Automated, Context-Aware Alt Text Generation for Educational Documents Using Large Language Models

Automated Alt Text Generation for Educational Materials

A Tool for Streamlining Accessibility Remediation in Microsoft Word and PowerPoint

Disha Baglodi

Georgia Institute of Technology, dbaglodi3@gatech.edu

Bella Martincic

The Sonification Lab, Georgia Institute of Technology, imartincic3@gatech.edu

Norah Sinclair

Center for Inclusive Design and Innovation, Georgia Institute of Technology, norah.sinclair@design.gatech.edu

Bruce N. Walker (PhD)

Georgia Institute of Technology, bruce.walker@psych.gatech.edu

This paper presents ALADDIN, a tool developed by [anonymized] to automate the generation of alternative text (alt text) for images in Microsoft Word and PowerPoint files, aimed at improving the accessibility of educational materials. The tool leverages Gemini Flash, a large language model (LLM), to generate context-aware and content-appropriate image descriptions by analyzing the surrounding text, slide content, existing metadata, and visual layout. Currently integrated into a user-friendly Google Colab notebook, ALADDIN has reduced alt text remediation time for large files of educational materials from weeks to hours. We describe the tool's implementation, evaluation, and future directions for broader accessibility enhancement.

CCS CONCEPTS • Human-centered computing • Computing methodologies • Applied computing

Additional Keywords and Phrases: Accessibility, Alternative Text, Document Remediation, Educational Technology, Generative AI, Large Language Models, Gemini Flash

ASSETS '25, October 26–29, 2025, Denver, CO, USA

ACM ISBN 979-8-4007-0676-9/2025/10.
https://doi.org/10.1145/3663547.3759748

1 INTRODUCTION

Individuals with disabilities may face steep hurdles to obtaining a college degree, especially in STEM fields. As of 2020, cognitive, mobility, and perceptual disabilities were formally diagnosed among 21% of undergraduates and 11% percent of postbaccalaureate students [8]. As high as they are, these percentages are lower than the national average of around 28% of adults who have diagnosed disabilities, indicating that there are barriers preventing adults with disabilities from completing higher education [1]. One of the many such barriers is the inaccessibility of teaching materials, such as syllabi, exams, lecture slides, readings, assignments, and homework. Such documents come in a variety of file formats (e.g., pptx, pdf, html, docs), which themselves can contain accessibility challenges; this issue is compounded by the addition of inaccessible charts, graphs, videos, pictures, diagrams, maps, simulations, etc. A lack of both knowledge and tools for the educational content creators (i.e., instructors) means course materials are often inaccessible and of diminished utility for learners with disabilities. Higher education, as a whole, cannot cope with the widespread scale of this issue, including gaps in knowledge, limited staff, and the potentially unfathomable cost of remediating all of the accessibility issues posed by these artifacts.

AccessCORPS (AC) is a training and service group at the Georgia Institute of Technology to combat issues of course material accessibility. AC trains a sizable number of undergraduates in accessibility best practices, along with the practical skills needed to remediate documents for higher education. The remediation for each course is a multi-week labor-intensive process with multiple phases and quality control checks. The professor starts by sharing all of the course's documents with AC. For instance, the instructor may provide all the MS PowerPoint files used in the lectures, and all the Word documents for the syllabus, homeworks, and assignments. AC team members then compile "metadata" about each of the files, such as the number of images of different types, such as graphs, charts, photographs, maps, images of tables, images of equations, as well as decorative images. There is great variability in the format and contents of these course materials files, from course to course, and from instructor to instructor.

The process of manually remediating the files comes in 2 phases, where each phase takes about 1 to 2 weeks, but for large files that are heavily inaccessible, it can take even longer to work through challenges. Quality remediation is foundational to the core of AC, but with all of the unique ways professors design and create course materials, many/most of which have accessibility issues, this can be laborious and time consuming. The success of AC lies, in part, in the availability of the large group of trained remediators. However, this pool of skilled labor is not infinite, and the time spent remediating files is a major hindrance in how briskly a course can be addressed. One of our priorities in AC is to reduce the intensity of labor that goes into remediation, starting with tackling the most common and time-consuming issues. The most common file types that professors upload to AC for remediation are Microsoft Word and PowerPoint files, with the most time consuming task being to generate alternative text for pictures. The context surrounding the images within the slide deck also makes a difference in what the alternative text should be.

To address this issue, we have built an AI-supported human-in-the-loop tool, Automated LLM-Assisted

Document Descriptions for INclusion (ALADDIN), to help streamline image classification and generation of alt text. ALADDIN takes a Word or PowerPoint file, looks at each image, classifies the image type, and then uses generative AI to create suitable and context-aware alt text for each image. The key is that we cannot simply rely on an AI engine to create generic, context-agnostic descriptions of images.

