v7.n11.2026

REVISTA CIENTÍFICA - RECIMA21 ISSN 2675-6218

CORTEX: PLATAFORMA PARA APLICAR INTELIGÊNCIA ARTIFICIAL NA SELEÇÃO E EDIÇÃO DE TRECHOS RELEVANTES DE VÍDEOS

CORTEX: A PLATFORM FOR APPLYING ARTIFICIAL INTELLIGENCE TO THE SELECTION AND EDITING OF RELEVANT VIDEO SEGMENTS

CORTEX: PLATAFORMA PARA APLICAR INTELIGENCIA ARTIFICIAL EN LA SELECCIÓN Y EDICIÓN DE FRAGMENTOS RELEVANTES DE VIDEOS

Daniel Bogiam Alves[1], Felipe Dallilo[2]

e7119597
https://doi.org/10.47820/recima21.v7i11.9597

PUBLICADO: 11/2026

RESUMO

Transformar um vídeo longo — um *podcast*, uma entrevista ou uma aula gravada — em dezenas de clipes curtos para redes sociais exige assistir ao material inteiro, escolher os trechos, cortar, reenquadrar para o formato vertical e legendar cada um, tarefa repetitiva que consome horas de trabalho por vídeo. Este trabalho apresenta o Cortex, plataforma *web* que automatiza esse fluxo aplicando modelos de Inteligência Artificial. O objetivo foi executar todas essas etapas sem intervenção humana, com critérios de decisão explícitos e documentados. A metodologia foi a pesquisa aplicada, qualiquantitativa, conduzida como estudo de caso do artefato. A plataforma encadeia quatro modelos pré-treinados de terceiros, empregados como extratores de características — reconhecimento automático de fala, representação semântica de sentenças, análise de energia sonora e detecção facial — e aplica sobre suas saídas um critério de pontuação implementado pelo autor, denominado Viral Score, que ordena os trechos candidatos e descarta os sobrepostos. Nos ensaios, a plataforma converteu um vídeo de uma hora em vinte clipes legendados e reenquadrados em menos de dez minutos; a comparação com a edição manual de uma amostra de três clipes sugere redução de tempo superior a 95%, valor indicativo. A avaliação qualitativa, conduzida pelo autor, tem caráter exploratório. Conclui-se que o fluxo é viável com bibliotecas de código aberto e critérios de decisão auditáveis.

PALAVRAS-CHAVE: Cortex. Edição automatizada de vídeos. Inteligência artificial aplicada. Processamento de linguagem natural. Reconhecimento automático de fala. Viral Score.

ABSTRACT

Turning a long video — a podcast, an interview or a recorded lecture — into dozens of short clips for social media requires watching the entire material, choosing the segments, cutting, reframing to the vertical format and captioning each one, a repetitive task that consumes hours of manual work per video. This paper presents Cortex, a web platform that automates this workflow by applying Artificial Intelligence models. The objective was to perform all these steps without human intervention, under explicit and documented decision criteria. The methodology was applied research, qualitative and quantitative, conducted as a case study of the artifact developed. The platform chains four third-party pre-trained models, used as feature extractors — automatic speech recognition, sentence semantic representation, sound energy analysis and face detection — and applies to their outputs a scoring criterion implemented by the author, named Viral Score, which ranks the candidate segments and discards overlapping ones.

[1] Universidade de Araraquara, graduando.
[2] Universidade de Araraquara, doutor.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

REVISTA CIENTÍFICA - RECIMA21 ISSN 2675-6218

In the trials, the platform converted a one-hour video into twenty captioned and reframed clips in under ten minutes; a comparison with the manual editing of a sample of three clips suggests a time reduction of more than 95%, an indicative figure. The qualitative evaluation, carried out by the author, is exploratory. It is concluded that the workflow is feasible with open-source libraries and auditable decision criteria.

KEYWORDS: *Applied artificial intelligence. Automated video editing. Automatic speech recognition. Cortex. Natural language processing. Viral Score.*

RESUMEN
Transformar un video largo —un pódcast, una entrevista o una clase grabada— en decenas de clips cortos para redes sociales exige ver el material completo, elegir los fragmentos, cortar, reencuadrar al formato vertical y subtitular cada uno, una tarea repetitiva que consume horas de trabajo por video. Este trabajo presenta Cortex, plataforma web que automatiza este flujo mediante la aplicación de modelos de Inteligencia Artificial. El objetivo fue ejecutar todas estas etapas sin intervención humana, con criterios de decisión explícitos y documentados. La metodología fue la investigación aplicada, cualicuantitativa, conducida como estudio de caso del artefacto. La plataforma encadena cuatro modelos preentrenados de terceros, empleados como extractores de características —reconocimiento automático del habla, representación semántica de oraciones, análisis de energía sonora y detección facial— y aplica sobre sus salidas un criterio de puntuación implementado por el autor, denominado Viral Score, que ordena los fragmentos candidatos y descarta los superpuestos. En los ensayos, la plataforma convirtió un video de una hora en veinte clips subtitulados y reencuadrados en menos de diez minutos; la comparación con la edición manual de una muestra de tres clips sugiere una reducción de tiempo superior al 95%, valor indicativo. La evaluación cualitativa, realizada por el autor, tiene carácter exploratorio. Se concluye que el flujo es viable con bibliotecas de código abierto y criterios de decisión auditables.

PALABRAS CLAVE: *Cortex. Edición automatizada de videos. Inteligencia artificial aplicada. Procesamiento del lenguaje natural. Reconocimiento automático del habla. Viral Score.*

1. INTRODUÇÃO

O avanço das tecnologias digitais impulsionou a produção e o consumo de conteúdo audiovisual, sobretudo em plataformas de vídeos curtos como TikTok, Instagram e YouTube. Criadores de conteúdo frequentemente transformam vídeos extensos — *podcasts*, entrevistas e transmissões ao vivo — em múltiplos clipes curtos para ampliar o alcance e o engajamento, tarefa que exige analisar o material inteiro, identificar os trechos relevantes e editá-los manualmente, o que a torna repetitiva e pouco escalável.

Técnicas de aprendizado profundo aplicadas ao áudio, à linguagem natural e à visão computacional viabilizaram a automação dessas tarefas: transcrição da fala, legendas sincronizadas e identificação de trechos com maior potencial de engajamento. Ferramentas como o Opus Clip (OPUS CLIP, 2026), o CapCut (CAPCUT, 2026) e o Descript (DESCRIPT, 2026) já incorporam, em graus variáveis, algoritmos de Inteligência Artificial para recortar,

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

legendar e reenquadrar vídeos longos, combinando o que é dito, como é dito e o que aparece em quadro para decidir, sem intervenção humana, o que recortar e como apresentar.

O objeto deste estudo é o Cortex, plataforma *web* que executa esse fluxo de ponta a ponta: recebe um vídeo longo, transcreve a fala, pontua os segmentos, seleciona os melhores trechos, reenquadra o vídeo para o formato vertical acompanhando o rosto do locutor e queima legendas sincronizadas palavra a palavra. A plataforma não treina modelos próprios: emprega quatro modelos pré-treinados de terceiros como extratores de características e decide a partir de suas saídas por meio do Viral Score, critério de pontuação implementado pelo autor. A seleção apoia-se em fala, texto, semântica e energia sonora; a modalidade visual atua apenas no reenquadramento, por detecção facial.

