Abordagem para Segmentação Temporal de Vídeos Educacionais

Eduardo Rocha Soares[1], **Eduardo Barrére**[1], **Jairo Francisco de Souza**[1]

[1] LApIC Research Group
Programa de Pós-Graduação em Ciência da Computação – (UFJF)
36.360-900 – Juiz de Fora – MG – Brasil

{eduardosoares, eduardo.barrere,jairo.souza}@ice.ufjf.br

Abstract. *This paper discusses the main aspects related to segmenting educational videos approaches. It presents the EasyTopic, a novel approach that allows the use of several approaches for the process of temporal segmentation of video classes into topics (semantic segments), in a configurable and generalist way. Aiming to prove the functioning of the proposal, it presents its use in the implementation of a solution to the segmentation problem in educational video topics, with results and considerations on the use of the framework.*

Resumo. *O artigo discute os principais aspectos relacionados a abordagens para segmentação de vídeos educacionais. Apresenta uma abordagem entitulada como EasyTopic que permite o uso de diversas técnicas para o processo de segmentação temporal de videoaulas em tópicos (segmentos semânticos), de forma configurável e generalista. Visando comprovar o funcionamento da abordagem, o artigo apresenta o seu uso na implementação de uma solução para o problema de segmentação em tópicos de vídeos educacionais, com resultados e considerações sobre o uso da proposta.*

1. Introdução

Com o aumento no uso de *e-learning*, os vídeos educacionais (ou simplesmente videoaulas) têm se apresentado como uma mídia de grande eficácia enquanto conteúdo educacional. Estes vídeos oferecem várias vantagens para os discentes, por exemplo, a possibilidade de revisar o conteúdo ministrado nas aulas regulares ou preencher a lacuna devido à sua ausência. Além disso, as videoaulas permitem que sigam seu próprio ritmo de aprendizado, pausando e ”navegando” no vídeo [Ronchetti 2010]. Apesar da alta disponibilidade de videoaulas na internet, muitos discentes ainda têm problemas para localizar uma videoaula apropriada para seus estudos, pois diversos conteúdos irrelevantes são retornados em suas pesquisas [Mitra and Srivastava 2020].

Também é comum que o discente tenha dificuldade em acessar um conteúdo específico desejado, mesmo quando finalmente encontra uma videoaula que supostamente aborda o tema, pois infelizmente as videoaulas geralmente não fornecem *links* que permitam acessar rapidamente a um tópico específico. Portanto, para encontrar o início de um tópico em uma videoaula de longa duração, ele deve assistir ao vídeo inteiro ou tentar avançar/retroceder o vídeo até encontrar o que deseja. Esse processo é demorado e contribui negativamente para a sua experiência de interação [Yang and Meinel 2014].

DOI: 10.5753/cbie.sbie.2020.972

Um repositório específico pode fornecer mecanismos para resolver tal desafio. Por exemplo, é necessário permitir que o discente acesse rapidamente tópicos específicos de uma videoaula e também navegue de maneira não linear entre esses tópicos. Este tipo de navegabilidade instantânea e não linear entre os tópicos da videoaula é ideal para o aprendizado e pode melhorar significativamente a experiência do discente em sistemas de *e-learning* [Pavel et al. 2014]. A maneira mais comum de fazer isso é segmentar a videoaula em tópicos, criando marcações temporais para cada unidade menor, onde cada uma delas represente um assunto específico abordado no vídeo (tópico). Embora a segmentação de tópicos criada pelo homem seja a mais precisa, ela é muito demorada e difícil de fazer em grandes repositórios de vídeos [Lin et al. 2005]. Portanto, é importante automatizar essa tarefa para torná-la viável na prática. A segmentação temporal automática de videoaulas tem sido um desafio para as áreas de recuperação de informações e multimídia devido à natureza do problema que envolve processamento de conteúdo e entendimento semântico.

