UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Juarez Angelo Piazza Sacenti

Impactos da representação e da sumarização de grafos de conhecimentos em sistemas de recomendação

Florianópolis
2021

Juarez Angelo Piazza Sacenti

Impactos da representação e da sumarização de grafos de conhecimentos em sistemas de recomendação

Tese submetida ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Santa Catarina para a obtenção do título de doutor em Ciência da Computação.
Orientador: Prof. Roberto Willrich, Dr.
Coorientador: Prof. Renato Fileto, Dr.

Florianópolis
2021

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Sacenti, Juarez Angelo Piazza
Impactos da representação e da sumarização de grafos de conhecimentos em sistemas de recomendação / Juarez Angelo Piazza Sacenti ; orientador, Roberto Willrich, coorientador, Renato Fileto, 2021.
152 p.

Tese (doutorado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-Graduação em Ciência da Computação, Florianópolis, 2021.

Inclui referências.

1. Ciência da Computação. 2. Sistema de recomendação. 3. Ontologia. 4. Grafos de conhecimento. 5. Sumarização de grafos. I. Willrich, Roberto. II. Fileto, Renato. III. Universidade Federal de Santa Catarina. Programa de Pós Graduação em Ciência da Computação. IV. Título.

Juarez Angelo Piazza Sacenti

Impactos da representação e da sumarização de grafos de conhecimentos em sistemas de recomendação

O presente trabalho em nível de doutorado foi avaliado e aprovado por banca examinadora composta pelos seguintes membros:

Profª. Carina Friedrich Dorneles, Drª.
Universidade Federal de Santa Catarina

Prof. Luís Paulo Faina Garcia, Dr.
Universidade de Brasília

Prof. Marcelo Garcia Manzato, Dr.
Universidade de São Paulo

Certificamos que esta é a **versão original e final** do trabalho de conclusão que foi julgado adequado para obtenção do título de doutor em Ciência da Computação.

Documento assinado digitalmente
PATRICIA DELLA MEA PLENTZ
Data: 27/11/2021 08:49:28-0300
CPF: 959.243.700-97
Verifique as assinaturas em https://v.ufsc.br

Profª. Patricia Della Méa Plentz, Drª.
Coordenador do Programa

Documento assinado digitalmente
Roberto Willrich
Data: 27/11/2021 17:55:56-0300
CPF: 711.959.399-49
Verifique as assinaturas em https://v.ufsc.br

Prof. Roberto Willrich, Dr.
Orientador

Florianópolis, 2021.

Este trabalho é dedicado a três mulheres.
À minha mãe, quem deu-me luz, quem abriu-me o primeiro
livro e quem agora significa-me os conceitos
de persistência e resiliência a cada dia.
À minha tia, quem deu-me carinho, exemplo e orientação
quanto a importância da dedicação ao estudo e a família,
independente dos desafios enfrentados.
E à minha noiva, quem acompanha-me ao longo desta
e de tantas aventuras por vir, lá e de volta outra vez,
com quem compartilho minhas manias e alegrias,
a garota que esperou e, agora, é tempo de viver.

AGRADECIMENTOS

À Deus que provê saúde e esperança quando grandezas perdem dimensão e fundamentos se reconstroem.
Ao meu pai, que me ensina a superar limites e a seguir em frente.
À minha família, que me conforta em momentos difíceis.
Aos meus amigos que superaram meus queixumes, convertendo-os em gargalhadas e risos.
A todo aquele que compreendeu meus sacrifícios e apoiou-me nesta jornada.
Àqueles que tiraram meu sono e me relembraram quais bens são mais preciosos.
A todos que se foram, e do porvir iluminam corpo, mente e alma.

Ao meu orientador e coorientador, por todo apoio incondicional, dedicação, disponibilidade e incentivo durante minha formação.
Ao PPGCC e a UFSC por sua excelência e compromisso com o conhecimento.
À CAPES e ao CNPq pelo serviço e apoio a formação acadêmica tão necessária ao desenvolvimento de nosso país, grandioso tanto em riquezas quanto em desafios.
A todo aquele engajado no ensino, produção, revisão, veiculação, publicação, organização de conferências e eventos, e em demais tarefas que mantém a ciência viva e em evolução.
Em especial, aos Professores Dr. Adriano Ferreti Borgatto, Dr. Edson Cilos Vargas Júnior e Dr. Márcio Bastos Castro, pelo suporte teórico e de infraestrutura.

O meu profundo agradecimento.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001. Os resultados de Sacenti, Fileto e Willrich (2021) foram reproduzidos com permissão da Springer Nature.

O estudo da representação de conhecimento em sistemas de recomendação
e de seus impactos na eficiência e na eficácia desses sistemas oferece
relevantes oportunidades de pesquisa.

RESUMO

Um problema clássico que frequentemente compromete a qualidade de Sistemas de Recomendação (SRs) é a esparsidade de dados sobre as interações dos usuários com os itens a serem recomendados. A representação de conhecimento acerca dos usuários e dos itens (p.ex., no domínio do filmes, atores, diretores e gêneros), também chamado de informações laterais, por meio de ontologias e Grafos de Conhecimento (GCs) se mostrou eficaz para contornar esse problema. No entanto, o crescimento das informações laterais em termos de volume e complexidade dá origem a muitos desafios, incluindo a demanda por algoritmos de alto custo para lidar com grandes quantidades de dados. Enquanto isso, embora a Sumarização de Grafo (SG) tenha se tornado popular para dar suporte às tarefas de visualização e consulta de GC, seu uso ainda é relativamente inexplorado no domínio de SRs. Esta tese investiga os impactos da representação e sumarização do conhecimento em SRs, tanto a nível de eficácia como de eficiência. A eficácia neste contexto está relacionada à qualidade das recomendações geradas, e eficiência, por sua vez, está relacionada ao custo computacional. Mais especificamente, esta tese investiga duas abordagens para mitigar o problema do alto custo de treinamento de modelos de recomendação baseados em informações laterais. A primeira abordagem converte a representação das informações laterais baseada em ontologia numa matriz de preferência, eliminando a necessidade do uso de algoritmos de alto custo computacional baseados em inferências em ontologias ou na estrutura de rede de informação. Esta abordagem é aplicável a SRs baseados em filtragem híbrida clássica para considerar as informações laterais no processo de recomendação. Para definir melhor esta técnica de conversão, esta tese propõe um arcabouço conceitual, chamado de ORBS, que permite especificar a representação ontológica descrevendo os diferentes aspectos (características) dos itens e as hierarquias de entidades que ordenam e descrevem estes aspectos. Neste arcabouço, o conhecimento é representado usando ontologias de diferentes níveis de abstração, diferenciando conceitos relacionados à tarefa de recomendação, ao domínio do item e à aplicação. A segunda abordagem foca os SRs baseados em GCs (SRGCs), onde é proposto a sumarização do GC utilizando uma técnica que combina *embeddings* com clusterização de nodos para reduzir o volume das informações laterais. Esta técnica adota duas estratégias de sumarização: a única-visão, que sumariza o GC como um todo, e a multi-visão, que separa o GC em múltiplas visões, sumariza cada visão e, então, unifica-as em um único sumário de GC. Estas visões são subgrafos do GC contendo entidades relacionadas a um determinado aspecto de item. Os experimentos desta tese analisaram as duas abordagens propostas usando os dados do MovieLens 1M e informações laterais de Movie Ontology, IMDb e DBpedia. Os resultados demonstram que a especificação de múltiplos aspectos e hierarquias têm o potencial de melhorar a eficácia de SRs. Além disso, experimentos subsequentes avaliam o impacto das duas estratégias da técnica de sumarização de GC na eficiência e eficácia de quatro SRGCs. Os resultados mostram que a sumarização proposta pode melhorar a eficiência do SRs sem mudar significativamente a eficácia.

Palavras-chave: Sistema de recomendação. Ontologia. Grafos de conhecimento. Sumarização de grafos. *Embeddings* de grafo de conhecimento.

ABSTRACT

A classic problem that often compromises the quality of Recommender Systems (RSs) is the sparsity of data about user interactions with the items to be recommended. The representation of knowledge about users and items (e.g., in film domain, actors, directors, and genres), also called lateral information, through ontologies and Knowledge Graphs (KGs), has proven effective to circumvent this problem. However, the growth of lateral information in terms of volume and complexity gives rise to many challenges, including the demand for costly algorithms to handle large amounts of data. Meanwhile, although Graph Summary (GS) has become popular to support KG visualization and query tasks, its use is still relatively unexplored in the recommendation domain. This thesis investigates the impacts of knowledge representation and summarization in recommendation systems, both in terms of effectiveness and efficiency. The effectiveness in this context is related to the quality of the generated recommendations, and efficiency, in turn, is the computational cost. Specifically, this thesis investigates two approaches to mitigate the high cost of training recommendation models based on lateral information. The first approach converts the representation of ontology-based RSs into a preference matrix, removing the need of using high computational cost algorithms based on ontology inference or the structure of information network. This approach is applicable to RSs based on classical hybrid filtering to consider the lateral information not the recommendation process. To better define this mapping technique, this thesis proposes a conceptual framework, called ORBS, which allows specifying the ontological representation describing the different aspects (characteristics) of items and the hierarchies of entities that order and describe these aspects. In this framework, knowledge is represented using ontologies of different levels of abstraction, differentiating concepts related to the recommendation task, item domain and application. The second approach focuses on KG-based RSs, where the KG summarization is proposed using a technique that combines embeddings with node clustering to reduce the volume of lateral information. This method takes two strategies: single-view, which summarizes the KG as a whole, and multi-view, which separates the KG into multiple views, summarizes each view, and then unifies them into a single KG summary. In this context, views are subgraphs of the KG containing entities related to a particular item aspect. The experiments in this thesis analyzed the two proposed approaches using data from MovieLens 1M and side information from Movie Ontology, IMDb and DBpedia. The results demonstrate that specifying multiple aspects and hierarchies has the potential to improve the effectiveness of RSs. Furthermore, subsequent experiments assess the impact of the two KG summarization technique strategies on the efficiency and effectiveness of four KG-based RSs. The results show that the proposed summarization can improve the efficiency of RSs without significantly changing the effectiveness.

Keywords: Recommender system. Ontology. Knowledge graph. Graph summarization. Knowledge graph embeddings.

