder information contain the

information of the investment relationship between enterprises and investors.

However, due to the technical limitations, most of the collected data are very sparse. After simple processing, there are still many quality problems in the information. Therefore, during the experiment, we will artificially select the information with better quality for the training and testing of the system.

Project Plan

1. Create an algorithm for generating massive graph in Neo4j (Commonly-used graph database) and use it to build the enterprise knowledge graph for later training and testing.
2. Implement and debug Strongly Connected Components Algorithm.
3. Implement and test different kinds of model of Graph Neural Network.
4. Demonstrate our automatic detection system with pre-built knowledge graph.

Expected Results

1. An efficient algorithm for constructing graph with ten million nodes and relationships.
2. An automatic detection system which could distinguish different entities of same names with state-of-the-art precisions and recalls.

Task Assignment

Name	Task
Zhou Zhanpeng	Create algorithm for constructing graph; Design and test different models of GNN;
Tian Yuchuan	Implement and debug Strongly Connected Components Algorithm;
Jin Haoxiang	Label the test dataset and measure the performance of system on test dataset;

Other tasks like **report writting** and **poster designing** will be evenly distributed to group members.

Reference

[1] Dong, Xin Luna, Xiang He, Andrey Kan, Xian Li, Yan Liang, and others. 2020. “AutoKnow: Self-Driving Knowledge Collection for Products of Thousands of Types,” in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (New York, NY, USA: ACM)

[2] Mark Dredze, Paul McNamee, Delip Rao, Adam Gerber, and Tim Finin. Entity Disambiguation for Knowledge Base Population. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 277-285, August 2010.

[3] Silviu Cucerzan. Large-Scale Named Entity Disambiguation Based on Wikipedia Data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 708-716, June 2007.

[4] Xianpei Han, Jun Zhao. Named Entity Disambiguation by Leveraging Wikipedia Semantic Knowledge. In *CIKM*, pages 1-10, November 2009.

[5] Zhengyan He, Shujie Liu, Mu Li, Ming Zhou, Longkai Zhang, Houfeng Wang. Learning Entity Representation for Entity Disambiguation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 30-34, August 2013.

[6] Andrew Chisholm, Ben Hachey. Entity Disambiguation with Web Links. In *Transactions of the Association for Computational Linguistics*, volume 3, pages 145-156, October 2014.

[7] Hongzhao Huang, Larry Heck, Heng Ji. Leveraging Deep Neural Networks and Knowledge Graphs for Entity Disambiguation. Pages 1-10, April 2015.

[8] Ganggao Zhu, Carlos Iglesias. Exploiting Semantic Similarity for Named Entity Disambiguation in Knowledge Graphs. In *Expert Systems with Applications*, pages 8-24, February 2018.

[9] Falk Brauer, Michael Huber, Gregor Hackenbroich, Ulf Leser, Felix Naumann, Wojciech Barczynski. Graph-based Concept Identification and Disambiguation for Enterprise Search. In *Proceedings of the 19th International Conference on World Wide Web*, pages 171-180, April 2010.

[10] Yang Li, Shulong Tan, Huan Sun, Jiawei Han, Dan Roth, Xifeng Yan. Entity Disambiguation with Linkless Knowledge Bases. In *Proceedings of the 25th International Conference on World Wide Web*, pages 1261-1270, April 2016.
