Chrome Summarizer Extension

NLP Final Project Report
https://github.com/najalee/nlp-project

Naja-Lee Habboush

May 6, 2025

1. Introduction

Often times while doing research for a project, I take breaks and eventually come back to the same few articles. Instead having to re-read the whole article, I wanted a quick, easy way to get the gist of it. That was the inspiration for this project.

The goal was to create a Chrome extension (with the hopes of expanding to other browsers soon) that could summarize any text a user highlighted on a webpage. This incorporates NLP directly into the browser and allows users to get concise summaries of the content without needing to switch tabs, open a new window, or use any other tool. It makes summarization fast and easy, helping to increase productivity and improve a user's workflow.

2. Technical Approach

2.1 Dataset

I decided to use the CNN/DailyMail dataset after some research. It is a well-established benchmark for abstractive summaries and consists of news articles paired with summaries written by humans. Since this project was meant to help me perform research in the context of school and work, a dataset centered around formal summaries would be more helpful.

2.2 Model

The model is based on google-t5/t5-small, and was trained and fine-tuned on the dataset using HuggingFace Transformers.

2.3 Preprocessing and Training

- Tokenized using T5 tokenizer.
- Trained for 1 epoch on a GPU-enabled environment.
- Used teacher forcing and label smoothing.

2.4 Chrome Extension

The frontend consists of three files, a manifest.json and a popup.html/js

- **manifest.json** - Defines extension permissions and scripts.
- **popup.html/js** - Handles user interface and communicates with backend.

2.5 Backend API

Initially I planned on using FastAPI for the backend, but since I wanted this to be a summarizer that anybody could use, without the background of a developer, I ended up utilizing HuggingFace spaces. The backend sends a JavaScript POST to https://huggingface.co/spaces/najale it, and grabs the output to send back. This is then sent to popup.js.

3. Contributions Over Base Model

This project goes beyond using a base model by:

- Fine-tuning T5 on task-specific data.
- Deploying the model through HuggingFace spaces.
- Creating a usable Chrome extension interface.

4. User Feedback Summary

Question	Response Options	Summary of Results
Was this extension easy to use?	Yes / No	Yes(6) No(0)
Did you find the summaries helpful?	Not helpful / Somewhat / Helpful	Helpful(5) Somewhat(1)
Did you encounter any bugs or issues?	Yes / No	Yes(5) No(1)
Would you use this in your daily browsing?	Never / Occasionally / Frequently	Occasionally(3) Frequently(3)

General Feedback (Qualitative)

In addition to the structured survey responses, participants provided the following open-ended feedback about their experience with the Chrome summarizer extension:

- *"If you try to summarize too large of a text it won't load."*
- *"Can't scroll through the popup on long summaries."*
- *"No feedback to whether or not the extension was processing."*

- *"The extension does have a habit of repeating details."*
- *"Some indication of a queue of requests and the ability to cancel requests if they haven't started processing."*
- *"Aesthetically very adorable...name is catchy and memorable while still being on topic, I LOVE THE SPINNY MOUNTAIN"*

In general, users liked the design and aesthetic of the popup. However, a few highlighted bugs, like not being able to scroll for long summaries, and general quality of life improvements, like no feedback if the request was being processed.

5. Lessons Learned

- Summarization models are effective, but require training to orient it to the intended goal.
- HuggingFace spaces are great to use as a free and easy to use alternative to APIs, but can be slow.
- There are real-time constraints like latency and the size of the input that need to be considered, especially in this context.

6. Potential Improvements

- Improving on all the bugs brought up by users.
- Allow users to select different summary styles i.e. bulleted lists
- Further train the model to allow for better summaries.
- Add features to show the user when the input is being processed or not.

7. Contributions

This project was completed entirely by myself, I did this project solo.

8. Self-Scoring Table

Category	**Max Points**	**Score Given**
Significant Exploration Beyond Baseline	80	70
Innovation or Creativity	30	10
Highlighted Complexity	10	10
Lessons Learned	10	10
Exceptional Visualization/Repo	10	10
External Testing	10	10
Earned Money with Project	10	0
Total	**160**	**120**