LISTA DE ILUSTRAÇÕES

LIST OF ALGORITMOS

LISTA DE ABREVIATURAS E SIGLAS

AD Agrupamento de Dados. 31, 34, 148

AM Aprendizado de Máquina. 30, 34, 74, 148

BC Base de Conhecimento. 41, 82, 148

DAC Dados Abertos Conectados. 44, 148

DC Dados Conectados. 28, 39, 43, 66, 79, 148

FBC Filtragem Baseada em Conteúdo. 27, 53, 148

FC Filtragem Colaborativa. 27, 52, 72, 96, 148

FH Filtragem Híbrida. 53, 148

FM *Factorization Machine*. 73, 117, 148

GC Grafo de Conhecimento. 28, 39, 56, 61, 75, 107, 120, 129, 148

HIN *Heterogeneous Information Network*. 61, 73, 148

IMDb *Internet Movie Database*. 28, 45, 56, 65, 148

KGE *Knowledge Graph Embedding*. 61, 72, 148

kNN *k-Nearest Neighbor*. 96, 148

MAE *Mean Absolute Error*. 62, 95, 97, 148

mAP@N *Mean Average Precision at N*. 62, 113, 148

ML1M *MovieLens 1M*. 96, 148

MO *Movie Ontology*. 43, 84, 148

nDCG@N *Normalized Discounted Cummulative Gain at N*. 62, 113, 118, 148

nll *Negative Loss Likelihood*. 116, 148

p@N *Precision at N*. 62, 113, 118, 148

PLN Processamento de Linguagem Natural. 46, 148

POU Perfil Ontológico de Usuário. 30, 35, 57, 129, 133, 148

PU Perfil de Usuário. 27, 49, 66, 148

r@N *Recall at N*. 62, 113, 118, 148

RC Representação de Conhecimento. 34, 148

RDF *Resource Description Framework*. 40, 148

RMSE *Root-Mean-Square Error*. 62, 95, 97, 148

SG Sumarização de Grafo. 31, 47, 74, 107, 130, 148

sGC Sumário de Grafo de Conhecimento. 33, 108, 148

SR Sistema de Recomendação. 27, 34, 48, 49, 63, 79, 148

SRC SR baseado em Conhecimento. 28, 48, 49, 63, 129, 148

SRGC SR baseado em Grafo de Conhecimento. 48, 49, 60, 65, 107, 120, 129, 148

SRO SR baseado em Ontologia. 48, 49, 57, 65, 79, 129, 148

SVD Singular Vector Decomposition. 148

LISTA DE SÍMBOLOS

$a \in A$	a é a aresta de um grafo e A é um conjunto de arestas
$c \in C$	c é um conceito, categoria ou classe e C é um conjunto de conceitos
$d \in D$	d é uma declaração RDF e D é um conjunto de declarações
$\langle s, p, o \rangle$	Declaração RDF na forma de tripla sujeito-predicado-objeto (respectivos s, p e o)
$e \in E$	e é uma entidade (conceito ou indivíduo) e E é um conjunto de entidades
$f \in F$	f é um fator de interesse e F é um conjunto de fatores de interesse
$G(N,A)$	Grafo de conhecimento
$H(E,D)$	Hierarquia de entidade
H_f	Hierarquia do fator de interesse f
$Int_u(H)$	Interações do usuário u organizadas (conectadas) pela hierarquia H
$freq(e,u,H)$	Frequência de ocorrência da entidade e em interações do usuário u conforme a hierarquia H
$i \in I$	i é um item a ser recomendado e I é um conjunto de itens
$ind \in Ind$	ind é um indivíduo de uma ontologia e Ind é um conjunto de indivíduos
$int \in Int$	int é uma interação usuário-item e Int é um conjunto de interações
$n \in N$	n é um nodo de um grafo e N é um conjunto de nodos
$p \in P$	p predicado de uma declaração RDF ou uma propriedade de uma ontologia e P é um conjunto de propriedades
$peso(f,u)$	Peso do fator de interesse f para o usuário u
$r \in R$	r é uma restrição de uma ontologia e R é um conjunto de restrições
$sn \in Sn$	sn é um supernodo e Sn é um conjunto de supernodos
$u \in U$	u é um usuário do sistema alvo da recomendação e U é um conjunto de usuários
$util_{i,u}$	$util$ é o valor de utilidade de um item i para um usuário u
$v_{e,u}$	Vetor de entidades ponderadas do usuário u
V	Conjunto de valores alfanuméricos representando informações sobre uma interação como tipo e intensidade

SUMÁRIO

1 INTRODUÇÃO

O uso de Sistemas de Recomendação (SRs) (AGGARWAL et al., 2016) tornou-se parte de nossa experiência *online* diária. Estes sistemas ajudam os usuários a encontrar, dentre muitos outros, itens de consumo (MESAS; BELLOGÍN, 2020), outros usuários (RODRíGUEZ-GARCíA et al., 2019), lugares (ZHENG et al., 2018) e anotações (YU et al., 2018). Ao longo dos anos, os SRs contribuíram para mitigar a sobrecarga de escolhas (*Choice overload*) (CHERNEV; BöCKENHOLT; GOODMAN, 2015), aumentando a satisfação dos usuários e a receita de muitos negócios de *streaming* e *e-commerce*, incluindo Netflix, Amazon, Youtube e Last.fm.

Para prover recomendações personalizadas, os SRs apoiam-se em um modelo de preferência que permite estimar interesses dos usuários, chamado modelo de Perfil de Usuário (PU, em inglês *User Profile*). O PU mantém informações diretamente ou indiretamente relacionadas às preferências do usuário, ao seu comportamento e ao seu contexto (KADIMA; MALEK, 2010; KANOJE; GIRASE; MUKHOPADHYAY, 2014).

Existem diversas técnicas de recomendação propostas na literatura. Elas podem ser classificadas segundo três abordagens clássicas: técnicas de Filtragem Colaborativa (FC) que utilizam um PU na forma de uma matriz de avaliação Usuário-Item mantendo as avaliações (*ratings*) dos itens acessados pelos usuários; técnicas de Filtragem Baseada em Conteúdo (FBC), que utiliza um PU mantendo características dos itens bem avaliados pelos usuários; e técnicas híbridas, que combinam técnicas de FC e FBC para minimizar os problemas do uso individual das duas primeiras.

Dentre outros problemas, as técnicas de recomendação clássicas podem perder qualidade devido à falta de dados de avaliações, a chamada esparsidade de dados (BOBADILLA; SERRADILLA, 2009). Esta esparsidade de dados faz com que as técnicas de FC produzam recomendações de baixa qualidade para os usuários que fornecem poucas avaliações. As técnicas de FBC tentam mitigar o problema da esparsidade de dados de avaliação através do uso de características dos itens (mesmo daqueles que nunca foram avaliados).

Existem diversos trabalhos na literatura que visam mitigar o problema da esparsidade de dados em SRs. Um exemplo é a solução proposta por Fernandes, Sacenti e Willrich (2017), que teve a participação do autor desta tese. Nela, foi definida uma técnica híbrida inspirada na técnica de FC clássica, que utiliza informação lateral para a determinação dos vizinhos próximos. Estes vizinhos não são determinados pela similaridade entre suas avaliações, mas pela similaridade da frequência de ocorrência de valores de certas características relacionadas aos itens (p.ex., no domínio de filmes: gêneros e ano de lançamento).

Além das técnicas clássicas de recomendação, Aggarwal et al. (2016) descreve outros SRs, como os baseados em conhecimento, demográficos, baseados em contexto, sensíveis ao tempo, baseados em localização, sociais, multicritérios, baseado em combinação de resultados (em inglês, *ensemble*) e estruturais. Diferente de Aggarwal et al. (2016), que considera como SR baseado em conhecimento apenas aqueles que explicitamente solicita os requisitos do usuário para recomendar determinado item, Burke (2007) define SRs baseados em Conhecimento

(SRCs) de modo mais abrangente: como sistemas que apoiam-se no conhecimento explícito do domínio dos itens e dos usuários para determinar como os itens satisfazem as necessidades do usuário. Os SRCs incorporam este conhecimento adicional nos PUs por meio de diversas abordagens e artefatos (BOBADILLA et al., 2013): raciocínio baseado em casos (*Case-Based Reasoning*), restrições (*Constraint-Based Reasoning*), consultas, métricas de alinhamento (*Matching Metrics*), redes sociais (SHOKEEN; RANA, 2020), vetores de conhecimento, ontologias e Grafos de Conhecimento (GCs).

De maneira similar às técnicas de FBC, as informações sobre usuários e/ou itens consideradas pelo SRC têm o potencial de mitigar o problema da esparsidade de dados. Estas informações, também chamadas de informações laterais (em inglês, *side information*), complementam os dados sobre as interações entre usuários e itens, permitindo novas abordagens para a recomendação. As informações laterais podem ser coletadas no sistema alvo da recomendação (p.ex., valores de metadados de filmes, livros ou músicas) ou em fontes externas, como dados conectados (DCs, como p.ex., DBpedia, Freebase, LinkedMDb), dados (semi)estruturados de domínios específicos como filmes (p.ex., Internet Movie Database - IMDb, Rotten Tomatoes) e mídias sociais (p.ex., Facebook Graph, Twitter). Esta tese tem como objeto de estudo SRC segundo a definição de Burke (2007) e no contexto de informação lateral.

As fontes externas, e o próprio sistema alvo da recomendação, podem oferecer uma quantidade imensa de informações sobre usuários e itens, podendo estar relacionadas aos mais diversos aspectos dos usuários (p.ex., informações demográficas) e dos itens (no domínio de filme, pode-se citar gêneros, atores, diretores, data de lançamento). Um dado SRC pode selecionar os tipos de informações que irão compor o PU, que idealmente deveriam ser aqueles que os usuários frequentemente levam em conta quando buscam pelos itens (ADOMAVICIUS; KWON, 2007; MANOUSELIS; COSTOPOULOU, 2007). Além da seleção dos tipos de informações que irão compor o PU, os SRCs devem escolher como estas informações serão representadas e utilizadas pelo SR. Em alguns SRCs, as informações laterais relacionadas a um ou mais aspectos dos itens são organizadas na forma de hierarquias (taxonomias). Por exemplo, Sieg, Mobasher e Burke (2010) e Pan et al. (2010) consideram como informação lateral uma hierarquia de gêneros de livros e filmes, respectivamente. Outros SRCs representam informações laterais na forma de ontologias e Grafos de Conhecimento, (GCs) os chamados SRs baseados em Ontologia (SRO) e SRs baseados em GCs (SRGCs).

Ontologia é uma especificação explícita e formal de uma conceitualização compartilhada (STUDER; BENJAMINS; FENSEL, 1998). Uma ontologia especifica conceitos (classes), indivíduos (instâncias), propriedades (relacionamentos) e axiomas (declarações), que permitem representar elementos de determinado domínio. As declarações de uma ontologia descrevem fatos do mundo real por meio de uma estrutura bem definida no formato de triplas $\langle s, o, p \rangle$ onde s é o sujeito, p é o predicado e o é o objeto da declaração. Por exemplo, a declaração $\langle \textit{Terminator: Dark Fate}, \textit{temTipo}, \textit{Filme} \rangle$ evidencia que o indivíduo *Terminator: Dark Fate* está relacionado ao conceito *Filme* por meio da propriedade *temTipo* (tipo).

Não há ainda amplo consenso na definição de GC (EHRLINGER; WÖSS, 2016). Mui-

tos autores consideram que GC (p.ex., Google's Knowledge Graph) enfatiza a estrutura na forma de um grafo rotulado cujos nodos denotam entidades de diferentes tipos (p.ex., *Usuário*, *Filme*, *Gênero*, *Diretor*, *Ator*) e as arestas denotam relações entre eles (p.ex., *temTipo*, *avalia*, *assiste*, *temGênero*, *temDiretor*, *temAtor*). O GC pode conter uma ou mais ontologias que formam um esquema parcial ou integral dos conceitos, indivíduos e das relações por ele conectados.

A publicação de ontologias ou GCs na *Web* seguindo um conjunto de diretrizes definidas e estudadas pela área da Web Semântica dá origem ao conceito de dados conectados (BERNERS-LEE et al., 1998b). Este tipo de representação de conhecimento potencializa o compartilhamento e o reuso dos dados representados. Desta forma, é adequado o uso de dados conectados (p.ex., DBpedia, Freebase, YAGO) como uma fonte de informação lateral adicional sobre usuários e itens de um SR.

Tanto propostas de SROs (PASSANT; HEITMANN; HAYES, 2009; NOIA et al., 2012; OSTUNI et al., 2013; BELLINI et al., 2017; ANGELIS et al., 2017) quanto SRGCs (MIRIZZI et al., 2012; CAO et al., 2019; SUN et al., 2018) reusam conhecimento oriundo de fontes externas como dados conectados ou estruturados. Por exemplo, um SR de filmes pode explorar dados da DBpedia sobre gêneros, diretores, atores, palavras-chave e prêmios de filmes, atores e diretores. Outro exemplo são os SRs que exploram dados de mídia social, como perfis de usuário, relações sociais, anotações e postagens (SHOKEEN; RANA, 2020).

Diversos trabalhos recentes demonstram que os SRCs são eficazes, ou seja, que a informação lateral tem potencial para melhorar a qualidade de recomendações. No entanto, estas técnicas em geral sofrem do problema da baixa eficiência. O termo eficiência de um SR é usado aqui para se referir principalmente ao custo computacional e ao tempo necessário para treinar o modelo de PU necessário à geração da recomendação. Apesar de haver poucos estudos desta limitação dos SRCs (PAUN, 2020), esta baixa eficiência é devida principalmente à incorporação da informação lateral que, acrescida dos dados acerca das interações, geram um grande volume de dados a serem analisados pelas técnicas baseadas em conhecimento. Além disso, entre os dados laterais adicionados pode haver dados irrelevantes (supérfluos) ou até mesmo ruidosos para a recomendação. O problema do volume e relevância da representação de conhecimento tem grande importância para SRs reais porque a atualização do modelo de preferência deve ser executada tão frequentemente quanto possível para considerar as últimas interações de usuários e novas adições ao catálogo de itens. Portanto, o treinamento do SR deve ter execução rápida e robusta para produzir modelos de preferência atualizados com frequência e rapidez.

Esta tese estuda a representação do conhecimento em sistemas de recomendação baseados em conhecimento, especificamente SROs e SRGCs e seus impactos na eficácia e eficiência. Mais especificamente, esta tese investiga duas abordagens para mitigar o problema do alto custo de treinamento de modelos de recomendação baseados em informações laterais.