Embora existam muitas abordagens na literatura para automatizar essa tarefa, a maioria delas se baseia em recursos de materiais complementares, como slides, livros didáticos e legendas; tornando essas abordagens muito dependentes da disponibilidade de recursos específicos, dificultando assim o uso dessas abordagens em boa parte das videoaulas disponíveis na internet. Visando minimizar essas limitações, desenvolvemos uma solução baseada no discurso do professor para a segmentação temporal em tópicos de qualquer videoaula [Soares and Barrére 2019] que possua ou não recursos adicionais. Para dar suporte, flexibilidade e escalabilidade a essa solução, foi desenvolvida uma solução que permite incorporar módulos para obtenção de diversas características prosódicas e semânticas do vídeo e também algoritmos de segmentação diversos. Esses módulos podem ser executados de forma separada ou dentro de um fluxo pré-determinado (técnica ou algoritmo existente). Desta forma, a solução possibilita auxiliar pesquisadores e especialistas na criação/avaliação de soluções para segmentação de vídeos e também extração de informações (prosódicas ou semânticas) dos vídeos.

2. Trabalhos Relacionados

Em relação a *frameworks* para segmentação temporal de vidoaulas, algumas características principais são usadas para classificá-los: o domínio das videoaulas em que eles se aplicam, a natureza dos recursos extraídos e a maneira como esses recursos são combinados para gerar a segmentação de tópicos. Nesta seção, as principais abordagens do estado-da-arte são discutidas e classificadas de acordo com suas características principais.

O problema da segmentação temporal das videoaulas pode ter definições diferentes, dependendo do autor. Essa dificuldade vem principalmente do conceito subjetivo de tópico. Uma das definições mais difundidas e assumida neste trabalho é que um tópico consiste em uma unidade lógica e semanticamente significativa da videoaula, sendo contíguo no tempo [Tuna et al. 2015, Galanopoulos and Mezaris 2019]. Assumindo esta definição podemos representar um tópico por seus limites de tempo, ou seja, considerando um conjunto de tópicos T de uma videoaula V, podemos representar cada tópico t_iT como o intervalo fechado $[init_i, end_i]$, em que $init_i$ e end_i representam respectivamente o horário inicial e final de t_i na videoaula, sendo $end_i > init_i$ e $init_{i+1} > end_i$. Portanto, a segmentação temporal pode ser simplificada da seguinte forma: dada uma videoaula V como entrada, encontrar automaticamente os limites de tempo $[init_i, end_i]$ de todos os tópicos t_i de V, onde $i = 1, 2, 3, 4, ..., N$ e N é o número de tópicos em V.

De forma resumida, é possível destacar as abordagens conforme o recurso utilizado para a segmentação, como legendas, Optical Character Recognition (OCR), imagem ou fala. Em [Galanopoulos and Mezaris 2019], os autores utilizam o agrupamento de palavras para calcular a similaridade semântica entre janelas de texto das legendas e, assim, identificam possíveis pontos de transição entre tópicos. São utilizadas legendas geradas via transcrição automática, porém sem analisar o impacto dos erros dessa transcrição. Por sua vez, em [Tuna et al. 2015] os autores realizaram uma investigação sobre o impacto no uso de fontes textuais para a segmentação de videoaulas e concluíram que o uso do OCR (Reconhecimento Óptico de Caracteres) nos *slides* fornece melhores resultados que o ASR (Automatic Speech Recognition), em grande parte porque o ASR tem muitos erros envolvidos. No entanto, o ASR superou o OCR como a melhor fonte de dados para a segmentação de videoaulas quando os autores corrigiram manualmente as transcrições automáticas. Em [Davila and Zanibbi 2017], os autores focaram em videoaulas onde o texto escrito no quadro é apresentado no vídeo. Os quadros-chave são extraídos e binarizados. Em seguida, esses quadros são segmentados para separar o plano de fundo do conteúdo manuscrito. Por fim, a segmentação de tópicos é feita para minimizar o conflito entre regiões de conteúdo no quadro branco. As regiões de conteúdo estão em conflito se ocuparem o mesmo espaço no quadro branco em intervalos de tempo diferentes. Portanto, eles não podem estar no mesmo tópico e precisam ser segmentados.

