Manuscribe AI: Learning Support System for People with Fine Motor Impairments

Raquel K. C. Santos[1], **Fábio S. da Silva**[2]

[1]SiDi Intelligence & Innovation Center

[2]Escola Superior de Tecnologia – Universidade do Estado do Amazonas (UEA)
Manaus – AM – Brazil

raquel.cs@sidi.org.br, fssilva@uea.edu.br

Abstract. *This paper presents and evaluates Manuscribe AI, a web-based system built on open-source speech-to-text (STT) and Large Language Models (LLMs) to support students with fine motor impairments through automatic lecture notes from recorded classes, using local execution and prompt engineering techniques. The comparative evaluation focused on small and medium-sized models for transcription and summarization tasks, using a custom dataset of Brazilian Portuguese lecture transcriptions. Results show competitive performance, with Whisper Small achieving the best transcription result and GPT-OSS 20B standing out in summary generation, highlighting the potential of lightweight open-source models for assistive educational applications.*

Resumo. *Este artigo apresenta e avalia o Manuscribe AI, um sistema web baseado em modelos abertos de speech-to-text (STT) e Large Language Models (LLMs) para apoiar estudantes com limitações motoras finas por meio da geração automática de anotações de aulas gravadas, utilizando execução local e técnicas de engenharia de prompts. A avaliação concentrou-se em modelos de pequeno e médio porte para tarefas de transcrição e sumarização, utilizando um dataset próprio de transcrições de videoaulas em português brasileiro. Os resultados demonstram desempenho competitivo, com destaque para o Whisper Small na tarefa de transcrição e o GPT-OSS 20B na geração de resumos, evidenciando o potencial de modelos abertos e leves para aplicações educacionais assistivas.*

1. Introdução

Atualmente, as aplicações baseadas em inteligência artificial têm-se destacado no mercado educacional por possibilitarem a personalização da aprendizagem, um aspecto cada vez mais valorizado na educação. Esse fator é relevante, considerando que o ensino escolar é frequentemente direcionado a todos os alunos de forma generalizada, sem atender às suas necessidades individuais [Barbosa et al. 2021].

Nesse contexto, destaca-se também a relevância das Tecnologias Assistivas (TAs), que contribuem para tornar o aprendizado mais inclusivo e adaptado às necessidades específicas dos estudantes. Essas tecnologias visam eliminar barreiras sensoriais, motoras ou cognitivas que dificultam o acesso a informações e comprometem a construção do conhecimento [Correa et al. 2021].

Apesar das contribuições proporcionadas pelas TAs, muitos estudantes com dificuldades motoras finas ainda enfrentam adversidades no contexto educacional. Prestar atenção ao assunto lecionado pelo professor e escrever anotações ao mesmo tempo já representa uma tarefa complexa para a maioria dos alunos [Boyle 2010, Boyle 2012]; para esse público, entretanto, pode tornar-se uma tarefa exaustiva ou até inviável, devido ao ritmo acelerado das aulas, à sobrecarga de informações e à alta exigência cognitiva e motora, levando o processo de aprendizagem a ocorrer de forma ainda mais desigual para esses estudantes [Tavares and Cardoso 2016]. Além disso, estudos apontam que apresentar o conteúdo de forma multimodal, por exemplo, a disponibilização da matéria escrita em paralelo à exposição oral, favorece a compreensão e a retenção da informação, especialmente para alunos com desafios de aprendizagem [Xu et al. 2024].

Diante deste cenário, este artigo apresenta o *Manuscribe AI*, um sistema que integra modelos *open source* de Reconhecimento Automático de Fala (ASR - Automatic Speech Recognition) e Modelos de Linguagem de Grande Escala (*LLMs* - Large Language Models), em um pipeline de transcrição, revisão textual e geração de anotações automáticas de aulas em português brasileiro, utilizando execução local e técnicas de engenharia de *prompts*. O software desenvolvido caracteriza-se como uma modalidade de Tecnologia Assistiva voltada à ampliação das formas de representação do conteúdo, estimulando a autonomia dos estudantes e contribuindo para a redução das desigualdades no processo de aprendizagem [Svensson et al. 2019].

Para garantir a viabilidade da solução proposta em contextos reais de sala de aula, este trabalho também realiza uma análise comparativa entre os modelos avaliados, buscando identificar quais apresentam maior adequação às necessidades do sistema desenvolvido. Como contribuição adicional, foi construído um *dataset* composto por transcrições revisadas manualmente, de aulas em português brasileiro disponibilizadas na plataforma *Youtube*[1] sob a licença *Creative Commons*, utilizado na avaliação comparativa deste trabalho.