A primeira abordagem (SACENTI; WILLRICH; FILETO, 2018) converte a representação das informações laterais baseada em ontologia numa matriz de preferência, eliminando a necessidade do uso de algoritmos baseados em inferências em ontologias ou na estrutura de

rede de informação. Esta abordagem é aplicável a SRs baseados em filtragem híbrida (FERNANDES; SACENTI; WILLRICH, 2017) para considerar as informações laterais no processo de recomendação. Para definir melhor esta técnica de conversão, esta tese propõe um arcabouço conceitual, chamado de ORBS. Este arcabouço organiza o conhecimento em ontologias de diferentes níveis de abstração, diferenciando conceitos relacionados à tarefa de recomendação, ao domínio do item e à aplicação. A ontologia de tarefa de recomendação é genérica, de maneira a não depender de conceitos do domínio do item. Assim, a ontologia de tarefa de recomendação pode ser reusada em SRs aplicados a diferentes domínios (tipos) de item, p.ex. filmes, músicas, produtos e pessoas. Por consequência, um SR projetado para considerar esta ontologia de tarefa, pode ser facilmente adaptado a diferentes domínios dos itens, bastando importar a(s) ontologia(s) do domínio do item a recomendar. Nesta tese, a capacidade de reuso ou adaptação de um SR em aplicações com itens de diferentes domínios é chamada de independência de domínio.

Além disso, este arcabouço permite especificar a representação ontológica das preferências de usuários, chamado de Perfil Ontológico de Usuário (POU), descrevendo os diferentes aspectos (características) dos itens e as hierarquias de entidades que ordenam e descrevem estes aspectos. Por exemplo, a data de lançamento do filme e os gêneros do filme são aspectos adotados pela construção de POUs para representar fatores de interesse do usuário que influenciam no processo de recomendação. Além disso, gêneros do filme podem ser organizados em uma taxonomia de gêneros. Este arcabouço permite analisar diferentes representações de informações laterais sobre itens e usuários, que especificam quais aspectos e hierarquias são relevantes para a recomendação, com o objetivo de identificar qual representação proporciona maior eficácia aos algoritmos de recomendação.

Experimentos demonstram que a especificação de múltiplos aspectos e hierarquias tem o potencial de melhorar a eficácia de SRs (SACENTI; WILLRICH; FILETO, 2018) e que o aumento da generalidade dos conceitos diminui o erro de predição e aumenta o número de vizinhos próximos (FERNANDES; SACENTI; WILLRICH, 2017). Além disso, os resultados obtidos motivaram a segunda abordagem proposta sobre a sumarização da representação de conhecimento baseada em GCs.

Na primeira abordagem, tentou-se reduzir o custo computacional via um mapeamento da ontologia na matriz de preferência, eliminando altos custos computacionais para o treinamento que consideram ontologias. Outra abordagem alternativa para mitigar o problema da eficiência dos SRCs é via a redução do volume de dados e da eliminação de dados irrelevantes e ruidosos. Trabalhos que optam pela seleção manual ou pela seleção semiautomática de dados conectados geralmente restringem a seleção às triplas diretamente relacionadas aos itens a recomendar. Alguns trabalhos recentes investigam a seleção automática de características de dados conectados (MUSTO et al., 2015; RAGONE et al., 2017). Outros trabalhos (GARCIA et al., 2012; ARSHADI; JURISICA, 2004; MAATEN; POSTMA; HERIK, 2009; LIU; CHENG; QU, 2020) reduzem o tamanho e o ruído dos dados de entrada de algoritmos de Aprendizado de Máquina (AMs – em inglês, *machine learning*) através de técnicas como a poda de instâncias

(*instance prunning*), edição baseada em casos (*case-based editing*), redução de dimensionalidade, Agrupamento de Dados (AD – em inglês, *clustering*) e abordagens de Sumarização de Grafo (SG).

Os algoritmos de SG (LIU et al., 2018) podem transformar grafos em representações mais compactas, preservando propriedades que são úteis para a aplicação ou domínio. Alguns trabalhos (SYDOW; PIKUŁA; SCHENKEL, 2013; ALI et al., 2020) aplicam a SG para acelerar o processamento de consultas em grafos e oferecer suporte à visualização e análise de grafos. Esses trabalhos apontam que a SG tem vários benefícios, incluindo redução do volume de dados, aceleração de algoritmos ou consultas em grafos e redução de ruído. Embora a SG tenha se tornado popular para esses propósitos, sua aplicação para sumarizar GCs que representam informações laterais em SRGCs é relativamente nova e inexplorada. Até onde sabemos, poucos trabalhos (RAGONE et al., 2017; WU et al., 2015) aplicaram a SG em certas tarefas de um SR específico, e nenhum estudo propôs abordagens de SG e avaliou seus impactos na redução de custos computacionais e/ou melhoria da qualidade de recomendação em SRGCs.

Um algoritmo de SG também pode se beneficiar da representação de informação lateral no GC em termos de aspectos e hierarquias por meio da delimitação de subgrafos, chamados nesta tese de visões. Métodos recentes de agrupamento baseados em grafos exploram estratégias multi-visão para aproveitar a diversidade, a precisão e a robustez das partições (Yang; Wang, 2018). Em tais métodos, subgrafos com diferentes pontos de vista sobre os mesmos dados são primeiro fundidos e depois aglutinados. Outras abordagens tentam encontrar uma maneira de maximizar a qualidade do agrupamento dentro de cada visão enquanto mantém a consistência do agrupamento através das diferentes visões.

A segunda abordagem proposta nesta tese para mitigar o problema da eficiência foca especificamente os SRGCs e tem como princípio a redução dos GCs representando infomações laterais via SG. Para tal, é proposto um método de GC combinando *embeddings* com clusterização de nodos usando o algoritmo *K-Means*. Este método de SG é usado em uma etapa de pré-processamento para SRGCs. Dado que os algoritmos SG são específicos para aplicações, propomos um método de SG que visa a redução do tamanho do GC para acelerar o tempo de treinamento dos SRGCs sem impacto significativo na eficácia. O método GS proposto adota a premissa de que a similaridade semântica entre entidades pode ser usada para sumarizar GCs para SRs através da aglutinação de entidades, onde entidades semanticamente similares são aglutinadas formando nodos mais genéricos, denominamos de supernodos.

Além disso, o método de SG proposto adota duas estratégias quanto aos tipos de entidades que podem ser aglutinadas em supernodos: a sumarização única-visão e a multi-visão. Na estratégia única-visão, o GC é sumarizado como um todo, de forma que os supernodos podem aglutinar entidades similares independente de qualquer forma de categorização dessas entidades.

A Figura 1 ilustra a estratégia única-visão aplicada à recomendação de filmes. O lado esquerdo desta figura representa os usuários interagindo com filmes. No centro estão representadas em GC das informações laterais relacionadas os filmes. Finalmente, no lado direito

contém uma ilustração do GC sumarizado. A título de exemplo, considere que os gêneros *War* e *Drama*, e o ator *S. Stallone* são mais similares entre si do que com as demais entidades, que faz com que estas entidades sejam aglutinadas em um supernodo (*SN-2* na Figura 1). Assim, os filmes relacionados com essas entidades tornam-se mais próximos entre si (por exemplo, *The expendables 3*, *Braveheart*) por causa da nova representação de conhecimento. Analogamente, o ator *A. Schwarzenegger* e o diretor *T. Miller* são aglutinados em *SN-3* devido a sua similaridade. Do mesmo modo, a estratégia de única-visão também permite aglutinar entidades homogêneas, como os gêneros *Action* e *Adventure* (*SN-1*), ou os diretores *P. Hughes* e *M. Gibson* (*SN-4*).

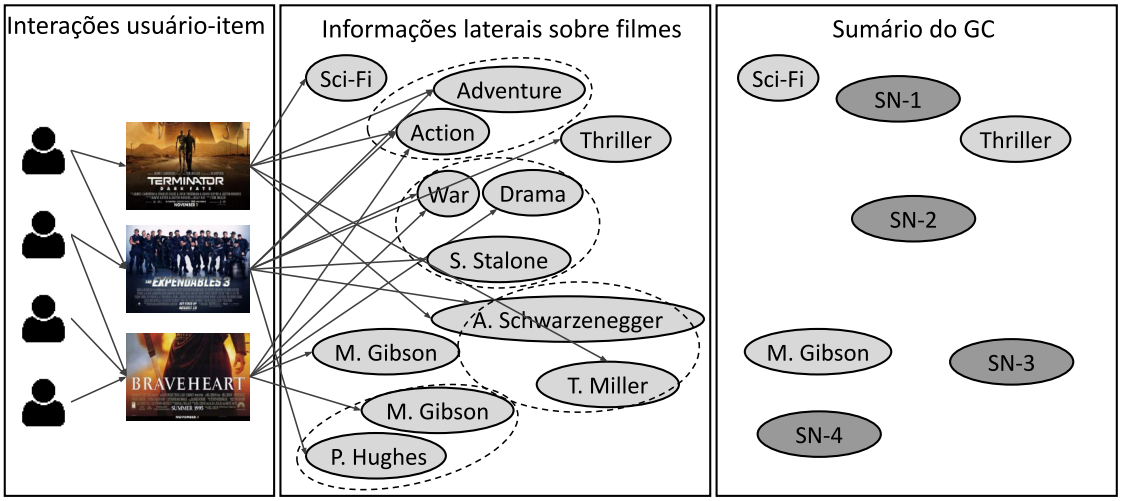

Figura 1 – Exemplo de sumarização de GC multi-visão para recomendação de filmes baseada em GC

Fonte: criado pelo autor. Publicado pela primeira vez em Journal of Intelligent Information Systems, 2021 pela Springer Nature.

Visando propor uma alternativa mais restritiva para a criação de supernodos, esta tese propõe a estratégia multi-visão de geração de supernodos. Nesta estratégia, o GC é segmentado/particionado em um número de visões distintas (subgrafos). Cada visão representa um aspecto (faceta, subconjunto) da informação lateral sobre os itens a serem recomendados. A Figura 2 ilustra a estratégia multi-visão de SG aplicada à recomendação de filmes baseada em GC. Nesta estratégia, o conteúdo do GC pode ser organizado em três visões (subgrafos) distintas: *Gêneros*, *Atores* e *Diretores*. Cada visão pode ser sumarizada de forma independente e, posteriormente, os sumários obtidos são combinados em um único GC, no qual apenas entidades homogêneas (considerando a visão pertencente) podem ser aglutinadas em um mesmo supernodo com base na similaridade semântica entre si. Diferente da estratégia única-visão, a multi-visão não permite a formação de um supernodo com entidades heterogêneas. Por exemplo, na visão de gênero, *Ação* e *Aventura* são aglutinadas no supernodo *SN-1* representando esses dois gêneros, devido a estes gêneros serem mais similares entre si do que aos demais gêneros. Assim, os filmes pertencentes a esses gêneros tornam-se mais próximos (por exemplo, *Terminator: Dark Fate*, *The expendables 3*, *Braveheart*). Analogamente, na visão dos atores *A.*

Schwarzenegger e *S. Stallone* são aglutinados em *SN-3*. Finalmente, na visão dos diretores, *P. Hughes* e *M. Gibson* são aglutinados em *SN-4*.

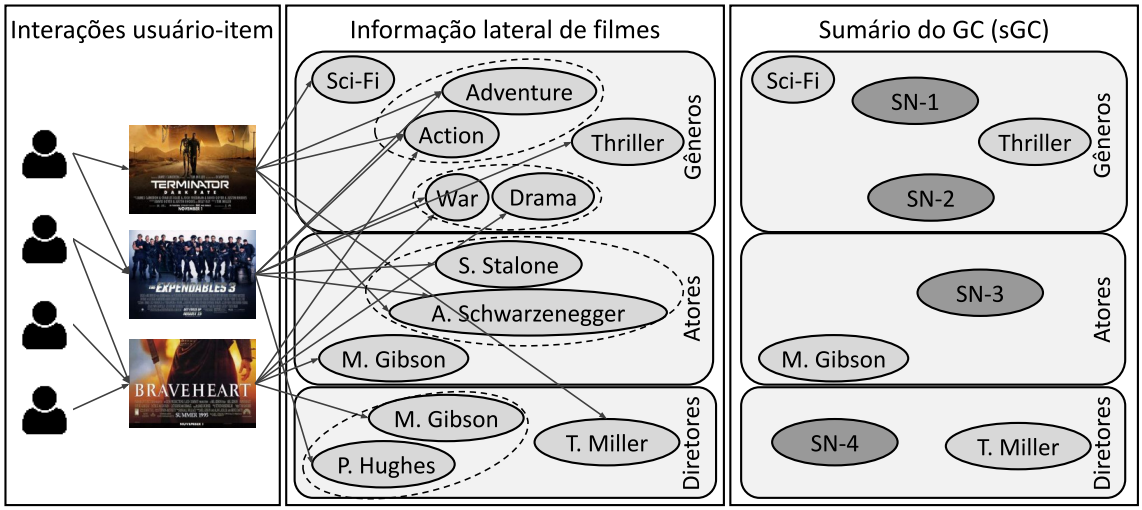

Figura 2 – Exemplo de sumarização de GC multi-visão para recomendação de filmes baseada em GC

Fonte: criado pelo autor. Publicado pela primeira vez em Journal of Intelligent Information Systems, 2021 pela Springer Nature.

