

Melhorando recomendações baseadas no contexto com classificação de texto para sistemas de recomendação de domínios cruzados

Jen Horng Liu¹, Douglas Vêras e Silva¹, Luciano Demétrio Santos Pacífico¹

¹ Departamento de Computação – Universidade Federal Rural de Pernambuco (UFRPE)
CEP 52.171-900 – Recife, PE – Brasil

{jen.horng,douglas.veras,luciano.pacifico}@ufrpe.br

Abstract. *Context-aware recommender systems (CARS) have gained substantial attention for their ability to enhance recommendation accuracy by considering various contextual factors. However, effective integration of contextual information across diverse domains remains a challenge. This paper presents a novel methodology for inferring contextual information using text classification techniques and assesses its impact on the performance of a Cross-Domain Context-Aware Recommender System (CD-CARS). Our methodology involves preprocessing textual data and utilizing Support Vector Machines (SVM) for context inference. Through an extensive evaluation, we analyze the effects of integrating inferred context on the CD-CARS framework. Experimental results demonstrate that the proposed methodology yields improved recommendation accuracy and relevance across different domains. The findings highlight the potential of leveraging text classification techniques to enhance context-aware recommendations, thus contributing to the advancement of recommendation systems in cross-domain scenarios.*

Resumo. *Os sistemas de recomendação sensíveis ao contexto (CARS) ganharam atenção substancial por sua capacidade de melhorar a precisão das recomendações, considerando vários fatores contextuais. Contudo, a integração eficaz de informações contextuais em diversos domínios continua a ser um desafio. Este artigo apresenta uma nova metodologia para inferir informações contextuais usando técnicas de classificação de texto, e avalia seu impacto no desempenho de um Sistema de Recomendação Sensível ao Contexto de Domínios Cruzados (CD-CARS). Nossa metodologia envolve o pré-processamento de dados textuais e a utilização de Support Vector Machines (SVM) para inferência de contexto. Através de uma avaliação extensa, analisamos os efeitos da integração do contexto inferido nos algoritmos do CD-CARS. Os resultados experimentais demonstram que a metodologia proposta produz maior precisão e relevância das recomendações em diferentes domínios. As conclusões destacam o potencial de aproveitar técnicas de classificação de texto para melhorar as recomendações sensíveis ao contexto, contribuindo assim para o avanço dos sistemas de recomendação em cenários de domínios cruzados.*

1. Introdução

A crescente oferta e produção de serviços e itens por parte de empresas e plataformas on-line [Hong and Kim 2016] torna cada vez mais complexa a escolha de itens que atendam às necessidades dos clientes. Os Sistemas de Recomendação (SR) são uma solução para esse problema, auxiliando os usuários na descoberta de informações com base em seus comportamentos e preferências prévias [Resnick and Varian 1997].

Os SR tradicionais, como Sistemas de Recomendação de Filtragem Colaborativa (SRFC) [Ricci et al. 2015], operam em duas dimensões (Usuário X Item) e utilizam classificações numéricas (como avaliações de 5 estrelas) para calcular a semelhança entre usuários e itens, com propósito de gerar recomendações. Em contraposição, os Sistemas de Recomendação Sensíveis ao Contexto (SRSC) [Panniello et al. 2014], incorporam informações contextuais (Usuário X Item X Contexto) para aprimorar a precisão das sugestões. A informação contextual pode ser qualquer coisa, desde a localização do usuário até o clima, por exemplo [Adomavicius and Tuzhilin 2010].

No entanto, esses sistemas também apresentam alguns obstáculos que precisam ser superados para que possam ser implementados de forma eficaz. Um dos principais desafios dos SRSC é a identificação e coleta de informações contextuais relevantes [Yujie and Licai 2010], já que essas informações nem sempre estão disponíveis ou são escassas. Essa carência de dados contextuais pode impactar diretamente na capacidade dos sistemas de fornecer recomendações personalizadas e precisas aos usuários [Champiri et al. 2015], uma vez que o contexto desempenha um papel crucial na compreensão das preferências individuais e na adaptação das sugestões de acordo com diferentes cenários e necessidades.

Para preencher essa lacuna nas informações contextuais, diversos estudos passaram a inferir dados de contextos a partir das avaliações feitas por usuários, visando aprimorar os modelos de recomendação [Lahlou et al. 2013, Bauman and Tuzhilin 2014, Campos et al. 2017]. Por exemplo, um sistema pode identificar o clima, a hora do dia e a localização de um usuário a partir de uma revisão e usar essas informações para recomendar itens relevantes [Adomavicius et al. 2005]. Entre as pesquisas que lidam com inferência contextual de companhia, a de [Lahlou et al. 2013] se destacou por seus bons resultados e implementação acessível.

Outro problema é que os SR tradicionais costumam ser projetados com foco em um único domínio, como filmes ou livros [Cantador et al. 2015]. Contudo, em muitas situações, os usuários demonstram interesse por itens que pertencem a múltiplos domínios [Cremonesi et al. 2011]. Por exemplo, um usuário pode manifestar interesse tanto por filmes quanto por livros, ou ainda por música e restaurantes.

Para lidar com o desafio de recomendar itens de diversos domínios, os pesquisadores [Cantador et al. 2015, Hu et al. 2018] têm optado por implementar Sistemas de Recomendação de Domínio Cruzado (SRDC). Essa escolha se baseia na capacidade dos SRDCs de aprimorar a precisão das recomendações, aproveitando as relações entre diferentes domínios, como já destacado em estudos anteriores [Cantador et al. 2015].

[Véras et al. 2019] apresenta um Sistema de Recomendação de Domínio Cruzado Sensíveis ao Contexto (CD-CARS). Este sistema gera recomendações considerando as avaliações dos usuários, os itens em questão e os contextos associados a esses itens nos

domínios de origem e alvo. A abordagem do CD-CARS segue a modelagem de contexto baseada em [Bauman and Tuzhilin 2014], incorporando informações contextuais, como tempo, local e companhia, nas recomendações. Contudo, o autor [Véras et al. 2019] menciona que não obteve resultados satisfatórios na inferência de informação contextual de companhia.

Portanto, é essencial incorporar abordagens mais precisas de inferências contextuais das revisões de usuários para o CD-CARS, visando aprimorar a acurácia das recomendações [Véras et al. 2019]. Diante desse cenário, torna-se imprescindível também abordar a necessidade de avaliar a eficácia dessa abordagem em diferentes domínios, considerando as variações nas características de contexto e nas preferências dos usuários, a fim de garantir sua aplicabilidade ampla e efetiva no campo das recomendações personalizadas.

O objetivo desta pesquisa é aplicar a metodologia de inferência contextual de companhia proposta por [Lahlou et al. 2013], pelos resultados positivos e reprodução acessível da metodologia, no CD-CARS desenvolvido por [Véras et al. 2019], e fazer a análise dos resultados obtidos, a fim de verificar se houve melhoria nas recomendações.

O artigo está organizado da seguinte forma: na Seção 2, discutimos trabalhos relacionados sobre inferência contextual para SR, abordagem proposta por [Lahlou et al. 2013] e sobre o funcionamento da metodologia apresentada por [Véras et al. 2019], o CD-CARS. A Seção 3 aborda a metodologia empregada deste estudo. Isso inclui a descrição da base de dados para a classificação, os algoritmos de inferência contextual de companhia, algoritmos de recomendação. Na Seção 4 apresentamos o detalhamento dos experimentos realizados e resultados obtidos. Por fim, na Seção 5, apresentamos nossas conclusões.

2. Trabalhos Relacionados

Recentemente, muitos trabalhos têm sido propostos para extrair dados de avaliações e integrá-los ao processo de recomendação. Esta seção apresenta alguns dos trabalhos feitos sobre inferência contextual para SRSCs.

No estudo realizado por [Hariri et al. 2011] foi introduzido um SRSC projetado para aproveitar as avaliações de hotéis feitas por usuários, para capturar informações contextuais. Esses dados contextuais são integrados às classificações históricas para calcular uma função, oferecendo posteriormente recomendações de itens aos usuários. A abordagem trata o contexto de forma semelhante a um problema de modelagem de tópico supervisionado e desenvolve um classificador de contexto por meio de Labeled Latent Dirichlet Allocation (Labeled-LDA). O algoritmo K-Nearest Neighbors (KNN) é empregado para previsão de classificação dentro do sistema. No entanto, este trabalho não detalhou os resultados das inferências contextuais, apenas da predição de recomendação.