A lacuna que o trabalho pretende preencher é técnica e metodológica: as soluções comerciais operam como sistemas fechados, sem documentar quais sinais combinam nem como os ponderam. Este trabalho descreve, de forma aberta e reproduzível, como as saídas de modelos pré-treinados podem ser alinhadas no tempo e combinadas em um critério de decisão explícito.

A questão de pesquisa que orienta este trabalho é: como combinar sinais multimodais extraídos por modelos de Inteligência Artificial pré-treinados para identificar automaticamente, com precisão suficiente para uso prático, os trechos mais relevantes de um vídeo longo para a geração de conteúdo curto, sem recorrer a modelos proprietários e sem depender de anotação humana prévia?

Como hipótese, considera-se que a combinação de sinais heterogêneos — energia sonora, ritmo e pausas da fala, marcadores lexicais característicos de abertura de narrativa e mudança semântica entre segmentos consecutivos — permite selecionar automaticamente trechos coerentes e relevantes sem treinamento de modelos nem anotação prévia, e que a delimitação das fronteiras do clipe pela estrutura do discurso produz cortes mais coerentes do que o janelamento por tempo fixo.

O objetivo geral é desenvolver e avaliar o Cortex, plataforma *web* que aplica modelos de Inteligência Artificial pré-treinados para selecionar automaticamente os trechos de maior potencial de engajamento em vídeos longos e editá-los em clipes curtos, tendo como núcleo de decisão o Viral Score. Constituem objetivos específicos:

- Integrar modelos pré-treinados de reconhecimento de fala, representação semântica de sentenças, análise de energia sonora e detecção facial em um fluxo único, alinhado pela transcrição com marcação temporal palavra a palavra;

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

- Desenvolver o algoritmo Viral Score, que pontua os segmentos a partir de sinais lexicais, prosódicos, semânticos e de energia sonora, com pesos explícitos e documentados;
- Selecionar os clipes por ranqueamento das pontuações e supressão de sobreposições;
- Automatizar o reenquadramento para os formatos vertical e quadrado com base na detecção do rosto do locutor;
- Gerar legendas dinâmicas sincronizadas, customizáveis quanto a fonte, cor e posição;
- Construir a plataforma com arquitetura desacoplada — interface de usuário, interface de programação e fila de processamento assíncrono —, com suporte a múltiplos usuários;
- Avaliar a plataforma por meio de medições de desempenho e de análise qualitativa dos clipes gerados.

A justificativa apoia-se em três dimensões. Socialmente, a automação reduz a barreira técnica de produção de conteúdo audiovisual para criadores independentes, educadores e pequenas organizações sem equipe de edição. Economicamente, o mercado de ferramentas de recorte automático está em expansão, com soluções pagas por assinatura ou por crédito de processamento, o que evidencia demanda real. Academicamente, a contribuição reside na documentação explícita do critério de decisão: enquanto as ferramentas comerciais entregam uma pontuação de viralidade sem revelar o que a compõe, este trabalho apresenta os pesos adotados, o código que os aplica e as razões de cada escolha, demonstrando que modelos pré-treinados de diferentes modalidades podem ser combinados por uma regra transparente, sem treinamento próprio nem dados anotados.

2. REFERENCIAL TEÓRICO

Esta seção apresenta o referencial teórico da solução: edição audiovisual, Inteligência Artificial aplicada a vídeo, reconhecimento de fala, processamento de linguagem natural, análise de áudio, visão computacional e as ferramentas de engenharia de *software* empregadas.

2.1. Edição de vídeos e o formato vertical

A edição de vídeos consiste na manipulação e organização de conteúdos audiovisuais — cortes, sincronização de áudio, legendas, transições e ajustes visuais — para transmitir

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

informações com eficiência e aumentar o engajamento. A consolidação do consumo em dispositivos móveis tornou dominante o formato vertical 9:16, o que impõe um problema técnico: o material horizontal precisa ser reenquadrado, e o simples corte central frequentemente descarta o locutor quando ele não está no centro do quadro.

Outro requisito do formato é a legendagem: como parcela expressiva do consumo ocorre com o som desligado, a legenda torna-se condição de compreensão. O padrão difundido emprega poucas palavras por quadro, alto contraste e destaque progressivo sincronizado com a fala, o que exige marcação temporal em nível de palavra.

2.2. Inteligência Artificial

Segundo Russell e Norvig (2021), a Inteligência Artificial ocupa-se do desenvolvimento de agentes capazes de perceber seu ambiente e agir racionalmente para atingir um objetivo definido. Com o aprendizado profundo, redes neurais passaram a ser amplamente usadas em visão computacional, reconhecimento de fala e interpretação textual, viabilizando, na área audiovisual, a geração de legendas, a classificação de cenas e a recomendação de conteúdo.

Cabe uma distinção que atravessa todo o trabalho: a plataforma não treina modelos próprios. Ela emprega modelos pré-treinados como extratores de características — a transcrição, os vetores de sentença e as regiões de rosto — e decide a partir dessas saídas, prática corrente na Inteligência Artificial aplicada, já que treinar um reconhecedor de fala exigiria dezenas de milhares de horas de áudio rotulado. O que é próprio deste trabalho é o critério que combina essas saídas: quais grandezas extrair, que peso atribuir a cada uma e como convertê-las em uma decisão de corte. Esse critério é o Viral Score, apresentado na seção 3.5.4.

2.3. Sumarização automática e detecção de *highlights*

A seleção automática dos trechos mais representativos de um vídeo é tratada na literatura como sumarização automática de vídeos. Zhang *et al.* (2016) formularam o problema como predição sequencial, mostrando que redes recorrentes com memória de longo prazo capturam dependências temporais entre quadros que a similaridade visual isolada não alcança. Panda e Roy-Chowdhury (2017) estimaram a relevância de uma tomada pela recorrência de conceitos visuais em vídeos do mesmo tema, argumentando que a relevância depende do contexto. Ambos operam sobre características visuais, ao passo que a seleção no Cortex apoia-se em fala, texto e áudio.

Há ainda uma diferença de escopo: esses métodos são supervisionados e dependem de dados anotados quadro a quadro, onerosos de produzir e praticamente inexistentes em

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

português. O Cortex adota uma abordagem não supervisionada e heurística, com pesos definidos empiricamente (seção 3.5.4); perde-se a garantia de otimalidade, mas ganha-se a dispensa de anotação e a transparência do critério de decisão.

2.4. Reconhecimento automático de fala

O reconhecimento automático de fala converte o áudio em texto e constitui a etapa inicial da plataforma. Radford *et al.* (2022) apresentaram o Whisper, treinado sobre grande volume de áudio multilíngue com supervisão fraca, cuja escala lhe confere robustez a ruído e sotaque sem necessidade de ajuste fino. Entre as variantes de tamanho do modelo, adotou-se a *medium*, pelo equilíbrio entre acurácia em português e tempo de inferência no *hardware* disponível.

Utilizou-se a reimplementação faster-whisper (SYSTRAN, 2026), construída sobre o motor de inferência CTranslate2, que aplica quantização de pesos — inteiros de oito bits na unidade central de processamento (CPU) e ponto flutuante de meia precisão na unidade de processamento gráfico (GPU) — e reduz substancialmente o tempo de transcrição, a etapa mais custosa do fluxo, sem alterar a arquitetura do modelo. As decisões tomadas sobre ela são descritas na seção 3.5.1.

