194.207 Generative AI VU WS25

Project Technical Report

Group Number 21

Burak Erel Akgün
12347560

Berke Baris Balli
12433742

Tyler Heaney
12500177

Pedro Silva
12433779

January 21, 2026

1 Abstract

The project addresses the problem of students lacking feedback on the completeness and correctness of their lecture notes. The system primarily targets university students who take free form notes, and want to understand which lecture topics they may have misunderstood or omitted. The app is developed as an interactive web application that takes student written notes as input, automatically summarizes them with the help of an integrated LLM, and compares the results against the lecture provided PDFs. The system returns the highlights of the missing points and provides a score based on rouge-bleu hybrid metric. You can access the repository.

2 Description

The first user to upload their notes for a lecture should login/sign up, then navigate to the *Edit* page, and upload the lecture slides/official notes (as a PDF). Then, they should go back to the *Home* page, and upload their notes for the lecture. This will generate summary of places your notes contradict the lecture, and where your notes miss topics covered in the lecture. Once multiple users have uploaded notes, a user can utilize the *Comparison* page to regenerate their comparison to the lecture, while also viewing how other student notes have captured different information than your own.

Our app consists of the Streamlit UI, chosen for simplicity, and a Mongo Database, chosen because we are working mainly with text data, and it would scale easily if this became more than a prototype. Finally, and most importantly, the LLM; we originally imagined a combination of LLM and traditional NLP to improve results, but Gemini's excellent summarization, combined with traditional NLP's bad results on the fragmentary nature of PDF Lecture slides caused us to transition to pure LLM logic to summarize and compare documents.

The biggest challenge was in determining scope. We initially overestimated the time we had and the difficulty of creating a polished project with useful features while worrying about robustness. The mid-way interview really helped put what is important into perspective and focus on making a meaningful final project.

3 Evaluation

The system was evaluated using real lecture summaries and student-written notes of varying completeness. For each submission, the student notes were summarized by the system and compared against a ground-truth lecture summary provided by the instructor. The evaluation focused on two aspects; the system's ability to identify missing or incorrect concepts, and the behavior of the quantitative knowledge coverage score under partial, paraphrased, and incomplete notes. The scoring was achieved via a hybrid Rouge–Bleu metric, in which Rouge-1, Rouge-2, and Rouge-L capture content recall and structural overlap between the student and lecture summaries, while Bleu provides a secondary precision signal, with the weighted combination producing a normalized knowledge coverage score. This method was more beneficial compared to a single score metric because a hybrid Rouge–Bleu metric balances recall and precision, allowing the system to reward student notes that capture key lecture concepts even when phrased differently, while still penalizing overly vague or incomplete summaries. Relying on either metric alone would bias the evaluation toward exact wording (Bleu) or raw content overlap without structure (Rouge), making the combined approach more robust for paraphrased educational notes.

4 Reflection

The app can increase human agency, though this depends on the user. Students must use the tool with care because, like any Generative AI, it is prone to errors. As seen in our evaluation, the system can still be improved to output better descriptions of missing or incorrect content. A student might reduce their own agency if they do not reflect on the feedback; thus, they must reason whether the output makes sense and check the original materials whenever in doubt. However, provided they do this, the system helps them gain significant time in the learning process.

Finally, we would surely develop the app further. It is already highly useful, but performance can be improved. Beyond refining the text generation, the app could accept different types of media. Currently, notes must be in Markdown, but they could eventually be hand-written, pictures, or diagrams. The same applies to the official lecture content, which could expand from simple PDFs to a wider variety of formats.