Note que a SG aplicada ao SRGC proposto não pode ser considerada uma sobrecarga de trabalho adicional ao SR. No contexto de SRGCs, a frequência de sumarização do GC é menor do que a do retreinamento do modelo de recomendação. O retreino deve ser executado com a maior frequência possível para levar em consideração os novos dados de interação do usuário. A inserção de novas informações laterais no GC ou de itens a serem recomendados costuma ser muito menos frequente do que a inserção de novas interações do usuário. Portanto, a sumarização do GC pode ser executada apenas uma vez para produzir um sumário do GC (sGC) que pode ser usado para retreinar o modelo de preferência várias vezes, capturando as novas interações. Assim, a sumarização como etapa de pré-processamento não pode ser considerada como uma sobrecarga adicional no retreino. Ao contrário, espera-se que o pré-processamento do GC reduza o tempo de treinamento geral dos modelos de SRGCs, enquanto mantém a eficácia do SR ou até mesmo melhora-a ligeiramente em alguns casos.

Para avaliar a segunda abordagem proposta para mitigar o problema da eficiência em SRGCs via SG, foram realizados experimentos considerando quatro SRGCs (citar referências) e dois conjuntos de dados obtidos pela combinação do conjunto de dados do Movielens 1M enriquecido com informações laterais obtidas da DBpedia e IMDb. Estes experimentos visaram avaliar os impactos em termos de eficácia e eficiência da SG quando aplicada nesses quatro diferentes SRGCs. Os resultados mostram que a sumarização do GC pode aumentar a eficiência do SRGC pela redução do tempo de treinamento do modelo sem mudanças significativas na eficácia da recomendação, que varia de acordo com a estratégia, a taxa de sumarização e o SR.

O restante deste capítulo descreve a hipótese e as perguntas de pesquisa, os objetivos geral e específicos, o escopo da pesquisa, o método de pesquisa e a organização da tese.

1.1 HIPÓTESE E PERGUNTAS DE PESQUISA

Observado a diversidade de representação de informação lateral e seus impactos em termos de eficácia e eficiência em SRCs, esta seção descreve a hipótese (Hipótese 1) e as perguntas de pesquisa.

Hipótese 1. *Múltiplos aspectos, hierarquias e visões para representar e selecionar informação lateral em grafos de conhecimento e aliados à sumarização destes podem melhorar a eficiência do treinamento e a eficácia de sistemas de recomendação baseados em grafos.*

Esta tese se propõe a responder às seguintes perguntas de pesquisa relacionadas com tal hipótese:

1. A representação de informação lateral por meio de características relacionadas aos múltiplos aspectos do item resulta na melhoria da eficácia dos resultados de SRs?
2. A representação de informação lateral por meio de características organizadas em taxonomias ou ontologias hierárquicas resulta na melhoria da eficácia dos resultados de SRs?
3. A sumarização de informação lateral resulta na melhoria da eficiência do treinamento sem prejuízo na eficácia dos resultados de SRs?
4. A estratégia de sumarização multi-visão resulta na melhoria da eficiência do treinamento e/ou eficácia dos resultados de SRs, quando comparada com a estratégia de única-visão?

1.2 OBJETIVOS

O tema de pesquisa desta tese é a representação e sumarização de conhecimento em SROs e SRGCs. Desta forma, a tese está relacionada às linhas de pesquisa de Sistemas de Recomendação (SR – em inglês, *recommender systems*), Representação de Conhecimento (RC – em inglês, *knowledge representation*), Sumarização (em inglês, *Summarization*) através de Agrupamento de Dados (AD – em inglês, *clustering*) e Aprendizado de Máquina (AM – em inglês, *machine learning*). Esta seção apresenta os objetivos geral e específicos dentro destas linhas de pesquisa.

1.2.1 Objetivo Geral

O objetivo geral desta tese é investigar a representação de conhecimento baseada em ontologia e GC que será utilizada como entrada do treinamento do modelo de preferência de SRCs, tendo como meta melhorar a eficiência e/ou a eficácia da recomendação.

Com o intuito de identificar os impactos da representação das informações laterais em SRs, é necessária a comparação de diferentes representações de conhecimento. Nesta tese, avaliamos os impactos causados pela variação da representação de conhecimento baseada na

reorganização de características de itens (em termos de aspectos e hierarquias de entidades) e baseada na sumarização. É importante observar que esta tese pretende avaliar a representação do conhecimento de SR, i.e. contribuir para o pré-processamento dos dados de entrada do treinamento do modelo de recomendação, e não pretende propor uma nova técnica de recomendação.

1.2.2 Objetivos Específicos

São estabelecidos os seguintes objetivos específicos para esta tese:

1. Identificar ou definir uma técnica de recomendação que determine as preferências dos usuários considerando diferentes configurações de informação lateral em um modelo de Perfil Ontológico de Usuário (POU), em termos das características organizadas em aspectos e hierarquias. Se necessário, definir um mapeamento do POU para modelos de PU empregados por técnicas clássicas de recomendação, como a já inicialmente investigada por Fernandes, Sacenti e Willrich (2017).

2. Definir e desenvolver uma arquitetura de implementação de um SRO com base em POU e na técnica de recomendação como definida no objetivo específico 1.

3. Definir e realizar experimentos utilizando o SRO desenvolvido. Estes experimentos deverão ser definidos de maneira averiguar as perguntas de pesquisa 1 e 2 desta tese.

4. Identificar ou definir uma técnica de SG para reduzir o tamanho do GC e eliminar dados irrelevantes e ruidosos.

5. Definir e desenvolver uma arquitetura de implementação de um SRGC que adote a técnica de SG no pré-processamento dos dados de entrada do SR.

6. Definir e realizar experimentos. Estes experimentos deverão ser definidos de maneira averiguar as perguntas de pesquisa 3 e 4 desta tese.

1.3 ESCOPO DA PESQUISA

Esta tese investiga especificamente a representação de conhecimento em modelos de PU de SROs e SRGCs. A adoção exclusiva destas duas categorias de SRs se deve ao conhecimento prévio do autor sobre tecnologias da web semântica e às abordagens recentes de recomendação baseada em GCs(CAO et al., 2019; SUN et al., 2018; ZHANG et al., 2018; ZHANG et al., 2016; PIAO; BRESLIN, 2018). Outras técnicas de recomendação como as baseadas em casos, restrições, consultas, métricas de alinhamento, redes sociais e vetores de conhecimento não são abordadas nesta tese. A representação baseada em ontologias e GCs é mais rica e complexa, de construção e processamento mais custosos, visto que explicita os relacionamentos entre as entidades. Nem sempre esta representação obtém a melhor eficácia (SHERIDAN

et al., 2019). Entretanto, esta representação pode ser mais adequada em cenários específicos, assim como otimizações e novas formas de uso podem superar as limitações atuais e melhorar a qualidade das recomendações.

O problema superespecialização (*Over-Specification*) (AGGARWAL et al., 2016), potencializado pela incorporação de informação lateral não é investigado nesta tese. Porém, métricas de diversidade podem ser adotadas para averiguar este fenômeno. Do mesmo modo, esta tese não investiga o problema da análise de conteúdo limitado (MUSTO et al., 2017), embora a integração do modelo de preferência com outras fontes externas de dados possa ser apontada como uma possível solução. Tal integração inclui o enriquecimento com dados conectados, realizados em Sacenti (2016) e Cao et al. (2019), ou dados estruturados, como em Sun et al. (2018). Além disso, embora a representação de conhecimento por meio de ontologia favoreça a independência de domínio do item ao SR, a avaliação desta contribuição também não é um objetivo desta tese, apesar de ser discutida em Sacenti, Willrich e Fileto (2018).

Dentre as técnicas para redução do tamanho e eliminação de ruído em GCs, esta tese limita seu escopo na SG, especificamente na SG baseada no uso do conceito de supernodos. Outras técnicas baseadas em poda de instâncias (*instance prunning*), edição baseada em casos (*case-based editing*), redução de dimensionalidade não serão investigados aqui.

O método de GC proposto nesta tese utiliza o K-Means como algoritmo de agrupamento de entidades no processo de geração de supernodos. O algoritmo K-Means foi adotado por ser uma técnica bem consolidada, porém indica-se que outras técnicas sejam avaliadas em trabalhos futuros. De modo análogo, não foi objetivo desta tese identificar a técnica de *embedding* mais adequada para a sumarização proposta. A técnica ComplEx foi escolhida por ter obtido resultados mais promissores em comparação com o TransE durante experimentos preliminares, porém indica-se a extensão desta avaliação em trabalhos futuros.

Finalmente, ressalta-se aqui que esta tese não tem como objetivo propor uma técnica de recomendação baseada em GC inédita. O objetivo aqui é propor técnicas de pré-processamento do GC que podem ser aplicadas em SRGCs existentes.

1.4 MÉTODO DE PESQUISA

Para atingir seus objetivos específicos, e segundo a classificação de pesquisa em ciência da computação (WAZLAWICK, 2015), esta tese propõe realizar uma pesquisa: (i) empírica (não formal), cujos objetos de estudo são os SRCs, especificamente SROs e SRGCs; (ii) aplicada (não pura), que estuda como representar a informação lateral no modelo de preferência destes SRs; (iii) exata, que avalia seus resultados através de métricas precisas (p.ex., erro, precisão e cobertura para avaliar a eficácia da recomendação; tempo em segundos para avaliar a eficiência de treinamento).

O método desta pesquisa é composto pelas etapas seguintes, sendo que as etapas de 1 à 6 são referentes as perguntas de pesquisa 1 e 2, enquanto que as etapas de 7 à 12 são referentes as perguntas de pesquisa 3 e 4:

Etapa 1. Revisão da literatura sobre SRs baseados em taxonomia, SROs e SRs baseados em dados conectados, e a análise de trabalhos relacionados.

Etapa 2. Desenvolvimento da técnica de recomendação e/ou função de mapeamento do POU para o modelo de PU empregado por uma técnica clássica de recomendação, como em (FERNANDES; SACENTI; WILLRICH, 2017).

Etapa 3. Desenvolvimento de um SRO baseado em POU e na técnica de recomendação adotada.

Etapa 4. Planejamento e realização de experimentos com o SRO desenvolvido na Etapa 3, utilizando os dados disponíveis em conjuntos de dados (*Datasets*) oriundos de projetos como o MovieLens 1M[1] (HARPER; KONSTAN, 2015). Para o enriquecimento semântico, serão consideradas as fontes de dados externas, como a MovieOntology[2].

Etapa 5. Avaliação dos resultados obtidos nos experimentos realizados durante a Etapa 4.

Etapa 6. Desenvolvimento e submissão de um artigo com resultados obtidos pela Etapa 5.

Etapa 7. Pesquisa do estado da arte sobre SRGCs e SG e a análise de trabalhos relacionados.

Etapa 8. Desenvolvimento das estratégias única-visão e multi-visão para a SG aplicada em SRGCs.

Etapa 9. Identificação e implantação de SRGCs incorporando a SG como etapa de pré-processamento.

Etapa 10. Planejamento e realização de experimentos com os SRGCs identificados na Etapa 9, utilizando os dados disponíveis em conjuntos de dados (*Datasets*) criados em projetos anteriores (CAO et al., 2019; SUN et al., 2018), que combinam dados de projetos como o MovieLens com fontes de dados conectados, como a DBpedia[3], e fontes de dados estruturados, como o IMDb[4].

Etapa 11. Avaliação dos resultados obtidos nos experimentos realizados durante a Etapa 10.

Etapa 12. Desenvolvimento e submissão de um artigo com resultados obtidos pela Etapa 11.

O restante deste trabalho é estruturado em 6 capítulos. O Capítulo 2 contém breve fundamentação teórica sobre representação de conhecimento, especificamente ontologias, taxonomias, GCs, dados conectados e *embeddings* de GC, e sobre manipulação de conhecimento, especificamente a sumarização de GC. O Capítulo 3 apresenta a fundamentação sobre SRs clássicos, SROs e SRGCs. O Capítulo 4 descreve e analisa os trabalhos relacionados aos SRs baseados em taxonomias, SROs, SRs baseados em dados conectados, SRGCs e à sumarização de GCs. O Capítulo 5 apresenta a primeira abordagem para mitigar o problema da eficiência de SRs baseados em Filtragem híbrida via a conversão de ontologia em matriz de preferência. As principais contribuições são uma técnica de formação de vizinhança baseada nas características

[1] Acesso: https://movielens.org/, em: 22/11/2021.
[2] Acesso: http://www.movieontology.org:80/2010/01/movieontology.owl, em: 14/06/2018.
[3] Acesso: https://wiki.dbpedia.org/, em: 22/11/2021.
[4] Acesso: https://datasets.imdbws.com, em: 22/11/2021.

de itens (FERNANDES; SACENTI; WILLRICH, 2017) e um arcabouço conceitual para desenvolvimento e aperfeiçoamento de SROs (SACENTI; WILLRICH; FILETO, 2018). O Capítulo 6 apresenta a segunda abordagem para mitigar o problema de eficiência de SRGCs via a sumarização de GCs (SACENTI; FILETO; WILLRICH, 2021). Este capítulo detalha as técnicas de SG propostas para SRGCs, bem como resultados dos experimentos. O Capítulo 7 apresenta as conclusões desta tese, detalhando as contribuições, os resultados obtidos, as limitações e os trabalhos futuros.