2.5. Processamento de linguagem natural e representação semântica

O processamento de linguagem natural analisa o texto transcrito; entre as tarefas da área descritas por Jurafsky e Martin (2023) estão a segmentação do discurso e a identificação de fronteiras temáticas, aplicadas neste trabalho. Para representar o significado das sentenças, empregam-se *sentence embeddings*: Reimers e Gurevych (2019) propuseram o Sentence-BERT, arquitetura siamesa que produz vetores comparáveis por similaridade de cosseno a custo muito inferior ao do uso direto de modelos do tipo BERT. Utiliza-se o all-MiniLM-L6-v2, variante compacta e destilada dessa família.

Na plataforma, essas representações têm uso duplo: a queda abrupta de similaridade entre segmentos consecutivos marca uma fronteira de assunto, e a mesma variação compõe um bônus da pontuação, sob a premissa de que a introdução de um tema novo tende a iniciar um trecho autocontido. O limiar de similaridade é escolha deste trabalho (seção 3.5.3).

2.6. Análise de sinais de áudio

Rabiner e Schafer (2011) apresentam a energia de curto prazo como descritor elementar da intensidade da fala. A plataforma utiliza a biblioteca librosa, descrita por McFee *et al.* (2015), para calcular a raiz do valor quadrático médio do sinal ao longo do tempo; o critério que converte

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

essa curva em picos é deste trabalho (seção 3.5.2). Os picos são tratados como indício de ênfase — elevação de voz, riso ou reação —, e a duração das pausas entre segmentos, obtida da transcrição, como indicador de fronteira natural do discurso.

2.7. Detecção facial e reenquadramento automático

Para converter material horizontal em vertical, a plataforma decide qual região preservar por meio de detecção facial em amostras de quadros. Adotou-se o detector neural distribuído com o OpenCV (OPENCV, 2026), que implementa a arquitetura Single Shot MultiBox Detector, proposta por Liu *et al.* (2016), sobre uma rede ResNet de dez camadas, detectando em uma única passagem pela rede. Ele substituiu os classificadores em cascata de Haar inicialmente testados, por ser mais robusto a rostos de perfil, iluminação insuficiente e faces pequenas. A amostragem, a escolha entre múltiplos rostos e a conversão em ponto de corte são decisões deste trabalho (seção 3.5.7).

2.8. Ferramentas de engenharia de *software*

Além das bibliotecas de Inteligência Artificial, a plataforma, escrita em Python (PYTHON *SOFTWARE* FOUNDATION, 2026) no lado do servidor, apoia-se em tecnologias de infraestrutura escolhidas a partir da documentação oficial de cada projeto. O FastAPI (RAMÍREZ, 2026) oferece validação automática dos dados de entrada, documentação interativa gerada a partir do código e suporte nativo a operações assíncronas; o Celery (CELERY PROJECT, 2026), com o Redis como intermediário, desacopla o envio da execução, que leva minutos, e permite reportar o progresso; o FFmpeg (FFMPEG PROJECT, 2026) executa corte, reenquadramento, redimensionamento e queima de legendas em uma única passagem de codificação, evitando perdas sucessivas de qualidade; o MySQL, com o mapeamento objeto-relacional do SQLAlchemy, persiste usuários e histórico; e o Next.js (VERCEL, 2026), com React, permite componentizar a interface e revalidar automaticamente os dados, adequado ao acompanhamento do estado das tarefas.

2.9. Ferramentas correlatas no mercado

Ferramentas como CapCut e Descript oferecem legendas automáticas e cortes por detecção de silêncio, e plataformas especializadas, como o Opus Clip, selecionam trechos por pontuação de viralidade e reenquadram com rastreamento do locutor. Todas operam como sistemas fechados: os critérios de pontuação não são publicados, o que impede a auditoria do resultado e a adaptação a domínios específicos — uma aula expositiva e um programa de humor

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

demandam critérios distintos de relevância. Soma-se a isso o modelo comercial baseado em créditos de processamento, que onera o uso continuado. Essas limitações motivam uma solução cujos critérios sejam explícitos e ajustáveis por quem opera a plataforma — na versão descrita, por alteração do código-fonte.

3. METODOLOGIA

Quanto à natureza, a pesquisa é aplicada; quanto aos objetivos, exploratória e descritiva; e quanto à abordagem, qualiquantitativa — qualitativa na avaliação da coerência dos clipes e quantitativa na medição dos tempos de processamento. Quanto aos procedimentos, combinou-se a pesquisa bibliográfica, sobre reconhecimento de fala, processamento de linguagem natural, sumarização de vídeos e análise de áudio, apoiada também na documentação oficial das bibliotecas, ao estudo de caso da plataforma desenvolvida.

O universo de análise é composto por vídeos longos com predominância de fala em português — *podcasts*, entrevistas e aulas gravadas. Os instrumentos de coleta foram a instrumentação do próprio sistema, que registra os tempos de cada etapa, e a inspeção manual dos clipes; a análise comparou os tempos medidos ao tempo estimado de execução manual e apreciou qualitativamente a completude do trecho, a sincronia da legenda e o enquadramento. A avaliação qualitativa foi conduzida pelo próprio autor, sem avaliadores independentes, e o tempo de execução manual foi estimado a partir de uma amostra de três clipes; ambos os procedimentos têm, portanto, caráter exploratório, e seus resultados indicam tendências, e não estimativas generalizáveis (seções 4.4 e 4.5). O desenvolvimento foi incremental, partindo de um protótipo de linha de comando para transcrição e corte e incorporando, a cada iteração, o reenquadramento, as legendas dinâmicas, a fila assíncrona, a autenticação e a interface *web*.

Esta seção descreve a construção do Cortex, cujo código está organizado em dois repositórios: um para a interface de usuário e outro para a interface de programação e o processamento de mídia. Sempre que uma etapa combina biblioteca de terceiros e decisão própria, o texto separa o que a biblioteca entrega do que foi implementado neste trabalho. Os excertos de código de cada etapa do fluxo de processamento, com o detalhamento da implementação, estão reunidos no material suplementar deste artigo.

3.1. Visão geral da arquitetura

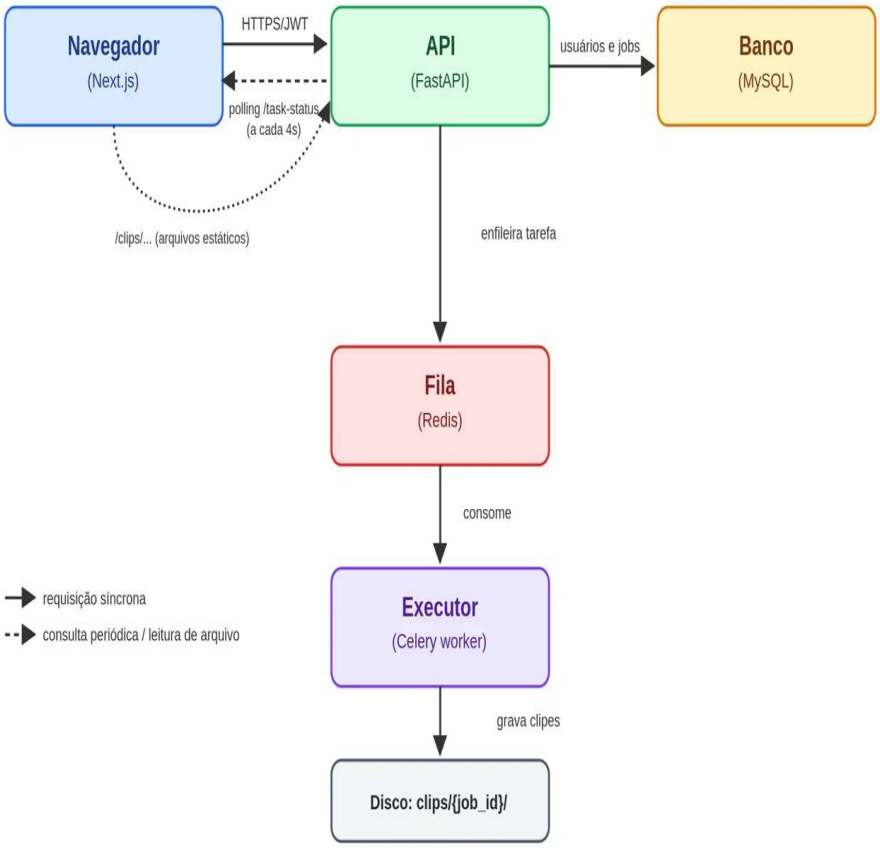
A plataforma foi estruturada em cinco componentes desacoplados, que se comunicam por protocolos de rede. A interface de usuário, em Next.js, comunica-se por requisições HTTP

