Shortest Paths Exploration Between Two Papers Based on Citation Network

Hou Chen
School of Electronic Info. & Electrical Engieering
Shanghai Jiao Tong University
Shanghai, China 200240
Email: vamdawn@sjtu.edu.cn

Abstract—Acemap shows a map consist of papers and citation network. It is necessary to show paths based on citation network when we research the relation between two papers. In this report, we introduce the database of paper citation network, the algorithm of path exploration and how to apply the function in our Acemap. We choose one part of the database that only includes the references relation, because the other parts are not necessary to get the path. To Explore the shortest path based on citation network, the breadth-first-search(BFS) algorithm is used. For the academic map is made up of papers and citation network, paths could be easily shown on the map. Paths exploration is a useful tool for analyzing the relation between two papers.

1 Introduction

Our acadimic information system — Acemap is based on papers' citation networks, and then it is necessary to talk about the similar network structure — Social Network. A social network is a social structure made up of a set of social actors(such as individuals or organizations), set of dyadic ties, and other social interactions between actors. Social networks could be intuitionistic exhibited in social softwares, such as Facebook, Twitter and LinkedIn: one user in a node in social networks, and the ”Follow” relation between users are paths in social networks, which are in the similar structure with citation network. A great number of researches have done on social network and discover some properties of social networks: small-world, also six degree of separation; degree distribution: power-law; network resilience and other properities. Therefore, it is meaningful to research paper citation network that has the similar structure with social network.

To research papers' citation network, paths exploration between two papers is a good breakthrough point. Acemap system has many superiorities for implementing paths exploration. Firstly, there is tremendous amount of data in the Acemap's database, which includes complete infomation of papers, and then paths exploration could be the operation on the existing database. Secondly, academic map is made up of papers and citation network, and then paths are shown on the map, which is clear and direct. Thirdly, paths exploration could be shown as a function of the academic map, adding information of the map.

The following part of the report introduces detailedly the path exploration implements. In section 2, it introduces the database of Acemap system and how to utilize data. In section 3, it introduces paths exploration algorithm. To get the shortest path, I use the BFS algorithm to explore paths. In section 4, it introduces the application of path exploration in the webpage. The function enriches the academic map.

2 Database Structure

PaperID	PaperReferenceID
ID_A0	ID_A1
ID_A0	ID_A2
ID_A0	ID_A3
ID_B0	ID_B1
ID_B0	ID_B2
ID_B0	ID_B3
...	...

Table 1. Data stored in the database

To build up a system like Acemap, much data about paper information is needed. Our Acemap system certainlly has a great database, which includes name, references, citations, year and much other information. In this report, I only need references and citations relation of papers in the database. For serving the academic map, I choose the set of papers in the map as the objects for path exploration, which means it could not include all papers in the database, but includes all papers on the map. There are a total of 5138887 papers in the references relation database, including 79306989 references relations, which take up 1.8GB storage in the database. Therefore, it is important to choose the appropriate algorithm to explore paths.

Papers' references and citations are stored in the database like Table 1. Each paper has one unique ID identifying itself, so I can locate one paper and get relative information by paper ID. That is to say, when I get the paper ID, I get the paper. The database takes a method that one ID targets multiple IDs, which means a paper my have several references. On the other hand, one paper ID may appear more than one time in row ”PaperReferenceID”, which means a paper may have several citations. In this data structure, we could get references of one paper using mysql query *”SELECT PaperReferencesID FROM PaperReferences2 WHERE PaperID = ?”*, where PaperReferences2 is the name of a schema that stores the references relations between papers. Similarly, we could get citations of one paper using mysql query *”SELECT PaperID FROM PaperReferences2 WHERE PaperReferenceID = ?”*. After I have those basic operations, I could get references and citations of one paper for next analysis.

The database only stores direct reference relations between papers, one exploration may involve a great number of rows, which means costing much time. Because mysql query is processed on the server, when large number of complex queries are excuted at the same time, the server make break down. Therefore it's a good idea to fetch part of the data to local client before dealing with it.