1.1 Previous Solutions

In the latest version of Microsoft Office, there is an Accessibility Checker tool that scans the document and performs many of the checks that are included in the AC remediation checklist. Part of the basic AC remediation process is checking for the presence (and quality) of alt text for all images and figures. For images, AC team members may either input an alt text description or click a button to generate alt text using Microsoft's built-in Image Analysis tool [2], a service that has the capability to extract many visual features of an image [3]. However, because that generic text generation tool is only focused on analyzing an image without consideration of how the image is used in its academic materials context, using the built-in tools to generate alt text does not always produce accurate or useful results. In any case, AC team members still have to manually go through each file, and for each image click the "generate alt text" button, which can be tedious for files with many images.

We have determined that Large Language Models (LLMs), such as Chat-GPT, Gemini, and Claude, are usually better than the built-in Office tools at generating suitable alt text descriptions, since the user can modify the prompt to shape the resulting description. We can take advantage of these generative AI tools, while also including automation, bulk file handling, consideration of document/educational context and image type, and more. While AI cannot completely replace humans in generating alt text, it can certainly be used as a supportive tool to improve efficiency and supplement the accuracy of this process.

2 IMPLEMENTATION

ALADDIN is currently implemented as a Google Colab notebook that is organized into sections, including instructions, a Setup section, and sections to input either Word or PowerPoint documents. Initially, this tool was created to generate image descriptions for each image in the input file and then put each generated description back into the corresponding image's alt text container within the original file, then output a copy of the file with the modified alt text. However, we have expanded its capabilities and quality of output. The LLM we used throughout the development of ALADDIN is Gemini Flash 2.0 due to its capabilities for analyzing, captioning, and detecting objects in images [4], which proves useful for both tagging and generating alt text.

2.1 Preprocessing

ALADDIN starts by generating context for the entire document, as a preprocessing step. This is done by scanning the document for certain properties that are fed into the LLM (Gemini Flash), which is asked to use these properties to generate a summary of the file.

For Word documents, the tool extracts the headings, number of paragraphs, frequent keywords, and existing alt texts. This is used to generate a concise summary and overview of what the document is about, the likely target audience, and the main themes or key points. For PowerPoint files, the tool extracts the title, the number of slides, the slide headers, frequent keywords, and existing alt texts to generate a similar summary. After the document has been preprocessed, the tool then scans through each image in the file, tags it (see below), and generates alt text for each image (again, see below).

2.2 Image Classification/Tagging

Image tagging using ALADDIN is done by feeding the image into Gemini and carefully and intentionally prompting the LLM to generate a set of tags regarding what the category of the image is, in which the categories are already defined in the prompt (such as

Graph, Map, Table, Equation, and more). Multiple tags can be identified for a given image, if appropriate.

2.3 Context Extraction

Generating an image description while considering the image's context can help to enhance the quality of the generated description; a study in 2024 found that blind and low vision users of AI description tools largely preferred context-aware generated descriptions to context-free ones [6]. Thus, we follow image tagging with context extraction. The first element of this context is the **nearby text**, which, for Word documents, is the paragraph the image is in; for PowerPoint , it is all the text on the slide containing the image. The second component of the context is the **summary of the file** (which is especially useful if the nearby text does not exist or is minimal). The third component of context is the set of **generated tags** for a given image. For PowerPoint, the fourth component is **the image of the slide** itself. The fifth component is the image's **existing alt text**. This set of context elements is used to prompt Gemini to generate a context-aware description for the specified image, and that description is then inserted into the alt text attribute of the corresponding image, in the original Word or PowerPoint file. This process is automated and completed for each image in an entire file.

2.2.1 Visual Context

We note that in many cases there is limited text near an image, especially in a Powerpoint slide. Additionally, presentations are often composed of many slides, and sometimes a simple summary is not sufficient to capture the full context of an image in a given slide. Thus, for an image in a PowerPoint slide, we also take as context the image of the entire slide. With this feature, we can better realize "outlier" images that don't necessarily fit into the bigger context of the presentation.

Another use case of taking the slide image as context is when handling a slide with multiple images that relate to each other. Then, ALADDIN will be able to "see" the relationship between images. Adding the image of the slide as context gives the LLM a better perspective on how each image fits into the presentation.