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

autenticadas com a interface de programação, em FastAPI, que recebe o vídeo, registra o processamento no banco de dados e enfileira a tarefa. O executor Celery consome a fila e realiza o processamento pesado. O Redis atua como intermediário de mensagens, repositório do estado das tarefas e armazenamento dos contadores de limitação de requisições. O MySQL, acessado por meio do SQLAlchemy apenas pela interface de programação, persiste os usuários e o histórico; o executor recebe da fila tudo de que precisa.

Essa separação responde a uma restrição do domínio: processar um vídeo de uma hora leva minutos, e fazê-lo dentro do ciclo de requisição HTTP manteria a conexão aberta e esgotaria as conexões do servidor. Com a fila, o envio retorna imediatamente um identificador de tarefa, consultado periodicamente pelo navegador (Figura 1).

Figura 1. Arquitetura geral da plataforma



Fonte: o autor, 2026.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

3.2. Interface de usuário

A interface foi construída com Next.js, React e Tailwind CSS e tem três rotas: cadastro, autenticação e painel de controle.

No painel, o usuário envia o vídeo e define quatro parâmetros aplicados a todos os clipes: o formato de corte; a faixa de duração — automática (10 a 180 s, acompanhando o encerramento natural da fala), menos de 30 s, 30 a 59 s ou 60 a 89 s; o estilo da legenda; e sua posição na tela — topo, meio ou base.

O estilo da legenda é escolhido entre dez predefinições de fonte e cor, restritas às fontes presentes no sistema do executor, pois o renderizador substitui silenciosamente uma fonte ausente, divergindo da pré-visualização. O formulário é apresentado na Figura 2, na seção 4.2.

O histórico é atualizado por consulta periódica a cada quatro segundos enquanto houver tarefa em andamento; cada processamento concluído exibe a grade de miniaturas dos clipes e, em caso de falha, o detalhamento técnico do erro.

3.3. Interface de programação e persistência

A interface de programação expõe cinco rotas: cadastro e autenticação, limitadas a cinco requisições por minuto; envio do vídeo, que registra e enfileira o processamento, limitado a cinco requisições por hora; consulta do estado de um processamento; e listagem do histórico do usuário. As rotas de processamento exigem autenticação por *token*.

A autenticação usa *tokens* JSON Web Token assinados com o algoritmo HS256, e as senhas são armazenadas apenas como resumo bcrypt. O modelo de dados tem duas entidades, usuário e processamento, e as rotas de estado e de histórico verificam a propriedade do processamento antes de devolver qualquer caminho. A limitação de requisições usa a biblioteca slowapi, com contadores no Redis, aplicada por usuário quando há *token* válido e por endereço de rede caso contrário.

3.4. Processamento assíncrono

A tarefa assíncrona recebe o caminho do arquivo e os parâmetros, delega a execução ao orquestrador do fluxo e atualiza seu estado com a descrição da etapa corrente, exibida ao usuário. Um bloco de finalização remove o arquivo bruto original tanto em caso de sucesso quanto de falha, evitando que vídeos longos e falhas repetidas acumulem arquivos em disco.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

3.5. Fluxo de processamento

O fluxo de processamento é o núcleo do Cortex. Para o usuário, tem três fases — transcrição, detecção de momentos e geração dos clipes —, mas internamente compõe-se de dez etapas, descritas a seguir, cada uma acrescentando estrutura a um fluxo contínuo de áudio e imagem, da transcrição ancorada no tempo à conversão de cada trecho em arquivo publicável.

Das dez etapas, quatro são integralmente de autoria própria; nas demais, a biblioteca fornece a operação de base e a decisão sobre o que fazer com ela é deste trabalho.

3.5.1. Transcrição

O restante do fluxo consome o texto ancorado no tempo, com precisão suficiente para que um corte não decepe uma palavra e a legenda acompanhe a fala. O reconhecimento é feito pelo faster-whisper (seção 2.4); são decisões deste trabalho a parametrização do modelo, a seleção de dispositivo e a normalização da saída.

O modelo *medium* foi preferido ao *large* porque o ganho de acurácia não compensou o acréscimo de tempo, e o idioma foi fixado em português para que silêncio ou música no início do vídeo não levem o detector a inferir outro idioma. A execução usa a GPU quando disponível e, em caso de falha no carregamento, degrada para a CPU, em inteiros de oito bits, em vez de interromper o processo.

A saída da biblioteca é convertida em uma lista de segmentos com texto, início, fim e marcação temporal de cada palavra, formato de intercâmbio de todas as etapas seguintes; é a marcação por palavra que viabiliza a precisão dos cortes e o destaque progressivo das legendas (seção 3.5.8).

3.5.2. Detecção de picos de energia

A transcrição registra o que foi dito, mas não como: riso, elevação de voz e reação do locutor não deixam rastro no texto, e são justamente esses momentos que definem um bom clipe. A biblioteca librosa calcula a raiz do valor quadrático médio do sinal ao longo do tempo, preservando a taxa de amostragem original, já que a envoltória de energia é insensível a ela.

O limiar de pico é fixado em 1,5 vez a energia média do próprio vídeo, e não em valor absoluto, o que normaliza implicitamente o volume; o fator foi ajustado por inspeção — valores menores marcavam a fala normal como pico, e maiores só reagiam a saturações do microfone. Como um trecho de volume sustentado gera dezenas de instantes acima do limiar, o bônus de pico é concedido uma única vez por segmento.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

3.5.3. Segmentação por fronteira de assunto

Nada no sinal indica onde um assunto termina e outro começa, e cortar por tempo fixo produz clipes que começam no meio de um raciocínio e terminam antes da conclusão. Além disso, há blocos tematicamente coesos que duram vários minutos e não cabem no formato de saída. O modelo all-MiniLM-L6-v2 (seção 2.5) converte cada segmento em um vetor; a medida de similaridade, o limiar de fronteira e a subdivisão dos blocos longos são deste trabalho.

A similaridade de cosseno entre segmentos consecutivos mede a continuidade do assunto, e a fronteira é marcada quando fica abaixo de 0,65, valor calibrado entre dois extremos: limiares mais altos fragmentam o discurso em blocos curtos demais, e mais baixos o reduzem a poucos blocos longos.

Quando um bloco excede o limite superior da faixa, é percorrido por uma janela desse tamanho com avanço de 70%, e a sobreposição de 30% entre candidatos vizinhos oferece ao ranqueamento mais de uma delimitação para o mesmo trecho. As janelas, porém, são posicionadas por tempo, sem alinhamento às fronteiras de fala, de modo que início e fim podem cair no meio de uma frase.

