BAX-422 Data Design and Representation
(Final Project)

Food Network Database

Group Members

Kshitij Karan

Priyanka Murugan

Suzana Amer

Table of Contents

Executive Summary

Our client is a food technology startup that aims to launch a food recommender app in partnership with food networks. The goal of the app is to provide a one-stop shop for health enthusiasts, amateur chefs, and dieticians to find quick and easy recipes based on their preferences, limitations, and goals. The app will offer personalized and nutritionally sound recommendations, making it easier for customers to achieve their dietary goals. As consultants, we proposed creating a database for the app by scraping data available on the web, focusing on the Food Network website first, to gather valuable information about the recipes, their ingredients, and nutritional values. This database will be used to train machine learning algorithms that can make personalized recipe recommendations based on user input.

We have scraped over 3000 recipe pages from the Food Network website to capture relevant information related to their diet, based on nutritional value, and find similar recipes based on categories and other celebrity chef recipes. We used Python's BeautifulSoup and Requests packages to scrape the website pages. Due to the huge volume of the data that we wanted to scrape we utilized several custom-built functions and built a scraping routine that would scrape as well as process the data to ensure that we have top-notch data quality, with standardized data formats, and also make it easier for users to query and access our data. We choose to use MongoDB over others to store the data due to its flexibility and high performance. Overall, our web scraping and database design processes have resulted in a comprehensive recipe database that can be used to build a successful food recommender app.

Introduction

1.1 Background, Context, and Domain Knowledge

The food industry is a highly competitive sector that requires constant innovation and adaptation to changing customer needs and preferences. With the rise of health awareness and increasing focus on health and nutrition, many customers are looking for quick and easy ways to find healthy and nutritious recipes. This has led to the growth of food technology startups that aim to simplify the process of finding and preparing healthy meals.

To meet these changing demands, the food industry has to evolve. Food networks are now partnering with technology startups to provide their customers with a personalized experience and access to a wider range of recipes. This partnership not only benefits the customers but also provides an additional revenue stream for food networks.

Our client is a food technology startup that is planning to launch a food recommender app in partnership with food networks. The goal of the app is to provide a one-stop shop for health enthusiasts, amateur chefs, and dieticians to find quick and easy recipes based on their preferences, limitations, and goals. The client is also planning to gather data from other websites to improve the app's recommendations.

In this context, our client's food recommender app is a promising business opportunity. By partnering with Food Network and other websites to gather data, the app can provide customers with personalized and nutritionally sound recommendations, making it easier for them to achieve their dietary goals. This can help our client gain a competitive edge in the food technology market.

To achieve our goal, we will be scraping data available on the web, for now we focus on the Food Network's website which will be highly useful in creating a successful food recommender app. The collected data will enable our client to gather valuable information about the recipes, their ingredients, nutritional values, and categories. It can also be used to train machine learning algorithms that can make personalized recipe recommendations based on user input.

Analyzing the data will help our client understand the food network industry better by identifying the most popular recipes and their ingredients. We can thus create a recipe database that caters to a wide range of customer needs and preferences. This can be used to create more accurate and personalized recommendations and provide insights into customer behavior and trends.

Web Scraping, Database Design and Data Cleaning Process

2.1 Introduction to data source

Our data source is Food Network's website from where we scraped 3000+ recipe pages and 200+ chef pages to capture relevant information. Food Network is a popular website and TV network that focuses on various cuisines and their cooking procedures. The website features a wide range of recipes and food-related content that appeals to both amateur and professional chefs. The site is organized into several categories, including recipes, and food shows.

The recipe section of the website includes thousands of recipes that are searchable by ingredients, course, cuisine, and dietary restrictions. Each recipe includes a list of ingredients, step-by-step instructions, nutritional data, and the categories it falls under. The shows section includes information about the various cooking shows that air on the Food Network TV network and the recipes that were broadcasted in each episode. Overall, Food Network is a comprehensive resource for anyone interested in cooking and food.

We have scraped the website by building a custom web scraping routine to create a comprehensive recipe database that is designed to cater to the needs of our users. This will act as a one-stop-shop solution for food lovers who are looking for quick and easy recipes, want to structure their diet based on nutritional value, and find similar recipes based on categories and other celebrity chef recipes.