2. Trabalhos Relacionados

Nesta seção são apresentadas soluções semelhantes à proposta do *Manuscribe AI*. Há diversas aplicações tecnológicas disponíveis no mercado que apresentam funcionalidades relacionadas às do *Manuscribe AI*, embora possuam finalidades e públicos-alvo distintos. Ferramentas como *Transkriptor*[2] e *Otter.ai*[3] combinam transcrições automáticas e geração de resumos [Transkriptor 2024, Lai 2025, Otter.ai 2025], enquanto *Quizlet*[4] e *Knowt*[5] oferecem recursos de estudo baseados em inteligência artificial, como geração de cartões de memorização (*flashcards*), questionários, resumos e modos de estudo personalizados [Sutherland 2010, Quizlet 2024, Ravaglia 2024, Knowt 2025]. *Transcribe*[6] e o dispositivo *Plaud.AI*[7], concentram-se principalmente na captura e processamento de áudio [Kodalovic 2025]. Apesar da diversidade de funcionalidades, não foi identificado,

[1] https://www.youtube.com
[2] https://transkriptor.com/pt-br/
[3] https://otter.ai
[4] https://quizlet.com
[5] https://knowt.com
[6] https://transcribe.com
[7] https://www.plaud.ai

entre as ferramentas analisadas, um apoio dedicado à acessibilidade educacional para estudantes com dificuldades motoras finas, integrando transcrição, revisão e sumarização em uma solução de execução local com suporte ao português brasileiro.

3. Fundamentação Teórica

Esta seção apresenta os principais conceitos teóricos relacionados às tecnologias empregadas no *Manuscribe AI*, introduzindo os assuntos de *LLMs*, sumarização e *STTs*. Modelos de linguagem são sistemas capazes de prever a próxima palavra com base no contexto, permitindo aplicações como geração de texto, tradução automática e sumarização [Zhang et al. 2023]. Esses modelos possuem papel central no Processamento de Linguagem Natural (*NLP*) [Ranzato et al. 2016] e, por meio de treinamento em larga escala, podem ser utilizados em diferentes domínios, incluindo a educação [Xiao and Zhu 2025, Hadi et al. 2023].

Os Modelos de Linguagem de Grande Escala (*LLMs*) são baseados majoritariamente na arquitetura *Transformer*, introduzida por [Vaswani et al. 2017]. Essa arquitetura substituiu abordagens baseadas em redes recorrentes ao permitir o processamento paralelo de sequências por meio do mecanismo de *self-attention*, que atribui pesos distintos aos elementos da entrada conforme sua relevância contextual.

A sumarização automática de texto permite a produção de versões mais concisas de documentos ao reunir as informações mais relevantes, reduzindo o esforço necessário para processar grandes volumes de conteúdo. Com a evolução dos *LLMs*, a sumarização passou a explorar modelos de propósito geral capazes de executar a tarefa com pouca instrução, reduzindo a dependência de modelos treinados especificamente para uma única tarefa [Zhang et al. 2025]. Em aplicações práticas, a sumarização envolve não apenas a escolha do modelo, mas também etapas de aquisição e pré-processamento dos dados, modelagem e avaliação dos resultados. Para documentos extensos, estratégias de processamento por partes podem ser empregadas para lidar com limitações de contexto, gerando resumos intermediários que posteriormente são combinados ou refinados.

Nos últimos anos, os (*LLMs*) também passaram a desempenhar um papel importante no processamento de fala [Peng et al. 2025]. Em aplicações tradicionais, o reconhecimento automático de fala (*ASR - Automatic Speech Recognition*), também denominado *Speech-to-Text* (STT), converte o sinal de áudio em texto, que pode posteriormente ser processado por modelos de linguagem [Radford et al. 2023]. Esses *Speech LLMs* têm obtido resultados expressivos, apresentando potencial para diferentes aplicações.

4. Metodologia e Solução Desenvolvida

Esta seção descreve o desenvolvimento do sistema *Manuscribe AI*, a abordagem de avaliação adotada para o *pipeline* e os modelos de *LLM* e *STT* selecionados para o experimento.