[Lahlou et al. 2013] conduziu um estudo para avaliar a precisão da inferência de informações contextuais a partir das avaliações dos usuários, empregando aprendizado supervisionado e técnicas de pré-processamento textual. A pesquisa se concentrou na detecção da intenção de compra nas avaliações de hotéis e carros. A abordagem sugere a remoção de termos únicos, a combinação de n-gramas e o uso de ponderação Term Frequency - Inverse Document Frequency (TF-IDF) [Qaiser and Ali 2018] para melhorar a

precisão da classificação. Diferentes configurações de representação de texto e algoritmos de classificação, como o Multinomial Naive Bayes (Multinomial NB) e SVM foram testados nos conjuntos de dados do TripAdvisor¹ e Cars². Contudo, a precisão dos algoritmos de classificação pode ser influenciada pela quantidade dos dados de treinamento, pela seleção de atributos contextuais e ajustes de parâmetros.

O trabalho de [Bauman and Tuzhilin 2014] introduz um método para extrair dados contextuais relevantes de avaliações de usuários, categorizando-as em “específicas” e “genéricas”, com foco nas avaliações específicas que detalham visitas a estabelecimentos. Dois métodos, “word-based” e “LDA-based”, são usados para extrair contextos de avaliações específicas. A validação em dados do Yelp³ revelou alta acurácia na separação de revisões por categoria. No entanto, a pesquisa também destaca limitações de dependência da qualidade e quantidade das avaliações, também na possibilidade de subjetividade introduzir ruído nos atributos contextuais extraídos.

[Kim et al. 2016] propôs um novo modelo de SR chamado fatoração de matriz convolucional (ConvMF). O ConvMF funciona primeiro aplicando Redes Neurais Convolucionais (CNNs) ao texto do documento para extrair características do texto. Esses recursos são então usados para treinar um modelo Probabilistic Matrix Factorization (PMF) para prever classificações. As CNNs auxiliam na captura das informações contextuais dos documentos, o que pode melhorar a precisão do modelo PMF. O ConvMF tem se mostrado eficaz na recomendação de itens aos usuários, especialmente nos casos em que os dados de classificação são escassos.

O estudo de [Campos et al. 2017] propõe um método para extrair dados contextuais de avaliações de usuários usando uma taxonomia genérica baseada na DBpedia⁴. Essa taxonomia é criada automaticamente e vincula palavras-chave em avaliações a categorias da DBpedia. A taxonomia é enriquecida com informações de fontes como WordNet [Miller 1995], melhorando sua qualidade e utilidade, permitindo a identificação do contexto. No entanto, os dados do DBpedia, que podem estar desatualizados, comprometendo a eficácia do método.

[Ding et al. 2019], os autores propõem uma abordagem para melhorar recomendações Online-to-Offline, em que é necessário inferir informações de contexto dos usuários em tempo real. Para isso, a abordagem apresentada é a Mixture Attentional Constrained Denoise AutoEncoder (MACDAE). No MACDAE, é explorado as interações entre usuários, itens e contextos explícitos a fim de inferir os contextos implícitos dos usuários.

[Sitkrongwong and Takasu 2019] propõe uma nova técnica de extrair contexto para SRSCs. A metodologia abrange duas etapas: extração de contexto na forma skipgrams de palavras usando a técnica de Region Embedding e previsão de classificação usando redes neurais. Embora os resultados sejam promissores, a técnica apresentada necessita de uma base de dados grande para conseguir inferir um contexto relevante.

Diante das propostas dos estudos previamente discutidos, resumida na Tabela 1,

¹<https://www.tripadvisor.com/>

²<https://www.cars.com/>

³<https://www.yelp.com/>

⁴<http://dbpedia.org/resource/Category>

a abordagem de [Sitkrongwong and Takasu 2019] mostrou-se eficiente, contudo empregamos a metodologia elaborada por [Lahlou et al. 2013], pois pudemos aproveitar a disponibilidade da implementação do algoritmo por [Silva 2019]. Além disso, planejamos aplicar os resultados da classificação nos algoritmos abordados no CD-CARS [Véras et al. 2019], pois o autor não obteve bons resultados na classificação contextual ao adotar a metodologia apresentada no trabalho de [Bauman and Tuzhilin 2014]. Também iremos comparar os resultados obtidos pela nossa metodologia proposta, com o que foi apresentado originalmente no CD-CARS.

Tabela 1. Resumo dos trabalhos relacionados

Trabalhos	Abordagem	Técnica de classificação	Classifica contexto de companhia	Base de dados
Hariri et al. [Hariri et al. 2011]	Supervisionada	CS e LDA	Sim	TripAdvisor
Lahlou et al. [Lahlou et al. 2013]	Supervisionada	NB e SVM	Sim	TripAdvisor e Cars
Bauman e Tuzhilin [Bauman and Tuzhilin 2014]	Não supervisionada	Word e LDA	Não	Yelp
Kim et al. [Kim et al. 2016]	Supervisionada	CNN e PMF	Não	MovieLens e Amazon Instant Video
Campos et al. [Campos et al. 2017]	Semi-supervisionado	CoreNLP, AL e DL	Sim	Amazon Reviews
Ding et al. [Ding et al. 2019]	Não supervisionada	MACDAE	Sim	Yelp, Dianping e Koubei
Sitkrongwong e Takasu [Sitkrongwong and Takasu 2019]	Não Supervisionada	Region Embedding	Sim	TripAdvisor

CS - Cosine Similarity

Word - Word-based method

PMF - Probabilistic Matrix Factorization

MACDAE - Mixture Attentional Constrained Denoise AutoEncoder

AL - Apache Lucene (Software)

DL - Damerau-Levenshtein distance

2.1. Inferência contextual

Para inferir informações contextuais, a abordagem proposta por [Lahlou et al. 2013], primeiramente os documentos passarão por etapas de pré-processamento. Após essa fase, cada documento será convertido em um modelo vetorial e, por fim, será realizada a classificação contextual das avaliações feitas por usuários.

2.1.1. Pré-processamento

O autor aplicou diversas técnicas de pré-processamento de texto, como tokenização, stemming, remoção de stopwords, Part-of-Speech Tagging (POS tagging), n-gramas e Term Frequency Thresholding (TFT), para preparar os documentos.

- Tokenização é o processo de dividir um texto em unidades menores, chamadas de *tokens*. Os tokens podem ser palavras, frases ou outras unidades significativas [Kannan et al. 2014].
- Stemming é a redução de palavras à sua forma base ou radical, a fim de lidar com variações morfológicas de uma palavra [Vijayarani et al. 2015].
- Remoção de stopwords é o processo de remover palavras comuns que não contribuem para o significado do texto [Aggarwal and Aggarwal 2015].
- POS tagging é a atribuição de rótulos gramaticais a cada palavra no texto [Petrov et al. 2011].
- N-gramas são sequências contíguas de N palavras, que ajudaram a capturar informações contextuais, como bigramas (duas palavras consecutivas) ou trigramas (três palavras consecutivas) [Robertson and Willett 1998].

- TFT é uma técnica que remove termos que ocorrem com baixa frequência no texto [Lahlou et al. 2013].

A normalização incluiu a conversão das palavras para minúsculas, remoção de acentos e pontuação, e algumas stopwords. Os pronomes foram mantidos durante a remoção de stopwords, pois eram relevantes. POS tagging ajudou na remoção controlada de stopwords. Stemming reduziu variações de palavras, e a técnica TFT eliminou termos pouco relevantes que ocorreram apenas uma vez no conjunto de dados.

2.1.2. Transformação

Nesta etapa, a transformação dos comentários pré-processados é realizada utilizando o modelo de espaço vetorial, necessária para permitir o uso de algoritmos de classificação que requerem entrada numérica. E para calcular a relevância das palavras é aplicada a técnica de TF-IDF [Lahlou et al. 2013].

2.1.3. Classificação

O autor [Lahlou et al. 2013] emprega dois classificadores amplamente reconhecidos, o Naive Bayes e o Support Vector Machine (SVM) [Ceci and Malerba 2007], para realizar previsões sobre a categoria de cada avaliação. No caso do Naive Bayes (NB), optam pela variante Multinomial NB, que é mais adequada para lidar com tarefas como a contagem de palavras. Quanto ao Support Vector Machine, utilizam o algoritmo Sequential Minimum Optimization (SMO), uma escolha conhecida por sua rapidez e eficiência na implementação do SVM [Bharti et al. 2022]. Para avaliar o desempenho dos classificadores, os autores adotam o método 10-fold cross-validation. Esse método implica a divisão do conjunto de dados em 10 partes iguais, treinando o modelo em nove delas e testando-o na parte restante [Arlot and Celisse 2010]. Esse processo é repetido 10 vezes, com cada parte servindo como conjunto de teste uma vez. A avaliação do desempenho dos classificadores é baseada nas métricas de acurácia⁵, precisão⁶, sensibilidade⁷ e Medida-F⁸.