No trabalho de [Che et al. 2018], as frases podem ser obtidas a partir de legendas ou ASR. Em seguida, são extraídos recursos prosódicos de cada frase, como tom, volume, taxa de pausa e duração, para obter pontos de destaque na videoaula. Como cada sentença tem um registro de tempo associado, uma segmentação temporal pode ser obtida. Por fim, propostas que utilizam diferentes fontes de informação para executar a tarefa de segmentação de tópicos são comuns na literatura. Em [Shah et al. 2014] os autores apresentam o sistema ATLAS como solução para o Grande Desafio ACM Multimedia 2014 sobre segmentação e anotação temporais automáticas. Essa abordagem baseia-se na fusão de indicadores de transição visual e textual. Para obter os marcadores de transição visual, dois modelos Support Vector Machines (SVM) foram treinados. Outro exemplo é o trabalho de [Mahapatra et al. 2018], cujo objetivo final é a geração de um índice para as videoaulas, utiizando a tecnologia OCR e os recursos de fala para encontrar a hora em que o professor começa a falar sobre os tópicos que aparecem na apresentação de slides.

Os trabalhos apresentados trazem uma característica em comum, ter um bom desempenho apenas para um subgrupo específico de videoaulas, pois acabam sendo dependentes de muitos recursos e condições pré-existentes para o seu bom funcionamento. A principal contribuição deste trabalho é propor uma solução, tanto do ponto de vista teórico quanto prático, que seja versátil o suficiente para ser empregada em grande parte das videoaulas presentes da *web*, pois pode ser configurada de acordo com o cenário, suportando a ativação ou desativação de etapas e recursos de acordo com a necessidade e/ou disponibilidade dos mesmos.

3. EasyTopic

De forma geral, as soluções da literatura convergem para um *framework* conceitual básico que realiza uma pequena variação do *pipeline* apresentado na Figura 1 com: (i) **Vídeo e demais recursos**: diversos *frameworks* realizam ações de conversão do vídeo (resolução, codificação etc.) e demais recursos (legenda, slides etc.) para viabilizar o processamento

nas demais etapas; (ii) **Extratores de carcterísticas**: obter informações do frames do vídeo, do áudio, dos slides, das legendas etc.; (iii) **Pré-processamento**: processamento prévio das características obtidas de forma a tratar/configurar parâmetros que venham a ser utilizados no algoritmo de segmentação; (iv) **Fluxo 1 - vermelho**: inicialmente as características obtidas são fundidas, conforme pesos e algoritmos previamente determinados, e com base no resultado desta fusão é executado o algoritmo de segmentação em si; (v) **Fluxo 2 - azul**: inicialmente é aplicado um algoritmo de segmentação para cada característica considerada e posteriormente essas segmentação são fundidas em uma única solução; (vi) **Pós-processamento**: de posse da segmentação final, alguma etapa de refinamento ou mesmo geração de informações complementares pode ser utilizada; (vii) **Documentos gerados**: o resultado de todo o processo é bem variado, mas basicamente consiste em fornecer o tempo inicial de cada segmento e algumas informações associadas a ele.

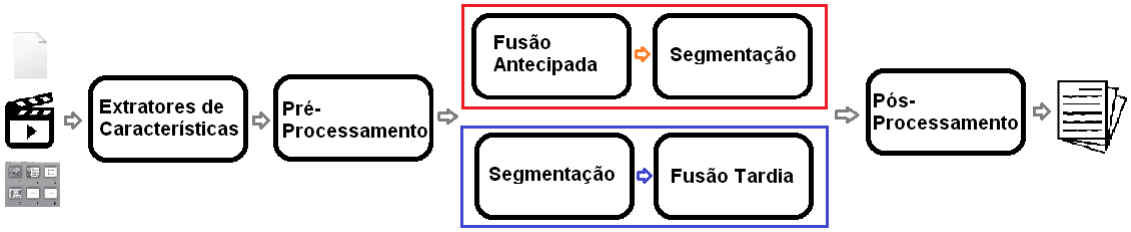

Figure 1. Esquema geral do EasyTopic para segmentação de vídeos.