A mesma medida, tomada como valor contínuo — a diferença entre um e a similaridade —, compõe o bônus de variação semântica do Viral Score. Registra-se como limitação que o all-MiniLM-L6-v2 foi treinado predominantemente em inglês, e seu uso em português apoia-se na transferência parcial da tokenização em subpalavras.

3.5.4. Pontuação dos segmentos: o algoritmo Viral Score

As etapas anteriores entregam sinais de naturezas incompatíveis — texto com marcações temporais, uma lista de instantes e um valor entre zero e um —, e nenhum deles responde isoladamente qual trecho merece virar clipe. É preciso reduzi-los a um único número por segmento. Treinar um modelo de regressão exigiria dados anotados de clipes em português, inexistentes (seção 2.3); adotou-se, por isso, uma função aditiva de pesos fixos, que abre mão da garantia de otimalidade em troca de um critério que pode ser lido, criticado e ajustado linha a linha. É essa função que se denomina Viral Score.

A pontuação tem duas parcelas. A parcela intrínseca avalia o segmento isoladamente, pelo texto e pela duração; a parcela contextual depende da comparação com o segmento anterior e dos picos de energia. A contextual só é calculada no caminho alternativo de geração de candidatos (seção 3.5.5), enquanto o ranqueamento final usa apenas a intrínseca. O Quadro 1 reúne os dez sinais: os cinco primeiros formam a parcela intrínseca, e os cinco últimos, a contextual.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Quadro 1. Componentes do algoritmo Viral Score

Sinal	Peso	Fundamento
Pontuação expressiva (! e ?)	2 por ocorrência	Indicam entonação não neutra.
Padrões de abertura de narrativa	6 por ocorrência	Expressões como “ninguém te conta” e “presta atenção” sinalizam início de trecho autocontido.
Padrões de forte ênfase	4 por ocorrência	Expressões como “eu descobri” e “isso mudou” indicam revelação ou virada narrativa.
Vocabulário de alto interesse	3 por ocorrência	Termos como “segredo”, “erro” e “estratégia”, recorrentes em conteúdos de maior alcance.
Velocidade da fala	2 por palavra/segundo	Aceleração acompanha momentos intensos; limitada a 4 palavras/s.
Variação de ritmo	4 (limiar 1,5)	Mudança brusca de velocidade em relação ao segmento anterior.
Pausa precedente	5 (limiar 0,6 s)	Pausa longa antes do segmento sugere início de nova unidade de sentido.
Encerramento pontuado	2	Segmento terminado em ponto ou exclamação tende a constituir frase completa.
Variação semântica	5 × variação	Proporcional à mudança de assunto em relação ao segmento anterior.
Pico de energia sonora	8	Concedido uma única vez se o segmento contém um pico.

Fonte: o autor, 2026.

Os pesos foram fixados pelo autor, e não por otimização, segundo três critérios: ordinal, pois importa a hierarquia entre os sinais de um mesmo vídeo; de escala, entre dois e oito, para que nenhum sinal isolado determine sozinho a posição de um segmento; e empírico, comparando os clipes produzidos antes e depois de cada alteração. O peso do pico de energia, por exemplo, subiu de cinco para oito ao se verificar que reações audíveis coincidiam com os melhores trechos, mas raramente os levavam ao topo.

Exclamações e interrogações são indício indireto de prosódia, pois o transcritor as insere a partir da entonação. As listas de expressões seguem uma hierarquia de especificidade — abertura de narrativa, forte ênfase e vocabulário de alto interesse, este o mais fraco, por operar sobre palavras isoladas. A velocidade da fala é limitada a quatro palavras por segundo, pois segmentos muito curtos com marcação imprecisa produziam velocidades desproporcionais que dominavam a pontuação.

Como exemplo, considere um segmento de 412,8 s a 419,6 s (6,8 s) com o texto “Olha isso: eu descobri o erro que quase todo mundo comete!”, de onze palavras. O segmento anterior

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

encerra em 411,9 s, com velocidade de 3,4 palavras por segundo; a variação semântica é de 0,62; e há um pico de energia em 414,2 s. O Quadro 2 detalha a composição da pontuação.

Quadro 2. Exemplo de cálculo do Viral Score para um segmento

Sinal	Verificação no segmento	Parcela
Pontuação expressiva	um "!" × 2 pontos	2,00
Padrões de forte ênfase	"eu descobri" × 4 pontos	4,00
Padrões de abertura de narrativa	"olha isso", "eu descobri" e "erro que" × 6 pontos	18,00
Vocabulário de alto interesse	"erro" × 3 pontos	3,00
Velocidade da fala	11 ÷ 6,8 = 1,62 pal./s × 2 pontos	3,24
Variação de ritmo	|1,62 − 3,40| = 1,78 > 1,5	4,00
Pausa precedente	412,8 − 411,9 = 0,9 s > 0,6 s	5,00
Encerramento pontuado	o segmento termina em "!"	2,00
Variação semântica	0,62 × 5 pontos	3,10
Pico de energia sonora	pico em 414,2 s dentro do intervalo	8,00
Total		**52,34**

Fonte: o autor, 2026.

O valor 52,34 só tem significado em comparação com os demais segmentos do mesmo vídeo, razão pela qual a etapa seguinte ordena as pontuações em vez de aplicar um limiar fixo. Na ordenação final, que usa apenas a parcela intrínseca, o valor comparado é de 30,24 pontos (as cinco primeiras linhas do Quadro 2), dos quais 27 são lexicais: na ausência de dados rotulados, o texto é o sinal mais barato de extrair e o mais fácil de auditar.

3.5.5. Ranqueamento e supressão de sobreposições

Os candidatos se sobrepõem por construção, já que a janela da seção 3.5.3 avança apenas 70% do próprio tamanho; sem filtro, os melhores clipes seriam versões do mesmo momento. Antes da ordenação, escolhe-se a origem dos candidatos: se a segmentação por assunto produz mais de cinco blocos, eles são adotados diretamente; caso contrário, típico de conteúdo monotemático, os segmentos são agregados guiados pelos picos de energia e pelas pausas, até atingir o limite superior da faixa ou uma pausa longa. É nesse caminho alternativo que a parcela contextual é calculada.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Cada candidato recebe como pontuação final a soma da parcela intrínseca dos segmentos que abrange, e a lista é ordenada de forma decrescente; assim, os cinco sinais contextuais não influem na ordem final dos clipes na versão avaliada.

Para eliminar a redundância, aplica-se supressão de não máximos sobre a lista ordenada: percorrendo-a do melhor para o pior, um candidato é descartado se a sua interseção sobre união (IoU) — o tempo compartilhado por dois trechos dividido pelo tempo que ocupam juntos — com algum candidato já aprovado exceder 50%. Assim, candidatos de 100 s a 160 s e de 130 s a 190 s (IoU de 0,33) são ambos mantidos, enquanto candidatos de 100 s a 160 s e de 110 s a 170 s (IoU de 0,71) representam o mesmo trecho deslocado, e o de menor pontuação é descartado.

Essa estratégia substituiu um agrupamento por proximidade temporal que, como os candidatos da janela deslizante são contíguos, colapsava o vídeo inteiro em um único grupo e produzia um só clipe, qualquer que fosse a duração do material. A IoU mede a sobreposição efetiva, e não a distância, preservando candidatos vizinhos que cobrem conteúdos distintos.

