Smart Document Retrieval System:

The objective is to create and implement an information retrieval project that utilizes Elasticsearch for the indexing and retrieval of documents. This project will involve extracting temporal expressions and georeferences from the documents. The ultimate aim is to facilitate spatiotemporal and textual queries, allowing users to search for information based on both time-related and geographical aspects, in addition to traditional textual queries. This comprehensive approach enhances the capability of the system to handle a wide range of queries.

Objectives and Expected outcome

The expected outcomes of the project are as follows:

1. ElasticSearch Mappings and Settings: Students should build the index mapping and settings after understanding the project objectives specified below.

2. A Document indexing tool that reads a collection of documents that has the following capabilities:
   - ingestIng and indexing a collection of documents into an Elasticsearch index, ensuring proper storage of titles, content and other entities.
   - Utilizing third-party tools to process the content of documents and finally extract temporal and georeferenced tokens. One tool (temporal extractor) to extract temporal expressions like dates. Another tool to identify place names (georeference) like the name of streets, cities, countries etc. These extracted entities should be included within each document in the index as an array of entities.
   - Use a proper temporal expression extractor
   - User a proper georeferenced extractor

A Document indexing tool that reads a collection of documents that has the following capabilities:

Try this tool English · spaCy Models Documentation

Have a look at this geocoder: geocoders: to convert place names into coordinates

Building An ElasticSearch index where each document has the following suggested structure:

- *Title*: the title of the document.
  - It should be analyzed to provide autocomplete search.
- *Content*: The content of the document, which should be analyzed such that
  - stop words, tokens of length less than 3 characters, html are eliminated
  - words are stemmed

- *Authors*: the authors of the documents
    - Use nested arrays to store a list of authors.
    - Each array element is a nested object with first name, last name, and email
- *Date*: the date and time of publication. Use date object for this field
- *Geopoint:* used to save the longitude and latitude coordinates of a suggested location associated with the document.
- *TemporalExpressions*: a list of temporal expressions extracted from the documents
- *georeferences*: a list of georeferenced expressions extracted from the documents

NOTE: If the document to be indexed has no explicit (*Date or Geopoint*), use the extracted temporal expressions and georeferences to approximate them.

3. A smart query processing and analytics engine that is able to answer spatio-temporal queries and provide related analytics.
    - Providing an autocomplete service to return a list of top-10 documents based on their titles. The system should start suggesting titles after the 3rd typed character. Take into consideration that the user might write some misspelled words.
    - Lexical and semantic Retrieval of relevant documents by considering both title and content, with greater emphasis placed on the title. Also, take into consideration the recency and localization factors while ranking the results. Note that the query from the user is represented as a tuple (query, temporal expression, georeference)
    - Return the top-10 mentioned georeferences across the entire index.
    - Return the distribution of documents over time, with a time aggregation of 1 day.