1. What do you think will be your contributions?

Our main contribution will be a study on the effectiveness of Pregel-like computational model (e.g., Apache Giraph) in solving retrieval problems on graph, such as ranking articles from Wikipedia, finding influential author in publication graph, or influential user in Twitter graph. Particularly, i) its effectiveness in indexing the graph (i.e., running PageRank-like algorithm to obtain ranking score of vertices) and ii) its ability to quickly response to queries (e.g., finding the most influential author in the field of Machine Learning from publication graph).

Additionally, while classic PageRank algorithm only deals with homogeneous graph (i.e., vertices are in one type), we need to derive appropriate PageRank-like algorithm for heterogeneous graph (i.e., vertices are of multiple types). Thus, another contribution is to study the performance of Pregel-based computational model on heterogeneous graphs.

2. How you will be experimentally (or otherwise) evaluating your projects?

Experiment: We are doing experiments with Giraph running on a Hadoop cluster with real graph data from SNAP project. The experiment is done as follow (step-by-step):

- Given a graph data, a Map-Reduce program is written to convert the data into Giraph-supported format
- Implement a vertex-based class with all PageRank-based computation needed to be done at each vertex
- Run Giraph with converted data and implemented vertex-based class
- Given the result, which is PageRank scores of vertices, write a program is answer the retrieval query and return a ranked list of vertices

**Evaluation**: some evaluation metrics:
- Correctness of retrieval results: this requires human judgement -- to mark each returned result whether it is relevant to the query or not
- Convergence rate of PageRank algorithm: how many iterations needed to be executed to get converged scores
- Execution time

**Current status**:
- Although we planned to run a Hadoop cluster on AWS, the free account only gives us limited configuration. Therefore, at the moment, we are running a pseudo Hadoop cluster on our own laptop using Cloudera CDH4 virtual machine image.

- We have installed Apache Giraph on the cluster and have implemented a simple version of PageRank algorithm for homogeneous graph with the computation for each vertex is implemented as simple as follow:

```java
public class PageRankVertex {
    public void compute(Iterator messages) {
        if (getSuperstep() > 0) {
            // recompute own PageRank from the neighbors messages
            pageRank = sum(messages);
            setVertexValue(pageRank);
        }
        if (getSuperstep() < k) {
            // send updated PageRank to each neighbor
            sendMessageToAllNeighbors(pageRank / getNumOutEdges());
        } else {
            voteToHalt(); // terminate
        }
    }
}
```

- We have been able to experiment with some sample graphs using Giraph to calculate PageRank score. A Map-Reduce program is written to converted input graph into the data format supported by Giraph, such as the following (which essentially contains adjacent list for each vertex):

```
[0,0,[[1,1],[3,3]]]
```
3. Whether you can complete in time and how you will scope their project if you cannot?

With current status, we believe that we could complete the planned objectives in-time.

4. What are the major difficulties, what has been straightforward?

The main difficulty is limited computational resource. At the moment, we could not run with the full set of data and can only run a sample of data. We might want to request more premiere AWS resources, or access to CCT to do experiments.