DBLP

The DBLP Computer Science Bibliography evolved from an early small experimental Web server to a popular service for the computer science community. In June 2009 the DBLP Computer Science Bibliography from the University of Trier contained more than 1.2 million bibliographic records. For computer science researchers the DBLP web site is a popular tool to trace the work of colleagues and to retrieve bibliographic details when composing the lists of references for new papers. Ranking and profiling of persons, institutions, journals, or conferences is another sometimes controversial usage of DBLP. The DBLP data may be downloaded. The bibliographic records are contained in a huge XML file.

Aim

Our aim in this project was to retrieve required information from the dataset. We had to accomplish the following tasks:

- Parse the XML File and store the data using collections.
- Finding publications by:
  Title Tags
  Author Names (includes entity resolution)
- Finding names of authors with more than 'k' publications. (includes entity resolution)
- Predict the number of publications for 5 authors with +-20% range for the next year given a year 'k'. (includes entity resolution)

Approach:

1. Parsing was done using SAX parser. SAX Parser is different from DOM parser because it doesn't load complete XML into memory and read xml document sequentially. The dataset being huge, DOM parsing would have been inadequate.

2. Entity Resolution – <www> tag in the xml file contains all the aliases of an author's names that are possible. So, a list 'b' of author type is made, it parses through only the <www> tags of the file and adds to this list, author objects containing all the possible aliases for a particular author.
Then a list 'a' of String type is made. User input is appended to this list. After this, we check in the previous list for whether the user input have any aliases or not. If yes, then these aliases are appended to list 'a'. Now we have a list 'a' containing all the possible aliases of the user input including the user input. Now, we take this list, pass it in the constructor of the parser class. Now while parsing, it checks for authors of that particular publication, and if any of the names in the list 'a' is contained in that publication's authors, the publication is appended as the publication of the searched user input.

3. Prediction - For the purpose of prediction we have used the approach of Prediction intervals. This method is capable of predicting the prediction interval of the next sets of data upto an appreciable accuracy. For this method we first iterated over the xml file to get the no. of publications by each author over the span of all these years. We first find the mean of the data given. Next we find the variance. After this, we find the prediction interval using the appropriate algorithm and get the most probable next sets of data.

Done By-

Akhil Goel (2015126)
Contributions: GUI code, implementation, logic (Including all the checks for GUI exception cases)
+ Bonus logic
Mohit Agarwal (2015060)
Contributions: Backend code, implementation, logic (Including parsing, entity resolution, and all queries)