SEMINAR IN ALGORITHMS: THE TWITTER PROBLEM (CONTINUED)

09/09/11

PROBLEM STATEMENT:

Consider a business model (say twitter) gives information about different topics (personalities) based on different queries. Assume that at most 'q' queries are allowed per day and each query returning at most 'k' results. And if there and 'N' topics –T_1; T_2, …. , T_N of our interest. Propose an algorithm to structure and organise the queries in such a way that with minimum number of effective queries we should maximise the amount of information for analysis. In other words, propose an effective and efficient algorithm which requires minimum queries to extract information about almost all or majority of the topics.

NOTATIONS USED IN SOLVING THE PROBLEM:

q ➔Total number of Queries allowed by Twitter per day

k➔ Maximum number of results a query can return

T_1, T_2, ... T_N ➔ Topics of interest

X_i ➔ Expected number of tweets returned for topic T_i.

Utility Functions used:

U (X_i) ➔How good the tweet is to gather the information about the topic T_i and whose expected value is X_i

U(X,i)➔ i is the topic's index, X is the tweet.

ASSUMPTIONS:

A1. Twitter can repeat randomly or not repeat the tweet for a query.

A2.1 U(X) = c

A 2.2 U(X) is decreasing

A3. All the topics (T_i's) are independent.

A4. No repetition of tweets is allowed.

A5. X_is are exact number of tweets that a query 'q' returns on T_i.

The first part of Problem discussion containing Introduction, detailed description of problem, solution approaches can be found at http://www.cs.sunysb.edu/~jgao/CSE642/Alg9-2-11.pdf

SOLVING THE PROBLEM:

Considering few of the above said assumptions, one of the approaches to solve the problem is as follows.

Considering the case where assumption A1, A2.1, A3, A4, A5 hold good.

Start with T_1 and continue adding topics to each query until

$X_1 + X_2 + X_3 + ... + X_n >= K$

So always adjust (or gather) X's which satisfy the above condition and add new topics as necessary. If X_i's are 'exact number' of tweets that a query 'q' returns on T_i, then we can get min {qk,n} every time.

REMARKS:

Remark1: Above said algorithm works optimally with above assumptions but so does querying

T_1 UT_2 UT_3 U... T_n every time.

Response: There is no agreement or contradiction in group's response to this question.

Remark2: Can we change the assumptions and still have a good algorithm?

Change to Utility function: No comments.

Repetition allowed: The solution proposed above is a good algorithm if repetitions are allowed.

Remark3: There is query if q should be less than n.

Response: There is no such constraint.

The first part of Problem discussion containing Introduction, detailed description of problem, solution approaches can be found at http://www.cs.sunysb.edu/~jgao/CSE642/Alg9-2-11.pdf

DISCUSSION:

There is proposal of constraining the Expectation value X_i such that the whole problem can be boiled to bin packing problem or 3 partition problem.

3 Partitions Problem:

> **Given '3n' numbers**
>
> **a_1, a_2, a_3 ,...., a_{3n} such that $(1/4 <= a_i <= 1/2)$**
>
> **Can we divide the numbers into a group of 3 where each group adds up to exactly 1?**

Remark 4: This problem is strongly NP-hard.

Now, can we use 3 partitions problem to prove the hardness of our problem?

REDUCTION:

> **Suppose there are 3q topics where N=3q and k/4<=Xi<=K/2**
>
> **Consider a case where we can divide the topics into a group of 3 where each group expectation value adds up to exactly K?**
>
> **Hence with all above constraints our original problem is reduced to a special case of 3 partition problem.**

Conclusion:

Solving our problem in any instance proposes a fast algorithm which can be used to solve the 3 partition problem. Since as 3 partitions problem is NP hard, there will not be any fast algorithm which can solve it and hence there can be no fast algorithm which can solve our problem. Hence, trying to solve our problem is as hard as solving 3 partitions problem.

The first part of Problem discussion containing Introduction, detailed description of problem, solution approaches can be found at http://www.cs.sunysb.edu/~jgao/CSE642/Alg9-2-11.pdf