2.2. Cross domain context-aware recommender system (CD-CARS)

O CD-CARS (Figura 1), conforme proposto por [Véras et al. 2019], tem como objetivo aprimorar as recomendações de filtragem colaborativa em vários domínios, incorporando técnicas sensíveis ao contexto. Para alcançar esse propósito, o CD-CARS recomenda itens do domínio alvo ao explorar as similaridades entre usuários, considerando suas classificações atribuídas aos itens e os contextos associados a esses domínios. A abordagem do CD-CARS integra algoritmos sensíveis ao contexto, especificamente a Pré-Filtragem Contextual (PreF) e a Pós-Filtragem Contextual (PostF), adaptados para operar em múltiplos domínios, com o intuito de melhorar o desempenho do algoritmo base de domínio-cruzado, NNUserNgrbr-transClosure [Cremonesi et al. 2011].

⁵Accuracy

⁶Precision

⁷Recall

⁸F-measure

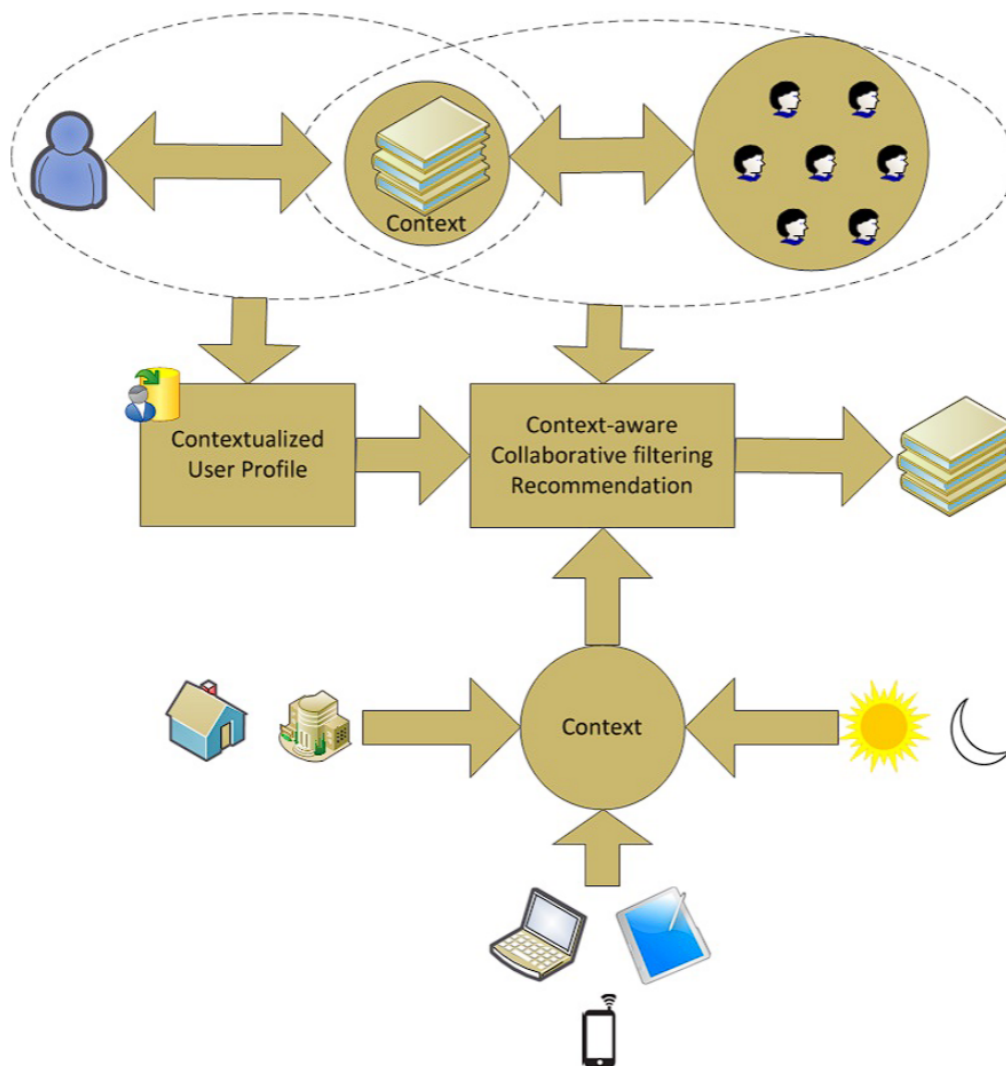


Figura 1. Funcionamento do CD-CARS

Fonte: [Véras et al. 2019]

O algoritmo de Pré-Filtragem Contextual [Adomavicius and Tuzhilin 2010] inicia por aplicar um critério contextual para identificar avaliações relevantes de usuários em um domínio particular, resultando na criação de um conjunto de dados contextualmente definido. Em sequência, o algoritmo NNUserNgb-transClosure é imediatamente empregado nesse conjunto de dados gerado, resultando em previsões de avaliações no domínio alvo da recomendação.

O algoritmo Pós-Filtragem Contextual [Adomavicius and Tuzhilin 2010] primeiro aplica o algoritmo de filtragem colaborativa em todo o conjunto de dados para gerar um conjunto de recomendações. Em seguida, ele filtra as recomendações com base nos critérios contextuais do domínio alvo. Por exemplo, se o domínio alvo for um sistema de recomendação de música e as informações contextuais forem a localização do usuário, o algoritmo filtrará as músicas que são mais bem avaliadas na localização do usuário das recomendações geradas.

2.2.1. Cenários de SRDC

Recomendações multidomínios são baseadas em conjuntos de preferências de usuários, que podem ser expressas por avaliações de itens. Os usuários podem avaliar itens de um único domínio ou de diferentes domínios [Cantador et al. 2015]. Nesse último caso, pode haver sobreposição de usuários em diferentes níveis. Existem dois cenários de recomendação em relação a avaliações de usuários em multidomínios [Cremonesi et al. 2011]:

- Com sobreposição de usuários: Neste caso, um único usuário fornece avaliações abrangendo pelo menos dois domínios diferentes. Por exemplo, podemos considerar um usuário que avalia tanto filmes quanto livros em uma mesma plataforma de compras online.
- Nessa situação, não há avaliações provenientes do mesmo usuário nos domínios considerados para a recomendação. Em outras palavras, cada usuário manifesta suas preferências apenas dentro de um único domínio específico de itens.

3. Metodologia

O modelo proposto é dividido em duas partes. Na primeira, utiliza-se a implementação de [Silva 2019] para extrair contexto de companhia, baseada no trabalho de [Lahlou et al. 2013]. Na segunda parte, emprega-se a técnica de SR proposta por [Véras et al. 2019], para prever classificações e gerar recomendações com base nos resultados da primeira parte. A seção detalha a obtenção dos conjuntos de dados para treinamento e avaliação, explora a extração de contexto de companhia da base de dados e aborda as técnicas utilizadas no CD-CARS.

3.1. Inferência contextual de companhia

Nesta abordagem, os documentos são pré-processados, convertidos em modelos vetoriais e, em seguida, classificados contextualmente em relação à companhia.

Primeiro aplicamos técnicas de pré-processamento de texto nas revisões classificadas manualmente, que inclui: tokenização, remoção de stopwords, stemming e TFT. Em sequência aplicamos *TfidfVectorizer* fornecido pela biblioteca *Scikit Learn* nos dados pré-processados, para transformar os dados em vetor. Por fim, utilizamos o algoritmo de aprendizado supervisionado SVM para a classificação de contexto, pois no estudo de [Silva 2019], o algoritmo mostrou-se mais preciso ao comparado com o *NB*.

No entanto, [Silva 2019] ressalta que na metodologia proposta por [Lahlou et al. 2013], os parâmetros de configuração utilizados pelos modelos classificadores não foram fornecidos. Portanto, para definir o conjunto ótimo de parâmetros, foi necessário recorrer à técnica de otimização de hiperparâmetros conhecida como *grid search* [Bergstra and Bengio 2012].

3.2. Geração de recomendações

Nesta etapa, o conjunto de dados gerado na extração de contexto é utilizado no sistema do CD-CARS. O autor [Véras et al. 2019] disponibilizou a estrutura completa do CD-CARS, incluindo a arquitetura e o código-fonte, para que pudéssemos realizar nossos experimentos.

