Distributed Computing Project Report

G. Savitha, 20150109 Instructor name: Dr. Rajendra Prasath

Research Topic: Reasoning with Knowledge

Paper 1: Scalable Distributed Semantic Network for knowledge management in cyber physical system

Authors: Shengli Song, Yishuai Lin, Bin Guo,Qiang Di , Rong Lv
Year: 2017
Journal: Journal of Parallel and Distributed Computing

The accelerated development of emerging technologies in the internet and the digital physical frameworks yield enormous mass of information sources. Data sources can contain complementary or semantically equivalent information stored under different
Formats. This can influence semantics and meaning. This paper proposes a new scalable model, named **Distributed Semantic Network (DSN)**, for heterogeneous data representation and can extract more semantic information from different data sources.

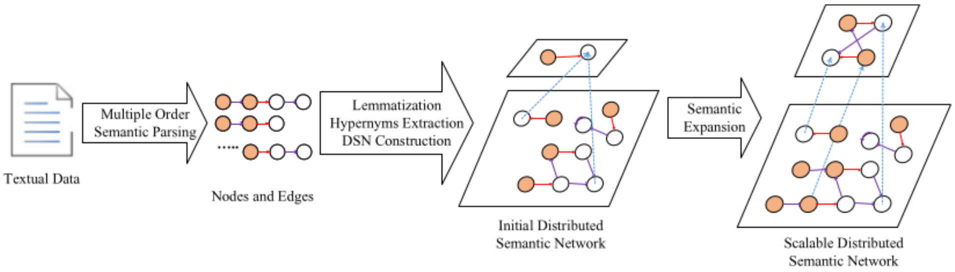

Textual Data
Multiple Order Semantic Parsing
Nodes and Edges
Lemmatization Hypernyms Extraction DSN Construction
Initial Distributed Semantic Network
Semantic Expansion
Scalable Distributed Semantic Network

Fig 1: Construction of Distributed Semantic Network

The paper describes the construction of multilevel distributed semantic network. Once network is constructed, each layer of the network is given as input to the Hadoop mapper, which extracts triplets, and assigns it a unique key. The reducer shuffles and removes and the duplicates. After obtaining the result from reducer, clustering is performed and the semantic similarity of the generated triplets from the existing knowledge base is calculated. If the triplet is is significantly different from triplets in knowledge base, the knowledge base is updated otherwise the triplet is discarded.

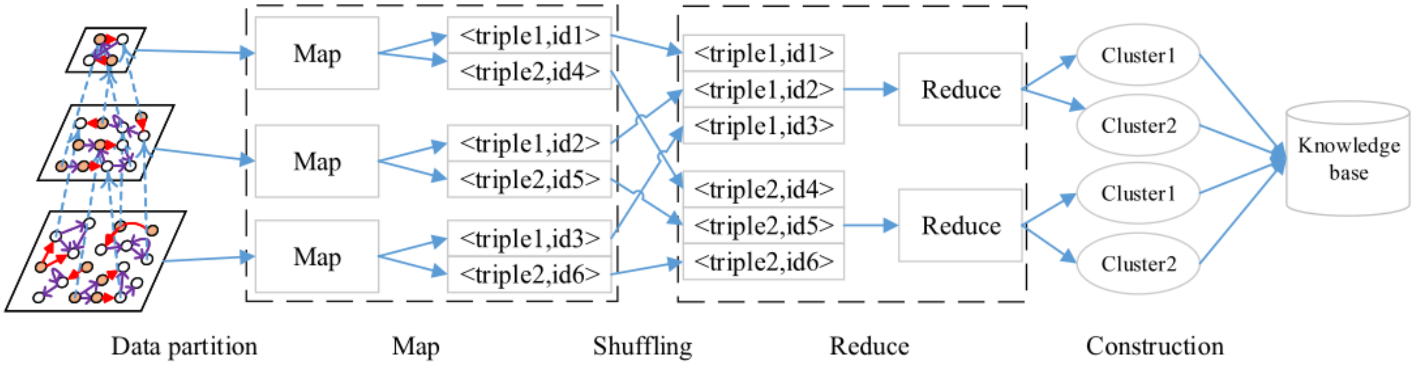


Fig 2: Proposed Architecture

Implementation:

1. Multiple Order Semantic Parsing

Extraction of NP-VP-NP triplets using Stanford Open Information Extraction

Example Sentence: Wall Street is trading broadly flat as investors eye some key speeches from Federal Reserve chief Janet Yellen and other central bankers.