2.4 Alt Text Generation

After inputting the context, which consists of the generated tags, the generated summary, the visual context (for PowerPoint), the textual context, and any existing alt text, the final step is to input the image itself into the prompt. The prompt has also been engineered to include guidelines for the LLM to follow when creating an alt text description, including describing key visual elements and their relationship to the context, using clear, academic language, and describing the image's key components and purpose if the image includes a diagram or figure. These guidelines provide rules of thumb to follow when creating alt text descriptions, but they are not permanent and will be subject to change as AC guidance and best practices for alt text evolves. Nevertheless, by using a contextual approach in generating alt text descriptions, ALADDIN can give those who interact with the content a better understanding of the purpose of the image, which can elevate understanding of course content [9].

3 ADDITIONAL FUNCTIONALITIES

3.1 User Input

ALADDIN also allows users to have more input on what goes into the document. For example, there is the option for the user to check each generated image description and either approve it, keep the old description, or input their own description. This would be most useful for smaller documents with not too many images. If approving each description would be too tedious, there remains the option to let the program run fully automated, without prompting the user for input. These options are displayed as subsections of the Word and PowerPoint sections of the notebook, so the user will expand whichever section corresponds to their needs.

If the user input/approval is enabled, then, after the file has been uploaded, for each image in the file, the program displays a screenshot of the slide (if processing a PowerPoint file), This screenshot contains the current alt text, the image itself, the AI-generated tags, and the AI-generated alt text. It will then prompt the user to either: approve of the generated alt text by inputting nothing; keep the old alt text; remove the alt text; or input their own alt text. Once the user submits their input, the alt text is inserted into the image, and the process continues until all images have been processed. The updated file is then downloaded to the user's machine. While this "supervised" option allows for more direct human input, this process takes more time. This option is better suited for less image-heavy files that have undergone one round of remediation but still require some review.

If supervision is not activated, then the program will always insert the newly generated alt text into each image. This fully automated option is more useful for when it would be too tedious to manually go through each image and input alt text descriptions, such as for files with hundreds of images. When inputting a PowerPoint presentation that consisted of almost 300 images, the program took less than an hour to run through the entire file without user input (including all the context generation stages, etc.), whereas AC members took two weeks to manually complete the remediation of the same file.

4 EVALUATION

ALADDIN has been developed via an iterative design process, where we run through cycles of implementing new functionalities and evaluating them. In our multiple evaluations, we have considered such factors as the tool's runtime, output quality, and usability. The runtime depends on the number of images in the input file, but for files with many images, it outperforms the manual process by a large margin (as discussed previously, in section 3.1). Additionally, the tool's generated descriptions are more closely related to the topic of the document than those of Microsoft's tools. However, the descriptions are not perfect and can tend to exaggerate an image's relevance to its context even when the image is merely included for decorative purposes. Of course, while AC team members have been using the tool, they have also provided notes for ways we can further improve its ease of use (see section 5.2).

5 ONGOING WORK

ALADDIN already provides a major convenience, but we still continue to improve it. Alt text is not the only factor that goes into making a file accessible, and it is also not the only accessibility issue that can be tedious. It remains important to constantly improve both the capabilities and usability of ALADDIN.

5.1 Expanding Capabilities

As but one example of how we are expanding ALADDIN capabilities, one common accessibility issue in PowerPoint files is the absence and repetition of slide titles. Using similar logic as that for generating image descriptions, we can generate slide titles using the context of the PowerPoint file and of the slide, insert that title into the slide, and move the title outside of the slide so that it is hidden, which enhances the accessibility of the file without changing its appearance. Another planned expanded capability is to generate alt text not just for images, but for all objects that need them, including shapes and groups. ALADDIN already iterates over every object in the document, so it certainly can expand from checking for images to checking for shapes and groups. Then, using the same context as was used for generating alt text for images, we can generate alt text for the object by inputting the name of the object, such as whether it is a star shape or a group of five rectangles or pictures, as well as any text inside the object.

5.2 Usability Improvement

Google Colab is convenient for quickly writing and running Python code that is easy to share [5]. However,

it can still be jarring for those who are unfamiliar with Colab to have to learn how to use it; for example, it could be easy to accidentally edit a line of code or forget to run a cell and encounter errors, which can make the user more frustrated. Additionally, the layout of Colab notebooks is sometimes not very user-friendly (and not as accessible).

To eliminate such issues, we are developing a web app version of ALADDIN, to optimize the usability and accessibility of the user interface. With the web app, users will not need to interact with the code behind the tool, which should make the web app less intimidating. This web app will be realized within a web design framework (such as React) on the frontend, and with a Python Flask-based backend that largely reuses the code in the Colab notebook. The app will then be served out via a web server, for access by any authorized user..