O grande diferencial do **EasyTopic** em relação aos demais é que ele foi projetado a permitir grande liberdade a cerca de qual algoritmo ou técnica utilizar em cada etapa ou mesmo a realização de somente uma etapa, desde que os parâmetros de entrada para o módulo estejam corretos. Por exemplo, a etapa (ii) pode ser composta por um único módulo de extração de características ou mesmo um conjunto de módulos para obter um conjunto de diferentes características. Na nossa implementação, cada módulo é representado por um *container*, facilitando assim sua instanciação e escalabilidade (várias instâncias do mesmo *container*).

3.1. Prova de conceito

Como prova de conceito, implementamos o algoritmo de segmentação baseado em fala proposto por [Soares and Barrére 2019] utilizando nossa proposta. Esse algoritmo pode ser visto como um pipeline de processamento que depende única e exclusivamente da presença da fala do professor no vídeo para seu funcionamento.

Nesse *pipeline*, a trilha de áudio da videoaula passa por uma série de processamentos para a extração de recursos semânticos e prosódicos do discurso do professor. Em seguida, esses recursos são usados para modelar a segmentação temporal de videoaulas como um problema de programação linear que é para encontrar uma solução ótima ou sub-ótima. Para uma melhor compreensão da visão geral desse algoritmo, ele pode ser dividido em 4 estágios de processamento, onde cada um deles recebe uma entrada e produz uma saída. Descrevemos brevemente cada um a seguir: (1) toda a informação necessária para este algoritmo está presente no discurso do professor. Portanto, na primeira etapa do processamento, a faixa de áudio da videoaula é extraída e fornecida para a próxima etapa;

(2) embora a trilha de áudio tenha todas as informações úteis para o algoritmo, também possui muitas informações irrelevantes, como ruído de fundo. Para aliviar esse problema, um processo chamado de remoção do silêncio é aplicado para que se obtenha trechos de áudio que contenham o máximo possível de fala em primeiro plano de forma contínua; (3) em seguida, recursos semânticos e prosódicos são extraídos dos trechos de áudio obtidos pelo estágio de remoção do silêncio; (4) por fim, é feita a fusão desses recursos para compor um modelo de programação linear que é otimizado para obter uma segmentação temporal para a videoaula.

Com esse *pipeline* de processamento, podemos obter uma segmentação temporal das videoaulas através apenas dos recursos de sua trilha de áudio. Utilizando-se do conceito da nossa solução, tornamos todos os estágios desse *pipeline* configuráveis, permitindo o uso de quaisquer modelos ou algoritmos de escolha do usuário para executá-los. A Figura 2 ilustra a visão geral desse *pipeline* de processamento.

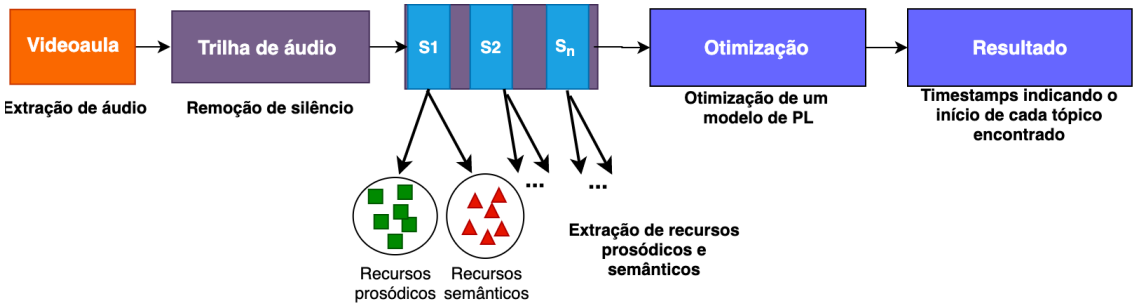

Figure 2. Visão geral do *pipeline* de processamento implementado utilizando o EasyTopic.

3.2. Implementação

Implementamos a solução proposta neste trabalho como uma arquitetura de software distribuída. Nesta arquitetura, cada etapa de processamento do algoritmo a ser incorporado é um módulo totalmente independente que funciona em um esquema de produtor / consumidor [Zhang et al. 2009]. Cada processo que executa as tarefas de um módulo é um *worker* que consome mensagens de uma fila de tarefas e, no final do processamento, insere os resultados em uma fila de saída (que pode ser uma fila de tarefas de outro módulo). O número de *workers* de cada módulo pode ser configurado e dimensionado conforme a necessidade. A principal vantagem do uso desse padrão de *design* é que podemos gerenciar mais facilmente processos que produzem e consomem dados de forma assíncrona em velocidades diferentes.