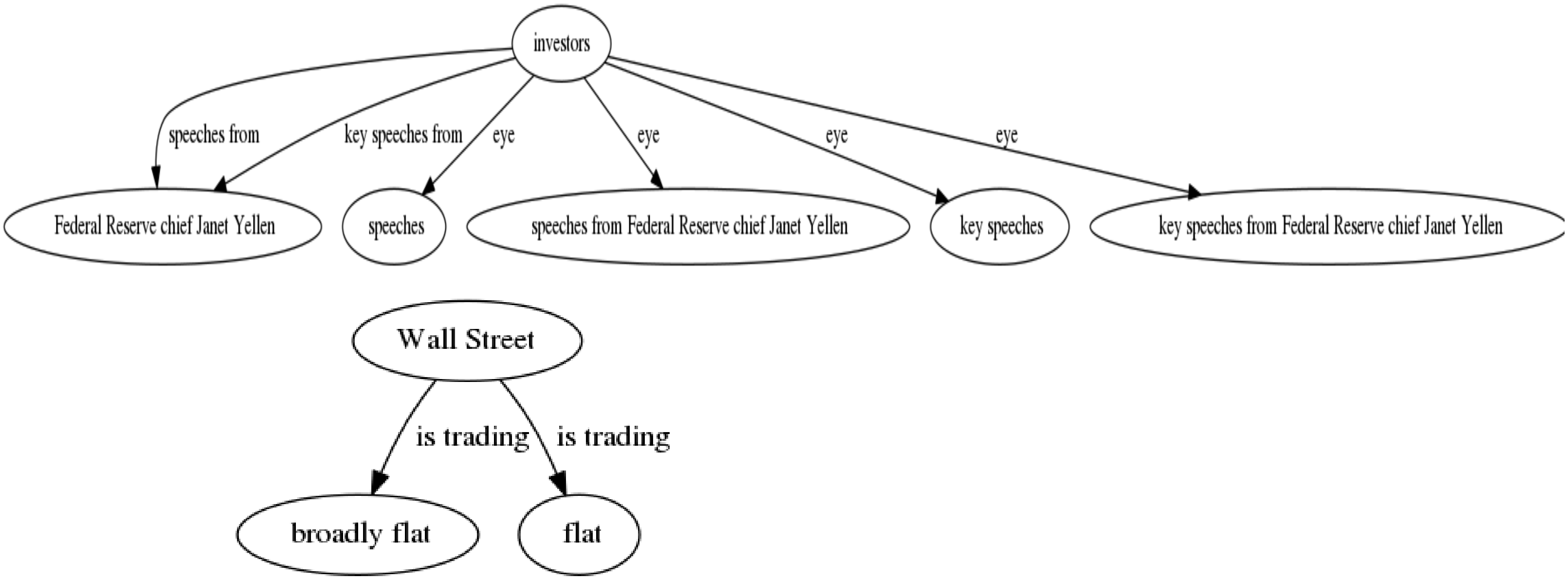
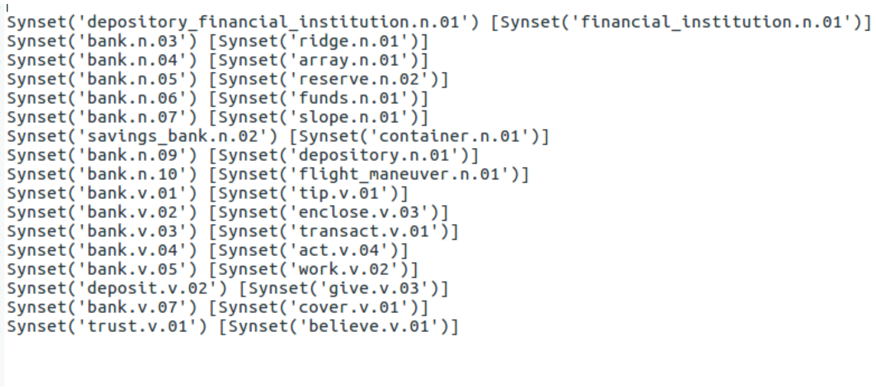

investors
speeches from
key speeches from
eye
eye
eye
eye
Federal Reserve chief Janet Yellen
speeches
speeches from Federal Reserve chief Janet Yellen
key speeches
key speeches from Federal Reserve chief Janet Yellen
Wall Street
is trading
is trading
broadly flat
flat

Fig 3: Graph denoting triplets genereated through Multiple Order Parsing.