2 REPRESENTAÇÃO E MANIPULAÇÃO DE CONHECIMENTO

Este capítulo apresenta a fundamentação teórica sobre representação e manipulação de conhecimento necessária ao entendimento desta tese. Serão apresentados conceitos gerais na área de ontologias, taxonomias, dados conectados (DCs), Grafos de Conhecimento (GCs), *embeddings* de GC e sumarização de GC.

2.1 ONTOLOGIAS, TAXONOMIAS E DADOS CONECTADOS

Ontologia é tradicionalmente definida como uma especificação explícita e formal de uma conceitualização compartilhada (STUDER; BENJAMINS; FENSEL, 1998). Os requisitos desta definição podem ser esclarecidos da seguinte maneira (FREITAS, 2003; SILVA, 2015):

- **Conceitualização**: representa um modelo abstrato de determinada área do conhecimento, onde os conceitos relevantes e suas relações são identificados.
- **Explícita**: os elementos e restrições são claramente definidos.
- **Formal**: estrutura bem definida, tornando possível o processamento automático da ontologia (compreensível por agentes humanos e computacionais).
- **Compartilhada**: a ontologia utiliza conhecimento consensual em sua modelagem, i.e. aceito por um grupo de pessoas.

Em termos de conceitualização, uma ontologia define conceitos (categorias ou classes), indivíduos (instâncias), propriedades (relacionamentos) e axiomas (declarações), que representam elementos de determinado domínio (universo de discurso) para um grupo de pessoas em consenso. Para ilustrar estes termos, considere a Figura 3 que representa um exemplo de ontologia baseada na ontologia FOAF[1]. Este exemplo de ontologia especifica três pessoas que se conhecem (Fernandes, Sacenti, Willrich), seus interesses em tópicos (Ontology, Recomender System) e de um documento relacionado a esses tópicos (OBRS:YAS). O conhecimento descrito neste exemplo é separado em dois níveis: intencional e extensional. No nível intencional, os conceitos (elipses roxas, nomes em negrito) *owl:Thing*, *foaf:Person* e *foaf:Document* representam respectivamente o entendimento abstrato de coisa, pessoa e documento. A propriedade *rdfs:subclassof* indica que *Person* e *Document* são subclasses de *Thing*. No nível extensional, os indivíduos (elipses verdes) *ex:Person1*, *ex:Person2* e *ex:Person3* são três pessoas, *ex:Topic1* e *ex:Topic2* são duas coisas, e *ex:Document1* um documento. A propriedade *rdf:type* indica a qual conceito um indivíduo é categorizado, assim como a propriedade *foaf:knows* relaciona pares de pessoas que se conheçam, a propriedade *foaf:topic_interest* indica um tópico que uma pessoa tem interesse, *foaf:topic* indica um tópico de um documento, e *foaf:name* e *foaf:lastName* determinam nomes dos indivíduos (literais representados por retângulos azuis).

[1] Acesso: http://xmlns.com/foaf/spec/, em: 22/11/2021.

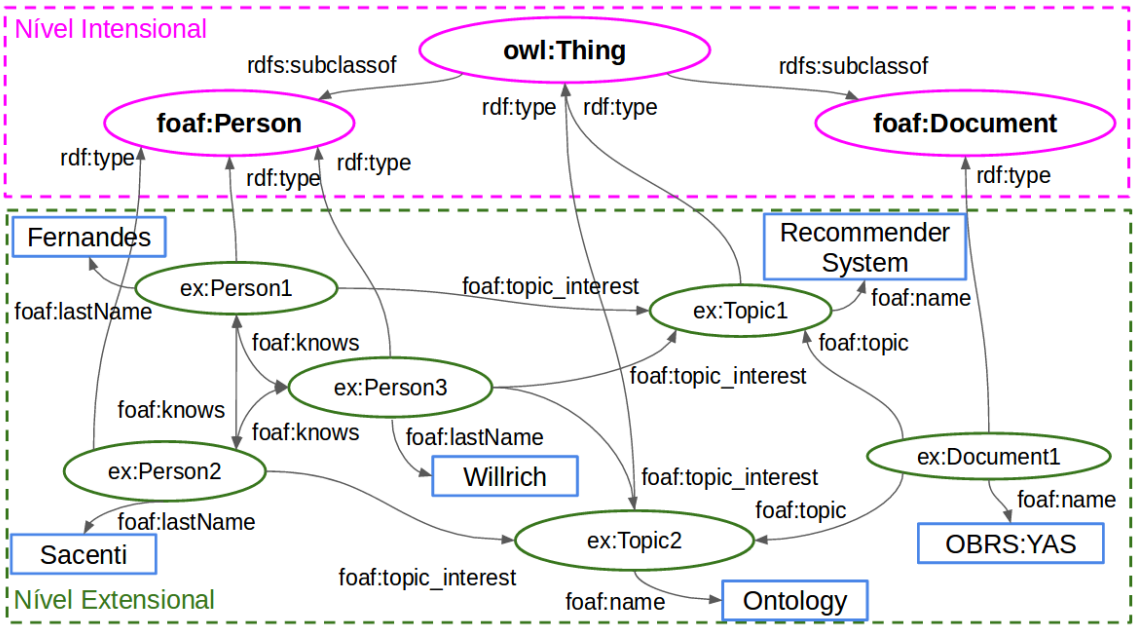

Figura 3 – Exemplo de ontologia

Fonte: criado pelo autor.

Algumas das vantagens da aplicação de tecnologias de representação semântica originam-se na representação formal do conhecimento que permite desambiguação e clareza na definição de conceitos. Além disso, é possível reusar representações semânticas e há interoperabilidade de representações semânticas entre diferentes aplicações.

Ontologias são descritas por meio de formatos para representar conhecimento derivados de formalismos como a lógica descritiva e empregados na *Web* Semântica. O propósito principal destas tecnologias é permitir que agentes computacionais (máquinas) possam participar e auxiliar a comunicação de agentes humanos via internet (BERNERS-LEE et al., 1998b). O núcleo do formato representacional destas tecnologias é contemplado pelos identificadores uniformes de recursos (em inglês, *Uniform Resource Identifier* - URIs) e pelo *Resource Description Framework* (RDF).

URIs (MASINTER; BERNERS-LEE; FIELDING, 2005) são utilizadas para identificar unicamente conceitos, instâncias e propriedades. Ao dereferenciar uma URI, é obtida a descrição RDF daquilo que foi identificado. O modelo RDF (MANOLA et al., 2004) é descentralizado, baseado em grafo e extensível, projetado para a representação integrada de dados de fontes diversas. Uma descrição RDF é uma tripla que relaciona um sujeito a um objeto por meio de um predicado, como formalmente descrito pela definição 1, inspirado em Sacenti (2016).

Definição 1. Uma declaração RDF é uma tripla $d = \langle s, p, o \rangle$, onde s é o sujeito, p é o predicado e o é o objeto da declaração. Sujeito e predicado são representados por um recurso unicamente identificado por uma URI u. O sujeito s representa uma entidade e o predicado p representa uma propriedade desta entidade. O objeto assume o valor da propriedade p, podendo ser repre-

sentado por outro recurso ou por um valor alfanumérico.

No exemplo da Figura 3, a tripla ⟨*ex:Person2, foaf:topic_interest, ex:Topic2*⟩ descreve que Sacenti (especificado por *ex:Person2*) tem interesse em ontologias (especificado por *ex:Topic2*). Nesta tripla, $ex:Person2$ é o sujeito, *foaf:topic_interest* (ter interesse) é o predicado e o objeto é o tópico *ex:Topic2*.

A tecnologia de representação semântica *Ontology Web Language* (OWL) é um padrão para o desenvolvimento de ontologias da *Web* (MOTIK; PATEL-SCHNEIDER; PARSIA, 2012). Por meio de um vocabulário próprio e padronizado em cima do RDF, a linguagem OWL permite descrever classes (conceitos), indivíduos, propriedades de objeto, propriedades de dados, tipos de dados, entre outros. Além disso, a OWL também permite definir restrições como subclasses, classes equivalentes, classes disjuntas, propriedades inversas, domínio e imagem de uma propriedade, propriedades transitivas, entre outras. Estas restrições proporcionam a aplicação de regras de inferência e raciocínio da lógica descritiva em declarações de uma ontologia descrita usando OWL.

Definição 2. Uma ontologia descrita em OWL é um conjunto de declarações RDF $d \in D$ (Definição 1) que descrevem entidades $e \in E$ como indivíduos $ind \in Ind$ ou conceitos $c \in C$ por meio de propriedades $p \in P$ e por meio de restrições $r \in R$ aplicadas a estas entidades e propriedades. Os conjuntos $Ind \subset E$ e $C \subset E$ são mutuamente disjuntos, i.e. $Ind \cap C = \emptyset$.

No exemplo da Figura 3, a declaração RDF ⟨*ex:Person2, rdf:type, foaf:Person*⟩ descreve que Sacenti é uma pessoa. Nesta declaração, a propriedade *rdf:type* define uma função cujo conjunto domínio (os sujeitos de declarações) são indivíduos e cujo conjunto imagem (os objetos de declarações) são conceitos. Além disso, o conceito *foaf:Person* possui uma restrição de subclasse *rdfs:subclassof* indicando que todo indivíduo deste conceito também é do conceito *owl:Thing*.

Declarações RDF são armazenadas em Bases de Conhecimento (BCs, em inglês *Knowledge Bases*) e podem ou não incluir ontologias descritas em OWL. Por exemplo, BCs do tipo Perguntas & Respostas geralmente não incluem ontologias, são apenas compostas de triplas interligando perguntas e respostas.

2.1.1 Tipos de ontologias

As ontologias podem ser classificadas quanto o nível de generalidade ou especificidade (GUARINO et al., 1998):

- **Ontologia de alto nível (*top-level ontology*)**: descreve conceitos genéricos (p.ex., espaço, tempo, objetos, ações), independentes de um problema ou domínio particular.
- **Ontologia de tarefa (*task ontology*)**: descreve o vocabulário relacionado a uma atividade ou tarefa específica, como recomendar ou diagnosticar, ao especializar conceitos pertencentes a uma ontologia de alto nível.

- **Ontologia de domínio (*domain ontology*)**: descreve o vocabulário relacionado a um domínio (p.ex., filmes, medicina, relações sociais), ao especializar conceitos pertencentes a uma ontologia de alto nível.

- **Ontologia de aplicação (*application ontology*)**: descreve conceitos dependentes de domínio(s) e tarefa(s) específicos, frequentemente exigindo especializações de ontologias de domínio e tarefa. Por exemplo, ontologias de aplicação são utilizadas para descrever papéis desempenhados por entidades do domínio durante a execução de determinada tarefa.

2.1.2 Taxonomias e hierarquias de entidades de uma ontologia

Pieterse e Kourie (2014) classificam cinco tipos de sistemas de organização de conhecimento, incluindo ontologias e taxonomias. Para os autores, uma taxonomia organiza o conhecimento com relacionamentos hierárquicos que classificam as entidades (conceitos ou indivíduos) em entidades rasas ou abrangentes. Já a ontologia é classificada por estes autores como um sistema mais complexo que descreve formalmente o conhecimento com relacionamentos hierárquicos e associativos, além de adicionar regras de inferência na forma de metarrelações, restrições, regras condicionais ou regras de produção. É importante observar aqui que uma ontologia pode conter uma ou mais taxonomias.

As declarações RDF de ontologias podem descrever relações de ordenamento parcial entre conceitos ou instâncias, compondo taxonomias. Neste trabalho, o termo hierarquia de entidades é adotado para nomear uma coleção de conceitos ou instâncias hierarquicamente organizadas por propriedades representando este tipo de relacionamento. Uma hierarquia de entidades é formalmente definida em (3), inspirado em Sacenti (2016).