Em relação aos módulos, todos são embarcados em contêineres Docker [1] para facilitar a reprodutibilidade e a implantação de nossa proposta. Para implementar o algoritmo proposto por [Soares and Barrére 2019] foi necessária a instanciação de 7 módulos. A Figura 3 ilustra o esquema de comunicação em fila entre os módulos de arquitetura. Além disso, disponibilizamos todo o código de nossa implementação no GitHub [2] para todos os pesquisadores interessados em contribuir ou reproduzir os resultados relatados neste trabalho.

[1]https://www.docker.com/
[2]https://github.com/eduardorochasoares/easytopic

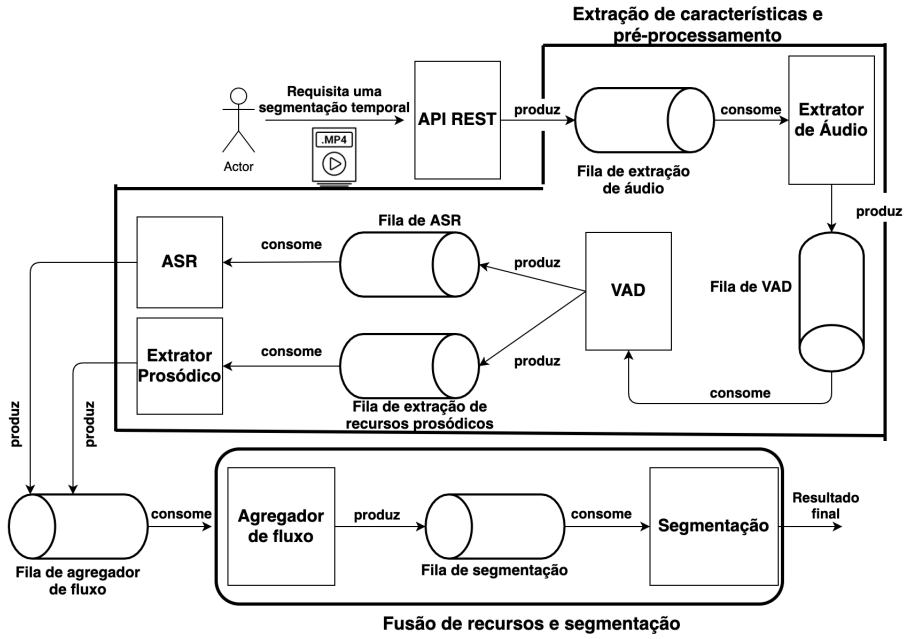

Figure 3. Esquema de comunicação entre os módulos da arquitetura implementada.

3.2.1. Extração de recursos

A **API REST** é o ponto de entrada da arquitetura e responsável por receber dos usuários a videoaula a ser segmentada e parâmetros de configuração através do método HTTP POST. Após receber a solicitação de segmentação, o módulo API REST armazena o vídeo em um banco de dados e insere uma mensagem de solicitação na fila de tarefas do módulo Extrator de áudio. O **Extrator de áudio** extrai o áudio da videoaula recebida. No final da extração, o Extrator de áudio armazena o áudio da videoaula em banco de dados. Além disso, ele insere mensagens de solicitação na fila de tarefas do **módulo VAD**. Este módulo executa a detecção de atividade de voz no arquivo de áudio extraído para remover o silêncio e o ruído. Ao final do processamento, o VAD armazena os trechos de áudio com voz gerados no banco de dados. Além disso, o VAD insere mensagens de solicitação nas filas de tarefas dos módulos ASR e Extrator Prosódico. Em seguida, o módulo **ASR** transcreve automaticamente o discurso contido em cada pedaço de áudio gerado pelo VAD. O módulo ASR gera e armazena um arquivo de texto no banco de dados para cada pedaço de áudio transcrito. Depois disso, ele produz uma mensagem de solicitação na fila de tarefas do Agregador de Fluxo. Por sua vez, o **Extrator Prosódico** é responsável por extrair os recursos prosódicos de cada pedaço de áudio. Como o módulo ASR, este módulo armazena o resultado do seu processamento no banco de dados e insere uma mensagem de solicitação na fila de tarefas do Agregador de Fluxo.