2. Find Hypernyms to construct second layer of the graph.

Finding hypernym of a given word, is well known research problem falling under word disambiguation. The scope of the paper does not address this. For this project provided if hypernyms exist, the first hypernym is taken. To find hypernyms, WordNet is utilised.

For a given words it's sysnsets are generated. A particular synset disambiguates the term. For the first synset generated, hypernym is found and stored.



```
Synset('depository_financial_institution.n.01') [Synset('financial_institution.n.01')]
Synset('bank.n.03') [Synset('ridge.n.01')]
Synset('bank.n.04') [Synset('array.n.01')]
Synset('bank.n.05') [Synset('reserve.n.02')]
Synset('bank.n.06') [Synset('funds.n.01')]
Synset('bank.n.07') [Synset('slope.n.01')]
Synset('savings_bank.n.02') [Synset('container.n.01')]
Synset('bank.n.09') [Synset('depository.n.01')]
Synset('bank.n.10') [Synset('flight_maneuver.n.01')]
Synset('bank.v.01') [Synset('tip.v.01')]
Synset('bank.v.02') [Synset('enclose.v.03')]
Synset('bank.v.03') [Synset('transact.v.01')]
Synset('bank.v.04') [Synset('act.v.04')]
Synset('bank.v.05') [Synset('work.v.02')]
Synset('deposit.v.02') [Synset('give.v.03')]
Synset('bank.v.07') [Synset('cover.v.01')]
Synset('trust.v.01') [Synset('believe.v.01')]
```

Fig 4: Possible synsets and hypernyms of the word "bank"

3. Indexing the network

The DSN is indexed on Solr.

The paper doesn't address how the graph or the adjacency list is stored. An adjacency list is given as input to the mapper. This adjacency list can grow exponentially, I feel it's impractical to store as plain text files. Indexing aids in faster retrieval.

4. Markov Clustering

Markov Clustering is an unsupervised cluster algorithm for graphs based on simulation of stochastic flow in graphs is applied to the network. It simulates flow within a graph and promotes flow in a highly connected region and demotes otherwise, thus revealing natural groups within the graph.

Markov Clustering is performed on the graph with inflation factor-1.4

Time Complexity of Markov Clustering: $O(N^3)$ where N is the number of vertices.

N^3 cost of one matrix multiplication on two matrices of dimension N. Inflation can be done in $O(N^2)$ time. The number of steps to converge is not proven, but experimentally shown to be ~10 to 100 steps, and mostly consist of sparse matrices after the first few steps.

Possible Extension:

1. Figure out how to perform Hadoop map reduce from solr indexed data.

Observations:

1. Open IE extraction provides redundant triplets. Deduplication method has to be devised.
2. The paper addresses the fact that Hadoop map reduce creates overhead. But since the paper hasn't used indexed data performance needs to be compared.

   The lowest time reported by the paper is 23 seconds for 200 statements in NYT dataset.

Paper 2: Resource discovery for distributed computing systems: A comprehensive survey

Authors: Javad Zarrin , Rui L. Aguiar , João Paulo Barraca

Computing infrastructure such as Grid, Clusters, Cloud have become increasingly popular. These varied methods have a fundamental common key property; the ability to share resources/services. This paper addresses the challenge of discovering resource in large-scale distributed computing environments which often contain significant amount of either homogenous or heterogenous computing resources from different sources. This paper is literature survey on current state of resource discovery protocols, mechanisms, and platforms for large-scale distributed environments, focusing on the design aspects.

This report focuses primarily on search algorithms.

1. Informed vs. Uninformed

   The primary distinction between informed and uniformed search methods is, uniformed search is a blind search – it has no prior knowledge about other nodes or resources in the network whereas informed search has each requester has some sort of heuristic or probabilistic measure of where the resource is located.

