User Documentation

Usage

To use the Ailixir application, you first need to specify input sources to scrape data from. A `config.json` file will be generated from them. Then, we start the data acquisition pipeline by executing the orchestrator component, which manages this process. Ensure that the specified data sources are accessible (free to use), as they significantly impact the AI's data quality and overall performance.

Configuration

Before using the main commands in the following section, update the scraping targets in `src/backend/Config/main.py` for example like this:

```
if __name__ == '__main__':
    c = Config()
    c.add_target(YouTubeTarget(
       url='https://www.youtube.com/@NutritionFactsOrg')
    )
    c.write_to_json()
```

Data Sources

Here is the list of available data scraping targets with example usage:

- Recipes from allrecipes.com
- Articles from arXiv
- Podcasts from the Peter Attia Podcast Archive
- Articles from PubMed
- Youtube channel (retrieve information from all videos on the channel)
- Nutrition related blog posts from NutritionFacts.org

```
# Without any parameters, it scrapes from allrecipes.com
AllRecipesTarget()
# The first argument specifies which keywords to look for
# and the second one limits the number of results for each keyword
ArchiveTarget(keywords=['nutrition', 'health'], max_results=1)
# The first parameter is the base URL link for scraping, don’t change it
# and the second one limits the number of podcasts to be retrieved
PodcastTarget(
   url='https://peterattiamd.com/podcast/archive/', num_podcasts=1
)
# The first argument specifies which keywords to look for
# and the second one limits the number of results for a set of keywords
```

```
PubMedTarget(keywords=['food', 'exercise'], max_results=1)
# The first parameter specifies the channel to scrape data from (all
videos of this channel will be scraped)
YouTubeTarget(url='https://www.youtube.com/@NutritionFactsOrg')
# The first argument specifies the base URL link, don’t change it
# and the second one limits the number of pages to scrape
NutritionTarget(url='https://nutritionfacts.org/blog/', max_pages=1)
```

Running the Data Acquisition Pipeline

To start the data acquisition pipeline, execute the following commands:

```
# Build data/config.json file and auxiliary folder structure
pdm build-config
# Scrape all targets specified in src/backend/Config/config.py
pdm run-orchestrator
```

The data will be stored in the `data/` directory. For example for youtube scraped data, you can find the scraped data under `data/youtube/raw/`. Unique ids of scraped youtube videos are stored in the file `data/youtube/index.json`.

Environment Variables

API keys and other sensitive information cannot be stored in the public code repository. When running the application for the first time, the user will be prompted to provide these values. The environment setup script ensures these values are stored in a `.env` file. The data acquisition pipeline only requires the `YOUTUBE_DATA_API_V3` key. We obtained some of these keys from our industry partner.

```
YOUTUBE_DATA_API_V3=""
GOOGLE_GEMINI_API=""
OPEN_AI_API=""
ASTRA_DB_TOKEN=""
ASTRA_DB_URL=""
```

Additional Commands

To perform specific scraping tasks, execute the corresponding commands:

```
# Scrape AllRecipes
pdm scrape-allrecipes
# Scrape Podcast
pdm scrape-podcast
# Scrape PubMed
pdm scrape-pubmed
# Scrape YouTube
```

```
pdm scrape-youtube
# Scrape Archive
pdm scrape-archive
# Scrape NutritionFacts
pdm scrape-nutritionfacts
```