

Of course. I understand your project is to build an **AI-powered email voice assistant** that can draft emails, summarize your inbox, and manage tasks by identifying and scheduling them.

Based on that, I have gone through all the papers you provided. Below is a simple explanation of the limitations for each one and a clear guide on which of those limitations you can realistically tackle in your final year project.

---


## 1. base-0.pdf (GlassMail: On-the-Go Email Creation on Smart Glasses)

This is your base paper. It's about an AI assistant on smart glasses for writing emails while walking.

### Limitations Explained Simply:

- **It only creates new emails:** The system was designed only for composing brand new emails and does not support reading the inbox, replying to threads, or forwarding messages<sup>1</sup>.
- **It can't handle attachments or links:** The prototype does not support adding things like files, bullet points, or hyperlinks to the emails it drafts<sup>2222</sup>.
- **The AI sounds robotic:** The emails generated by the system often sound too formal and unnatural, especially for casual conversations, requiring a lot of manual editing by the user to fix the tone<sup>3333</sup>.
- **Editing is clumsy:** Making small, precise changes to the text using only voice commands is difficult and often inaccurate<sup>4</sup>.
- **It's for smart glasses only:** The entire system was designed and tested for smart glasses, which most people don't have, limiting its real-world use<sup>5</sup>.

### How You Can Overcome These in Your Project:

-  **Expand the Scope (HIGHLY RECOMMENDED):** This is the biggest opportunity for your project. Your plan to **summarize the inbox and handle replies** directly solves the primary limitation of GlassMail. You can position your project as a *complete* email assistant, not just a composer.

- **✅ Improve Personalization:** You can directly tackle the "robotic tone" problem. The paper itself suggests that the system could learn from a user's past emails<sup>6</sup>. You can implement this by designing a feature that takes a few examples of a user's sent emails and uses them to guide the AI's writing style, making the generated content sound much more like the actual user.
- **✅ Build for an Accessible Platform:** Instead of building for smart glasses, develop your assistant as a **mobile application**. This makes your project more practical and immediately relevant to a wider audience, directly addressing the platform dependency limitation.

---

## 2. base-1.pdf (AI-Powered Reminders for Collaborative Tasks)

This paper studies how Microsoft's AI-powered email reminders (Viva Daily Briefing) help people remember tasks.

### Limitations Explained Simply:

- **Limited interactions:** The tool only lets users mark a task as "Done" or "Remind me"<sup>7</sup>. It doesn't allow for more useful actions like quickly replying or scheduling the task directly in a calendar<sup>8888</sup>.
- **It's not always accurate:** Users reported that the system sometimes surfaces reminders for tasks that are already completed or provides information that is out of date<sup>9999</sup>.
- **Focuses only on one tool:** The research is based entirely on Microsoft's specific daily digest email, so the findings might not apply to other kinds of reminder systems<sup>10101010</sup>.

### How You Can Overcome These in Your Project:

- **✅ Create Smarter, Actionable Reminders:** This is a perfect fit for your project. When your assistant identifies a task (e.g., "Can we meet on Friday at 2 PM?"), you can overcome the interaction limitation by having it ask, **"I've found a meeting request. Would you like me to add this to your calendar?"** This makes your assistant proactive and more useful than the system studied in the paper.
- **✅ Improve Accuracy with User Confirmation:** To address the inaccuracy problem, your assistant can confirm tasks with the user before scheduling them. This

human-in-the-loop design makes your system more reliable.

---



### 3. base-2.pdf (Emails by LLMs: A Comparison of Language)

This paper compares emails written by humans to those generated by AI, finding that AI emails are often too wordy, formal, and complex<sup>11</sup>.

#### Limitations Explained Simply:

- **It only studied 100% AI-written emails:** The research looked at emails generated entirely by AI<sup>12</sup>. In the real world, people often use AI to get a first draft and then edit it themselves (co-writing)<sup>13</sup>.
- **The user study was small and biased:** The feedback on the emails came from a small group of mainly university students, so their opinions might not represent everyone<sup>14</sup>.

#### How You Can Overcome These in Your Project:

-  **Focus on a "Co-writing" Experience:** Design your assistant to be a collaborative partner. After drafting an email, allow the user to give follow-up voice commands like, "Make that sound more casual," or "Shorten the second paragraph." This directly addresses the paper's limitation by creating a more realistic and useful co-writing tool.
-  **Use its Findings to Guide Your Goals:** This paper gives you a clear target. You can state in your project report: "Prior research shows AI emails are often verbose and impersonal<sup>15</sup>. Therefore, a primary goal of our project is to use personalization techniques to generate emails that are more concise and match the user's natural style."

---



### 4. base-6.pdf (LaMPost: AI-assisted Email for Adults with Dyslexia)

This paper is about an AI email-writing tool designed to help adults with dyslexia.

#### Limitations Explained Simply:

- **AI accuracy was not good enough:** The language models (from early 2022) were not reliable enough for real-world use, sometimes generating irrelevant or factually incorrect content ("hallucinations")<sup>1616161616161616</sup>.
- **Too many choices were overwhelming:** The "Rewrite" feature provided up to 15 different options, which users found confusing and cognitively demanding to read through<sup>17171717</sup>.

### How You Can Overcome These in Your Project:

-  **Leverage Newer, Better Models:** The models used in this 2022 study are older. You can state that your project will use a state-of-the-art model (like Gemini), which has significantly improved accuracy and is less prone to errors, directly addressing their primary limitation.
-  **Design a Better User Experience:** You can easily solve the "too many choices" problem. Design your assistant to provide only the **top 2 or 3 suggestions** for a rewrite. This is a simple but important design choice that directly improves upon the user experience described in the paper.

---


## 5. base-4.pdf (ShortMail: An email summarizer system)

This paper describes a system that uses older AI models (like BERT) to create summaries of emails.

### Limitations Explained Simply:

- **It only works in English:** The system was only designed to summarize English-language emails and would need to be re-engineered for other languages<sup>18181818</sup>.

### How You Can Overcome These in Your Project:

-  **Use a Modern, Multilingual API:** This is an easy win. The models used by ShortMail have limited language support. Modern LLM APIs like Gemini are inherently multilingual. You can state that by using a modern API, your system naturally overcomes this limitation and can handle summarization for a wider range of languages.
-


## 6. base-5.pdf (TellTime: An AI Calendar with a Voice Interface)

This paper is about a voice-controlled calendar where an AI parses spoken narratives (e.g., "I woke up at 7, then worked out for 15 minutes...") into scheduled events.

### Limitations Explained Simply:

- **The AI often got the time wrong:** The most common failure was the AI misinterpreting durations and scheduling events at the wrong time on the calendar<sup>19</sup>.
- **The system was slow:** Users felt that waiting for the AI to process a command was slow, especially for making small corrections<sup>20202020</sup>.

### How You Can Overcome These in Your Project:

-  **Use a Hybrid Approach for Task Scheduling:** This is directly applicable to your task management feature. The paper found that users preferred a hybrid approach: voice for initial input and manual controls for corrections<sup>21</sup>. When your assistant extracts a task like "Let's meet next week," you can overcome the "wrong time" limitation by having it ask a clarifying question:

**"I see a meeting request. What day and time should I schedule it for?"** This makes your system more accurate and robust.