4.1. Desenvolvimento do Software

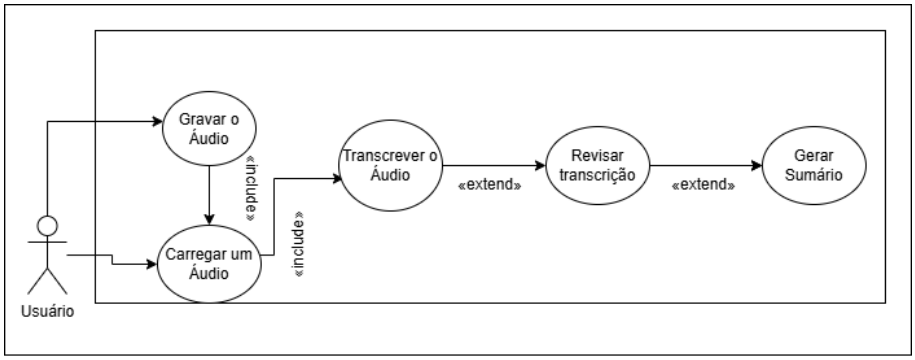
O *Manuscribe AI* foi desenvolvido como uma aplicação web utilizando a linguagem de programação *Python* em conjunto com o *framework FastAPI*[8], escolhido por sua leveza

[8] `https://fastapi.tiangolo.com`

e desempenho na criação de *APIs RESTful*. O sistema foi projetado para apresentar uma interface simples ao usuário, mantendo a integração com os modelos de IA na camada de *backend*. Entre as funcionalidades disponíveis estão gravação de áudio, envio de arquivos e transcrição automática. Além disso, o sistema realiza uma etapa de revisão da transcrição, removendo ruídos e trechos irrelevantes, reduzindo a quantidade de informações processadas na etapa de sumarização.

O sistema recebe áudios nos formatos *.mp3* ou *.wav*, que são processados por modelos de *STT* e *LLMs* para transcrição, revisão e sumarização. O diagrama de caso de uso do sistema pode ser visualizado na Figura 1.

Figura 1. Diagrama de caso de uso do sistema implementado



O sistema foi projetado para utilizar modelos locais, evitando limitações comuns de *APIs* públicas, como restrições de duração de áudio e número de requisições. Além disso, foram aplicadas técnicas de engenharia de *prompts* para orientar as *LLMs* na geração de resumos mais adequados ao contexto educacional.

A sumarização gerada apresenta o conteúdo da aula de forma estruturada, incluindo tópicos principais e materiais complementares relacionados ao assunto. Para a busca desses materiais, foi utilizada a *Google Custom Search JSON API*, permitindo recuperar conteúdos externos relevantes ao tema abordado.

4.2. Avaliação de modelos de inteligência artificial

O processo de avaliação foi baseado em abordagens clássicas de aprendizado de máquina e nas etapas metodológicas do *CRISP-DM* [Chapman et al. 2000], contemplando preparação dos dados, modelagem e avaliação. Além disso, foram consideradas práticas de *benchmarking* e avaliação de modelos descritas na literatura [Sculley et al. 2015, Chang et al. 2023]. As etapas da avaliação foram:

1. **Criação do *dataset*:** desenvolvimento de um conjunto de dados contendo transcrições de videoaulas de diferentes áreas e durações, simulando cenários reais de uso.
2. **Definição da suíte de testes:** construção de testes automatizados utilizando bibliotecas da linguagem de programação *Python* como *Pytest* e *Pandas*, permitindo padronização e reprodutibilidade experimental.
3. **Avaliação quantitativa:** utilização do *framework DeepEval* para avaliação das *LLMs*, além das métricas *WER (Word Error Rate)* e *CER (Character Error Rate)* para mensurar a qualidade das transcrições automáticas.

4. **Avaliação qualitativa humana:** análise humana dos resultados gerados pelos modelos, observando aspectos subjetivos e limitações não identificadas pelas métricas quantitativas.

5. Modelos avaliados

Os modelos avaliados foram selecionados considerando a limitação computacional dos recursos utilizados para inferência. Os seguintes critérios foram priorizados na escolha: modelos open source de famílias diferentes, suporte ao português e execução local. Para as LLMs, também foi considerada a capacidade de seguir instruções e gerar textos coerentes. Para os modelos STT, foram consideradas versões de diferentes tamanhos dos modelos Whisper e Vosk, permitindo observar diferenças de desempenho e demandas computacionais.

