Ski Genius Tech Review

Ashwin, Chakita, Kyle (1), and Kyle (2)

Background and Use Case

Our project is a recommendation platform for ski resorts that uses a dataset of resort details to provide answers to user queries.

The component we want to analyze is the database retrieval portion of the project. We need a way to browse our data and return the most similar results to the user's selected query.

The particular use case (workflow) here is: User provides us with a set of resort constraints via the front end interface, we query the backend database, and we display the returned results to the user.

Package Choices

Many database choices:

- SQL (MySQL, PostgreSQL, etc)
- NoSQL (MongoDB, Cassandra, etc)

NoSQL databases are more flexible for our needs

Python libraries to consider:

- Pymongo
- Cassandra-driver

Pymongo

- Better documentation
- Looks easier to set up
- More flexible schema
- MQL queries (very similar to SQL)
- Open source
- Built-in aggregation functionality
- Slower writes, faster reads
- More storage intensive

Cassandra-driver

- Better scalability for large datasets
- CQL queries (Cassandra Query Language), which are a little complicated
- Better error redundancy (peer to peer architecture)
- Open source
- Faster writes, slower reads

Our Chosen Package

After comparing the packages, it looks like **pymongo** is the better choice for a variety of reasons:

- It is simpler to set up and has better documentation (saves us time making the database)
- It has faster read queries, which is important since we will be almost exclusively reading from the dataset as opposed to writing.
- It has more straightforward querying (MQL > CQL)

Drawbacks and/or Remaining Concerns

Not too many drawbacks to pymongo. Almost all the functionality we want is provided for us and mongodb is a reliable database system that is commonly used.

- Possible Drawbacks:
  - The application is read intensive so there might be some latency induced due to multiple queries. Might need to optimize query if the latency times turn out to be too long.
  - MongoDB is storage intensive which likely won’t be an issue given the scale of our project, but important to note.

Package Demonstration

Colab notebook