3.2.2. Fusão de recursos e segmentação

O módulo Agregador de fluxo é responsável por combinar recursos que são processados de forma assíncrona por seus respectivos módulos e enviá-los ao algoritmo de segmentação que irá tomar as decisões de fato. Quando uma mensagem chega em sua

fila de tarefas (do módulo ASR ou Extrator prosódico, por exemplo), o módulo Agregador de fluxo inicia um *worker* para aguardar os recursos ausentes concluírem o processamento e, em seguida, agrega-os em uma única mensagem para enviá-los ao algoritmo de segmentação. O módulo Agregador de fluxo identifica mensagens do processamento de uma videoaula usando um ID exclusivo gerado para cada videoaula fornecida como entrada pelo usuário. Por fim, módulo de **Segmentação** contém o algoritmo de segmentação, responsável pelas tomadas de decisão para encontrar soluções para o problema utilizando-se dos recursos extraídos da videoaula.

4. Avaliação da Proposta

Para avaliar a proposta e demonstrar seu potencial, comparamos os resultados de segmentação temporal desenvolvido no contexto do nosso laboratório com os resultados de outros dois trabalhos da literatura.

4.1. Conjunto de dados

Para avaliar nossa solução (subseção 3.1) e fornecer evidências experimentais sobre a qualidade de nossa proposta, realizamos experimentos em dois conjuntos de dados de videoaulas com características diferentes. O primeiro conjunto de dados, que chamamos de "audio-based", consiste em videoaulas em inglês sobre assuntos científicos exatos. Nestas videoaulas, um professor fala de um palco para uma audiência sem qualquer apresentação de slides visível ou outro texto na tela. Assim, são videoconferências em que a informação está predominantemente no áudio, mais especificamente no discurso do professor. Extraímos essas videoaulas do site http://videolectures.net, onde estão disponíveis gratuitamente. Por outro lado, o segundo conjunto de dados consiste em videoaulas em inglês do canal do YouTube da ACM (Association for Computing Machinery) [3]. Nestas videoaulas, uma apresentação de slides em tela cheia é exibida enquanto um discurso em segundo plano explica o conteúdo, por isso chamamos de conjunto de dados "slide-based". Além disso, eles foram gravados em ambientes muito mais controlados e sem ruído do que aqueles do primeiro conjunto de dados.

4.1.1. Métricas de Avaliação

Para avaliar a qualidade da proposta, a segmentação obtida por nosso método foi comparada a uma segmentação de base verdade. Assim, é possível estimar a que distância os resultados estão do ideal. A abordagem foi avaliada usando as métricas de Precisão, Revocação e F1 Score [Goutte and Gaussier 2005]. Por meio delas, podemos analisar os principais aspectos das soluções para o problema de segmentação temporal, como a precisão ao definir os limites dos tópicos no tempo e a abrangência da proposta ao encontrá-los.

4.2. Resultados

Nesta subseção, avalia-se a qualidade da segmentação temporal alcançada pelo uso do EasyTopic. Seu desempenho, de acordo com as métricas de avaliação, foi comparado com duas outras abordagens da literatura. A avaliação foi dividida em duas partes: na primeira,

[3]https://www.youtube.com/user/TheOfficialACM

o desempenho de nossa proposta no conjunto de videoaulas "*audio-based*" foi comparado com os resultados obtidos pela proposta de [Galanopoulos and Mezaris 2019] (VFWE); na segunda parte, o desempenho da nossa proposta foi comparado com a proposta de [Biswas et al. 2015] (MMTOC) usando o conjunto de videos "slide-based".