Os modelos de *LLM* escolhidos foram: *LLaMa 3 8B* da (*Meta AI*)[9] [Dubey et al. 2024]; *Gemma 2 9B* da (*Google DeepMind*)[10] [Gemma Team 2024]; *Mixtral 8x7B Instruct* da (*Mistral AI*)[11] que utiliza a arquitetura *Sparse Mixture of Experts (SMoE)*, ativando apenas parte dos parâmetros durante a inferência, o que reduz custos computacionais sem comprometer o desempenho [Jiang et al. 2024]; e o *GPT-OSS 20B* (*OpenAI*)[12] que emprega arquitetura *Mixture-of-Experts (MoE)* e técnicas de quantização que permitem execução local com menor demanda de hardware [OpenAI 2025].

Os modelos *Speech-to-Text* selecionados são as versões *Small* e *Base* do modelo *Whisper* da (*OpenAI*); *Small* e *Large* do *Toolkit Vosk*[13].

O *ASR Whisper* é um modelo baseado em arquitetura *transformer encoder-decoder*, treinado com grandes volumes de dados multilíngues, apresentando robustez frente a sotaques, ruídos e diferentes idiomas, sendo amplamente utilizado em tarefas de transcrição automática [Graham and Roll 2024, Radford et al. 2023]. O *Vosk* é um *toolkit* de reconhecimento de fala offline compatível com diversos idiomas, incluindo português brasileiro [Alpha Cephei Inc. 2020]. O modelo possui baixo custo computacional e suporte a personalização para domínios específicos, permitindo aplicações em diferentes contextos [Soni 2025].

6. Resultados Obtidos

Esta seção apresenta os resultados obtidos durante o desenvolvimento e a avaliação dos modelos selecionados no *pipeline* de teste proposto. Serão abordados, a produção de um *dataset* para o experimento, os resultados quantitativos e qualitativos da avaliação comparativa entre os modelos de *LLM* e *STT*.

6.1. Dataset

O *dataset* produzido para este estudo é composto por 100 transcrições de aulas em português brasileiro, revisadas manualmente por um dos autores. As amostras foram

[9] `https://ai.meta.com`
[10] `https://deepmind.google`
[11] `https://mistral.ai`
[12] `https://openai.com`
[13] `https://alphacephei.com/vosk/`

distribuídas em 25 aulas entre as áreas de tecnologia, filosofia, administração e literatura. A escolha das disciplinas foi motivada pelo caráter predominantemente verbal das explicações.

As transcrições foram obtidas de videoaulas do *YouTube* sob licença *CC BY (Creative Commons)* e de conteúdos do canal "Prof. Fábio Santos"(autor do artigo). Os dados foram coletados entre agosto e novembro de 2025, com duração entre 3 e 49 minutos. A estrutura da base de dados é exibida na Tabela 1.

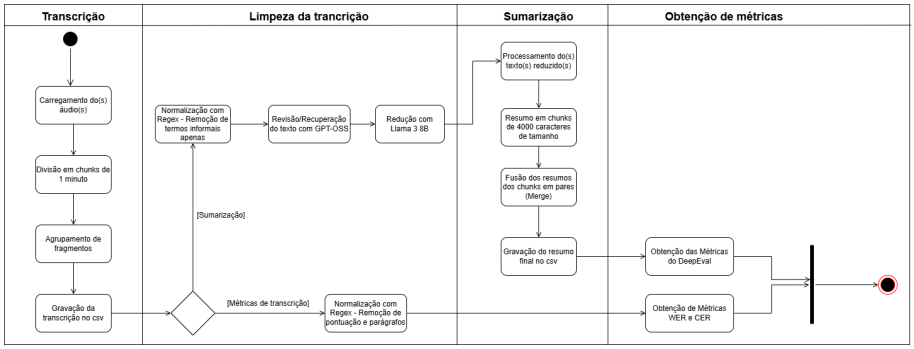
Tabela 1. Estrutura do dataset

Coluna do dataset	Exemplo de conteúdo
id	1
subject	Tecnologia
youtube link	URL do vídeo
video title	Aula 45 - Como Criar um Sistema...
channel	Nome do canal
license	CC BY
speaker gender	male
subject matter	AI programming
duration	11:53
human revised transcription	Olá pessoal, hoje vamos estudar conceitos básicos de programação...

6.2. Abordagem de avaliação

A abordagem de avaliação adotada é composta pelas seguintes etapas: inferência dos modelos de transcrição; técnicas de normalização e pré-processamento de texto; inferência de modelos de *LLM* para revisão e redução do conteúdo; e, por fim, obtenção de métricas dos modelos avaliados. Todas as amostras do *dataset* desenvolvido nesse trabalho foram utilizadas em todas as etapas do experimento. O fluxo de avaliação descrito é apresentado na Figura 2.

