Distributed Computing Final Project

G.Savitha

Scalable Distributed Semantic Network for Knowledge management in cyber physical system

- Data growing rapidly.
- The 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.
- Use the prior knowledge of WordNet and Wikipedia to scale out DSN horizontally and vertically.

The main contributions of this paper can be summarized as follows:

(1) We design Multiple Order Semantic Parsing (MOSP) to analyze heterogeneous data, that is, each piece of semantic information implied in the text is represented by a set of nodes and edges associated.

(2) We propose a hierarchical graph based scalable DSN to express semantic information implied in the text and absorb more prior external knowledge for expansion.

(3) We propose a MapReduce-based framework to extract semantic information from DSN and manage it for knowledge base construction.

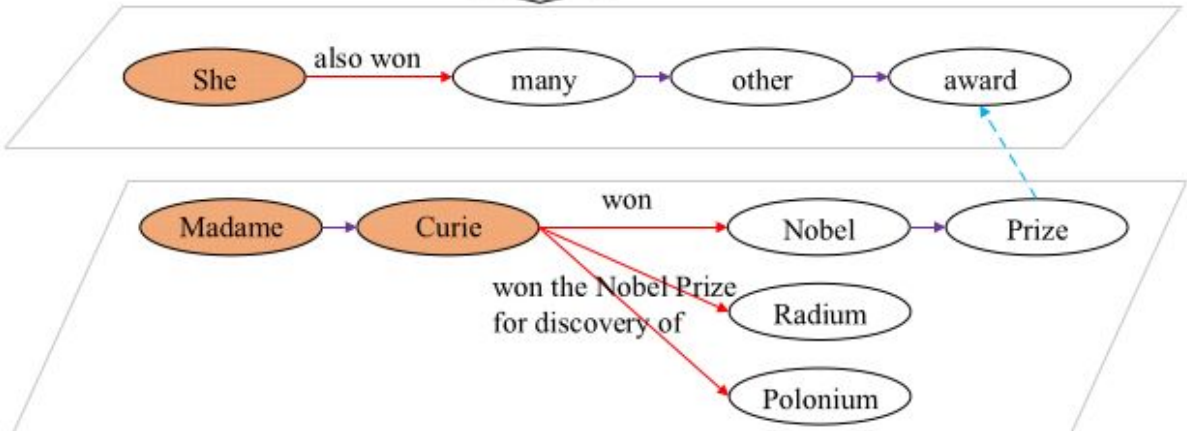
The original sentence:

Madame Curie won the Nobel Prize for her discovery of Polonium and Radium. She also won many other awards.

The triples:

"Madame Curie" "won" "the Nobel Prize"
"Madame Curie" "won the Nobel Prize for discovery of" "Radium"
"Madame Curie" "won the Nobel Prize for discovery of" "Polonium"
"She" "also won" "many other awards"

Lemmatization
Query hypernym
Construction



Algorithm

Algorithm 1.
Function Map (String key, String value)
Input: id-adjacency list
Output: triple-id
1: R_v *= FindRoot(value);*
2: foreach (r $\in R_k$*) do*
3: $p = DFS(r)$
4: if(p contains semantic information)
5: $P_v.add(p)$
6: foreach (p $\in P_v$*) do*
7: $TF_v = p.split$
8: foreach (tf $\in TF_v$*) do*
9: t *= generate triple from tf*
10: EMIT-INTERMEDIATE(t, key)
11: $TF_i = inference(TF_v)$
12: foreach (tfi $\in TF_i$*) do*
13: t_i *= generate triple from* tf_i
14: EMIT-INTERMEDIATE(t_i*, key)*

Algorithm 2.
Function Reduce (String key, Iterator values)
Input: triple-id list
Output: triple
1: if (there is a knowledge base K)
2: if(!K.contains(key))
3: EMIT(t)
4: else
5: EMIT(t)

Issues

- To identify Hypernyms for building another level in the semantic network, word sense disambiguation must be solved.

  Existing libraries do not have perfect accuracy (65% to 75%) which will lead to error in the network.

- Distributed hashing. Not addressed.

Resource Discovery Algorithms

Table 5
Comparison of some well-known synchronous search algorithms for state discovery.

Search algorithm	Time complexity	Communication complexity	
		Pointer complexity	Message complexity
Absorption [189]	$O(\log n)$	$O(n^2), O(n^2 \log n)$	$O(n), O(n \log n)$
Kutten & Peleg [185]	$O(\log n \log^* n)$	$O(n^2 \log^2 n)$	$O(n \log n \log^* n)$
ALG-Flooding [142]	$d_{initial}$	$\Omega(n \cdot m_{initial})$	$\Omega(d_{initial} \cdot m_{initial})$
Swamping [142]	$O(\log n)$	$\Omega(n^3)$	$\Omega(n^2)$
Random Pointer Jump [142]	$\Omega(n)$ in worst case	$number.of.cycles \cdot m_{initial}$	$number.of.cycles \cdot m_{initial}$
Name-Dropper [142]	$O(\log^2 n)$	$O(n^2 \log^2 n)$	$O(n \log^2 n)$
Fast-Leader [127]	$O(\log^2 n)$	$O(n^2)$	$O(n \log^2 n)$

Summary of bio-inspired search methods (SA: Search Algorithm).