2.2 Description of Scraping routine

We used Python's BeautifulSoup and Requests packages to scrape the website, which organizes chef and recipe webpages alphabetically. To ensure that we could extract all required attributes, we first built our scraping routine for a single alphabet page. Once we confirmed that the routine was working, we scaled it up to all 26 alphabet pages, with 150 recipes per page, resulting in over 3000 recipe pages. To create a balanced database with the same number of recipes from each alphabet, we only chose the first page of each alphabet to scrape. Given the volume of data we were working with, we had to avoid overloading the server with requests and encountering a timeout error. To address this, we created batches of four alphabet pages, with 600 recipes per batch. We added a two-minute sleep time between batches and a 10-second delay between each recipe to ensure smooth scraping and avoid any duplication of recipe pages. In contrast, we were able to scrape the roughly 200 chef websites without the need for batching or saving pages since it did not pose a risk of causing a timeout error.

2.3 Database Design Description

2.3.1 MongoDB as Database Choice

We decided to use **MongoDB** over others, primarily because it offers us the flexibility to store dynamic attributes as embedded documents within a collection. For instance, in the

"recipe_attributes" collection, the "ingredients" variable is dynamic and varies across different recipes. By saving it as an embedded document, we can easily expand and reveal all the ingredient items without any further processing by the users. This approach helps to keep the data organized and makes it easier to work with. We have also adopted the same design approach for other attributes such as "shows" and "episodes" in the "recipe_shows" collection and "categories" in the "recipe_categories" collection. By embedding these attributes as documents, we can store related information together and retrieve it efficiently when needed.

In addition to embedding, we have utilized linking in MongoDB to relate collections. For instance, we have linked the collections using unique identifiers such as “recipe_id” and “chef_id”. This allows us to access related information from different collections easily and avoid data duplication. Finally, MongoDB's high-performance capabilities make it an excellent choice for our project, especially since we are handling more than 3000 recipes. With MongoDB, we can ensure that our database is processing data quickly and accurately, without compromising performance or scalability.

2.3.2 Database Design

We have organized our database into 5 distinct collections to better serve our users and the ERD is available (Appendix, Fig 1). Our primary collection, **"recipe_attributes,"** contains essential information about each recipe, including its name, cooking time, required skill level, and the chef who created it. This collection serves as a quick reference for users who want to access this crucial information without being overwhelmed by other data. We also have additional collections to provide more in-depth information about each recipe. For example, our **"recipe_nutrition"** collection includes detailed nutritional information like calories, fats, protein, and carbohydrates, which is associated with each recipe in the "recipe_attributes" collection. Our **"recipe_categories"** collection is designed to make it easier for users to find recipes based

on their preferred cuisine or food category. From the embedded documents in the category variable, user can access the website URL of a specific category, and find other recipes with a similar profile. The **"recipe_shows"** collection contains information about the shows and episodes in which these recipes were featured and broadcasted on Food Network. Finally, we created the **"a_z_chefs"** collection to provide users with a comprehensive list of their favorite chefs' social media websites. This way, users can stay up-to-date with their favorite chefs' latest posts, recipes, and culinary adventures. By organizing our database in this manner, we believe that we have created a user-friendly and comprehensive database that caters to the needs of all users.

2.4 Data cleaning, Formatting, and Standardization Process

To ensure data consistency and improve data quality, we took several steps in data processing. Firstly, we addressed the issue of missing values by replacing them with "N/A" for categorical variables and "0.0" for numerical variables. We also implemented RegEx and custom functions to process and standardize certain string variables in our data. For example, in the "recipe_nutrition" collection, we converted variables containing various weight formats to "grams" and returned them as float values. We standardized all the time variables in the "recipe_attributes" collection by creating a function called "convert_to_minutes," which takes data in various time formats and returns the corresponding time in "minutes" as float values. To improve data readability, we renamed certain variable names to a more understandable format. Additionally, we created "recipe_id" and "chef_id" as unique identifiers in our collections and made them into an **index** for their respective collections, which improves performance in searching and querying. Overall, by implementing these data processing techniques, we were able to improve data quality, standardize data formats, and make it easier for users to query and access our data.

Business Impact of our proposed database solution

3.1 Business value of our solution

The food database that we put together will enable the business to build a recommendation system that will allow the users to find food recipes based on their preferences, dietary restrictions and goals. The main goal of this application is to serve all user needs when it comes to food recipes. It will be the one-stop shop for diverse food cuisines while offering various options in which every user will be able to find something for themselves, regardless whether they are trying to look for easy recipes, nutritious recipes, vegetarian/vegan recipes or even get a little more sophisticated with their cooking skills. The application will be able to tailor to each user experience depending on their preferences. The user will just need to answer a few simple questions related to their food interests and dietary goals which in that case the recommendation system will be able to provide the user with a list of unique food recipes meeting their requirements.