Figura 2. Diagrama *UML* de atividades do fluxo de avaliação



O processo de avaliação dos modelos *STT* inicia-se com a divisão do áudio de uma aula em trechos (*chunks*) de 1 minuto. Em seguida, o modelo de *STT* infere cada um desses trechos e, ao final, as inferências são agrupadas. Após esta etapa, as saídas são normalizadas através de expressões regulares. Com essa limpeza as métricas de *WER* e *CER* podem ser obtidas por meio da biblioteca *Jiwer* da linguagem *Python*.

A seguir são descritas as etapas de avaliação dos modelos de *LLMs*:

1. **Revisão via *LLM***: refinamento textual das transcrições geradas pelo modelo com melhor desempenho *(Whisper Small)*. O modelo *GPT-OSS 20B* foi selecionado para essa tarefa por apresentar boa capacidade para seguir instruções e gerar correções contextualmente coerentes em testes preliminares. As saídas demonstraram a correção de termos, recontextualização e recuperação de detalhes perdidos no processo de transcrição.
2. **Redução da revisão**: Esta etapa foi necessária, visto que, a fase anterior não foi suficiente para reduzir o conteúdo da transcrição a somente as explicações. O modelo selecionado para realizar essa tarefa foi o *Llama 3 8B*.
3. **Geração de resumo**: a geração de resumos ocorre por meio de um processo de hierarquização. Primeiro, o texto reduzido é dividido em *chunks* de cerca de 4000 caracteres, respeitando a capacidade de processamento dos modelos. Cada modelo de *LLM* gera resumos para cada *chunk* individualmente, em seguida, os resumos parciais são combinados em pares, em um processo de fusão *(merge)* sucessiva, até que não haja mais pares a unir. Por fim, o texto unificado é refinado para garantir clareza, fluidez, no formato *json*, resultando em um resumo final conciso.

O processo de inferência dos modelos de *LLM* seguiu uma estratégia de engenharia de *prompts* baseada em *instruções diretas (instruction prompting)*, evitando o uso de exemplos (*few-shot prompting*), uma vez que a inclusão de exemplos poderia induzir o modelo a produzir saídas curtas, padronizadas e enviesadas. Essa escolha buscou favorecer um nível maior de detalhamento e fidelidade ao conteúdo das aulas, características desejáveis para materiais educacionais. Os principais trechos dos prompts empregados em cada etapa são apresentados na Tabela 2.

Tabela 2. Estruturas de *prompts* das etapas de redução e sumarização

Etapa	Trecho do *Prompt*
Revisão	Corrija o texto sem alterar o conteúdo.
Redução	Limpe o texto transcrito, removendo ruídos e reduzindo a verbosidade sem alterar a estrutura da narrativa, seguindo estas regras estritas para a limpeza: remover ruído conversacional, reduzir conectivos desnecessários e retorne apenas o texto limpo.
Resumo em *chunks*	Siga as seguintes instruções: 1. Gere um resumo textual coeso com foco apenas no conteúdo ensinado. 2. Mantenha o tom impessoal e técnico, como um material de estudo. 3. Não use o formato JSON.
Merge	Unifique dois resumos parciais em um único texto coeso, eliminando redundâncias, unificando tópicos relacionados e organize a informação. Use tom impessoal, técnico e didático. Gere APENAS o texto unificado, não adicione comentários, introduções ou qualquer formatação além do texto do resumo.
Merge final	Transforme um resumo longo em um documento estruturado e pronto para estudo, mantendo a voz impessoal, técnica e didática. Crie um objeto JSON válido com os campos especificados: título, resumo breve, resumo explicativo, tópico, descrição e pontos-chave.

6.3. Avaliação quantitativa dos *LLMs* com o *DeepEval*

A avaliação quantitativa contou com *framework DeepEval* para aplicar a técnica *LLM-as-a-judge*, que visa julgar o desempenho dos resumos gerados descritos na seção anterior. O modelo selecionado para o papel de juiz foi o *Qwen 2.5 72B Instruct*, que não faz parte

das famílias de modelos avaliados, de forma a reduzir possíveis vieses associados a esse fator nas pontuações atribuídas.

O *framework DeepEval* permitiu a criação de um prompt com métricas customizadas para a análise dos resumos gerados. São elas: *Pedagogical Clarity* (Clareza Pedagógica), que avalia clareza, organização e utilidade do resumo para estudo; *Answer Relevancy* (Relevância da resposta), que verifica o alinhamento do resumo aos conceitos centrais da aula; *Faithfulness* (Fidelidade), que mede a fidelidade das informações em relação à transcrição original; *Hallucination* (Alucinação), que identifica a inserção de informações inexistentes ou distorções conceituais. O avaliador retornava pontuações entre 0 e 1 para cada métrica, juntamente com justificativas textuais (*reasoning*) para auxiliar a análise qualitativa.