SA	Description
Bee colony [8,134,301,176,291]	A swarm-basedoptimization algorithm which is inspired by foraging mechanisms of honeybees. The scout bees stochastically search (global search) the new food sources, upon finding a new source, they will transfer the flower information by performing waggle or round dance. On the other hand, the worker bees evaluate the quality of the newly discovered flower sources through observing the dances and choose the elites (best sources) for foraging. Bees must avoid overcrowding and keep diversity, for this reason, they scatter in the proximity around the elite flowers (local search) while at the same time the scout bees start a new iteration of the global search. Bee Colony Algorithm (BCA) divides the search process into two steps, parallel implementation of the global search (parallel random search in variable space by scout bees) and the local search which is the local improvement of the current elites by worker bees. The elite selection mechanism in each iteration compares the quality of the discovered solutions from global search and local search with the quality of the elites in the last iteration.
Ant colony [99,101,182,1]	Ant Colony Search is a meta-heuristic search method based on Ant Colony Optimization (ACO) which is inspired by the behavior of the real ants searching their environment to find the food sources. The ants (queries) start their search by randomly parallel scattering around the nest. The successful ants whose found the food sources (resources) will return to the nest using their memory and mark the (successful) path (between nest and food source) through emitting a chemical substance called pheromone on trails. The other ants, coming across the trail, follow the marked path instead of wandering randomly to check the food source. If they become successful to find the food, they will return the nest and reinforce the pheromone on the trail. For selecting a trail, each ant makes a local decision by comparing its experience with the environmental information (trails) which is updated by other ants and finally it selects the strongest path(trail) in terms of the density of pheromone considering the fact that the pheromone evaporates over time. In other words, in the intersections, the ants prefer to choose the strongest trail through comparing the already available trails marked and modified by different ants (i.e., indirect communication or stigmergy) instead of direct communication and exchanging information with other ants. Using this approach the pheromone of the paths with the long distance source will be more evaporated than the shorter tracks since for the long paths it takes more time for the ant to reach the nest. Thus, the density of the pheromone in the shorter paths would be higher than the longer paths. This leads to discovering the closest resources with higher probability. The ACO based search methods make benefits [182] for resource discovery in terms of autonomy (the nodes do not have any global information), parallel search, proximity-awareness (quick convergence to near optimal solution), efficiency (prevents a large-scale flat flooding [87]), flexibility (supporting multi-attribute range query [87]) and robustness (concerning system workload). There are several proposals for ACO based search methods such as Semant [221], NAS(Neighboring Ant Search) [311], ACS(Ant Colony System) [100] and Max–Min Ant System [287].
Neural search [333]	Neural search is based on Artificial Neuron Network (ANN) [145] wherein a set of neurons (computing units, nodes or computing elements) are connected together to construct a network which mimics a biological neural network of the brain. Artificial neurons are the simple modelings of brain cells which are connected using both of feedback and forward links with different dynamic weights that create an adaptive system. The adaptive weights are the numerical parameters that are tuned by a learning algorithm during run time (or training/prediction phase), and conceptually they clarify the strength of the links which can be used for the link evaluation in the process of neighbor selection of the neural search. NeuroSearch [315] is an example Neural search which attempts to solve the overhead problem of the flooding based approaches such as BFS by selecting the best neighbor for query forwarding based on the individual evaluation of the output of the neural network (for a set of specific input parameters) for every one of the neighbors. ANN will create deterministic or probabilistic maps between input and out parameters which lead to specifying the weights for each one of neighbor nodes.
Viral search [125,45,137]	Proposes a meta-heuristic search method based on viral infection using a biological analogy. It takes the advantages of Multi Agent Systems (MASs) [96] and combine it with some well-known approaches in Artificial Intelligence (AI). The viruses in the system are considered as part of the widespread infection, while each individual tries to make its benefits but leading in the overall benefit of the viral system. It is desirable for viruses to infect the individuals with the minimum level of health. The efficient viral infection can be achieved by continuously searching the individual microorganisms (e.g., bacteria) that are appropriate for infection (i.e., the best solutions which are the individuals with the less level of healthy). In this way, the viruses optimize their search results by infecting less healthy individuals (weak solutions). The probability to extend the domain of infection (in other words, achieving higher access to the new potential results) can be increased by systematically propagating the viruses which are lodged in unhealthy cells.