Table 1. Comparação de desempenho no conjunto de videos "audio-based" entre o EasyTopic e o VFWE [Galanopoulos and Mezaris 2019]

Method	$Precisão_m$	$Revocação_m$	$F1Score_m$
EasyTopic	**0.41 +/- 0.18**	**0.51 +/- 0.16**	**0.42 +/- 0.13**
VFWE	**0.41 +/- 0.22**	0.14 +- 0.18	0.20 +/- 0.10

Table 2. Comparação de desempenho no conjunto de videos "slide-based" entre o EasyTopic e o MMTOC [Biswas et al. 2015]

Method	$Precisão_m$	$Revocação_m$	$F1Score_m$
EasyTopic	0.49 +/- 0.20	**0.50 +/- 0.26**	**0.44 +/- 0.15**
MMTOC	**0.83 +/- 0.22**	0.21 +/- 0.14	0.31 +/- 0.14

Antes de tirar qualquer conclusão comparativa sobre os resultados das Tabelas 1 e 2, primeiro calculamos a significância estatística deles através do teste t de Student bicaudal para duas amostras independentes [Lakens 2017]. Esse teste é usado para determinar se duas médias independentes são diferentes uma da outra, assumindo a hipótese nula de que são iguais. Assim, aplicamos o teste t de Student para cada medida de avaliação, considerando os resultados dos métodos que estamos comparando. Aplicando o teste t, obtemos os valores de p mostrados na Tabela 3.

Table 3. Resultados do teste t de Student para a média das medidas de avaliação

	Audio-based p-value	Slide-based p-value
Precisão	1.00	2.87×10^{-11}
Revocação	5.10×10^{-15}	8.40×10^{-7}
F1 Score	8.53×10^{-13}	9.00×10^{-3}

Definindo um intervalo de confiança de 95%, podemos afirmar, por convenção, que a hipótese nula é rejeitada se o valor de *p* for menor que 0.05 [Greenland et al. 2016]. Portanto, de acordo com os resultados da Tabela 3, podemos dizer que as diferenças entre os resultados obtidos pelos soluções da literatura e o EasyTopic são estatisticamente significativas. A única exceção diz respeito à comparação entre a Precisão média de nossa proposta e a do VFWE no conjunto de videoaulas "audio-based", do qual não podemos rejeitar a hipótese nula de que as médias são iguais.

Assim, podemos dizer que o uso do EasyTopic superou o VFWE e o MMTOC em Revocação e F1-Score. Em relação à Precisão, o EasyTopic obteve o mesmo desempenho do VFWE no conjunto de videoaulas "audio-based" e foi superado pelo MMTOC no conjunto de videoaulas "slide-based". Esses resultados mostram o potencial da solução proposta neste trabalho, uma vez que através de sua utilização foi possível implementar um algoritmo de segmentação temporal de videoaulas (desde a extração de features até a segmentação em si) capaz de superar os resultados de outras abordagens da literatura.

A grande vantagem do EasyTopic reside no fato de ele possibilitar a configuração de fluxos de processamento, bem como a configuração de cada módulo de forma independente. Assim, ele permite que possamos definir parâmetros e algoritmos a serem executados por cada módulo, possibilitando assim, uma maior flexibilidade para atacar os mais diferentes domínios de videoaulas. Além disso, caso seja necessário, é possível instanciar diferentes módulos com algoritmos distintos, por exemplo, para extração de recursos ou até mesmo para encontrar diversas soluções para o problema, bastando passar as configurações e modelos necessários para tal.

5. Conclusão e trabalhos futuros

Neste trabalho, foi apresentado o EasyTopic, uma abordagem para segmentação temporal de videoaulas. Essa solução permite a configuração de fluxos de processamento abrangendo desde a extração de recursos das videoaula e sua fusão até o algoritmo de tomada de decisão responsável por encontrar as soluções para o problema. Essa solução foi projetada de forma a possibilitar a fácil implementação, execução e avaliação de métodos baseados em fala para a segmentação temporal de videoaulas através da instanciação de módulos responsáveis por executar tarefas bem definidas presentes no método desejado.