Definição 3. Uma hierarquia de entidades $H(E,D)$ é um conjunto de entidades E descritas por um conjunto de declarações RDF $d \in D$ na forma $d = \langle s,p,o \rangle$, onde o sujeito e o objeto pertencem ao conjunto de entidades, i.e. $s,o \in E$ e a propriedade p representa uma relação de ordenamento parcial (p.ex., *is_a* ou *subClassOf*, *part_of* ou *contained*, *type*), conectando entidades como um digrafo (grafo dirigido, direcionado ou orientado) acíclico e fracamente conexo (*weakly connected directed acyclic graph* - DAG).

A título de exemplo, considere o sistema de classificação de trabalhos sobre computação *ACM Computing Classification System*[2] (ACM CCS). A Figura 3 descreve que o documento *OBRS:YAS* tem como tópico *Recommender System*. De acordo com ACM CCS, este tópico é subtópico de *Retrieval tasks and goals*, que é subtópico de *Information Retrieval*.

[2] Acesso: https://dl.acm.org/ccs/ccs_flat.cfm, em: 22/11/2021.

2.1.3 Classificações facetadas em uma ontologia

As propriedades de uma ontologia permitem descrever diferentes aspectos de um fato observado (indivíduo). Por exemplo, a ontologia Movie Ontology (MO)[3] descreve filmes especificando seus gêneros, atores, diretores, produtores, premiação, avaliação da crítica, data de lançamento, duração, entre outros. Usuários alvos de recomendações frequentemente levam em conta uma ou mais destas características quando selecionam itens (ADOMAVICIUS; KWON, 2007; MANOUSELIS; COSTOPOULOU, 2007). Estas informações laterais podem ser usadas em uma classificação facetada (JOUDREY; TAYLOR; MILLER, 2015), que permite organizar o conhecimento de maneira sistemática de segundo aspectos (facetas) distintos, cada qual com seu conjunto de valores (p.ex., entidades), que podem ser organizados em uma hierarquia de entidades (i.e., uma taxonomia). Nesta tese, uma classificação facetada é um conjunto de facetas mutuamente exclusivas e exaustivas em conjunto, cada uma feita isolando uma perspectiva (aspecto) da caracterização de indivíduos (p.ex., taxonomia de gêneros de filmes, atores, diretores, prêmios, níveis de avaliação, períodos, categorias de metragem). Quando combinadas, estas facetas descrevem completamente os indivíduos observados, sendo então úteis na análise, navegação e recuperação destes indivíduos. Desta forma, uma ontologia pode conter uma ou mais facetas descrevendo indivíduos de uma determinada classe.

Na Figura 3, os tópicos de interesse *ex:Topic1* (*Recommender Systems*) e *ex:Topic2* (*Ontology*) servem de facetas tanto para documentos *foaf:Document* quanto pessoas *foaf:Person*. Ainda, entidades de uma faceta podem ser organizadas por taxonomias.

2.1.4 Dados conectados e dados abertos conectados

Um dos principais objetivos de dados conectados (DCs) é estender a *Web* tradicional, onde documentos (páginas e *sites* da *Web*) são interconectados por hiperligações, para uma *Web* com dados que possam estar diretamente conectados (BIZER; HEATH; BERNERS-LEE, 2009). As diretrizes para a publicação destes dados conectados são (BERNERS-LEE et al., 1998a):

1. Usar URIs como nomes para coisas.

2. Usar URIs HTTP para que as pessoas possam procurar esses nomes.

3. Quando alguem procurar uma URI, prover informação útil, usando padrões (p.ex., RDF, SPARQL).

4. Incluir conexões para outras URIs de modo que possam permitir a descoberta de mais coisas.

[3] Acesso: http://www.movieontology.org:80/2010/01/movieontology.owl, em: 14/06/2018.

A publicação e consumo de BCs na *Web* é orientada por este conjunto de diretrizes, que permitem descrever conexões entre recursos de diferentes fontes de dados por meio de, por exemplo, propriedades como *owl:sameAs*, *owl:equivalentClass*.

A publicação de dados abertos conectados (DACs) é uma iniciativa comunitária cujo objetivo é fornecer dados públicos e livres sob os princípios dos dados conectados (BIZER; HEATH; BERNERS-LEE, 2009). Coleções de dados abertos conectados são disponibilizados sob uma licença aberta (p.ex. dados governamentais ou de domínio público) e no formato de dados conectados (PESKA; VOJTAS, 2013).

2.2 GRAFOS DE CONHECIMENTO

O termo Grafo de Conhecimento (GC, em inglês *Knowledge Graph*) tornou-se popular após a introdução do GC da Google em 2012 e, desde então, vem sendo usado sem amplo consenso quanto a sua definição (EHRLINGER; WÖSS, 2016). Muitas vezes, sua definição se confunde como a de BC ou ontologia. De certo modo, um GC enfatiza que a estrutura de uma BC pode ser representada e processada como um grafo que pode incluir ontologias que definam o seu esquema de dados de forma completa ou parcial. Nesta tese, é adotada a seguinte definição de GC:

Definição 4. Um grafo de conhecimento $G(N,A)$ é um conjunto finito de nodos (vértices) N e um conjunto de arestas $A \subseteq N \times P \times N$, cada uma ligando um par de nodos pertencentes a N por meio de uma propriedade particular $p \in P$. Cada nodo $n \in N$ refere-se a uma entidade $e \in E$ (conceito ou indivíduo), enquanto as arestas são ligações semânticas do tipo p entre dois nós *n_cabeça* e *n_cauda*. Note que, nesta tese, a notação de aresta é similar a de declaração RDF.

O modelo RDF permite a representação do GC como um conjunto de declarações RDF (Definição 1) na forma (s,p,o) ou $\langle n\_cabeça, p, n\_cauda \rangle$, descrevendo fatos (declarações), onde o *n_cabeça* (subject) refere-se a uma entidade, o *n_cauda* (object) refere-se a uma entidade ou literal (valor alfanumérico) e o p refere-se ao tipo de ligação semântica entre os primeiros.

O GC pode ser aplicado a uma ampla variedade de tarefas em diferentes áreas de interesse:

- identificar qual o sentido de uma palavra usada em uma frase (desambiguação);
- responder perguntas feitas em linguagem natural (Perguntas & Respostas);
- procurar informação pelo significado e não apenas pelo casamento léxico (busca semântica);
- detectar e classificar relações entre entidades (extração/mineração);
- inferir relações faltantes (predição de ligações, em inglês, *link prediction*);
- predizer a preferência de um usuário (SRs).

A entidades e relacionamentos apresentados na Figura 2 podem ser representados por um GC aplicado à recomendação de filmes. Os nodos deste GC representam usuários (à esquerda), filmes, gêneros, atores, diretores e inclusive estes conceitos, enquanto que as arestas representam relações entre essas entidades, como as interações usuário-item, a classificação dos filmes em um ou mais gêneros, os atores que aparecem nesses filmes e os diretores.

Um GC descrevendo informações laterais para SRs pode integrar informações oriundas de fontes de dados de diferentes tipos, por exemplo:

- **Fontes de dados de domínio-cruzado (em inglês, *cross-domain*)**: fornecem informações de diversos domínios. Por exemplo, DBpedia (LEHMANN et al., 2015) e Freebase (BOLLACKER et al., 2008) contêm um número expressivo de fatos sobre coisas variadas (por exemplo, pessoas, lugares, eventos, livros, filmes).

- **Fontes de dados de domínio-específico**: têm seu escopo de informações limitado a determinados domínios. Por exemplo, Internet Movie Database (IMDb) é um banco de dados online com informações relacionadas a filmes, programas de televisão, vídeos, videogames e conteúdo de streaming online - incluindo elenco, equipe de produção e biografias pessoais, resumos de enredo, curiosidades, classificações e críticas. Já o LinkedMDB (HASSANZADEH; CONSENS, 2009) foi o primeiro conjunto de DACs ligando recursos da *Web* sobre filmes de várias fontes (por exemplo, IMDb, Rotten Tomatoes, FreeBase, DBpedia).

- **APIs de redes sociais**: alguns GCs incluem dados coletados de mídia social por meio de APIs como Facebook Graph API e Twitter API.

2.3 ANOTAÇÕES SEMÂNTICAS

As anotações semânticas (BERNERS-LEE; HENDLER; LASSILA, 2001) associam uma informação de base (p.ex., um nome de um autor em um texto) a descrições com semântica bem definida (p.ex., o URI da descrição do autor da DBpedia). Estas descrições semânticas, também chamadas de informações sobrepostas (valores), são descritas em ontologias ou GCs. Por exemplo, um filme de um sistema de *streaming* pode ser a informação base de uma anotação semântica (aquilo que é anotado) e a URI de uma entidade descrita por uma ontologia (p.ex., *Movie Ontology* e *Music Ontology*[4]) ou GC pode ser a informação sobreposta a este filme. Tais declarações podem ser armazenadas em uma BC e que está armazenada em uma BC.

Definição 5. Uma anotação semântica é uma tripla (Definição 1) $as = \langle s, p, o \rangle$, onde s identifica a informação base, p representa um relacionamento de anotação e o identifica a informação sobreposta representada por uma entidade $e \in E$ descrita em uma ontologia OWL.

[4] Acesso: http://purl.org/ontology/mo/, em: 22/11/2021.

Por exemplo, na Figura 2 o filme *Terminator: Dark Fate* é a informação base de uma anotação semântica que relaciona-o por meio da propriedade *mo:belongsToGenre* com o gênero *mo:Sci-Fi*, descrito pela ontologia MovieOntology (mo).

2.4 EMBEDDINGS

Embeddings são representações latentes e de baixa dimensionalidade de entidades do mundo real na forma de vetores algébricos que capturam propriedades relevantes dessas entidades, tais como similaridade de significado. Embeddings podem ser utilizados em vários domínios. No Processamento de Linguagem Natural (PLN), *word embeddings*, como *word2vec* (MIKOLOV et al., 2013a; MIKOLOV et al., 2013b), são usados para representar em um espaço vetorial o significado de palavras durante a análise textual, onde a proximidade dos vetores é proporcional à similaridade dos significados (JURAFSKY; MARTIN, 2009). Estes modelos são também chamados de métodos de aprendizagem relacional estatística ou aprendizagem de máquina relacional.

Em um *embedding* de GC, cada nodo de GC é representado por um vetor de um espaço de nodos V_N e cada aresta, por sua vez, é representado por um vetor de um espaço de propriedades V_P. Ambos os espaços vetoriais ideais para representar V_N e V_P são desconhecidos, portanto, os nodos e propriedades do GC são mapeados para o espaço $\mathbb{R}^k$ por meio das funções de *embedding* $f_N : V_N \to \mathbb{R}^l$ e $f_P : V_P \to \mathbb{R}^m$, respectivamente, sendo que $k,l,m \in \mathbb{N}$ e representam as dimensões destes espaços reais. Os espaços gerados são uma estimativa dos espaços ideais V_N e V_P. No mundo real, k e l são definidos *a priori*, enquanto que f_N e f_P são estimados por meio de algoritmos de otimização (p.ex., gradiente descendente, aprendizado adaptativo) que minimiza uma função de perda definida pela função de *embedding* (WANG et al., 2017).

Existem duas classes principais de modelos para geração de *embeddings* de GC: i. modelos de distância de tradução, como TransE (BORDES et al., 2013), TransH (WANG et al., 2014), TransR (LIN et al., 2015); e ii. modelos de correspondência semântica, como DistMult (YANG et al., 2015) e ComplEx (TROUILLON et al., 2016).

TransE é um método que constrói um modelo e interpreta relações como translações no espaço de um *embedding* de entidades: caso exista uma tripla $\langle h,l,t \rangle$, então o vetor do *embedding* representando a entidade t deve estar próximo do vetor de h somado ao vetor que depende ou representa a relação l, isto é, $h + l \approx t$ (BORDES et al., 2013). Caso não exista uma tripla $\langle h,l,t \rangle$, os vetores $h + l$ e t devem estar distantes, segundo alguma medida de dissimilaridade como as normas $L1$ ou $L2$. TransE aprende a identificar triplas existentes por meio das triplas presentes no conjunto de treinamento, e aprende triplas inexistentes por meio de triplas geradas aleatoriamente e não presentes no conjunto de treinamento, chamadas de amostras negativas. A função de perda do treino minimiza a distância entre entidades presentes em triplas do conjunto de treinamento e maximiza a distância entre entidades de triplas geradas como amostras negativas. TransE adota como algoritmo de otimização o gradiente descendente estocástico com a restrição do valor da norma $L2$ dos vetores de entidades ser igual a 1.

