RecycLens - Team extra-fluffle

Proposal

Problem or question

Objective

Our general objective to build and deploy a web application that serves real use to people, and uses LLM technology for good. We decided the problem we want to focus on is when people don't know how to dispose things properly, whether it's recyclables, electronics, or hazardous materials. Another specific objective to have our web application be smooth, easy, and sensible for users to navigate.

LLM Usage

We want to address this problem using an LLM where users first can input their location (zipcode, county, etc.) and interact with a chatbot to ask their questions about recycling. Some examples we envision are:

- "How and where should I dispose batteries?" (if needed, the LLM should follow up with more questions about specific type of batteries and condition of batteries)
- "Is my Chipotle bowl recyclable? Does it go in paper only, cans, or plastic?"

The LLM's reponse should contain detailed answers along with a map of the closest relevant centers near the user's location. We plan to give structured data to the LLM so it can consistenly output things like the county's offical recycling page URL, contact information, and hours of operation. LLMs are suited for this problem because people face complex everyday scenarios and best describe their situation through natural language, and a normal predictive model can't really give as much catered information as possible to the user in the form of a series of instructions along with supplemental info. They're also useful since we can train them on a very particular knowledge base of recycling sites in New York, so they're able to summarize a lot information available scattered throughout the Internet (which is time-consuming for humans to do) in a concise manner on our web application.

Importance

By building this application, we hope to encourage better and safer recycling practices. This is a common confusion in people's daily lives, but by creating a quick solution to answer these questions, people can dispose things correctly more often instead of navigating through social media or other guidelines that aren't correct for their location. This would also ideally help recycling centers not have to sort as much misplaced recyclables.

Feasibility

We plan to start building our RAG with Tompkins County guidelines and focus on getting good performance with one county. Eventually though, we would like to expand our scope to all New York counties. We also hope to eventually integrate a feature where the chatbot can take in multimodal input such as a user's image of a container they want to dispose.

Data sources

Our data collection is all first-hand data that we manually collected. To build our RAG store, we looked at Wikipedia for a list of all 62 counties in New York. We then manually found relevant urls of the recycling and solid waste information of that county's website, and stored it in a CSV file. This is stored in the /data file.

Overall, we observed that most sites had a structured process to access the recycling info. For most sites, we clicked on the Departments tab, looked for either Recycling, Solid Waste, or Public Works links, and then these pages had a lot of sublinks to explore. Some sites had information embedded in PDFs or images, which could be a challenge for the LLM to recieve.

Final product

RecycLens is an AI-powered recycling assistant that helps users quickly determine whether an item is recyclable and where it can be disposed of properly - using only a photo and their location. Users upload an image, enter their city or ZIP code, and RecycLens handles the rest (they can also add additional context if they choose). The system uses the OpenAI Vision API (GPT-4o) to identify the materials and condition of the item in the uploaded image. This vision analysis is then passed to the OpenAI Responses API (GPT-4o with web search), which evaluates local recycling rules, determines whether the item can be recycled, and retrieves nearby recycling or disposal facilities. These locations are displayed both as interactive cards and on a Mapbox-powered map for visual clarity. While the MVP relies on web search for location-specific recycling rules, the final product will also incorporate a Retrieval-Augmented Generation (RAG) system covering all New York State counties. This will allow RecycLens to provide highly accurate, hyper-local recycling guidance based on county-specific regulations, materials lists, contamination requirements, and facility rules. RecycLens ultimately aims to make sustainable disposal intuitive by leveraging LLMS to turn a confusing process into a simple, guided experience.

Ethical concerns

Accuracy & Hallucinations - People may see our LLM as a source of truth regarding recycling. Incorrect intstructions may be detrimental to our user. To mitigate this, we should:

```
- Use RAG with verified county-level guidelines to reduce hallucinations.

- Provide structured, citation-style outputs that always link back to official
county resources.

- Clearly mark responses as informational rather than legally authoritative.
```

Warnings & User Verification - Because recycling rules differ by county and evolve over time, even correct model outputs may become outdated.

```
- Include automatic links to the official county recycling pages in every
response.

- Display a persistent banner reminding users to verify disposal instructions
on their local government website.

- Use UI cues (e.g., “Check local rules”) to maintain transparency.
```

Safety & Hazardous Waste - Batteries, chemicals, electronics, sharps, or other hazardous materials can create fire hazards, contamination, or injuries. - Include a small disclaimer on the UI which states things such as chemicals, batteries, and more should not be recycled

Project timeline

11/18–11/20: Idea Validation & Calling MVP

- Make simple MVP for calling to ensure it works using react and express (done).
- Integrate Mapbox API to calling MVP to ensure that the integration works (done)
- Make proposal document. (done)
- Talk to Professor Soltoff and get his feedback (done).

11/20–11/25: Build Knowledge Layer & SHINY MVP

Build MVP

- Incorporate current call system into a Shiny app
- Integrate Map into shiny

Build Knowledge Store

- Create a knowledge store for RAG.
- Collect county-level recycling data for all NY counties.
- Chunk, embed, and store data in a structured knowledge base.
- Validate coverage and ensure consistency with source documents.

11/26–12/04: Integrate RAG and Test

Integrate RAG

- Incorporate RAG into Shiny MVP.
- Modify backend pipeline to add retrieval before the Responses API call.
- Add fallback logic between RAG ⟶ API web search ⟶ default rules.

- Ensure county detection and knowledge routing work reliably.

Testing & Iteration

- Conduct functional testing across common item categories and multiple counties.
- Log retrieval quality and hallucination cases; refine chunking/metadata.
- Run small usability tests (clarity of instructions, flow, UI smoothness).
- Refine UX and overall design.
- Identify latency bottlenecks (image processing, embedding lookup, multi-step calls).
- Improve UI clarity (location input, image upload, results view).

Report Draft - Create rough draft of the report; ensure alignment with rubric.

- Begin writing sections for architecture, methods, challenges, and evaluation.
- Cross-check progress against rubric requirements.

12/04–12/11: Final Improvements

Convert Shiny to React & Express

- Convert SHINY application to react (front end is already built, as is backend)
- Integrate RAG into it

Peer Review Revisions

- Review peer comments and draft score.
- Create prioritized action-item list (accuracy, UX, clarity, etc.).
- Address major issues and document changes.

Product Adjustments

- Improve RAG accuracy, chunk retrieval, and county routing.
- Update UI components (cards, map display, confidence explanations).

12/11-12/18 Eval & Report

Finalize Report & Product

- Update all report sections with final architecture, evaluation data, screenshots, and metrics.
- Write conclusions, limitations, and future work.
- Ensure full rubric compliance.

Final Checks

- Confirm reproducibility and setup instructions.
- Perform a full end-to-end demonstration for instructors/TA.