CENG SURP PROJECT 2023

Integrating Human Expert Knowledge with OpenAI and ChatGPT: A Secure and Privacy-Enabled Knowledge Acquisition Approach

Jenny Wang and Ben Phillips

Cal Poly
Noyce School of Applied Computing
COLLEGE OF ENGINEERING

Problem Statement

Advanced Large Language Models (LLMs) struggle to produce accurate results and preserve user privacy for use cases involving domain-specific knowledge.

A privacy-preserving approach for leveraging LLM capabilities on domain-specific knowledge could greatly expand the use cases of LLMs in a variety of disciplines and industries.

Our Solution

- We used a Named Entity Recognition model to detect private information in text and alert the user to remove or replace it.
- We developed a system to extract relevant user information for a query and store user data in a knowledge base.
- Then, we used an LLM to answer a user's question with their desensitized data.

Use Cases

Computer Science Faculty:

A professor uploads course resources from her website and uses the system to create a chatbot that can answer students' course-related questions.

IME Faculty:

Faculty can upload domain-specific data to increase productivity with a virtual GPT-based assistant equipped with domain-specific knowledge.

Automatic Reply Agent:

Users can use the system to automatically produce helpful responses to private emails with information from relevant internal resources.

Ethical Implications

Our project project primarily addresses issues concerning:

- **User privacy**—by removing sensitive information from LLM prompts
- **Response Accuracy**—by leveraging domain-specific knowledge to improve LLM results

System Design

The system contains the following four components:

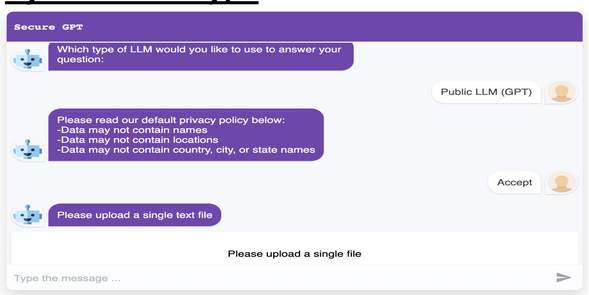
Component 0:
- Collects the user's privacy preferences.
- Determines use of either a public LLM or a private LLM.
- Collects user data and their questions

Component 1:
- Uses a ML model to scan the user's data for private info.
- Allows user to replace/remove private info
- re-evaluates data until no private information is detected in the user's data.

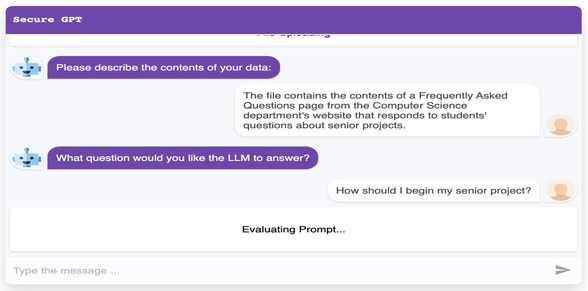
Component 2:
- Determines if the user's question has been previously answered by the LLM.
- If not, the system uses a similarity search to retrieve documents related to the query and sends them to Component 3.

Component 3:
- Sends data to LLM with the user's query
- Returns response generated by an LLM based on the user's data.
- Logs interaction details, including question, response, and feedback.

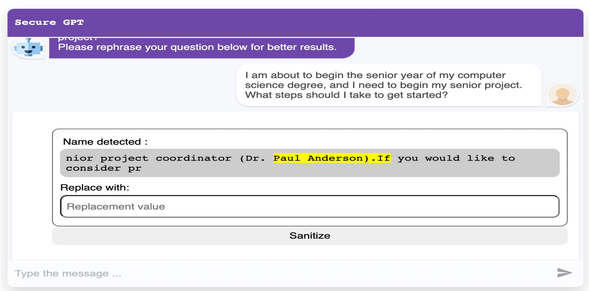
System Prototype



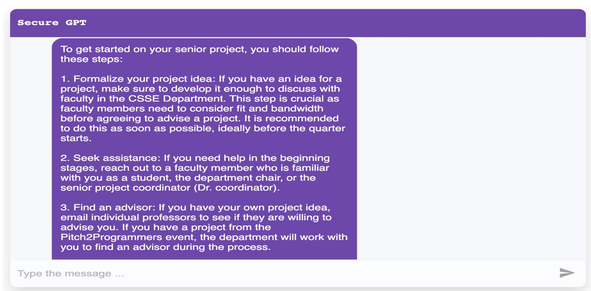
System collects user LLM preference and consent to the privacy policy.



User submits data, describes the file's contents, and asks a question.



System asks user to remove sensitive data detected by the model.



System generates a response to the user's question based on the data.

Resources and Models

- Frontend: React
- Backend: Flask, Firebase DB
- ML Models:
  - SpaCy NER for privacy detection
  - OpenAI GPT 3.5 Turbo for response generation
  - OpenAI Embeddings for semantic search

Limitations

Currently, our approach is limited by the accuracy and label types of our NER detection model. With more training tailored to particular use cases, the detection model could help our system detect sensitive data more accurately.

Potential Future Work

Future work should focus on improving the detection of private information and incorporating user feedback into the process to improve both the NER model and the LLM

Demo Our Project

1. Create an account
2. Submit a text document with data the LLM should use to answer your question
3. Ask a question