6 CONCLUSION

The [anonymized] document processing tool ALADDIN addresses one of the biggest and most time-consuming hurdles we have when remediating documents, that being generating quality alt text for a range of image types. Solely relying on human labor to generate alt text has been extremely time consuming in the context of some higher-education courses. Previous technological interventions are often inaccurate and do not follow the best practices that we know work for higher education course materials specifically. Tools such as ALADDIN improve the remediation process for course documents as accurately and efficiently as possible, while also ensuring the highest quality of remediation, and maximizing remediation efficiency. Humans will always need to be involved in remediation to ensure that documents are accessible, but also true to the original intent of the instructor. Using carefully and intentionally designed AI tools can effectively supplement the remediation process by doing the majority of the work for generating alt text in a fraction of the time. Our tool decreases remediation time, shortening the multi-week process of remediation. This allows the AC team to maximize labor and remediate as many courses per semester as possible. By remediating course documents and making courses more accessible, all types of learners have easier access to knowledge in higher education. AC team members will improve courses of all topics, giving learners with disabilities access to all manner of careers and disciplines.

REFERENCES

[1] Centers for Disease Control and Prevention. 2024. Disability and Health Data System (DHDS). Retrieved from https://www.cdc.gov/ncbddd/disabilityandhealth/infographic-disability-impacts-all.html

[2] P. Farley, E. Urban, and N. Mehrotra. 2025. Overview: Generate image alt text with Image Analysis. Microsoft. https://learn.microsoft.com/en-us/azure/ai-services/computer-vision/use-case-alt-text

[3] P. Farley, E. Urban, N. Mehrotra, and A. Jenks. 2024. What is Image Analysis? Microsoft. https://learn.microsoft.com/en-us/azure/ai-services/computer-vision/overview-image-analysis?tabs=4-0

[4] Google AI For Developers. 2025. Explore vision capabilities with the Gemini API. Google. https://ai.google.dev/gemini-api/docs/vision?lang=python

[5] Google Colaboratory. 2019. Google. https://colab.research.google.com/

[6] A. Gubbi Mohanbabu, and A. Pavel. 2024. Context-Aware Image Descriptions for Web Accessibility. ArXiv (Cornell University), 1–17. https://doi.org/10.1145/3663548.3675658

[7] MConverter. 2025. API Documentation. MConverter. https://dev.mconverter.eu/documentation

[8] National Center for Education Statistics. 2023. Table 311.10. Number and percentage distribution of students enrolled in postsecondary institutions, by level, disability status, and selected student characteristics: Academic year 2019–20 [Data table]. In *Digest of education statistics*. U.S. Department of Education, Institute of Education Sciences. Retrieved from https://nces.ed.gov/programs/digest/d22/tables/dt22_311.10.asp.

[9] WCAG 2.1 Understanding Docs. 2025. Identify Purpose (Level AAA). World Wide Web Consortium. https://www.w3.org/WAI/WCAG21/Understanding/identify-purpose.html

APPENDIX

Flowchart of ALADDIN's Process

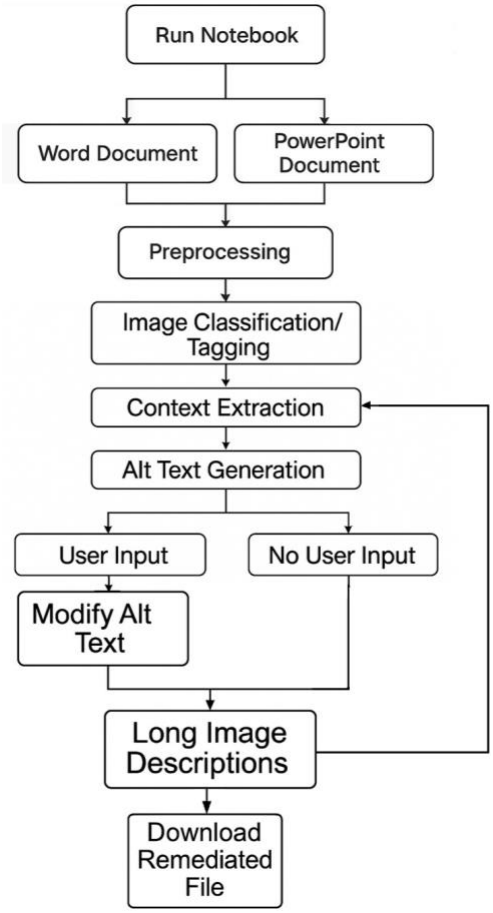

Figure 1: A diagram of the workflow of ALADDIN. Different branches appear when users are given a choice, such as the choice to either input a Word document or a PowerPoint document, as well as to provide input on the generated alt text.