6.4. Resultados quantitativos e qualitativos dos modelos *STTs*

A Tabela 3 exibe as taxas de erro obtidas através das métricas *WER* e *CER*, quanto menores os valores maior a precisão. Os modelos *Whisper* demonstraram superioridade em relação aos modelos *Vosk* no experimento.

O *Whisper Small* apresentou o melhor desempenho geral, alcançando *WER* médio de 17,8% e *CER* médio de 7,7%, enquanto o *Whisper Base* apresentou desempenho intermediário, resultado compatível com a diferença de capacidade entre as versões avaliadas. Já o desempenho inferior do *Vosk* pode estar relacionado às diferenças de arquitetura e treinamento, o que pode limitar seu desempenho em tarefas de transcrição mais complexas.

Tabela 3. Taxas de erro calculadas por modelo STT

Modelo	WER	CER	WER (%)	CER (%)
Vosk Large	0.434	0.191	43.4	19.1
Vosk Small	0.413	0.193	41.3	19.3
Whisper Base	0.255	0.108	25.5	10.8
Whisper Small	0.178	0.077	17.8	7.7

6.5. Resultados Quantitativos e Qualitativos das *LLMs*

Os resultados quantitativos indicam que o modelo *GPT-OSS 20B* obteve o melhor desempenho geral, alcançando pontuação média geral de 0.93 e valores superiores a 0.90 em todas as métricas avaliadas. Os modelos *Llama 3 8B* e *Gemma 2 9B* apresentaram resultados próximos, enquanto o *Mixtral 8x7B Instruct* obteve desempenho inferior (Tabela 4).

Tabela 4. Desempenho médio dos modelos na tarefa de sumarização

Modelo	Média geral
Gemma 2 9b	0.87
GPT-OSS 20b	0.93
Llama 3 8b	0.88
Mixtral 8x7 instruct	0.66

Na análise qualitativa, o modelo *GPT-OSS 20B* destacou-se pela geração de resumos descritivos que preservaram informações relevantes, *emojis* e palavras-chave compatíveis com os temas das aulas. Os modelos *Llama 3 8B* e *Gemma 2 9B* também foram capazes de produzir resumos com boa qualidade textual, contudo, foram encontradas inconsistências pontuais de formatação estrutural de saída (formato *JSON*) e a geração de *emojis* incompatíveis com o texto. Os mesmos problemas foram encontrados com maior frequência nos resultados das inferências do modelo *Mixtral 8x7B Instruct*, que apresentou alucinações como alterações no idioma original.

Considerando os resultados quantitativos e qualitativos, o *GPT-OSS 20B* apresentou o melhor desempenho geral e em todas as métricas de sumarização, incluindo relevância pedagógica (0.91). O resultado corresponde às expectativas, visto que o modelo possui um número maior de parâmetros em relação aos demais. Entretanto, *Gemma 2 9B* e *Llama 3 8B* apresentaram resultados próximos em desempenho geral e em métricas (relevância pedagógica igual a 0.86 e 0.87, respectivamente), indicando que modelos leves e médios podem ser utilizados em contextos educacionais com recursos computacionais limitados.

6.6. Ameaças à validade do estudo

Algumas limitações devem ser consideradas na interpretação dos resultados apresentados. O *dataset* utilizado no experimento é composto exclusivamente por videoaulas em português brasileiro dentro das áreas de conhecimento expostas, limitando a generalização a outros domínios e idiomas.

A avaliação qualitativa foi realizada por um único avaliador, o que pode introduzir subjetividade na análise. Além disso, a utilização de *LLMs* como juízes pode incorporar vieses do próprio modelo avaliador, favorecendo respostas alinhadas aos seus próprios padrões de geração [Wataoka et al. 2025]. Para mitigar esse efeito, a abordagem *LLM-as-a-judge* deve ser analisada em conjunto com a avaliação humana.

Por fim, os experimentos foram executados em dois ambientes computacionais distintos: um servidor Ubuntu com processador *Intel i7-8700* e duas GPUs *NVIDIA GTX 1080*, utilizado nos testes de transcrição e sumarização para os modelos *Llama 3 8B* e *GPT-OSS 20B*; e a plataforma *Kaggle*, com duas GPUs *NVIDIA Tesla T4*, empregadas nos testes de sumarização dos demais modelos. Em razão da diferença entre os ambientes, os tempos de inferência não foram considerados na comparação; a análise concentrou-se, portanto, na qualidade das saídas geradas e não no desempenho computacional.