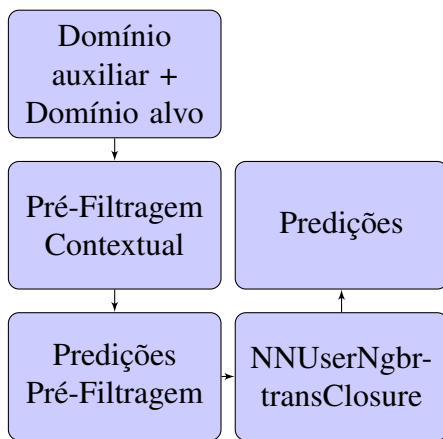


Figura 2. Etapas do PreF

Para gerar recomendações com o algoritmo PreF, mostrada na Figura 2, as preferências do domínio alvo da recomendação são filtradas usando informações contextuais de companhia. Isso forma um conjunto de preferências. Em seguida, aplicamos o algoritmo NNUserNgbr-transClosure ao conjunto formado pelo algoritmo PreF, gerando predições de avaliações a itens pertencentes ao domínio alvo.

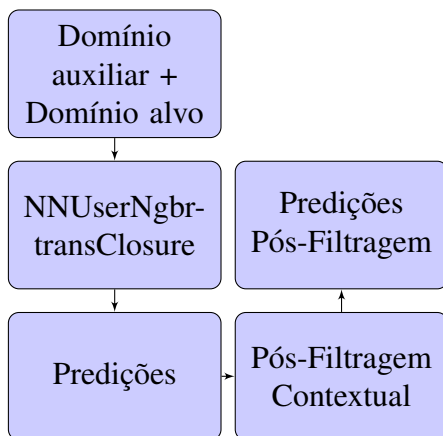


Figura 3. Etapas do PostF

Na geração das recomendações utilizando o algoritmo PostF (Figura 3), primeiro é aplicado o algoritmo NNUserNgbr-transClosure no conjunto de preferências formado pela agregação dos domínios auxiliares e alvo. Em seguida, as informações contextuais são usadas para filtrar os resultados no passo anterior. O PostF seleciona as melhores predições com base nos itens contidos no conjunto de categorias preferenciais dos usuários, gerando um novo conjunto de predições.

4. Experimentos

Os experimentos apresentados nesta seção (Figura 4) visam avaliar o impacto do uso de informações contextuais de companhias extraídas pela técnica apresentada por [Lahlou et al. 2013] no CD-CARS.

4.1. Configurações do Algoritmo de Recomendação

Para a operação do algoritmo *NNUserNgr-transClosure*, adotamos a mesma configuração estabelecida por [Véras et al. 2019]. Inicialmente, o tamanho da vizinhança foi estabelecido em 475, representado pelos “N” usuários mais próximos. Em seguida, a Distância Euclidiana foi selecionada como a métrica de similaridade, uma vez que apresentou o melhor desempenho nas previsões de avaliações, abrangendo diversos cenários de recomendação em ambientes de múltiplos domínios. Após a determinação da métrica de similaridade, o limiar do PostF foi definido em “2/3” da frequência das categorias preferidas dos usuários.

Para a avaliação das previsões, os conjuntos de treinamento e teste foram criados para cada domínio alvo e contexto de teste. Em todos os experimentos, 100% das preferências disponíveis nos domínios auxiliares foram selecionadas para o treinamento dos algoritmos de recomendação, enquanto no domínio alvo 90% das preferências foram selecionadas para treinamento e 10% para teste, utilizando o método de amostragem *Holdout cross-validation*.

Na avaliação da classificação, a configuração para a métrica *Medida-F* foi a seguinte: Selecionamos aleatoriamente 1,4% das classificações do conjunto de dados original para formar um conjunto de teste, enquanto o conjunto de treinamento consistiu em 98,6% das classificações restantes. O conjunto de teste incluiu apenas classificações de 5 estrelas, excluindo as demais. A abordagem foi adaptada ao domínio alvo e contexto de teste para garantir relevância nos conjuntos.

Tabela 2. Cenários do experimento

Cenários dos Experimentos	Sobreposição de usuário	Base de dados	Domínio Alvo
Cenário 1	10%	Book-television	Television
Cenário 2	50%	Book-television	Television
Cenário 3	100%	Book-television	Television
Cenário 4	10%	Book-television	Book
Cenário 5	50%	Book-television	Book
Cenário 6	100%	Book-television	Book
Cenário 7	10%	Book-music	Music
Cenário 8	50%	Book-music	Music
Cenário 9	100%	Book-music	Music
Cenário 10	10%	Book-music	Book
Cenário 11	50%	Book-music	Book
Cenário 12	100%	Book-music	Book

4.2. Bases de dados

Neste estudo, adotamos o conjunto de dados⁹ disponibilizados por [Véras et al. 2019]. Esses conjuntos compreendem dois cenários: um que inclui domínios mais relacionados, como livros e televisão, e outro que considera domínios menos relacionados, como livros e música. Cada cenário é caracterizado por diferentes níveis de sobreposição de usuários, variando entre 10%, 50% e 100%.

⁹<https://github.com/douglasveras/cd-cars-datasets>

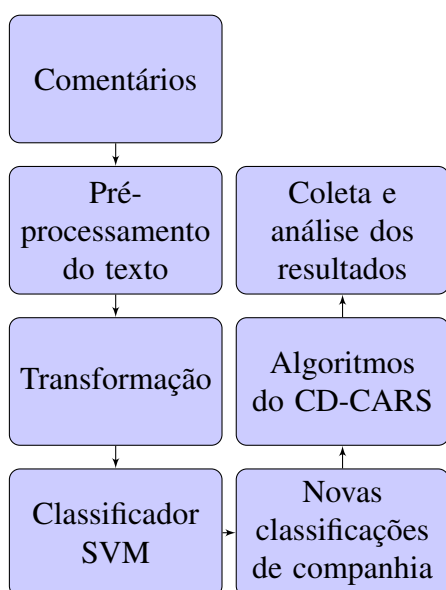


Figura 4. Etapas do experimento

Essas bases de dados, extraída da Amazon¹⁰ por [Véras et al. 2019], incluem metadados e análises de usuários para vários produtos pertencentes aos domínios citados anteriormente. Esses conjuntos de dados foram adaptados para incluir apenas as informações relevantes para este estudo, como as avaliações dadas aos itens (classificadas de 1 a 5 estrelas) e detalhes do contexto de companhia, que foram deduzidas com base nas avaliações fornecidas pelos usuários.

4.2.1. Base de dados das avaliações dos usuários

Neste conjunto de dados extraído por [Véras et al. 2019] inclui o ID do usuário, código do item e avaliações em inglês de uma variedade de produtos da Amazon, como livros, músicas e filmes. As bases de dados correspondentes são *datasetBooksReviews* (210.883 avaliações), *datasetMusicReviews* (18.757 avaliações) e *datasetTelevisionReviews* (118.071 avaliações), respectivamente. Utilizamos esta base para classificação do contexto de companhia.

4.2.2. Book-television dataset

Para este conjunto de dados, considerando apenas as avaliações com 100% de sobreposição de usuários entre os domínios de livro e televisão, contém no total 1.249.949 avaliações feitas por 15.341 usuários, sendo 805.102 avaliações de livros e 444.847 de filmes. Na base de dados, cerca de apenas 20% das avaliações possuíam contexto de companhia, resultando em aproximadamente 251.707 avaliações.

¹⁰<https://amazon.com>

4.2.3. Book-music dataset

Nessa base de dados também considerou apenas as avaliações com 100% de sobreposição de usuários entre os domínios de livro e música. A soma dos resultados entre os domínios resultou em 1.031.386 avaliações feitas por 13.189 usuários, dos quais 742.844 avaliações foram de livros e 288.542 de música. Contendo, aproximadamente 207.010 das avaliações tinham classificações do contexto de companhia.