As we previously mentioned, the database will not only contain the recipe information but it will also contain nutritional related information such as calories, total fat, carbohydrates and much more. This information will feed into the recommendation system allowing the business to have a nutritional section in their application that will attract health enthusiasts which will generate more user traffic on the app. In addition to that, the database contains information around the time it takes to complete the recipe such as total time, prep time and cook time. All of these attributes will provide the consumer with the flexibility to choose between easy, medium and hard recipes which will not only attract users that are looking for quick easy meals but also users that are interested in experimenting with a bit more complicated recipes. The recipes featured in various shows by famous chefs will attract youths and social media enthusiasts that are interested in experimenting with famously featured dishes. Overall, the food database contains

a variety of attributes that will have diverse consumer profiles providing them with unique and tailored food recipes based on whatever goals or needs that they are trying to achieve.

3.2 Business value with MongoDB as Database choice

As we previously stated, the database is quite comprehensive and includes vast amounts of data as well as attributes. For that reason, the database structure requires a high-performance database management system to ensure the efficient storage and retrieval of data. When it comes to selecting a database management system for the database, we had two popular options: Mongodb and MySQL. While both databases have their advantages, MongoDB is a better fit for this data due to its unique features. One of the key advantages of using MongoDB is its flexible document-oriented data model. In a document-oriented database, each record is stored as a document, which allows for easy manipulation and querying of data. However, MySQL uses a relational data model, which can be more rigid and less flexible when dealing with complex data structures. The designed database contains five collections, each with various embedded documents. MongoDB's document-oriented model allows for easy management of complex data structures such as this one. This allows for faster data retrieval and reduced processing times. Another benefit of MongoDB is its ability to handle unstructured data. The database contains description data which makes MongoDB better suited to handle such information than MySQL for example. Lastly, MongoDB offers better performance than MySQL for specific types of queries that the business will need to run when retrieving the various embedded documents. Overall, MongoDB has the unique ability to handle unstructured data and provides the business with various scalability options, if additional data needs to be scrapped and appended. All of these advantages will allow the company to ensure efficient storage, retrieval and processing of data, while providing a better user experience for its customers.

Conclusion

In conclusion, the food industry is constantly evolving to meet the changing demands of customers who are increasingly concerned with health and nutrition. Our client, a food technology startup, aims to launch a food recommender app in partnership with food networks that will provide personalized and nutritionally sound recommendations. The scraped data we provided will be used to train machine learning algorithms that make personalized recipe recommendations based on user preferences and limitations. The data will also help identify popular recipes, trending ingredients, and cooking techniques that cater to a wide range of customer needs and preferences. Overall, the food recommender app is a promising business opportunity that can help our client gain a competitive edge in the food technology market and create a sustainable revenue stream for food networks.

Appendix

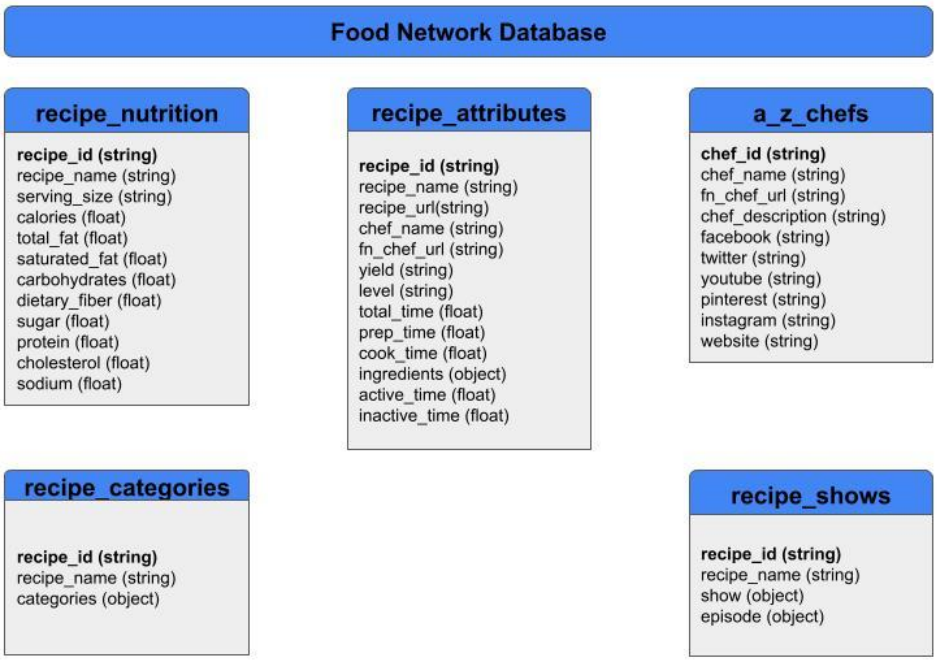

Fig 1: The food network database