7. Conclusões

O desenvolvimento da solução *Manuscribe AI* evidenciou que é possível construir uma ferramenta de transcrição e sumarização educacional de forma acessível utilizando modelos de inteligência artificial abertos e de menor porte, sem dependência de assinaturas ou serviços comerciais. A combinação entre transcrição, sumarização e materiais de estudo demonstra que soluções desse tipo podem ser implementadas de forma viável, beneficiando principalmente públicos que enfrentam limitações para registrar ou organizar conteúdos, como estudantes com dificuldades motoras finas.

A avaliação também evidenciou que modelos abertos podem desempenhar tarefas educacionais de forma satisfatória quando aliados à engenharia de *prompts*. A criação

do *dataset* em português contribui para pesquisas futuras , visto que a disponibilidade de recursos desse tipo é escassa.

8. Trabalhos futuros

Como desdobramentos futuros, o projeto prevê a inclusão de funcionalidades de geração de *flashcards* visuais e *quizzes* relacionados ao assunto das aulas, a implementação de um banco de dados para armazenar os materiais produzidos e a realização de testes com estudantes com dificuldades motoras finas. Tais melhorias visam ampliar a utilidade do sistema, tornando-o mais completo e possibilitando novas investigações sobre seu impacto em relação à acessibilidade e aos diferentes perfis de aprendizagem.

Agradecimentos

Os autores gostariam de agradecer ao SiDi Intelligence & Innovation Center e à Universidade Estadual do Amazonas (UEA) pelo apoio durante o desenvolvimento deste trabalho.

Referências

Alpha Cephei Inc. (2020). Vosk speech recognition toolkit. Software. Disponível em: https://github.com/alphacep/vosk-api. Acesso em: 4 jun. 2025.

Barbosa, F. D. D., Mariano, E. d. F., and Sousa, J. d. (2021). Tecnologia e educação: perspectivas e desafios para a ação docente. *Conjecturas*, 21(2):38–60.

Boyle, J. R. (2010). Note-taking skills of middle school students with and without learning disabilities. *Journal of Learning Disabilities*, 43(6):530–540. Disponível em: https://doi.org/10.1177/0022219410371679. Acesso em: 10 jun. 2025.

Boyle, J. R. (2012). Note–taking and secondary students with learning disabilities: Challenges and solutions. *Learning Disabilities Research & Practice*, 27(2):90–101. Disponível em: https://journals.sagepub.com/doi/abs/10.1111/j.1540-5826.2012.00354.x. Acesso em: 10 jun. 2025.

Chang, Y., Wang, X., Wang, J., Han, Y., Li, C., Xu, S., Zhu, H., Dong, B., Zhang, Y., Yao, K., Lu, Z., Wu, Y., and Zhang, W. (2023). A survey on evaluation of large language models. arXiv preprint arXiv:2307.03109.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., and Wirth, R. (2000). Crisp-dm 1.0: Step-by-step data mining guide. CRISP-DM Consortium.

Correa, Y., Moro, T. B., and Valentini, C. B. (2021). Tecnologia assistiva na educação inclusiva. *Revista Ibero-Americana de Estudos em Educação*, pages 2963–2970.

Dubey, A., Jauhri, A., Pandey, A., et al. (2024). The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Gemma Team (2024). Gemma 2: Improving open language models at a practical size. Disponível em: https://arxiv.org/abs/2408.00118. Acesso em: 06 jun. 2026.

Graham, C. and Roll, N. (2024). Evaluating openai's whisper asr: Performance analysis across diverse accents and speaker traits. *JASA Express Letters*, 4(2).

Hadi, M. U., Al-Tashi, Q., Qureshi, R., Shah, A., Muneer, A., Irfan, M., Zafar, A., Shaikh, M., Akhtar, N., Wu, J., and Mirjalili, S. (2023). Large language models: A comprehensive survey of its applications, challenges, limitations, and future prospects.

Jiang, A. Q. et al. (2024). Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Knowt (2025). Student's guide to getting started. Disponível em: https://help.knowt.com/en/articles/10722204-student-s-guide-to-getting-started. Acesso em: 7 jun. 2025.

Kodalovic, M. (2025). Análise do plaud.ai — principais recursos, prós e contras explicados. https://www.bluedothq.com/pt-br/blog/plaud-review. Acesso em: 30 nov. 2025.

