RC25332 (WAT1209-028) September 12, 2012
Computer Science

IBM Research Report

On the Accuracy of k-Nearest Neighbors in MongoDB

Dakshi Agrawal, Raghu Ganti, Mudhakar Srivatsa
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 208
Yorktown Heights, NY 10598
USA

Research Division
Almaden - Austin - Beijing - Cambridge - Haifa - India - T. J. Watson - Tokyo - Zurich

On the Accuracy of k-Nearest Neighbors in MongoDB

Dakshi Agrawal, Raghu Ganti, and Mudhakar Srivatsa
IBM T.J. Watson Research Center
Yorktown Heights, NY
agrawal,rganti,msrivats@us.ibm.com

ABSTRACT

MongoDB, a popular open source object-oriented database supports geospatial objects and querying on these objects. A frequently used geospatial query is that of k-Nearest Neighbors, finding the closest k neighbors. In this paper, we show that the geodetic implementation of the nearest neighbors in MongoDB version 2.2.0 is incorrect.

1. INTRODUCTION

MongoDB [1] is a scalable, high-performance, open source NoSQL database that is written in C++. This database provides support for storing and querying geospatial objects (GIS data). It supports data types such as geodetic points (specified as latitude/longitude), Cartesian points (specified as x/y), rectangles, circles, and polygons. The various types of queries supported[1] are nearest neighbor searches and bounds queries with support for both geodetic and Cartesian queries. The assumption of Cartesian coordinates/queries is that the earth is flat and the geodetic queries/coordinates is the earth is ellipsoidal (in some cases, spherical assumptions are also made). MongoDB provides an option for spherical earth model in its computations of nearest neighbors.

In this paper, we evaluate the accuracy of MongoDB's nearest neighbor searches. We show that the results of the nearest neighbor searches using the spherical earth model in MongoDB are inaccurate. We also show that both under-estimation (excluding neighbors that are in the search radius) and over-estimation (including neighbors beyond the search radius) can occur.

2. EVALUATION

In this Section, we will evaluate the accuracy of nearest neighbors using the geoNear command of MongoDB assuming a spherical earth model. An example query is of the form:

```
distances=db.runCommand(geoNear: "testKNN", near
: lon : 18.027665, lat : 59.299086, spheri-
cal : true, maxDistance : 0.3 / 6378, num : 20).re-
sults;
```

In the above query, `testKNN` is the table which stores points and the specified point (18.027665, 59.299086) is the point of interest, the search radius is 300 meters (earth radius is assumed to be 6378 km, as specified in the example in MongoDB geospatial indexing [2]).

We generate a dataset of 6692 points (represented as latitude/longitude pairs) from Stockholm, Sweden. This data were collected from taxicabs in Stockholm, Sweden using GPS devices. Points are chosen from a one hour time window in the month of January, 2008. The ground truth for the nearest twenty neighbors within a range of 300 meters for each selected point is computed using Vincenty's [4] formula for distance computation assuming a WGS84 ellipsoid [3]. We compare the results obtained by executing the query on MongoDB as described above with the ground truth results obtained from applying Vincenty's formula.

We compute the type I (false negative) and type II (false positive) errors. Type I errors occur when MongoDB returns results that are not part of the true result set and Type II errors occur when the true results are missing from the result set of MongoDB. We compute the average type I and type II errors across all the 6692 queries (corresponding to the 6692 points) and illustrate them in Table 1.

Type I	Type II
7.02	0.134

Table 1: Type I and type II errors for MongoDB nearest neighbor query

We observe from Table 1 that MongoDB returns many results that are not usually present in the true result set (Type I errors), whereas it misses only a few results from the true result set (Type II errors). But, we observe that MongoDB generates results that are inaccurate (with both types of errors). A closer inspection revealed that the distance (from the search boundary) for

[1]The description of MongoDB assumes the current version, ver. 2.2.0.

the Type I error points can be quite large (on an average of 1000 meters). This inaccuracy points to a possibly incorrect usage of the GeoHash based indexing mechanism, which is the geo-spatial indexing mechanism used when the points are represented as latitude/longitude pairs (in MongoDB). Thus, we conclude that based on extensive experimental results that the k-Nearest neighbor queries assuming the spherical earth model in MongoDB is inaccurate.

3. REFERENCES

[1] Mongodb. http://www.mongodb.org.

[2] Mongodb geospatial indexing. http://www.mongodb.org/display/DOCS/Geospatial+Indexing.

[3] N. G.-S. I. Agency. Dod world geodetic system 1984. *NIMA Technical Report*, (TR8350.2), July 1997.

[4] T. Vincenty. Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. *Survey Review XXIII*, 176:88–93, April 1975.