4.3. Métricas de avaliação

As métricas *Mean Absolute Error* (MAE) e *Root Mean Square Error* (RMSE) foram escolhidas para avaliar o desempenho das previsões de recomendações geradas pelo algoritmo de SR [Gunawardana et al. 2012]. E para o desempenho de classificação dos algoritmos, a *Medida-F* foi escolhida. Adotamos essas métricas para o nosso trabalho, pelo fato de que foram as métricas usadas no CD-CARS.

$$MAE = \frac{\sum_{i=1}^N |p_i - q_i|}{N} \quad (1)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (p_i - q_i)^2}{N}} \quad (2)$$

Nas equações 1 e 2, mostram detalhadamente as fórmulas das métricas. Onde p_i representa a classificação real do item i , q_i representa a classificação prevista do item i e N representa o número total de classificações a serem previstas. A métrica *Medida-F*, é calculada conforme os valores *Precisão* e *Sensibilidade* [Cremonesi et al. 2010]. São apresentadas da seguinte forma:

$$Sensibilidade(N) = \frac{TotalClassificadosCorretos}{|TotaldeItensNaClasse|}$$

$$Precisão(N) = \frac{TotalClassificadosCorretos}{N * |TotaldeItensNaClasse|}$$

$$Medida - F(N) = \frac{2 * Sensibilidade(N) * Precisão(N)}{Sensibilidade(N) + Precisão(N)}$$

4.4. Análise da significância estatística

Adotamos o teste não paramétrico de Wilcoxon [Wilcoxon et al. 1970], conforme usado por [Véras et al. 2019], para avaliar a significância estatística dos resultados da nossa abordagem. Este teste foi conduzido com um nível de significância de 5%. Utilizamos a implementação do teste (ranksums) da biblioteca *Scipy* e aplicamos o teste¹¹ nos resultados preditivos da base Book-television.

¹¹ranksums(CD-CARS, Metodologia-proposta, alternative='greater')

Tabela 3. Conjunto de dados classificados manualmente

Classe	Quantidade
Acompanhado	21
Família	108
Amigos	69
Casal	68
Sozinho	247
Total	513

4.5. Classificação das bases de dados

A base de dados de treinamento para as classificações das revisões foi gerada por [Silva 2019], que fez uma classificação manual dos dados usados no trabalho de [Véras et al. 2019]. Foram escolhidas 300 avaliações relacionadas a livros, outras 300 referentes a música e um total de 379 avaliações relacionadas a filmes. Assim, as três bases de dados totalizaram um conjunto de 979 revisões. Porém cerca de 47.2% das avaliações não possuem classe conhecida, ou seja, são avaliações que não descrevem nenhuma informação contextual de companhia, portanto, foram removidas.

As categorias contextuais das avaliações foram estabelecidas com base nas categorias contextuais de companhia definidas por [Véras et al. 2019]. As avaliações da classe contextual “Colega” foram excluídas, pois constituíam apenas 0,4% do conjunto de revisões. Devido à quantidade muito reduzida dessas revisões, não seria viável treinar os algoritmos de classificação de maneira satisfatória para prever esse tipo de classe e também foi removida da base. Consequentemente, o conjunto final de dados¹² é formado por um total de 513 revisões, conforme indicado (Tabela 3).

Nesta etapa de classificação contextual de companhia, aplicamos somente a configuração de pré-processamento: *Stemming + TFT + UniBiTrigrama*¹³, pois no experimento realizado por [Silva 2019], essa foi a configuração em conjunto com o classificador SVM que obteve melhor resultado nas métricas: acurácia (77.2%). Também aplicamos a técnica *grid search* com os valores para os parâmetros:

- C - [0.001, 0.01, 0.1, 1, 10]
- Gamma - [0.001, 0.01, 0.1, 1]
- Kernel - ['linear', 'poly', 'rbf', 'sigmoid']

Ao finalizar o *grid search* os parâmetros de configuração foram definidos como $C=1$, $gamma=0,1$ e $kernel=rbf$ para o modelo. Para validar o modelo, o método *K-fold cross-validation* com 10 *folds* foi utilizado.

Após a definição dos parâmetros do classificador, executamos o treinamento do SVM com a base de dados rotulada (Tabela 3). Na validação, inferimos o contexto de companhia nos conjuntos de dados *datasetBooksReviews*, *datasetMusicReviews* e *datasetTelevisionReviews*.

Após a execução da etapa anterior, substituímos as classificações contextuais de companhia antigas das bases *Book-music dataset* e *emphBook-television dataset*. Obser-

¹²https://github.com/douglashss/reviews_amazon_dataset

¹³Combinação de unigrama, bigrama e trigrama

vamos que para a base *Book-music* 117.226 das avaliações sofreram alteração, no total de 207.010, com isso a taxa de mudança da classificação contextual de companhia foi cerca de 56.6% (Tabela 7). Notamos que para a base *Book-Television* das 249.989 avaliações que contém contexto de companhia, 77.588 teve a classificação contextual alterada, resultando numa taxa de mudança de aproximadamente 31%, abaixo quando comparada com a da base *Book-Music*. A alta taxa de mudança da classificação contextual pode ter sido ocasionada pela diferença de precisão da inferência de contexto das metodologias adotadas por [Véras et al. 2019] e [Silva 2019] (Tabela 8).

4.6. Execução do CD-CARS

O algoritmo de recomendação foi aplicado nos conjuntos de dados descritos em 4.2.2 e 4.2.3, representando três cenários de recomendação em domínios cruzados: 10%, 50% e 100% de sobreposição de usuários. Cada conjunto de dados foram formados pelos domínios - livros, filmes e músicas - que foram alternados entre domínio alvo e domínio de origem.

Os critérios contextuais para PreF e PostF foram formados por atributos apenas do contexto de companhia. Ao todo foram explorados doze cenários distintos de recomendação, apresentado na Tabela 2. Cada cenário foi executado três vezes consecutivas para coletar os resultados e realização dos testes estatísticos, com as configurações especificadas na subseção 4.1.

4.7. Resultados

De acordo com os experimentos realizados, apresentamos os resultados da nossa abordagem comparados com os resultados originais do CD-CARS.

4.7.1. Book-Music

Nos gráficos (A) e (B) da Figura 5, apresentam os resultados com domínio de origem *Book* e domínio alvo *Music*. Podemos observar na Tabela 4 sobreposição 10%, os resultados obtidos CD-CARS + SVM, integrado com Pós-filtragem contextual, obteve uma acurácia maior em 8.4% (MAE) e em 38.2% (RMSE). Na sobreposição 50%, a metodologia proposta gerou um menor desempenho em 7.5% (MAE) e 11% (RMSE). Com sobreposição 100%, resultou em uma melhora de apenas 7% (MAE) e 8.5% (RMSE).

E com a aplicação do algoritmo de Pré-Filtragem, analisamos que na sobreposição de 10%, o desempenho inferior em 44.5% (MAE) e 17.2% (RMSE). Com sobreposição em 50%, houve uma pequena melhora de 3.5% (MAE) e 14% (RMSE). Observamos na Tabela 4 com sobreposição 100% que o algoritmo proposto foi um pouco inferior ao CD-CARS, em 5.9% (MAE) e 4.1% (RMSE).

Analisando os gráficos (C) e (D) e com o auxílio da Tabela 5, com domínio de origem *Music* e domínio alvo *Book*, o resultado do algoritmo de Pós-filtragem contextual, foi inferior em 32.4% (MAE) e superior em 14.7% (RMSE), com nível de sobreposição de 10%. Com nível de sobreposição de usuários em 50%, a diferença nos resultados das predições foram pequenas tanto para ambas métricas. Verificamos que na sobreposição de 100% os resultados foram inferior em 3% (MAE) e 7% (RMSE).

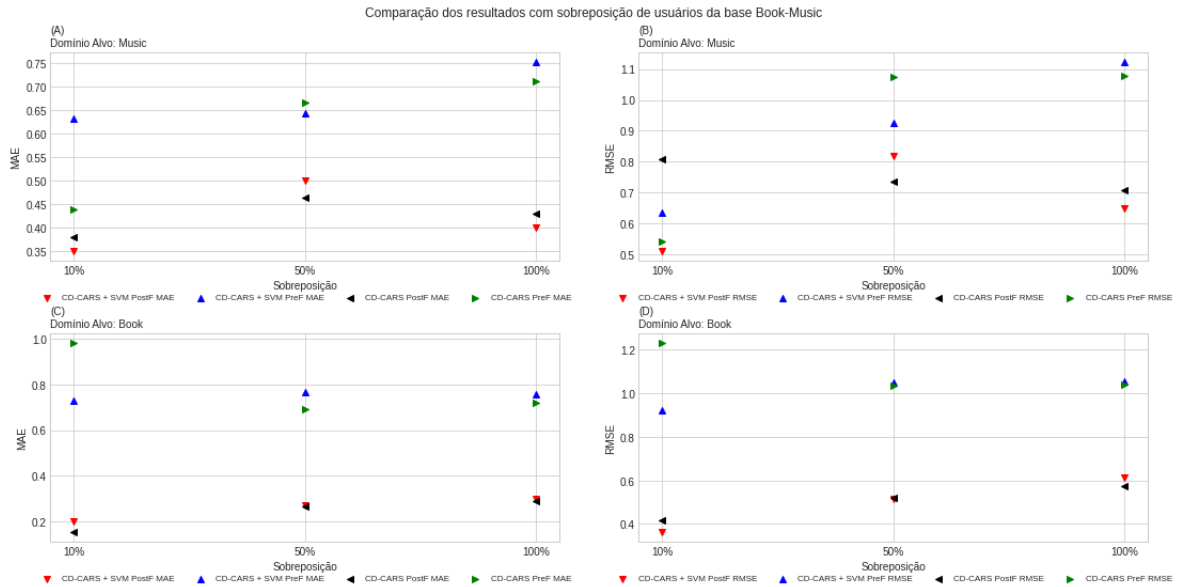


Figura 5. Comparação dos resultados (MAE / RMSE) do CD-CARS + SVM e CD-CARS e níveis de sobreposição. Base Book-Music