3.5.6. Ajuste das fronteiras de duração

Um trecho tematicamente coeso raramente mede o que a faixa de duração permite, e truncá-lo no limite superior corta frases ao meio e faz com que quase todos os clipes tenham a mesma duração. O tratamento adotado substitui a truncagem por uma busca: quando o clipe excede o limite superior, procura-se o encerramento de segmento mais tardio que ainda caiba na janela permitida; sem encerramento válido, recai-se sobre a truncagem no teto, e o início do clipe não é ajustado.

Nos candidatos em que atua, os cortes deixaram de ocorrer no meio de frases e as durações passaram a distribuir-se ao longo da faixa. Clipes que permanecem abaixo do limite inferior são descartados, e não estendidos, para não incorporar fala que o critério de seleção não considerou relevante.

3.5.7. Reenquadramento automático

O material é horizontal e a saída é vertical, e o corte centralizado só funciona quando o locutor ocupa o meio da cena, situação pouco frequente em entrevistas e *podcasts*. O detector do OpenCV (seção 2.7) responde, para um quadro isolado, onde há rostos e com que confiança; qual rosto adotar, quantos quadros examinar e o que fazer na ausência de detecção são decisões deste trabalho.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Descartam-se as detecções com confiança inferior a 0,5 e adota-se o centro horizontal do rosto de maior área, pois a média de todos os rostos, com duas pessoas lado a lado, converge para o centro da cena, frequentemente vazio.

Amostrar um quadro a cada quinze, com teto de trinta amostras válidas, limita o custo a trinta inferências por clipe, contra 1.800 se todos os quadros de um clipe de um minuto fossem examinados. A posição final é a média das detecções, normalizada pela largura do quadro; sem detecção válida, recai-se sobre o corte centralizado. Trechos que alternam entre um e dois interlocutores ainda podem gerar uma média que não centraliza nenhum deles (seção 4.4).

3.5.8. Geração das legendas

A legenda do formato vertical é lida com o som desligado e acompanha a fala palavra a palavra. O formato Advanced SubStation Alpha foi escolhido por suportar posicionamento absoluto, contorno de caractere e temporização por palavra com marcadores de karaokê; toda a etapa é de autoria própria.

Cada bloco tem no máximo três palavras e é fechado quando a pausa entre palavras consecutivas excede 0,4 s, o que mantém a legenda legível em tela pequena. A temporização recebe ajustes empíricos: o início do bloco é antecipado em 0,15 s, compensando o tempo de leitura; o fim é recuado em 0,2 s, evitando que o bloco persista sobre o silêncio; e impõe-se duração mínima de 1,5 s, sem sobreposição entre blocos. Na marcação de karaokê, o destaque avança ligeiramente à frente da fala.

As palavras são convertidas para caixa alta e despojadas de pontuação, seguindo a convenção visual do formato vertical; a cor escolhida exige permutar os canais vermelho e azul, pois o formato os representa em ordem inversa à da notação hexadecimal da *web*.

3.5.9. Renderização

Corte, reenquadramento, redimensionamento e queima das legendas aplicados em invocações sucessivas exigiriam recodificar o vídeo a cada etapa, acumulando perda de qualidade e tempo. O FFmpeg executa a codificação, e a cadeia de filtros que faz as quatro operações em uma única passagem é construída por código deste trabalho conforme o formato solicitado: formatos mais estreitos que altos recortam a largura seguindo a posição do rosto, e o horizontal recorta a altura de forma centralizada.

A cadeia de filtros recebeu duas correções resultantes de falhas em execução: o recorte é limitado ao tamanho efetivo do quadro, e a vírgula no interior da função de mínimo é escapada, pois o FFmpeg a interpreta como separador entre filtros. O progresso é lido da saída estruturada

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

do FFmpeg, e a fusão dos canais de saída e de erro eliminou um travamento causado pelo preenchimento do *buffer* do canal de erro sem consumidor.

3.5.10. Geração de miniaturas

Por fim, o FFmpeg extrai do clipe renderizado o quadro do segundo inicial, com 360 pixels de largura, como miniatura para a grade da interface; uma falha nessa etapa apenas registra um aviso.

4. RESULTADOS E DISCUSSÃO

4.1. Funcionalidades entregues

A plataforma implementa o fluxo proposto nos objetivos específicos: os quatro extratores de características — faster-whisper (*medium*), Sentence-BERT (all-MiniLM-L6-v2), librosa e o detector facial do OpenCV — são alinhados pela marcação temporal por palavra; o Viral Score combina dez sinais; a seleção usa ranqueamento e supressão de não máximos com limiar de 0,5; o reenquadramento atende aos formatos 9:16 e 1:1, com corte centralizado no 16:9; as legendas usam o formato ASS, com dez estilos e três posições; e a plataforma *web* integra interface de usuário, interface de programação, fila assíncrona, autenticação por *token* e limitação de requisições. A avaliação de desempenho e de qualidade é apresentada nas seções 4.3 e 4.4.

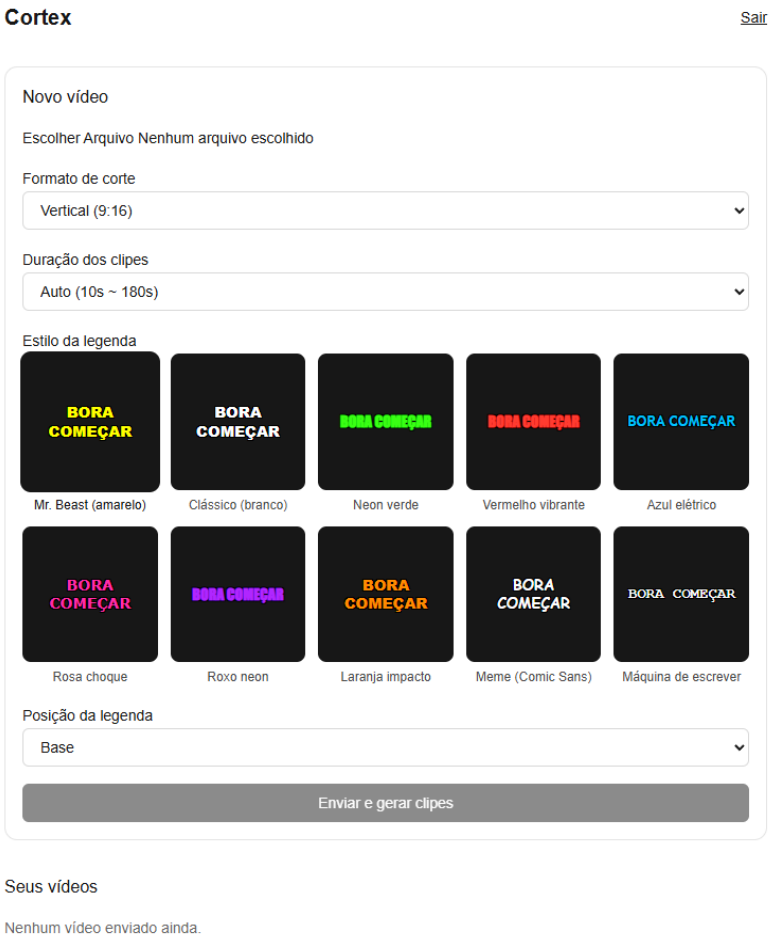
Além do escopo previsto, foram implementados a geração de miniaturas para pré-visualização e o histórico persistente de processamentos por usuário.

