Search in Networks

Udit Sajjanhar

03CS3011 {sajjanharudit@gmail.com}

I. Introduction

Search in itself can be classified broadly in to three domains:

1. Fast Message Movement
2. Clustering
3. link Analysis

Here we will be dealing with link analysis. First let us see why is link analysis needed. The query issued to a search system can be of two types: 1.Abundant and 2.Specific. Example of an abundant query is $Java$ or say $Harward$. A specific query may be of navigational type for example if I am looking for the blog page of Steve Jobs.

But when the query is an abundant one(say $Harward$), then how do we get the relevant results? We will look for all the pages containing the word Harward. But is that suffcient? No. There may be a link say www.harward.edu which may hardly mention the word $Harward$. Thus we see that Text Analysis alone can't solve all our problems. We need something else. So here comes in Link Analysis(Links have some information encoded).

Now we study the Hub and Authority Model. A node can be devided into two types: either hub or authority. Hubs are the nodes which point to a lot of pages that contain information. Thus a typical Hub has many outgoing links. An authority page is the one which contains the information about a subject. Thus an authority page has many incoming links.

So the results for a query should ideally contain the authority pages about that subject. One simple approach can be to take the first 100 results offered by an engine and then rank them according to their in-degree. But the problem is that there are some pages that are universally popular ex. www.yahoo.com and which offer no information about the query. These pages have to be filtered out.

One solution is as foloows. We take a smaller graph R_σ,which consists of the top 50 search results from a search engine say AltaVista. Now we $growthegraph$ to form a new graph S_σ. S_σ contains all the outlinks of R_σ and a K random incoming links. Thsi is justified by the fact that an authority page will contain links to other authority pages and will have links some other Autority pages. Now we remove from the new added links those which point to the same domain and all those which are more tha C in number and point to the same page. However this scheme doesn't remove the problem of Universal popular pages.

Another solution: In the starting each node is given a Hub Score(X_i) and an Authority Score(Y_i). Now consider this. For an Authority page its score is the nomalized summation of the score of the hubs that point to it and for a hub the score is the normalized summation of the score of the authorities that it points. So we can write the following equations:

$X = AY$

$Y = AX$

To find the solutions to the above equations we iteratively calculate the value of X_i and Y_i until the solutions converge. Now in the results we present the C best Hub pages and C best Authority pages. This gives the best result, eliminating our orignal problem of Universally Popular pages.