No experimento do algoritmo de Pré-filtragem, com nível de sobreposição de usuários em 10%, houve um aumento de 25.9% (MAE) e 25.2% (RMSE), na acurácia das predições. Com nível de 50% na sobreposição de usuários, os resultados obtidos foi inferior em 10.8% (MAE) e 1.5% (RMSE). Por fim, com sobreposição 100%, resultou desempenho um pouco inferior, 5.1% (MAE) e 1.3% (RMSE).

Os gráficos (A) e (B) da Figura 7 apresentam os resultados da métrica de precisão Medida-F. Analisando o gráfico e Tabela 6, notamos que para todos os níveis de sobreposição de usuários, a aplicação da Pós-filtragem foi superior em 7,6%, 11,7% e 12,2% respectivamente. Porém, na Pré-filtragem contextual, resultou desempenho inferior em todos os níveis de sobreposição de usuários. No gráfico B com o domínio alvo *Book*, observamos um desempenho melhor nos três níveis de sobreposição tanto com a Pós-filtragem, quanto com Pré-filtragem.

Tabela 4. Desempenho preditivo geral (MAE/RMSE) com desvio padrão variando o nível de sobreposição do usuário (domínio de origem: book e domínio alvo: music)

(*) Diferença entre o resultado obtido e o resultado original do CD-CARS

Abordagens de Recomendação	10% de sobreposição		50% de sobreposição		100% de sobreposição	
	MAE ± dp	RMSE ± dp	MAE ± dp	RMSE ± dp	MAE ± dp	RMSE ± dp
PostF com NNUserNbr-transClosure	0.349 ± 0.088 (8.4%)*	0.507 ± 0.082 (38.2%)*	0.500 ± 0.043 (-7.5%)*	0.818 ± 0.066 (-11.9%)*	0.401 ± 0.022 (7%)*	0.647 ± 0.048 (8.5%)*
PreF com NNUserNbr-transClosure	0.633 ± 0.262 (-44.5%)*	0.633 ± 0.262 (-17.2%)*	0.643 ± 0.116 (3.5%)*	0.926 ± 0.140 (14%)*	0.753 ± 0.078 (-5.9%)*	1.124 ± 0.125 (-4.1%)*

Tabela 5. Desempenho preditivo geral (MAE/RMSE) com desvio padrão variando o nível de sobreposição do usuário (domínio de origem: music e domínio alvo: book)

(*) Diferença entre o resultado obtido e o resultado original do CD-CARS

Abordagens de Recomendação	10% de sobreposição		50% de sobreposição		100% de sobreposição	
	MAE ± dp	RMSE ± dp	MAE ± dp	RMSE ± dp	MAE ± dp	RMSE ± dp
PostF com NNUserNbr-transClosure	0.200 ± 0.175 (-32.4%)*	0.360 ± 0.271 (14.7%)*	0.272 ± 0.028 (-3%)*	0.517 ± 0.069 (1%)*	0.298 ± 0.027 (-3%)*	0.614 ± 0.043 (-7%)*
PreF com NNUserNbr-transClosure	0.730 ± 0.170 (25.9%)*	0.922 ± 0.140 (25.2%)*	0.768 ± 0.010 (-10.8%)*	1.051 ± 0.042 (-1.5%)*	0.758 ± 0.017 (-5.1%)*	1.055 ± 0.021 (-1.3%)*

Tabela 6. Desempenho com a métrica *Medida-F* com desvio padrão da nossa proposta, com os algoritmos de recomendação do CD-CARS, na base Book-Music (contexto de companhia)

(*) Diferença entre o resultado obtido e o resultado original do CD-CARS

Domínio Alvo	Sobreposição	NNUserNgbr-transClosure + PostF	NNUserNgbr-transClosure + PreF
Music	10%	0.0280 ± 0.0014 (7.6%)*	0.0190 ± 0.0028 (-9.5%)*
Music	50%	0.0458 ± 0.0003 (11.7%)*	0.0289 ± 0.0007 (-12.5%)*
Music	100%	0.0494 ± 0.0010 (12.2%)*	0.0423 ± 0.0004 (-6%)*
Book	10%	0.0259 ± 0.0006 (3.6%)*	0.0188 ± 0.0010 (4.4%)*
Book	50%	0.0518 ± 0.0010 (12.6%)*	0.0391 ± 0.0002 (77.7%)*
Book	100%	0.0622 ± 0.0006 (19.6%)*	0.0544 ± 0.0002 (94.2%)*

Tabela 7. Taxa de mudança das classes contextuais de companhia nas bases de dados

<i>Dataset</i>	Taxa de mudança
<i>Book-Music</i>	56.6%
<i>Book-Television</i>	31%

4.7.2. Book-Television

Os gráficos (A) e (B) da Figura 6 e Tabela 9 apresentam os resultados do domínio de origem *Book* e domínio alvo *Television*. Com 10% de sobreposição de usuários, os resultados obtidos pela metodologia proposta integrado com Pós-Filtragem, resultou na melhoria de 14.7% e 15.1% nas acurácias de predições nas métricas MAE e RMSE. Com nível de sobreposição 50% dos usuários, o tivemos uma piora no desempenho em 10.5% (MAE) e 9.2% (RMSE). Com sobreposição de 100%, notamos que tivemos uma pequena piora na acurácia em 3.6% (MAE) e 0.7% (RMSE).

Os resultados obtidos com Pré-Filtragem com nível de sobreposição de usuários em 100%, obtemos um melhor desempenho de 2.7% tanto para a métrica MAE e RMSE. Nos níveis de sobreposição 10% e 50%, os resultados gerados pelo nosso modelo proposto tiveram uma perda de desempenho de 26.4% e 14.9% na métrica MAE, e de 17.2% e 10.6% para métrica RMSE.

Ao mudar o domínio alvo, vemos que nos gráficos (C), (D) e Tabela 10 os resultados com a Pós-filtragem foram melhores em 15% (MAE) e 29.5% (RMSE) com nível de sobreposição de 10%. Para nível de sobreposição 50%, a acurácia foi pior em 14.7% (MAE) e 13.6% (RMSE). Uma pequena melhoria de 7.5% e 4.7% foram observadas nos resultados, com sobreposição de 100% nas métricas MAE e RMSE, respectivamente.

Por fim, ao integrar a Pré-filtragem contextual no modelo, o desempenho foi inferior nos níveis de sobreposição 10% e 50%. Sendo 11.2% (MAE) e 5.9% (RMSE) para

Tabela 8. Comparação da precisão de classificação contextual de companhia das metodologias

Resultados das metodologias de inferência contextual	Precisão de classificação
CD-CARS [Véras et al. 2019] baseado em [Bauman and Tuzhilin 2014]	35.8%
Silva [Silva 2019] baseado em [Lahlou et al. 2013]	77.2%

o nível de sobreposição 10%. Com 0.6% para a métrica MAE e 1.2% para RMSE, para o nível de sobreposição em 50%. Com sobreposição de usuários em 100%, uma pequena melhora da acurácia de 2.8% foi observada para a MAE e 5.3% para RMSE.

Nos gráficos (C) e (D) da Figura 7 e Tabela 11 apresentam os resultados para a métrica Medida-F. Ao analisar o gráfico (C), notamos que O obtivemos desempenho inferior em todos os níveis de sobreposição, quando comparado à metodologia original do CD-CARS em no cenários Pré-filtragem, no Pós-Filtragem obtivemos uma pequena melhora. No gráfico (D), para Pós-Filtragem, notamos de desempenho inferior para níveis de sobreposição 10% e 100% , porém o resultado foi melhor na sobreposição de 50%. Na Pré-Filtragem, o desempenho melhorou para os níveis de sobreposição 10% e 100%, mas foi inferior na sobreposição de 50%.