4.2. Artefato de validação em funcionamento

A Figura 2 apresenta o formulário de envio, no qual o usuário seleciona o vídeo e define o formato de corte, a faixa de duração e o estilo e a posição da legenda.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 2. Formulário de envio e parametrização



Fonte: o autor, 2026.

Concluído o processamento, o painel exibe os clipes selecionados pelo Viral Score em uma grade de miniaturas; cada miniatura, ao ser acionada, é substituída pelo reprodutor do clipe. Nos formatos vertical e quadrado, a posição do recorte acompanha o rosto do locutor identificado pelo detector facial, com a adequação avaliada na seção 4.4. Os quadros dos clipes gerados não são reproduzidos neste artigo porque os vídeos de teste são de terceiros e retratam pessoas identificáveis.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

4.3. Desempenho do processamento

Foram processados vídeos de diferentes durações, medindo-se o tempo consumido por cada etapa do fluxo (Tabela 1).

Tabela 1. Tempo de processamento por etapa

Duração do vídeo	Modo	Transcrição	Detecção	Renderização	Total	Clipes gerados
10 min	GPU	0 min 21 s	0 min 04 s	1 min 12 s	1 min 37 s	20
10 min	CPU	5 min 17 s	0 min 03 s	1 min 16 s	6 min 36 s	20
30 min	GPU	2 min 32 s	0 min 04 s	1 min 40 s	4 min 16 s	20
30 min	CPU	30 min 06 s	0 min 10 s	2 min 00 s	32 min 16 s	20
60 min	GPU	6 min 42 s	0 min 15 s	2 min 34 s	9 min 32 s	20

Fonte: o autor, 2026.

Os tempos foram extraídos dos registros do executor Celery, que marca o início e o término de cada etapa. No vídeo de 10 minutos, a renderização domina (cerca de 74% do total), pois o corte e a legendagem dos vinte clipes têm custo praticamente fixo; à medida que a duração aumenta, a transcrição, cujo custo cresce linearmente com o áudio, passa a dominar, com cerca de 59% do total no vídeo de 30 minutos e 70% no de 60. Todos os vídeos geraram vinte clipes, limite máximo por processamento na implementação atual.

O ensaio foi executado em máquina com GPU NVIDIA GeForce RTX 5070 Ti (16 GB de VRAM), 32 GB de RAM e Windows 11. Para isolar o efeito da GPU, os vídeos de 10 e 30 minutos foram reprocessados com execução forçada em CPU. A transcrição foi a etapa mais sensível: 21 s contra 5 min 17 s no vídeo de 10 minutos (cerca de 15 vezes) e 2 min 32 s contra 30 min 06 s no de 30 minutos (cerca de 12 vezes). A detecção não apresentou ganho consistente, o que sugere que seu tempo é dominado pelo carregamento do modelo de *embeddings*, e não pelo volume de cálculo; a renderização, dependente do FFmpeg, variou menos de 20%. No total, a GPU reduziu o tempo de processamento de 4 a 7,5 vezes, ganho concentrado quase inteiramente na transcrição.

4.4. Avaliação qualitativa dos clipes

A avaliação qualitativa examinou quatro critérios: completude do trecho, sincronia entre legenda e fala, adequação do enquadramento e pertinência da seleção. A inspeção foi feita manualmente pelo autor sobre os quarenta clipes gerados a partir dos vídeos de 10 e 30 minutos;

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

um clipe foi considerado adequado quando não apresentou o problema descrito na Tabela 2. Por envolver um único avaliador, sem avaliadores independentes nem medida de concordância, essa avaliação está sujeita a viés e tem caráter exploratório: os resultados indicam tendências, e não estimativas generalizáveis da qualidade dos clipes.

Tabela 2. Avaliação qualitativa dos clipes gerados

Critério	Clipes adequados	Observações
Completude do trecho	40/40	Todos os trechos têm início e fim sobre um mesmo assunto.
Sincronia da legenda	36/40	Atraso da legenda em trechos de fala acelerada.
Adequação do enquadramento	32/40	A posição de corte é fixa por clipe (média das amostras); quando o locutor muda de lugar no quadro, por exemplo após um corte de câmera, a média não o centraliza em nenhum momento.
Pertinência da seleção	38/40	Os clipes de menor pontuação, ao final da ordenação, foram os de menor relevância percebida.

Fonte: o autor, 2026.

4.5. Comparação com a edição manual

Para dimensionar o ganho de produtividade, três clipes do vídeo de 60 minutos foram editados manualmente pelo autor — seleção do trecho, corte, reenquadramento e legendagem em um editor convencional —, com tempo médio de 11,5 minutos por clipe. Extrapolado para os vinte clipes gerados automaticamente, o trabalho manual demandaria cerca de 230 minutos (3 h 50 min), contra 9 min 32 s do processamento automático, o que sugere redução superior a 95%. O valor deve ser lido como indicativo, e não como medida de desempenho: a amostra de três clipes, editados por uma única pessoa, não permite inferência estatística; o estilo de edição, a familiaridade com a ferramenta e a complexidade de cada trecho podem alterar substancialmente o tempo real; e a comparação considera apenas o tempo de produção, sem confrontar a qualidade dos clipes manuais e automáticos.

4.6. Síntese das dificuldades enfrentadas

Boa parte das decisões da seção 3 foi tomada em resposta a falhas observadas em execução, e não antecipada em projeto. Entre os oito problemas de maior impacto, três interrompiam o processamento — o travamento na renderização, a cadeia de filtros inválida e o

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

erro de dimensão no recorte —, e os demais degradavam a qualidade sem impedir a entrega: a geração de um único clipe por vídeo, as durações uniformes com cortes no meio de frases, o enquadramento vazio com dois interlocutores, a pontuação dominada por segmentos curtos e a legenda renderizada com fonte divergente. As causas e as soluções de cada um foram descritas na seção 3.5.

Seis dos oito problemas só se manifestaram com o fluxo executado de ponta a ponta sobre vídeos reais, e nenhum seria detectado por inspeção do código, o que justifica a validação sobre vídeos completos.

4.7. Limitações identificadas

As limitações a seguir delimitam o alcance dos resultados obtidos. Os componentes lexicais da pontuação dependem de listas de expressões em português, e os pesos foram definidos empiricamente, sem otimização sobre dados anotados; o ranqueamento final usa apenas a parcela intrínseca do Viral Score; blocos mais longos que a faixa são subdivididos por janelas de tempo fixo, que podem começar ou terminar no meio de uma frase; pesos e limiares estão fixados no código-fonte, sem exposição na interface; o rastreamento facial atua só no eixo horizontal, com posição fixa por clipe; não há análise do conteúdo visual além da detecção de rostos; e a qualidade de todo o fluxo depende da transcrição, degradada por ruído intenso, vozes sobrepostas ou música de fundo. Quanto à validação, a avaliação qualitativa foi conduzida pelo próprio autor, sem múltiplos avaliadores independentes nem medida de concordância, o que a sujeita a viés; a estimativa de economia de tempo apoia-se na edição manual de apenas três clipes, por um único editor, e não compara a qualidade dos clipes manuais e automáticos; e o ensaio abrangeu três vídeos, de 10, 30 e 60 minutos, todos com predominância de fala em português, sem avaliar outros gêneros de conteúdo.