Enquanto que TransE compõe vetores de números reais, outros *embeddings* adotam números complexos para tratar uma vasta variedade de relações binárias, como as relações de simetria e assimetria. No modelo ComplEx (TROUILLON et al., 2016), uma entidade é representada por dois vetores complexos diferentes, dependendo se aparece como sujeito ou objeto da relação. Este modelo adota o produto interno *Hermitian* (ou sesquilinear) para definir a probabilidade de existir uma relação (tripla) envolvendo uma entidade sujeito h e outra entidade objeto t de modo assimétrico, isto é, considerando a ordem das entidades na relação. Matrizes normais, autovetores e autovalores são utilizados calcular esta probabilidade. Relações são aproximadas por meio de fatoração, usando números complexos para representar fatores latentes.

De modo análogo à *word embeddings*, a proximidade entre vetores de *embeddings* de GC pode ser usada para indicar a similaridade entre os respectivos nodos e/ou propriedades. Por exemplo, métricas de distância como as distâncias Euclidiana, *Manhathan* e de similaridade vetorial como similaridade de cosseno permitem comparar a proximidade/similaridade dois vetores e, consequentemente, estimar medidas de similaridade semântica entre nodos e propriedades do GC. A título de exemplo, a distância Euclidiana é definida pela Equação 2.1.

$$d = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{2.1}$$

Onde: x e y são dois vetores pertencentes a um mesmo espaço vetorial $x, y \in V$, este espaço vetorial é composto de n dimensões de números reais $V \subset \mathbb{R}^n$ e x_i e y_i são valores da i-ésima dimensão deste espaço vetorial. A distância Euclidiana d equivale à distância entre os pontos representados pelos vetores x e y.

2.5 SUMMARIZAÇÃO DE GC

O grande volume e a falta de relevância de dados representados em um GC prejudicam a eficiência e a eficácia de aplicações que adotam este tipo de formalismo para representar conhecimento. A Sumarização de Grafo (SG) permite reduzir o custo computacional destas aplicações, por meio da redução do volume do GC e da eliminação de dados irrelevantes e ruidosos. Os algoritmos de SG podem transformar grafos em representações mais compactas (sumários), preservando propriedades que são úteis para a aplicação ou domínio.

Os métodos de SG tem cinco desafios principais (LIU et al., 2018): (i) o volume de dados, pois o método de SG deve ser escalável ao processar grandes quantidades de dados; (ii) a complexidade dos dados – a grande quantidade de arestas entre os nodos do GC e a variedade de propriedades que as descrevem dificultam o processamento de operações no GC; (iii) a definição de relevância – a decisão do sumário mais adequado para a aplicação precisa considerar as compensações entre tempo, tamanho, informação preservada e complexidade do sumário; (iv) a avaliação do sumário – que depende do domínio de aplicação; (v) a atualização

do sumário – que envolve como e com que frequência o sumário deve seguir a evolução do GC de entrada.

Alguns métodos de SG (FIORUCCI; PELOSIN; PELILLO, 2020) tratam apenas sua estrutura de conectividade como um grafo não-rotulado (sem considerar propriedades das entidades representadas pelos nodos) ou uma matriz de adjacência. Outros métodos (LIU; CHENG; QU, 2020; ROZEMBERCZKI et al., 2019) exploram as semânticas de declarações RDF estruturadas pelos GC como grafos rotulados, ou seja, grafos cujos nodos e arestas são descritos por listas de atributos (com valores de propriedades). Além disso, os métodos de SG podem ser classificados de acordo com a abordagem adotada (LIU et al., 2018):

- **Baseados em agrupamento**: agrupam nodos em supernodos ou arestas em nodos virtuais, resultando em um sumário na forma de um supergrafo. Estes métodos adotam diversas abordagens como as baseadas em clusterização, agregação e quociente (ČEBIRIĆ et al., 2019);
- **Baseados em compressão de bits**: minimizam o número de bits usado para representar o GC;
- **Baseados na simplificação ou esparsificação**: removem nodos ou arestas pouco relevantes ou ruidosas. Estes métodos incluem abordagem como técnicas de amostragem de grafo, esparsificação de grafos e visualização (*sketching*) de grafos;
- **Baseados em influência**: identificam os papéis (descrições de alto nível) de nodos e arestas com base na propagação de influência.

2.6 CONSIDERAÇÕES FINAIS

Este capítulo apresentou a fundamentação teórica sobre tipos de ontologias, taxonomias e hierarquias de entidades, classificações facetadas, dados conectados (DCs), Grafos de Conhecimento (GCs), anotações semânticas, *embeddings* de GC e sumarização de GC. Estes conceitos são relevantes para a análise dos objetos de estudo desta tese, as variações de representação da informação lateral de SRs baseados em conhecimento (SRCs), especificamente SRs baseados em ontologia (SROs) e SRs baseados em grafo de conhecimento (SRGCs). A seguir, o Capítulo 3 apresenta a fundamentação teórica sobre Sistemas de Recomendação (SRs), SRCs, SROs, SRGCs e as métricas de avaliação de SRs que foram adotadas por esta tese.

3 RECOMENDAÇÃO USANDO INFORMAÇÃO LATERAL

Em geral, os Sistemas de Recomendação (SRs) apoiam-se em interações (p.ex., avaliações, acessos, compras) prévias dos usuários com os itens de consumo (p.ex., filmes, livros, músicas, lugares) para ajudar estes usuários a encontrar novos itens (AGGARWAL et al., 2016). Um dos problemas que afetam a qualidade das técnicas de recomendação é a falta de dados de avaliações, a chamada esparsidade de dados (BOBADILLA; SERRADILLA, 2009). A esparsidade de dados geralmente provoca a produção de recomendações de baixa qualidade para os usuários que fornecem poucas avaliações. Para mitigar este problema e melhorar a qualidade da recomendação, os SRs baseados em Conhecimento (BURKE, 2007) incorporam informações adicionais sobre itens e usuários, também chamadas de informações laterais.

Este capítulo apresenta primeiramente a fundamentação teórica sobre SRs, modelo de perfil de usuário, técnicas clássicas de filtragem e um exemplo de filtragem colaborativa. Na sequência, é apresentada a fundamentação sobre os SRs baseados em Conhecimento (SRCs), incluindo SRs baseados em multiatributo, SRs baseados em Ontologia (SROs) e SRs baseados em Grafo de Conhecimento (SRGCs). Finalmente, este capítulo descreve as métricas de avaliação de SRs que foram adotadas por esta tese.

3.1 SISTEMAS DE RECOMENDAÇÃO

SRs aplicam técnicas e ferramentas de software para fornecem sugestões de itens que são na maior parte do interesse de um usuário em particular (RICCI; ROKACH; SHAPIRA, 2015), e são úteis principalmente onde há um grande volume de opções, uma vez que nesta situação o processo de seleção pode se tornar difícil para o usuário (SENECAL; NANTEL, 2004).

SRs realizam recomendações personalizadas aos usuários, considerando as preferências e interesses destes usuários para recomendar itens específicos. A personalização tem grande importância para a área de Ciência de Computação, especificamente em aplicações de SRs. Sistemas personalizados são sensíveis ao contexto e aos perfis de seus usuários, adaptando serviços para suprir as necessidades e melhorar a experiência de uso para cada usuário. Um dos passos fundamentais da personalização é a construção do modelo de Perfil de Usuário (PU – em inglês, *user profile*, *user profiling*), detalhado a seguir.

3.1.1 Modelo de perfil de usuário

No contexto de SRs, um PU é uma construção estruturada, ou um modelo, contendo dados (ou informações) diretamente ou indiretamente relacionadas às preferências, comportamento e ao contexto de um usuário (KADIMA; MALEK, 2010). O modelo de PU apoia a recomendação personalizada ao armazenar comportamentos e interesses dos usuários.

A construção do PU pode ser definida como o processo de capturar dados sobre o domínio de interesse do usuário (KANOJE; GIRASE; MUKHOPADHYAY, 2014). Para esta construção do PU, são observadas as interações de usuários sobre itens, como visualização, compra ou avaliação. Ao capturar uma dessas interações, um SR pode obter outras informações, como o valor de compra, a nota de avaliação, ou o tempo indicando quando a interação ocorreu. Estas informações auxiliam a compreender as necessidades e comportamentos do usuário e podem estar representadas no PU. Nesta tese, a interação de um usuário sobre um item é definida em (6).

Definição 6. Uma interação usuário-item é uma tripla $int = \langle u, i, v \rangle$, tal que $u \in U$ é um usuário, $i \in I$ é um item e $v \in V$ é uma tupla de metadados descrevendo informações sobre a interação como o tipo (p.ex., compra, acesso ou avaliação) e a intensidade (p.ex., valor da avaliação, preço da compra).

A eficácia da recomendação personalizada está atrelada à qualidade da construção destes PUs, o que envolve três principais desafios (adaptados de Zhao e Shen (2016)):

1. Capturar de maneira eficiente, e mais completamente possível, os dados que compõem o PU. A quantidade e a qualidade de dados mantidos no PU implicam diretamente na precisão de um sistema de recomendação (CREMONESI; EPIFANIA; GARZOTTO, 2012; KADIMA; MALEK, 2010; KANOJE; GIRASE; MUKHOPADHYAY, 2014). Estes dados variam em termos de tipo e generalidade de acordo com a técnica de recomendação e com os dados disponíveis. Técnicas clássicas de recomendação geralmente baseiam-se nas interações prévias de usuários com itens (p.ex., avaliações, compras, acessos).

2. Converter os dados mantidos no PU em preferências e interesses do usuário no domínio dos itens a serem recomendados. Esta conversão é uma atividade complexa, pois envolve tentar estimar predileções individuais determinadas por diversos fatores.

3. Utilizar preferências estimadas para prever quais itens são possivelmente de interesse do usuário. Geralmente, um SR determina o chamado grau de utilidade de cada item para um dado usuário. Os itens de maior utilidade comporão a lista de itens recomendados.

Quanto à captura dos dados que compõem o PU, os SRs podem adotar uma das seguintes técnicas (MIDDLETON, 2003):

- **Captura Explícita**: os dados que compõem o PU são fornecidos diretamente pelos usuários. A maioria dos SRs se baseiam nas avaliações (em inglês, *rating*) que os usuários fornecem explicitamente sobre os itens (p.ex., a partir de uma classificação cinco-estrelas). Alternativamente, o SR pode solicitar informações mais explicitamente relacionadas com as preferências dos usuários, como temas de interesse deste usuário, ou informações de contexto, como as demográficas (p.ex., sexo, idade, informações de localização).

- **Captura Implícita**: os dados mantidos no PU são obtidos pelo SR sem a intervenção explícita dos usuários. Estes dados podem ser obtidos a partir do histórico de interações dos usuários com o sistema, incluindo dados de navegação, busca, visualização, compra, realizadas pelos usuários sobre os itens.

- **Modo híbrido**: os dados mantidos no PU são oriundos tanto das realimentações explícitas vindas do usuário quanto das informações capturadas implicitamente pelo sistema. O objetivo do modo híbrido é alcançar melhorias na qualidade das recomendações.

Um exemplo simples de representação de um PU gerado pela captura explícita de interações são as chamadas Matrizes de Avaliação Usuário-Item, onde cada usuário é representado por um vetor de itens. Uma matriz de avaliações mantém as avaliações que os usuários se dispuseram a prover sobre os itens. Nesta matriz, os vetores de itens são vetores de n dimensões, onde n refere-se à quantidade de itens disponíveis no sistema alvo da recomendação. A informação armazenada em cada posição do vetor representa o valor da avaliação atribuída pelo usuário para aquele item. O valor da avaliação pode ser quantificado utilizando diferentes tipos de dados, como o binário (curti/não curti), o numérico, ou o baseado em categorias (como em {bom, ótimo, excelente}).

Por exemplo, a Tabela 2 apresenta um exemplo de Matriz de Avaliação Usuário-Item com quatro usuários (*u2116*, *u835*, *u986* e *u5238*), seis itens (*i1019*, *i1089*, *i1097*, *i1253*, *i1301* e *i1320*) e avaliações na forma de valores numéricos na faixa de valores inteiros [1; 5]. A avaliação não fornecida (dado faltante) é representada pelo elemento vazio (na tabela ilustrado pelas células em branco).

Tabela 2 – Matriz de avaliação usuário-item construída via captura explícita

item/usuário	u53	u835	u986	u5238
i1019	4	3		
i1089				1
i1097	5	1		5
i2116		5		
i2402		4		
i2571	4		4	

Fonte: criada pelo autor.