Tabela 9. Desempenho preditivo geral (MAE/RMSE) com desvio padrão variando o nível de sobreposição do usuário (domínio de origem: book e domínio alvo: television)

(*) Diferença entre o resultado obtido e o resultado original do CD-CARS

(**) Não é estatisticamente significativo

Abordagens de Recomendação	10% de sobreposição		50% de sobreposição		100% de sobreposição	
	MAE ± dp	RMSE ± dp	MAE ± dp	RMSE ± dp	MAE ± dp	RMSE ± dp
PostF com NNUserNgr-transClosure	0.198 ± 0.038 (14.7%)(*)	0.418 ± 0.067 (15.1%)(*)	0.283 ± 0.038 (-10.5%)(**)	0.590 ± 0.043 (-9.2%)(**)	0.229 ± 0.015 (-3.6%)(**)	0.527 ± 0.023 (-0.7%)(**)
PreF com NNUserNgr-transClosure	1.177 ± 0.308 (-26.4%)(**)	1.534 ± 0.186 (-17.2%)(**)	0.986 ± 0.044 (-14.9%)(**)	1.332 ± 0.043 (-10.6%)(**)	0.828 ± 0.019 (2.7%)(**)	1.150 ± 0.036 (2.7%)(**)

Tabela 10. Desempenho preditivo geral (MAE/RMSE) com desvio padrão variando o nível de sobreposição do usuário (domínio de origem: television e domínio alvo: book)

(*) Diferença entre o resultado obtido e o resultado original do CD-CARS

(**) Não é estatisticamente significativo

Abordagens de Recomendação	10% de sobreposição		50% de sobreposição		100% de sobreposição	
	MAE ± dp	RMSE ± dp	MAE ± dp	RMSE ± dp	MAE ± dp	RMSE ± dp
PostF com NNUserNgr-transClosure	0.205 ± 0.056 (15%)(**)	0.367 ± 0.085 (29.5%)(*)	0.303 ± 0.029 (-14.7%)(**)	0.617 ± 0.062 (-13.6%)(**)	0.259 ± 0.020 (7.5%)(*)	0.548 ± 0.019 (4.7%)(*)
PreF com NNUserNgr-transClosure	0.680 ± 0.153 (-11.2%)(**)	0.924 ± 0.234 (-5.9%)(**)	0.762 ± 0.024 (-0.6%)(**)	1.082 ± 0.034 (-1.2%)(**)	0.767 ± 0.029 (2.8%)(**)	1.046 ± 0.042 (5.3%)(*)

Tabela 11. Desempenho com a métrica Medida-F com desvio padrão da nossa proposta, com os algoritmos de recomendação do CD-CARS, na base Book-Television (contexto de companhia)

(*) Diferença entre o resultado obtido e o resultado original do CD-CARS

Domínio Alvo	Sobreposição	NNUserNgr-transClosure + PostF	NNUserNgr-transClosure + PreF
Television	10%	0.0565 ± 0.0023 (2.7%)*	0.0272 ± 0.0010 (-9.4%)*
Television	50%	0.0736 ± 0.0006 (0.8%)*	0.0428 ± 0.0015 (-22.2%)*
Television	100%	0.0779 ± 0.0017 (3.8%)*	0.0621 ± 0.0016 (-17.2%)*
Book	10%	0.0301 ± 0.0010 (-3%)*	0.0183 ± 0.0018 (1.6%)*
Book	50%	0.0518 ± 0.0023 (7.9%)*	0.0300 ± 0.0010 (-3.3%)*
Book	100%	0.0620 ± 0.0008 (-1.6%)*	0.0437 ± 0.0007 (4%)*

4.8. Discussão

De acordo com os resultados da subseção anterior 4.7, a nossa metodologia proposta melhorou as recomendações do CD-CARS em alguns cenários, principalmente com pós-filtragem contextual e com 10% de sobreposição de usuários.

Uma possível explicação para esses resultados é que a base de treinamento para inferência contextual de companhia estava desbalanceada em relação à quantidade de comentários por classe (Tabela 3). Outro fator que pode ter influenciado é o tamanho limitado da base de treinamento, que pode não ter sido suficiente para treinar adequadamente

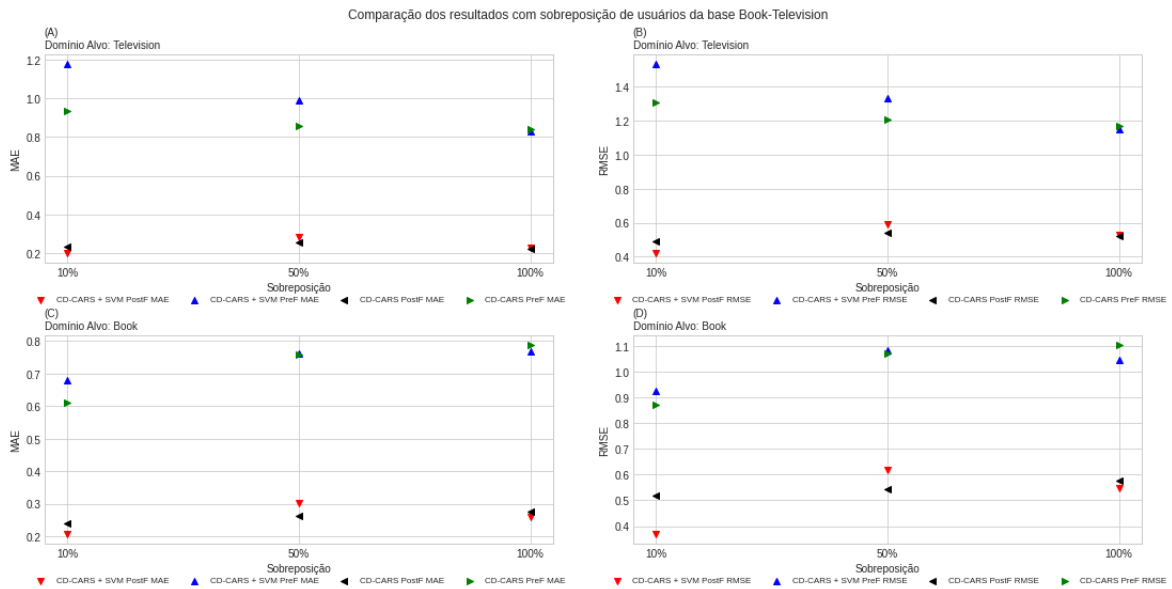


Figura 6. Comparação dos resultados (MAE / RMSE) do CD-CARS + SVM e CD-CARS e níveis de sobreposição. Base Book-Television.

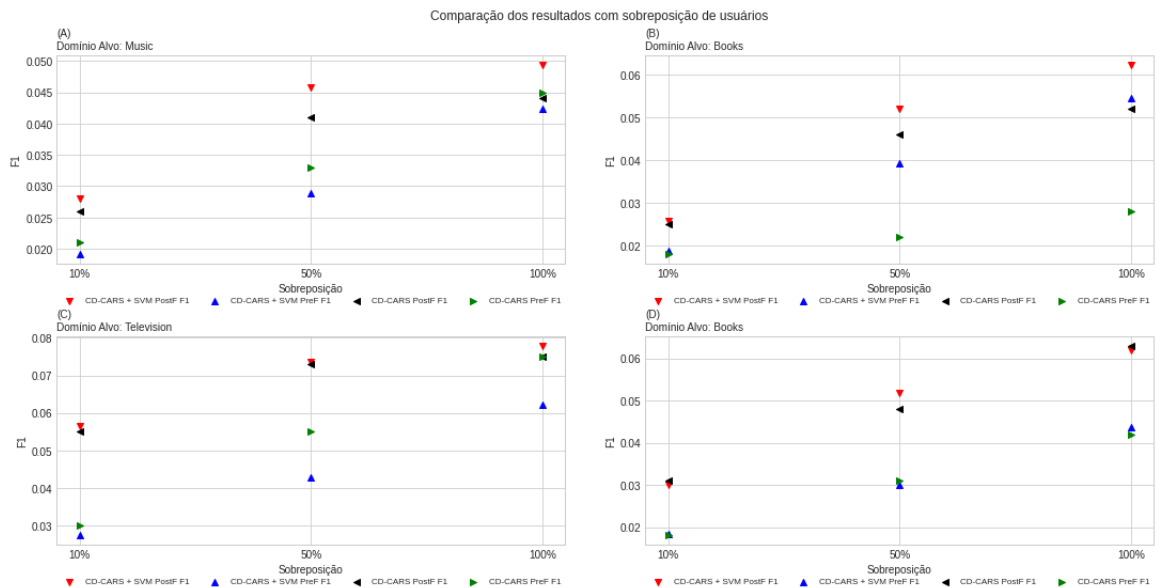


Figura 7. Comparação dos resultados (Medida-F) do CD-CARS + SVM e CD-CARS e níveis de sobreposição. Bases Book-Music e Book-Television

o modelo. Uma terceira hipótese é que os experimentos (Tabela 2) foram executados poucas vezes, o que pode ter reduzido a confiabilidade dos resultados.

