Group 3 :-

1. Mayank Goyal(1401CS25)
2. Naman Agarwal(1401CS28)
3. Dnyaneshwar Shendurwadkar(1401CS43)
4. Vipin Mavi(1401CS48)
5. Hitesh Golchha(1401CS56)

Paper Name:More Like This: Query Recommendations for SQL
Author: Christopher Miles , Dept. CSE,University of Washington

Data used:

- Exposing over 30 TB of data on approximately 500 million celestial bodies via SQL, The Sloan Digital Sky Survey (SDSS) is a widely used resource within the astronomical community.
- Over 90 tables, 200 user- defined functions, and 3,400 columns.
- *Requirement: context-aware suggestions based on partial information entered by the user. Recommendations are drawn from similar past queries authored by other users*

More Like This:

(a) Query representation as abstract feature strings:

- Feature representation:
  The feature set of a query q is defined as: features(q) = {f | f(q) = true} . Features are extracted from the FROM, SELECT, WHERE, ORDER BY, GROUP BY, HAVING, and DISTINCT clauses of a query (or nested query). Features have form: <clause>-<expression>
- Abstaction:
  To ensure structurally similar queries are recommended, constants from features are replaced with placeholders(abstracted). Recommendations should share maximum features with the query.

(b) Query Similarity:

- A simple ranking function is computed by summing the tf-idf scores for each feature extracted from the input query.
- Given a pair of queries Q_i , Q_j and their associated feature sets F_i , Fj , similarity is defined by the following formula.

$$similarity(F_i,F_j)=\sum_{f\in F_i} tf(f,F_j)\times idf(f)$$

$$idf(f)=\log\left(\frac{n_{QR}}{|\{Q\,|\,f\in features(Q)\}|}\right)$$

$$tf(f,F_j)=\frac{n_{f,F_j}}{|F_j|}$$

- Term frequency (tf) measures the importance of feature i to feature set Fj. Inverse document frequency (idf) measures the general importance of a feature f .

(c) Subjective, per-clause Weights:

Extension to tf idf to prevent bias towards selective features, to diversify recommended queries and to give additional weights to some features which may be difficult for users.

(d) Diversifying the Results:

Making sure the results are not too similar to queries, they are derived after being constrained to two conditions:

- Maximize similarity to the input query Q.

$$\sum_{i=1}^{K} similarity\left(Q,R_i\right)$$

- Maximize The diversity of the set of recommendations.

$$diversity\left(R\right)=\sum_{i<j}^{K}-similarity\left(R_i,R_j\right)$$

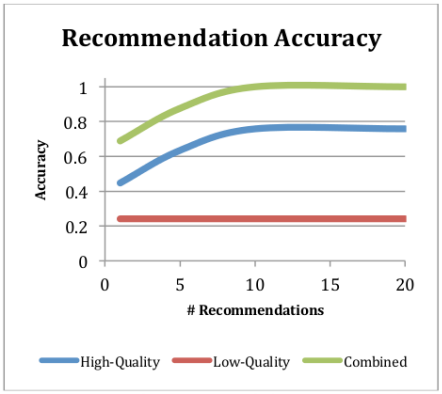

Figure 1. Depicts recommendation accuracy on the predicted user-query pairs as a function of the number of recommendations.

(e) *Evaluation and results:*

- random sample of 10,000 queries logged by the SDSS server
- invalid queries and exact matches removed.
- a simple heuristic based on edit distance was devised for identifying pairs of queries predicted to have been issued consecutively by the same user.

(f) *Future Work:*

- integrating MLT functionality within an actual RDBMS and known user query attribution must be used.
- Conducting a series of controlled user studies.
- Gaining a better understanding of the interplay between idf scores and subjective, per-clause weighting