   ALG-Flooding, Breadth First Search (BFS), and Depth First Search (DFS),are the most well-known systematic search methods. Moreover, there are several other systematic and random search methods such as Depth Limited Search , Iterative Deepening , Uniform Cost Search , Random Walk etcetera.

2. Synchronous vs. Asynchronous

Synchronous algorithms proceeds synchronously, in parallel rounds, where each round is defined as the time required for each node in the network to communicate with one or more other nodes, learning about them and gathering information. Asynchronous methods are based on the asynchronous communication model, where each node can send a message in arbitrary size to any of its neighbors in a way that the outgoing message eventually arrives in the destination node after an unbounded finite time. ALG-Flooding, Swamping, Random Pointer Jump and Name-Dropper are well known synchronous resource discovery search algorithms.

2.1. Random Pointer Jump Algorithm

In each round u∈ V contacts a random neighbor v∈ V. It then sends a message M(u) to V which then merges u with v. (Fig. 5)

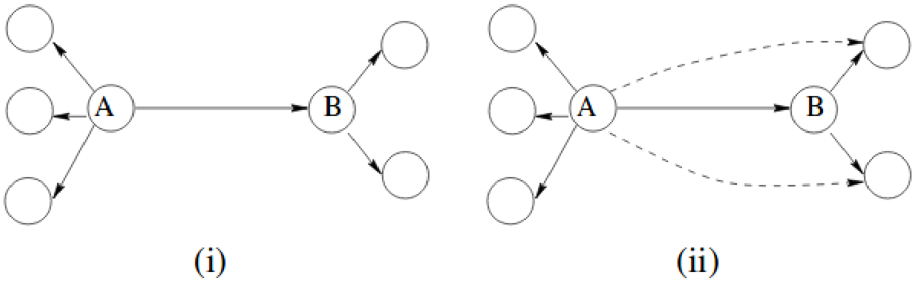


Fig 5: A randomly choses neighbor B. A sends a message to B. B is merged with A. New edges are shown as dotted lines.

2.2. Name dropper algorithm

Name dropper algorithm works similar to random pointer algorithm. In each round, each node contacts a random neighbor v. A message is passed and the nodes are merged.

3. Bio-inspired approaches

Bio inspired search algorithms are based on biological systems. Biological systems exhibit capability for self-management, autonomy and self-organization. Researchers have found correlation between behavior of any distributed architecture such as P2P and biological systems. Common phenomenon such as birthing/ addition of new member, death of a nodes, migration, replication, division etc. could possibly be replicated in our distributed systems.

A few well-known examples are Tabu Search, Ant Colony Algorithm, Immune Search, Neural Search, Viral Search and Bee Colony Algorithm.

References

1. Gabor Angeli, Melvin Johnson Premkumar, and Christopher D. Manning. Leveraging Linguistic Structure For Open Domain Information Extraction. In *Proceedings of the Association of Computational Linguistics (ACL)*, 2015.
2. Fellbaum, Christiane (2005). WordNet and wordnets. In: Brown, Keith et al. (eds.), Encyclopedia of Language and Linguistics, Second Edition, Oxford: Elsevier, 665-670.
3. Stijn van Dongen, *Graph Clustering by Flow Simulation*. PhD thesis, University of Utrecht, May 2000.
4. J. Taheri, Y. Choon Lee, A.Y. Zomaya, H.J. Siegel, A bee colony based optimization approach for simultaneous job scheduling and data replication in grid environments, Comput. Oper. Res. 40 (6) (2013) 1564–1578.
5. M. Harchol-Balter, T. Leighton, D. Lewin, Resource discovery in distributed networks, in: Proceedings of the Eighteenth Annual ACM Symposium on Principles of Distributed Computing, PODC ’99, ACM, New York, NY, USA, 1999, pp. 229–237.
6. Mor- Harchol Barter, Tom Leighton, Daniel Lewin, Resource Discovery in Distributed Networks
7. V.K. Garg, A. Aziz, An Efficient Deterministic Algorithm for the Resource Discovery Problem, Technical Report, Technical Report, ENS 527, The University of Texas, Austin TX 78712, 2000.