NoCap: Fact Checking with AI

Anthony Ciero, Joshua Pechan, Varun Doddapaneni, Thomas Chamberlain
Faculty Advisor: Professor Silaghi

Goals/Motivation

- Fact checking articles is important, as no one is always correct.
- Determining an article to be fact or fiction should be easy and comparative.
- After determining validity, it should be common knowledge and it should be spread to everyone.
- The goal is to create a website that checks articles, specifies what's fact and fiction, and gives it a rating.
- We then aggregate scores of articles by the same publishers to give them an "authenticity" rating and allow users to view their scores and the articles'.

Approach

- **AI Powered Claim Detection**
  - User insert article URL
  - AI gives article a rating based on authenticity
  - If article has not been used before, store in database
- **Publisher Authenticity**
  - Aggregate authenticity scores from the same publisher
  - Give publishers an average authenticity score based on articles in database
  - Allows users to browse by publisher, view their score, and database articles' scores
- **Chrome Extension**
  - Use webpage features on the article's link
  - Run extension to grab link and perform the usual features

Novel Features

- **Authenticity Graphical Representation**
  - Shows graph with misinformation rating
  - Shows which words are most shown up in true vs false articles
- **Web Content Accessibility Guidelines (WCAG)**
  - Ensure website follows accessibility standards
  - Being able to tab between important sections of the website
  - Accessible colors for background/text combinations

Algorithms/Tools

- Python (backend): primary server-side language for AI orchestration and services.
- FastAPI (API/backend web framework): lightweight, async-friendly framework to expose REST endpoints.
- LangChain & LangGraph (LLM/NLP modules): tooling to compose prompts, retrieval, and multi-step AI workflows.
- React (JavaScript UI): component-based interface for the extension popup and web dashboard.
- AWS Bedrock (Nova Lite): managed LLMs with model swapability for classification and analysis tasks.
- AWS Amplify (GraphQL with AppSync + DynamoDB): optional persistence layer for user preferences, cached verdicts, and analytics.

Technical Challenges

- **Frontend Design**
  - More familiar with backend development
  - Must ensure frontend is designed simply and accessible for all
- **LangChain/LangGraph**
  - Our familiarity with AI is not all at the same level
  - Important for managing prompts/reasoning chains in large language models
  - Start with smaller prototypes with simple prompts before going full-scale
- **AI Connection**
  - Must ensure proper procedure of managing API calls and handling authentication securely

Milestone 1

Tool Selection and Evaluation
- Compare and select technical tools for A, B, C
- Build small “hello world” demos to test feasibility and ease of use
- Address initial technical challenges related to A, B, C

Collaboration Setup/Chose tools for:
- Software development
- Document and presentation editing
- Communication
- Task/calendar management

Project Foundations
- Requirement Document
- Create Design Document
- Outlined Test Plan

Milestone 2

- Design Frontend
- Set up AI model on AWS
- Establish basic connection with AI
- Develop rudimentary backend and API
- Establish API endpoints

Milestone 3

- Create basic prompt engineering for AI
- Use LangChain to break down texts into tokens
- Output basic score from AI
- Set up database to store scores/rankings

Milestone 1 Task Matrix

Task	Thomas	Josh	Anthony	Varun
Compare and select technical tools for A, B, C	25%	25%	25%	25%
Provide small ("hello world") demo(s) to evaluate the tools for A, B, C	25%	25%	25%	25%
Resolve technical challenges: X, Y, Z	25%	25%	25%	25%
Compare and select collaboration tools for software development, documents/presentations, communication, task calendar	25%	25%	25%	25%

Milestone 1 Task Matrix Continued

Task	Thomas	Josh	Anthony	Varun
Create Requirement Document	Wrote 25%	Wrote 25%	Wrote 25%	Wrote 25%
Create Design Document	Wrote 25%	Wrote 25%	Wrote 25%	Wrote 25%
Create Test Plan	25%	25%	25%	25%