Como prova de conceito, implementamos o fluxo de um algoritmo de segmentação baseado em fala proposto anteriormente em [Soares and Barrére 2019] utilizando o EasyTopic. Também, comparamos os resultados obtidos por nossa implementação com outros dois *frameworks* da literatura. Como resultados, foi possível ver que, utilizando o EasyTopic, o algoritmo foi capaz de superar os outros dois tanto em Revocação quanto em F1 *score*, o que demonstra o funcionamento e potencial dessa proposta na solução do problema.

Como trabalhos futuros, pretende-se focar no aprimoramento da implementação do EasyTopic, permitindo a reutilização de fluxos de processamento pré-existentes, assim como na criação de uma interface *web* por onde será possível configurar e acompanhar a execução desses fluxos. Além disso, a realização de experimentos que levam em conta aspectos de performance computacional do EasyTopic também está no cronograma de execução do projeto.

References

Biswas, A., Gandhi, A., and Deshmukh, O. (2015). Mmtoc: A multimodal method for table of content creation in educational videos. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 621–630. ACM.

Che, X., Yang, H., and Meinel, C. (2018). Automatic online lecture highlighting based on multimedia analysis. *IEEE Transactions on Learning Technologies*, 11(1):27–40.

Davila, K. and Zanibbi, R. (2017). Whiteboard video summarization via spatio-temporal conflict minimization. In *Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on*, volume 1, pages 355–362. IEEE.

Galanopoulos, D. and Mezaris, V. (2019). Temporal lecture video fragmentation using word embeddings. In *International Conference on Multimedia Modeling*, pages 254–265. Springer.

Goutte, C. and Gaussier, E. (2005). A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. In *European Conference on Information Retrieval*, pages 345–359. Springer.

Greenland, S., Senn, S. J., Rothman, K. J., Carlin, J. B., Poole, C., Goodman, S. N., and Altman, D. G. (2016). Statistical tests, p values, confidence intervals, and power: a guide to misinterpretations. *European journal of epidemiology*, 31(4):337–350.

Lakens, D. (2017). Equivalence tests: a practical primer for t tests, correlations, and meta-analyses. *Social psychological and personality science*, 8(4):355–362.

Lin, M., Chau, M., Cao, J., and Nunamaker Jr, J. F. (2005). Automated video segmentation for lecture videos: A linguistics-based approach. *International Journal of Technology and Human Interaction (IJTHI)*, 1(2):27–45.

Mahapatra, D., Mariappan, R., and Rajan, V. (2018). Automatic hierarchical table of contents generation for educational videos. In *Companion of the The Web Conference 2018 on The Web Conference 2018*, pages 267–274. International World Wide Web Conferences Steering Committee.

Mitra, U. and Srivastava, G. (2020). A study on agent-based web searching and information retrieval. In *Intelligent Communication, Control and Devices*, pages 569–578. Springer.

Pavel, A., Hartmann, B., and Agrawala, M. (2014). Video digests: a browsable, skimmable format for informational lecture videos. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*, pages 573–582. ACM.

Ronchetti, M. (2010). Using video lectures to make teaching more interactive. *International Journal of Emerging Technologies in Learning (iJET)*, 5(2).

Shah, R. R., Yu, Y., Shaikh, A. D., Tang, S., and Zimmermann, R. (2014). Atlas: automatic temporal segmentation and annotation of lecture videos based on modelling transition time. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 209–212. ACM.

Soares, E. R. and Barrére, E. (2019). An optimization model for temporal video lecture segmentation using word2vec and acoustic features. In *Proceedings of the 25th Brazillian Symposium on Multimedia and the Web*, pages 513–520.

Tuna, T., Joshi, M., Varghese, V., Deshpande, R., Subhlok, J., and Verma, R. (2015). Topic based segmentation of classroom videos. In *2015 IEEE Frontiers in Education Conference (FIE)*, pages 1–9. IEEE.

Yang, H. and Meinel, C. (2014). Content based lecture video retrieval using speech and video text information. *IEEE Transactions on Learning Technologies*, 7(2):142–154.

Zhang, Y., Zhang, J., and Zhang, D. (2009). Implementing and testing producer-consumer problem using aspect-oriented programming. In *2009 Fifth International Conference on Information Assurance and Security*, volume 2, pages 749–752. IEEE.