3 Path Exploration Algorithm

In section 2, we know that there is much data to be dealt with in the database if path exploration is excuted. Therefore, I shold choose an algorithm that has the least cost. This report works on shortest path between two papers based on citation networks [1]. That is to say we concentrate on three things: paper, reference and citation. Reference and citation are similar relation, and their differences are not the key point, so we think them same things in this report. The structure of citation network is like a graph. If I think papers as nodes, references or citations as edges, the path exploration could be implemented using knowledges of graph theory. In graph theory, the algorithm to get shortest path is Dijkstra Algorithm and Floyd Algorithm. Because they are used to deal with shortest path problem of a wighted graph, and the graph consist of citation networks has all same edges. Then the breadth-first-search(BFS) algorithm could be used for path exploration.

Breadth-first search(BFS) is an algorithm for traversing or searching three or graph data strutures. It starts at the tree root(or some arbitrary node of a graph, sometimes refered to as a 'search key') and explores the neighbor nodes first, before moving to the next level neighbors. Implements of BFS are like this:

(1) Enqueue the root node.

(2) Dequeue one node from the queue, search its next level nodes and enqueue those nodes.

(3) If the end node is found, end the process, or return (2).

(4) If it cannot find the end node after the whole nodes are visited, end the process.

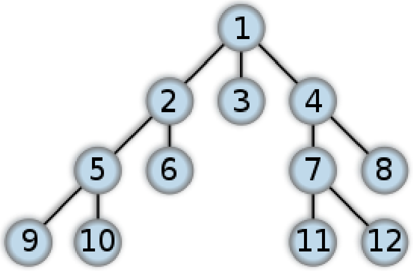

Fig. 1. An example to show the breadth-first-search(BFS) algorithm

If we think a graph like Fig 1, and the start node is 1, the end node is 7, the process of BFS is like this:
dequeue node 1, and enqueue its next level nodes 2, 3, 4.
dequeue node 2, and enqueue its next level nodes 5, 6.
dequeue node 3, and enqueue its next level nodes. (no node)
dequeue node 4, and enqueue its next level nodes 7 (end node), 8.
The example shows the detailed process of BFS algorithm. So the next problem is how to apply it in our circumstance.

Because the number of papers is 5138887, the number of reference or citation relation is 79306989, and total storage is 1.8GB, it is unrealistic to fetch all data to local client. Therefore Fetching part of data could save the cost and time. The time complexity of BFS can be expressed as $O(|V|+|E|)$, since every vertex and every edge will be explored in the worst case. —V— is the number of vertices and —E— is the number of edges in the graph. Because this report aims at paths between two papers, to decrease the time of path exploration, I choose to fetch three-level papers of both two papers and start comparing whther there is same paper. Here gives a definition, n-level papers of paper A means that A's references and citations are the 1st-level papers, the 1st-level papers' references and citations are the 2st-level papers and so on. Then the number of level needs to be fetched decreases to a half of original method, which could decrease the time of path exploration.

4 Application in Webpage

Path exploration could be a useful function of the academic map. As is known to us all, visualizations are more direct and easier to understand than a set of numerical value. The academic map is a good example, people can easily find which paper has more influences on the field (the paper with more citations is larger than other paper), besides, people could find the degree of correlation among papers from the distance among the paper, papers with longer distance have weaker relation. Similarly, paths between two papers can also shows their relation. For example, when we research two papers' similarities and distinctions, we could analyze them from topic, authors, contents, and references. From references we could know whose's though has a influence on them. There is no doubt that two papers with longer paths have less relations.

In the academic map, one point represents a paper, so paths could be easily shown on the map. Based on the path exploration, a new function of the academic map could be done: When users choose two papers on the map and click ”search”, all possible shortest paths will be shown on the academic map, which adds the information and enrich the use of Acemap.

5 Conclusion

The path exploration function uses BFS algorithm to get the shortest path. For a small amount of data, it could be done fast, but when there is a large amount of data, long time exploration is inevitable. It is a good way to optimize the algorithm to speed up the exploration. Therefore, one of future works is optimizing the path exploration algorithm.

Through this course project, I am able to master the use of these tools: Javascript, SQL and HTML language. I adapt myself to read academic papers in English, which was difficult and boring for me. The project also help add a function for Acemap website. Though the function still has some shortcomings, it is useful for comparing two papers' relation and analyzing citation networks.

Acknowledgements

Firstly, I would like to thank Professor Wang's teaching, without him, I will never have a chance to research on the system like Acemap. Then I wourld like to thank my group leader Tang Weijie. He gave me many useful suggestions and guidance. And I would like to thank all members of my Acemap group, they gave me many suggestions when I decided what to do.

References

[1] Breadth-first search, Wiki Pedia