Lai, A. (2025). What is otter? Disponível em: https://help.otter.ai/hc/en-us/articles/360035266494-What-is-Otter. Acesso em: 7 jun. 2025.

OpenAI (2025). gpt-oss-120b & gpt-oss-20b model card.

Otter.ai (2025). Otterai transcribe voice notes. Disponível em: https://play.google.com/store/apps/details?id=com.aisense.otter. Acesso em: 7 jun. 2025.

Peng, J., Wang, Y., Fang, Y., Xi, Y., Li, X., Zhang, X., and Yu, K. (2025). A survey on speech large language models. Disponível em: https://arxiv.org/abs/2410.18908. Acesso em: 4 jun. 2025.

Quizlet (2024). Como funciona o quizlet. Disponível em: https://quizlet.com/br/features/how-quizlet-works. Acesso em: 8 jun. 2025.

Radford, A. et al. (2023). Robust speech recognition via large-scale weak supervision. Online.

Ranzato, M., Chopra, S., Auli, M., and Zaremba, W. (2016). Sequence level training with recurrent neural networks. Disponível em: https://arxiv.org/abs/1511.06732. Acesso em: 4 jun. 2025.

Ravaglia, R. (2024). Quizlet using ai as a socratic tool to ask better questions. *Forbes*. Disponível em: https://www.forbes.com/sites/rayravaglia/2024/02/21/quizlet-using-ai-as-a-socratic-tool-to-ask-better-questions/. Acesso em: 8 jun. 2025.

Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.-F., and Dennison, D. (2015). Hidden technical debt in machine learning systems. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'15, page 2503–2511, Cambridge, MA, USA. MIT Press.

Soni, A. A. (2025). Improving speech recognition accuracy using custom language models with the vosk toolkit. Disponível em: https://arxiv.org/abs/2503.21025. Acesso em: 7 jun. 2025.

Sutherland, A. (2010). Quizlet's Andrew Sutherland on The Morning Show. YouTube video. Disponível em: https://www.youtube.com/watch?v=ilSPvgo0K38. Acesso em: 8 jun. 2025.

Svensson, I., Nordström, T., Lindeblad, E., Gustafson, S., Björn, M., Sand, C., Almgren/Bäck, G., and and, S. N. (2019). Effects of assistive technology for students with reading and writing disabilities. *Disability and Rehabilitation: Assistive Technology*, 16(2):196–208. Disponível em: https://doi.org/10.1080/17483107.2019.1646821. Acesso em 9 jun. 2025.

Tavares, A. A. and Cardoso, A. A. (2016). Inter-relações entre o desempenho no processo de aprendizagem escolar e o desenvolvimento das capacidades motoras: revisão da literatura. *Revista de Terapia Ocupacional da Universidade de São Paulo*, 27(1):88–93. Disponível em: https://revistas.usp.br/rto/article/view/111098. Acesso em: 9 jun. 2025.

Transkriptor (2024). Transkriptor: O melhor aplicativo de conversão de fala para texto com ia. Disponível em: https://play.google.com/store/apps/details?id=com.transkriptor.transcribe. Acesso em: 4 jun. 2025.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, volume 30.

Wataoka, K., Takahashi, T., and Ri, R. (2025). Self-preference bias in llm-as-a-judge. Disponível em: https://arxiv.org/abs/2410.21819. Acesso em: 06 jun. 2026.

Xiao, T. and Zhu, J. (2025). Foundations of large language models. Disponível em: https://arxiv.org/abs/2501.09223. Acesso em: 4 jun. 2025.

Xu, J., Wu, A., Filip, C., Patel, Z., Bernstein, S. R., Tanveer, R., Syed, H., and Kotroczo, T. (2024). Active recall strategies associated with academic achievement in young adults: A systematic review. *Journal of Affective Disorders*, 354:191–198. Disponível em: https://www.sciencedirect.com/science/article/pii/S0165032724004245. Acesso em: 9 jun. 2025.

Zhang, A., Lipton, Z. C., Li, M., and Smola, A. J. (2023). Dive into deep learning. Disponível em: https://arxiv.org/abs/2106.11342. Acesso em: 3 jun. 2025.

Zhang, Y., Jin, H., Meng, D., Wang, J., and Tan, J. (2025). A comprehensive survey on process-oriented automatic text summarization with exploration of llm-based methods. Disponível em: https://arxiv.org/abs/2403.02901. Acesso em: 12 jun. 2025.