DBLP – Search Engine

Project Report

By: Akash Kumar Gautam, 2015011
Nishant Gahlawat, 2015151

Entity Resolution

After analyzing the XML and looking through the documentation on the website, we discovered that all the names used by a particular author are stored under <www> tags with the attribute key containing the string “homepages/”.

We created a database which includes an arraylist of “Persons”. The Person class contains the names used by an author. Through the initial parsing of the program, we build up the database, thus creating the basis of our entity resolution.

When searching through the author name, if the tag supplied by the author gets a match from an author name, all the names used by that particular author would come into play. All the names used by the author would show up in the search result.

In the “More than k” searcher, all the names of the author with more than k publications would be shown in one field. In the prediction query, all the names of the author are taken into consideration when the publications per year are searched for.

Prediction

For the prediction part, we used the linear regression formula. By using the formula, we were able to get the formula for a line graph that shows the probable path that the author takes in writing his/her publications per year.

The Formula:
The line projected would be:
y=ax + b, where ‘y’ would be the number of publications in the year ‘x’.

b=sum ((xi-$\underline{x}$)*(yi-$\underline{y}$))/sum ((xi-$\underline{x}$) ^2), a=$\underline{y}$-b*$\underline{x}$ (sum(x) = sum of all x, $\underline{x}$=mean of all x, $\underline{y}$=mean of all y)

For an year to be predicted ‘P’, we used all the years preceding it for the particular author, mapped all the years to their respective no of publications and then used the formula.

Contributions

- Akash Kumar Gautam: GUI, DataHandling
- Nishant Gahlawat: Parsing, DataHandling

GUI Designing:

For designing basic layout of GUI which consists of two main panel query panel and result panel in which query panel automatically changes in accordance with type of query and thus implementing model view control design pattern

Doxygen Commenting:

For generating doxygen file important member fields and functions had to be commented out and thus running through terminal gave a doxy file in which parameters can be set for generating specific type of doxy file which was saved as html file that can be viewed through browser in order to get full view of doxy generated file.