5. CONSIDERAÇÕES FINAIS

Converter um vídeo longo em dezenas de clipes curtos consome horas de trabalho, e as ferramentas que automatizam a tarefa não revelam o critério que aplicam. Este trabalho construiu e avaliou o Cortex, plataforma *web* que executa esse fluxo de ponta a ponta com critério de decisão inteiramente documentado: quatro modelos pré-treinados de terceiros — reconhecimento de fala, representação semântica de sentenças, análise de energia sonora e detecção facial — atuam como extratores de características, e sobre suas saídas opera o Viral Score, apresentado com os pesos, o código e um exemplo numérico de cálculo.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Quanto à questão de pesquisa, na versão avaliada a resposta tem duas camadas: a mudança semântica entre segmentos consecutivos delimita os candidatos, e uma soma ponderada de sinais lexicais e de ritmo, acumulada por candidato, os ordena, seguida da supressão dos candidatos cuja interseção sobre união com um candidato melhor ultrapasse 50%. Todo o encadeamento usa bibliotecas de código aberto e dispensa material anotado. Os sinais contextuais do Viral Score foram implementados, mas ainda não participam da ordenação final (seção 3.5.5).

A hipótese recebe respostas distintas em suas duas partes. A delimitação pela estrutura do discurso é sustentada parcialmente: a busca pelo encerramento natural da fala eliminou os cortes no meio de frases onde atua, mas blocos longos ainda são subdivididos por janelas de tempo fixo. A viabilidade da combinação de sinais heterogêneos sem treinamento nem anotação está demonstrada para os vídeos avaliados, sem que se afirme superioridade sobre o uso isolado de cada sinal, o que exigiria um estudo de ablação. Observou-se, ainda, que o critério de combinação determina a qualidade da saída: trocar o agrupamento por proximidade temporal pela supressão por interseção sobre união levou de um único clipe por vídeo a um conjunto de trechos distintos, sem alterar nenhum sinal.

Os sete objetivos específicos foram cumpridos, conforme a seção 4.1. Com GPU, um vídeo de sessenta minutos foi processado em 9 min 32 s, gerando vinte clipes legendados e reenquadrados. Na avaliação exploratória conduzida pelo autor, os quarenta clipes avaliados apresentaram início e encerramento sobre um assunto determinado, e 38 foram considerados pertinentes; a sincronia da legenda falhou em quatro, e o enquadramento foi o critério de menor adequação, com 32. A comparação com a edição manual sugere redução de tempo superior a 95%, valor indicativo em razão da amostra de três clipes (seção 4.5).

A principal contribuição não está na superação do estado da arte em sumarização automática, o que exigiria dados anotados indisponíveis em português, mas na documentação explícita do critério de decisão — sinais, pesos, razões e código —, demonstrando que um critério equivalente ao das ferramentas comerciais pode ser construído com recursos acessíveis e exposto à crítica e ao ajuste. São vantagens a transparência, a ausência de custo por processamento e a possibilidade de ajustar os pesos a domínios específicos; são desvantagens a dependência do português, os pesos empíricos, a ausência de análise visual e o caráter exploratório da validação (seção 4.7).

Como trabalhos futuros, recomendam-se: unificar as parcelas do Viral Score no ranqueamento final; alinhar as janelas de subdivisão às fronteiras de segmento; expor os pesos na interface por perfis de domínio; adotar um modelo de sentenças multilíngue ou um modelo de

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

linguagem de grande porte no lugar da heurística lexical; realizar um estudo de ablação com otimização dos pesos sobre dados anotados; estender o rastreamento facial ao eixo vertical; incorporar a detecção de mudanças de cena; e, sobretudo, repetir a avaliação com múltiplos avaliadores independentes e medida de concordância, comparando a edição manual e a automática sobre amostra maior, com editores distintos.

REFERÊNCIAS

CAPCUT. **CapCut**: editor de vídeo online. [*S. l.*], 2026. Disponível em: https://www.capcut.com. Acesso em: 12 abr. 2026.

CELERY PROJECT. **Celery**: distributed task queue documentation. [*S. l.*], 2026. Disponível em: https://docs.celeryq.dev. Acesso em: 20 ago. 2026.

DESCRIPT. **Descript**: all-in-one video and podcast editing. [*S. l.*], 2026. Disponível em: https://www.descript.com. Acesso em: 12 abr. 2026.

FFMPEG PROJECT. **FFmpeg documentation**. [*S. l.*], 2026. Disponível em: https://ffmpeg.org/documentation.html. Acesso em: 20 ago. 2026.

JURAFSKY, D.; MARTIN, J. H. **Speech and language processing**: an introduction to natural language processing, computational linguistics, and speech recognition. 3. ed. preliminar. [*S. l.*: *s. n.*], 2023.

LIU, W. *et al*. SSD: single shot multibox detector. In: EUROPEAN CONFERENCE ON COMPUTER VISION, 14., 2016, Amsterdam. **Proceedings** [...]. Cham: Springer, 2016. p. 21-37.

MCFEE, B. *et al*. librosa: audio and music signal analysis in Python. In: PYTHON IN SCIENCE CONFERENCE, 14., 2015, Austin. **Proceedings** [...]. Austin: SciPy, 2015. p. 18-25.

OPENCV. **OpenCV**: open source computer vision library documentation. [*S. l.*], 2026. Disponível em: https://docs.opencv.org. Acesso em: 20 ago. 2026.

OPUS CLIP. **Opus Clip**: AI video clipping tool. [*S. l.*], 2026. Disponível em: https://www.opus.pro. Acesso em: 20 ago. 2026.

PANDA, R.; ROY-CHOWDHURY, A. K. Collaborative summarization of topic-related videos. In: IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION, 2017, Honolulu. **Proceedings** [...]. Honolulu: IEEE, 2017. p. 7083-7092.

PYTHON *SOFTWARE* FOUNDATION. **The Python language reference**. [*S. l.*], 2026. Disponível em: https://docs.python.org. Acesso em: 12 abr. 2026.

RABINER, L. R.; SCHAFER, R. W. **Theory and applications of digital speech processing**. Upper Saddle River: Prentice Hall, 2011.

RADFORD, A. *et al*. **Robust speech recognition via large-scale weak supervision**. [*S. l.*]: arXiv, 2022. Disponível em: https://arxiv.org/abs/2212.04356. Acesso em: 12 abr. 2026.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

RAMÍREZ, S. **FastAPI documentation**. [*S. l.*], 2026. Disponível em: https://fastapi.tiangolo.com. Acesso em: 20 ago. 2026.

REIMERS, N.; GUREVYCH, I. Sentence-BERT: sentence embeddings using siamese BERT-networks. In: CONFERENCE ON EMPIRICAL METHODS IN NATURAL LANGUAGE PROCESSING AND INTERNATIONAL JOINT CONFERENCE ON NATURAL LANGUAGE PROCESSING, 2019, Hong Kong. **Proceedings** [...]. Hong Kong: Association for Computational Linguistics, 2019. p. 3982-3992.

RUSSELL, S.; NORVIG, P. **Artificial intelligence**: a modern approach. 4. ed. Hoboken: Pearson, 2021.

SYSTRAN. **faster-whisper**: reimplementação do Whisper com CTranslate2. [*S. l.*], 2026. Disponível em: https://github.com/SYSTRAN/faster-whisper. Acesso em: 20 ago. 2026.

VERCEL. **Next.js documentation**. [*S. l.*], 2026. Disponível em: https://nextjs.org/docs. Acesso em: 20 ago. 2026.

ZHANG, K. *et al*. Video summarization with long short-term memory. In: EUROPEAN CONFERENCE ON COMPUTER VISION, 14., 2016, Amsterdam. **Proceedings** [...]. Cham: Springer, 2016. p. 766-782.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.