Para validar os resultados obtidos, é essencial conduzir experimentos adicionais utilizando uma base de treinamento mais equilibrada e maior. Além disso, os experimentos devem ser executados um número maior de vezes para assegurar a confiabilidade dos resultados.

5. Conclusão

Neste artigo, analisamos a integração da metodologia para classificação contextual de companhia, método proposto por [Lahlou et al. 2013], no CD-CARS. Usamos a base fornecida por [Silva 2019] para treinar o classificador e as bases de validação foi fornecida por [Véras et al. 2019].

Para inferir as informações contextuais, as análises passam por etapas de pré-processamento, transformação e classificação. Adotamos o *SVM* como o classificador, de acordo com o estudo de [Silva 2019], o algoritmo mostrou-se mais preciso em comparação com o *NB*. Após a classificação realizamos o experimento utilizando a nova base gerada na etapa anterior, em conjunto com a metodologia proposta no CD-CARS.

Os resultados mostraram uma melhora no desempenho do CD-CARS em situações de baixa sobreposição de usuários, embora o impacto foi menor em domínios de alta sobreposição. A métrica Medida-F mostrou que, em geral, nossos resultados foram melhores do que os da proposta original do CD-CARS.

5.1. Trabalhos Futuros

Para trabalhos futuros podemos revalidar o experimento de classificação contextual com uma base de dados maior para treino, pois com a base atual as classes contextuais de companhia estavam desbalanceadas. Adicionalmente, é interessante executar os experimentos um número maior de vezes para obtermos resultados mais confiáveis e com maior significância estatística. Também é válido implementar e adaptar outra metodologia de classificação contextual para SR de trabalhos mais recentes, como proposto pelos autores [Sitkrongwong and Takasu 2019] e integrar com o CD-CARS.

Referências

- Adomavicius, G., Sankaranarayanan, R., Sen, S., and Tuzhilin, A. (2005). Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information systems (TOIS)*, 23(1):103–145.
- Adomavicius, G. and Tuzhilin, A. (2010). Context-aware recommender systems. In *Recommender systems handbook*, pages 217–253. Springer.
- Aggarwal, C. C. and Aggarwal, C. C. (2015). *Mining text data*. Springer.
- Arlot, S. and Celisse, A. (2010). A survey of cross-validation procedures for model selection.
- Bauman, K. and Tuzhilin, A. (2014). Discovering contextual information from user reviews for recommendation purposes. In *CBREcsys@ recsys*, pages 2–9.
- Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2).
- Bharti, K., Cervera-Lierta, A., Kyaw, T. H., Haug, T., Alperin-Lea, S., Anand, A., Degroote, M., Heimonen, H., Kottmann, J. S., Menke, T., et al. (2022). Noisy intermediate-scale quantum algorithms. *Reviews of Modern Physics*, 94(1):015004.
- Campos, P. G., Rodríguez-Artigot, N., and Cantador, I. (2017). Extracting context data from user reviews for recommendation: A linked data approach. In *ComplexRec@ RecSys*, pages 14–18.

- Cantador, I., Fernández-Tobías, I., Berkovsky, S., and Cremonesi, P. (2015). Cross-domain recommender systems. *Recommender systems handbook*, pages 919–959.
- Ceci, M. and Malerba, D. (2007). Classifying web documents in a hierarchy of categories: a comprehensive study. *Journal of Intelligent Information Systems*, 28:37–78.
- Champiri, Z. D., Shahamiri, S. R., and Salim, S. S. B. (2015). A systematic review of scholar context-aware recommender systems. *Expert Systems with Applications*, 42(3):1743–1758.
- Cremonesi, P., Koren, Y., and Turrin, R. (2010). Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 39–46.
- Cremonesi, P., Tripodi, A., and Turrin, R. (2011). Cross-domain recommender systems. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pages 496–503. Ieee.
- Ding, X., Tang, J., Liu, T., Xu, C., Zhang, Y., Shi, F., Jiang, Q., and Shen, D. (2019). Infer implicit contexts in real-time online-to-offline recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2336–2346.
- Gunawardana, A., Shani, G., and Yoyev, S. (2012). Evaluating recommender systems. In *Recommender systems handbook*, pages 547–601. Springer.
- Hariri, N., Mobasher, B., Burke, R., and Zheng, Y. (2011). Context-aware recommendation based on review mining. In *ITWP@ IJCAI*.
- Hong, S.-E. and Kim, H.-J. (2016). A comparative study of video recommender systems in big data era. In *2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN)*, pages 125–127. IEEE.
- Hu, G., Zhang, Y., and Yang, Q. (2018). Conet: Collaborative cross networks for cross-domain recommendation. In *Proceedings of the 27th ACM international conference on information and knowledge management*, pages 667–676.
- Kannan, S., Gurusamy, V., Vijayarani, S., Ilamathi, J., Nithya, M., Kannan, S., and Gurusamy, V. (2014). Preprocessing techniques for text mining. *International Journal of Computer Science & Communication Networks*, 5(1):7–16.
- Kim, D., Park, C., Oh, J., Lee, S., and Yu, H. (2016). Convolutional matrix factorization for document context-aware recommendation. In *Proceedings of the 10th ACM conference on recommender systems*, pages 233–240.
- Lahlou, F. Z., Benbrahim, H., Mountassir, A., and Kassou, I. (2013). Inferring context from users’ reviews for context aware recommendation. In *Research and Development in Intelligent Systems XXX: Incorporating Applications and Innovations in Intelligent Systems XXI Proceedings of AI-2013, The Thirty-third SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence*, pages 227–239. Springer.
- Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

- Panniello, U., Tuzhilin, A., and Gorgoglione, M. (2014). Comparing context-aware recommender systems in terms of accuracy and diversity. *User Modeling and User-Adapted Interaction*, 24:35–65.
- Petrov, S., Das, D., and McDonald, R. (2011). A universal part-of-speech tagset. *arXiv preprint arXiv:1104.2086*.
- Qaiser, S. and Ali, R. (2018). Text mining: use of tf-idf to examine the relevance of words to documents. *International Journal of Computer Applications*, 181(1):25–29.
- Resnick, P. and Varian, H. R. (1997). Recommender systems. *Communications of the ACM*, 40(3):56–58.
- Ricci, F., Rokach, L., and Shapira, B. (2015). Recommender systems: introduction and challenges. *Recommender systems handbook*, pages 1–34.
- Robertson, A. M. and Willett, P. (1998). Applications of n-grams in textual information systems. *Journal of Documentation*, 54(1):48–67.
- Silva, D. H. S. d. (2019). Um estudo comparativo de técnicas para a classificação contextual de companhia para sistemas de recomendação sensíveis a contexto. B.S. thesis, Brasil.
- Sitkrongwong, P. and Takasu, A. (2019). Unsupervised context extraction via region embedding for context-aware recommendations. In *Proceedings of the 23rd International Database Applications & Engineering Symposium*, pages 1–10.
- Véras, D., Prudêncio, R., and Ferraz, C. (2019). Cd-cars: Cross-domain context-aware recommender systems. *Expert Systems with Applications*, 135:388–409.
- Vijayarani, S., Ilamathi, M. J., Nithya, M., et al. (2015). Preprocessing techniques for text mining-an overview. *International Journal of Computer Science & Communication Networks*, 5(1):7–16.
- Wilcoxon, F., Katti, S., Wilcox, R. A., et al. (1970). Critical values and probability levels for the wilcoxon rank sum test and the wilcoxon signed rank test. *Selected tables in mathematical statistics*, 1:171–259.
- Yujie, Z. and Licai, W. (2010). Some challenges for context-aware recommender systems. In *2010 5th International Conference on Computer Science & Education*, pages 362–365. IEEE.