Xapian's GSoC Student Application

About me

Name: Ou Jiao
E-mail address: ojiao1111@gmail.com
IRC nickname: Fannie
Personal website: http://blog.csdn.net/perhaps9329
Phone number: +86-18710249053
Biography:

2011.08~2015.07:I studied in Xidian University and majored in Software Engineering, finally I got a bachelor degree.

2015.09~now: I am studying in Institute of Computing Technology, Chinese Academy of Sciences, majoring in Technology of Computer Application. And our research group primarily research search engine and relative application.

Background Information

It is the first time I take part in GSoC or similar programmes.

But last year I took part in development of News Search System. First we collected three or five news websites, extracted every website information, made index construction for each. Finally it can implement web-page retrieval. This system has no less than 200 thousand website items. It can rank results by relevance, or popularity, or freshness. What is more, if there are a lot of search results , many of them are similar is to group them using clustering automatically via our system.

I am sorry to say I have no previous experience with Free Software and Open Source .But I hope I can obtain worthy experience by GSoC this year.

Last summer, I found a holiday job, mainly developed an android application about e-commerce, which includes a function that according to keywords by user, it returns relative result sets sorted by popularity. And in the home page, it needs to return the hottest commodity information based on what user concern about. Of course it includes so many functions , but only this two relate to the project what I want to apply in GSoC.

I usually prefer to use Linux and Windows platform.

The above two may be projects of a similar scope

I will be in GMT+8, China standard time during the coding period.

My Summer of Code project will be the main focus of my time during the program. And I hope to work from Monday to Sunday 9am-5pm. Actually as long as I have time ,I will do it. This project is the only one what I apply for in GSoC 2016.

Your Project

Motivations

On the one hand, I am interested in search engine, and use open source search engine library more than once in the development, thus I am curious about how it works and want to understand it. On the other hand, I take some courses about information retrieval, pattern recognition and machine learning in one recent year, I urgently hope to apply theoretical knowledge to actual projects.

Some developers using search engine libraries will benefit from my project. For example, If I find out an effective algorithm to improve Learning-to-Rank, then developers can use this interface directly without knowing details.

Project Details

I choose the project, Learning to Rank Stabilisation. When we approach to information retrieval at first, rank only involves term frequency, inverse document frequency and document length, so using these factors to overfit ordering formulas that are TF-IDF or BM25. But with arguments increase, machine learning may prefer to do it well.

The traditional retrieval model overfit ordering formulas by artificial. To obtain the best arguments, we need to experiment again and again. In contrast, the most reasonable ordering formula can be learned by learning to rank, what we should do is to provide training data.

Learning to Rank consists of labeling training data by artificial, document feature extraction, learning classifier and using the machine learning model in the actual search system. Firstly, find out some documents related to this query, and label the degree of relevant. As for machine learning, the input is query and a series of documents, it learns how to give a score about relevance. Then According to scores, it outputs results. Of course each document is represented by a vector of features. These features, besides term frequency, inverse document frequency and document length, also including web-page's in-link number, web-page's out-link number, pageRank and so on, try to distinguish the levels of relevancy between documents.

Learning to rank relates to three methods, pointwise approach, pairwise approach and listwise approach.

As for pointwise approach, it can cope with a single document. After each document is represented by a vector of features, it turns rank problem into classification or regression problem. This approach does not consider the relationship of documents relative order. It suppose relevance is not relevant to query. But it is incorrect!

As for pairwise approach, it abandons the above assumption and turns rank problem into binary classification. Therefore, so many methods such as Boost, SVM, neural network can be used in this projects.

Different from the above two approach, listwise regards the whole results corresponding query as one training sample. According to the training samples , it can obtain the optimal scoring function. In testing stage, based on a new query, scoring function scores every documents, and ranking the results by scores. How to obtain the optimal scoring function is what machine learning needs to do.

This project is about consolidating the work done so far to get to a stable, tested core of functionality that can be included in a future Xapian release. So to familiarize myself with git or svn and some test methods will make my work be more likely to succeed.

This project will not be as simple as just introducing code from on branch to another. So there are three aspects. First is to integrate work done on various branches. Second is write an automated test suite for the LTR code and fixing any bugs that it unearths. Third is writing some practical code examples. Now I am not sure which uncertainties there exists depend on further research and investigation.

When I do this project, I will divide it into some sub-projects, each of which can be reviewed and merged before moving on to next. And I will propose any changes to the user-facing APIs and create practical code examples for every sub-projects. Even though I cannot work out everything exactly as planned, most of work can be completed, it only consolidates the part of the work done so far , but have no bad influence on the project's stability, others can continue doing this easily. Of course I can do my best to do it well as planned.

Project Timeline

Before April 23:

-To familiarize myself completely with Xapian functionality and architecture.
-Study of provided relevant resources.
-To get used to the tools needed.

April 22 – May 22(Before the official coding time):

-To set up development environment and get myself into the swing of Xapian's coding and development style.
-During this period I will remain in constant touch with my mentor and the Xapian community. I will remain active on IRC and Mailing lists to discuss and finalize on the problem what I meet.
-With the help of my mentor I will become absolutely clear about my future goals and how to introduce enough code to make work together as a complete system.

May 23-June 21(Official coding period starts):

-Choose one of every option, and introducing enough code to make them work together as a complete system.
-Get the current master branch building cleanly.
-Create practical code examples that use the core features and API.
-Propose any changes to the user-facing APIs and write tests for API.

June 21 MID TERM EVALUATION

June 21-July 12

-To ensure correct use of C++ features and all code introduced match Xapian's style guide.
-Write unit test and documentation requirements for introduced code.
-To ensure the system continues to build without warnings.
-To improve the file layout of source code.

July 12- August 16:

-Proceed to introduce additional features, mostly variant rankers and scorers.
-Write a practical example of using the new features or class.
-Write tests for any user-facing API changes
-Write unit tests for any new functions, classes and so forth
-Update the user manual and the comments about API documentation, and ensure code quality and style guide

August 16-August 24:
-Clean code, clear up tests, improve documentation, and submit my work
-To be in constant touch with my mentor and modify documentation or something wrong according to feedback.

A Buffer of two weeks has been kept for any unpredictable delay.