A qualidade e quantidade de dados mantidos em um PU para representar as preferências e interesses de um usuário implicam diretamente na precisão do SR (KANOJE; GIRASE; MUKHOPADHYAY, 2014; CREMONESI; EPIFANIA; GARZOTTO, 2012; KADIMA; MALEK, 2010). Considera-se aqui que a captura explícita tende a prover dados mais confiáveis, e portanto, possibilitariam uma estimativa das preferências dos usuários mais próximas da realidade. Apesar disto, a captura explícita tem um aspecto negativo que é a exigência de um esforço por parte do usuário em avaliar os itens. Os usuários tendem a não prover avaliações

para os itens que eles interagem, devido carga extra de trabalho (SPERETTA; GAUCH, 2005; XIA et al., 2009).

SRs que adotam a captura explícita de dados são vulneráveis aos dados faltantes (avaliações não fornecidas), o que caracteriza o problema da esparsidade de dados (BOBADILLA; SERRADILLA, 2009). Este problema é ilustrado pela Tabela 2 que apresenta um número elevado de células em branco.

A captura implícita ou híbrida permite coletar um volume maior de interações que a captura explícita. Um exemplo de modelo de PU construído a partir da captura implícita ou híbrida é a Matriz de Interação Usuário-Item. A Tabela 3 exemplifica uma matriz de interação construída pela captura híbrida, que complementa as avaliações representadas pela Tabela 2 com interações de visualização representadas pelo símbolo *V* (valor de tipo booleano), diminuindo assim o número de células em branco.

Tabela 3 – Matriz de interação usuário-item construída via captura híbrida

item/usuário	u53	u835	u986	u5238
i1019	4	3	V	
i1089		V		1
i1097	5	1	V	5
i2116	V	5	V	V
i2402	V	4		
i2571	4	V	4	V

Fonte: criada pelo autor.

Os SRs aplicam técnicas de recomendação nos dados capturados e representados pelo modelo de PU para gerar recomendações personalizadas aos usuários. A seguir, são apresentadas algumas destas técnicas.

3.1.2 Técnicas clássicas de filtragem

As técnicas de recomendação podem ser categorizadas como subclasses de técnicas de Filtragem de Informação que visam filtrar informações irrelevantes ou redundantes sob a perspectiva de um ou mais usuários (HANANI; SHAPIRA; SHOVAL, 2001). De acordo com a técnica de filtragem de informação adotada, uma técnica de recomendação pode ser classificada em um dos três tipos básicos (CAZELLA; NUNES; REATEGUI, 2010):

- **Filtragem Colaborativa (FC)**: utilizam interações de múltiplos usuários para recomendar itens de forma colaborativa. Existem diferentes métodos de FC. Por exemplo, a FC baseada em usuário utiliza as interações usuário-item para determinar usuários com perfis similares, chamados de vizinhos próximos, para então calcular o grau de utilidade de um item a um usuário. De modo análogo, a FC baseada em item determina itens vizinhos ao item apreciado (bem avaliado) pelo usuário alvo da recomendação. Estas técnicas não consideram as características dos itens, nem dados pessoais sobre o próprio usuário.

Dentre outros problemas, as técnicas são vulneráveis ao problema da esparsidade de dados (BOBADILLA; SERRADILLA, 2009), ilustrado na seção 3.1.1. Além disso, estas técnicas geralmente não oferecem suporte a explicabilidade das recomendações, ou seja, essas técnicas não consideram quais aspectos do item levaram o usuário a bem ou mal avaliá-lo.

- **Filtragem Baseada em Conteúdo (FBC)**: além de interações usuário-item, as técnicas de FBC levam em consideração as características dos itens a serem recomendados. A descrição dos itens e o próprio conteúdo associado ao item (p.ex., conteúdo de documentos) permitem estimar preferências do usuário considerando, por exemplo, as características mais presentes nos itens melhor avaliados. As técnicas de FBC sofrem do problema da superespecialização (*over specification*), onde o SR recomenda apenas itens similares aos itens já acessados anteriormente pelo usuário (IAQUINTA et al., 2008); e do problema da análise de conteúdo limitado, que ocorre quando existe apenas um número limitado ou a ausência de características descrevendo os itens a serem recomendados (MUSTO et al., 2017). Além disso, estas técnicas geralmente consideram apenas características diretamente relacionadas aos itens, sem considerar relacionamentos indiretos (p.e., no domínio de filmes, o local de nascimento do diretor de um filme) ou relacionamentos entre valores de características (p.e., hierarquia ou taxonomia de gêneros de filme).

- **Filtragem Híbrida (FH)**: combinam técnicas FC e FBC para minimizar os problemas do uso individual destas técnicas. Por exemplo, explorando o conteúdo sobre itens para mitigar a esparsidade de dados.

Um problema clássico da FC é a esparsidade de dados, que torna difícil a recomendação de itens que tiveram pouco (ou talvez nenhum) acesso por parte dos usuários, visto que afeta as etapas de cálculo de similaridade, determinação de vizinhança e inferência de utilidade. De modo similar às técnicas de FBC que consideram características de itens para recomendar, os SRs baseados em Conhecimento (SRCs) mitigam o problema da esparsidade de dados ao incorporar informações laterais sobre itens e usuários.

3.1.3 Exemplo de filtragem colaborativa

Independente da técnica de filtragem utilizada, um SR é composto de três elementos principais: (i) um módulo para capturar das preferências do usuário; (ii) um módulo para analisar as preferências do usuário; (iii) um módulo que realiza a recomendação personalizada ao usuário (CHEN et al., 2011). O primeiro módulo observa no sistema alvo da recomendação os interações usuário-item e constrói os modelos de PU. O segundo módulo analisa os PUs para filtrar os itens mais relevantes para o usuário alvo da recomendação. Por último, o terceiro módulo realiza finalmente a recomendação ao usuário, por exemplo via reordenamento das buscas do usuário para considerar tanto os critérios de busca e o perfil de usuário. Outro tipo de reco-

mendação é a apresentação de uma lista de ranqueamento de itens relevantes de acordo com a previsão da avaliação (abordagens baseadas em avaliação) ou com a estimativa da preferência deste usuário (abordagens baseadas em utilidade).

Esta seção exemplifica estes 3 elementos considerando a FC baseada em usuário, que serviu como inspiração para a técnica de filtragem híbrida que compõe a primeira abordagem da proposta desta tese, descrita no Capítulo 5.

Em geral, o método de FC baseada em usuário utiliza a Matriz de Avaliações Usuário-Item construída via captura explícita (como a da Tabela 2) para determinação dos vizinhos próximos e cálculo da utilidade dos itens para cada usuário.

Com base na matriz de avaliações, um algoritmo de FC determina a similaridade entre cada par de usuários do sistema. Segundo (OWEN; OWEN, 2012), as métricas de similaridade mais comuns em SRs incluem a correlação de Pearson e de Spearman, distância euclidiana, medida dos cossenos, coeficiente Tanimoto e Log-Likelihood. A correlação de Pearson é definida pela Equação 3.1 e, a título de exemplo, é usada para a determinação da similaridade entre dois usuários representados por vetores de itens contendo o valor da avaliação ou o símbolo de vazio para representar um dado faltante.

$$\rho = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{3.1}$$

Onde: x e y são os vetores de item do dois usuários correlacionados, x_i é o valor fornecido pelo usuário x da avaliação relacionada ao i-ésimo item, y_i é o valor fornecido pelo usuário y de avaliação relacionada ao i-ésimo item, e n é o número de itens avaliados por ambos os usuários. E $\bar{x}$ e $\bar{y}$ representam respectivamente a média do valor das avaliações do usuário x e y considerando apenas os n itens avaliados por ambos os usuários. O coeficiente de Pearson resultante varia de -1 a 1. Por exemplo, considere as colunas referentes aos usuários *u53* e *u835* da Tabela 2. Os itens avaliados em comum por estes usuários são *i1019* e *i1097*, isto é, $n = 2$. Portanto, o vetor de itens do usuário *u53* é $x = (4,5)$ e o vetor de *u835* é $y = (3,1)$. A média do valor dos vetores destes usuários é $\bar{x} = 4,5$ e $\bar{y} = 2$. Conforme indicado pela Equação 3.2, o valor de correlação (similaridade) é $\rho = -1$, indicando que estes usuários têm opiniões totalmente opostas sobre os filmes que avaliaram. Note que a correlação de Pearson é indicada para vetores de dimensão $n > 2$, para obter resultados significativos, sendo este exemplo apenas ilustrativo. Note ainda que o cálculo de similaridade entre usuários pode adotar abordagens distintas que consideram a captura implícita ou híbrida (p.ex., para a matriz de interação usuário-item representada pela Tabela 3).

$$\rho = \frac{(4-4,5)(3-2)+(5-4,5)(1-2)}{\sqrt{(4-4,5)^2+(5-4,5)^2} \cdot \sqrt{(3-2)^2+(1-2)^2}} \tag{3.2}$$

$$\rho = \frac{-0,5-0,5}{\sqrt{(-0,5)^2+(0,5)^2}\cdot\sqrt{(1)^2+(-1)^2}}$$
$$\rho = \frac{-1}{\sqrt{0.25+0.25}\cdot\sqrt{1+1}}$$
$$\rho = \frac{-1}{\sqrt{0.5}\cdot\sqrt{2}}$$
$$\rho = \frac{-1}{1}$$

Calculada a similaridade entre usuários, a etapa seguinte é a determinação dos vizinhos próximos. Existem duas técnicas clássicas para determinar a vizinhança (DESROSIERS; KARYPIS, 2011): Filtragem *Top-N*, em que a vizinhança de um usuário alvo é formada por N vizinhos próximos com maior similaridade com o usuário alvo; e filtragem por limiar (*threshold filtering*), em que a vizinhança de um usuário u é formada pelos usuários com similaridade com u acima de um certo limiar.

A etapa seguinte após a formação das vizinhanças consiste em inferir a utilidade de um item para um dado usuário. A FC baseada em usuário recomenda a um usuário que pertence a uma vizinhança os itens que não foram avaliados por este usuário e que foram avaliados positivamente pelos vizinhos deste usuário.

A utilidade de um item não avaliado por um usuário u é tradicionalmente calculada por uma média ponderada das avaliações deste item realizadas pelos vizinhos próximos de u. A Equação 3.3 apresenta o cálculo de utilidade ($util_{i,u}$) de um item i para um usuário u, onde n é o número de vizinhos próximos de u que avaliaram o item i e representados por x_j, $\overline{u}$ é a média do valor de avaliações do usuário u, $x_{j,i}$ é o valor fornecido pelo usuário x_j como avaliação ao item i, $\overline{x_j}$ é a média do valor de avaliações do usuário x_j, $sim(u,x_j)$ é a medida de similaridade entre o usuário alvo u e o vizinho próximo x_j, e finalmente k é um fator que indica o quanto o valor de avaliação de vizinhos próximos influencia no valor de utilidade do item para o usuário alvo da recomendação.

$$util_{i,u} = \overline{u} + k\sum_{j=1}^{n} sim(u,x_j)(x_{j,i}-\overline{x_j}) \tag{3.3}$$

Após a determinação da utilidade dos itens para o usuário u, o SR gera uma lista ranqueada de itens recomendados para este usuário que é composta pelos itens de maior utilidade para o usuário u ordenados de forma decrescente ou ordenada por meio de um algoritmo de otimização de ranqueamento (p.ex., *Bayesian personalized ranking* – BPR). Esta lista geralmente é limitada aos n primeiros itens mais úteis.

3.2 SR BASEADO EM CONHECIMENTO

Os Sistemas de Recomendação baseados em Conhecimento (SRCs) apoiam-se no conhecimento explícito do domínio dos itens e dos usuários para determinar como os itens satisfazem as necessidades do usuário (BURKE, 2007). Estes sistemas capturam informações laterais

sobre usuários (p.ex., informações demográficas) e itens (p.ex., no domínio de filmes, pode-se citar gêneros, atores, diretores, data de lançamento, sinopse, críticas) oriundas do sistema alvo de recomendação ou de fontes externas como dados conectados (DCs – p.ex., DBpedia, Freebase, LinkedMDb), dados estruturados (p.ex., IMDb, Rotten Tomatoes) e mídias sociais (p.ex., Facebook Graph, Twitter).

As informações laterais são incorporadas ao SRCs durante a construção de modelos de PU e de diferentes maneiras (BOBADILLA et al., 2013): casos (*Case-Based Reasoning*), restrições (*Constraint-Based Reasoning*), consultas, métricas de alinhamento (*Matching Metrics*), redes sociais (SHOKEEN; RANA, 2020), vetores de conhecimento, ontologias (apresentada na Seção 2.1) e Grafos de Conhecimento (GCs – apresentado na Seção 2.2). Um dado SRC pode selecionar os tipos de informações que irão compor o PU, que idealmente deveriam ser aqueles que os usuários frequentemente levam em conta quando buscam pelos itens (ADOMAVICIUS; KWON, 2007; MANOUSELIS; COSTOPOULOU, 2007). Além da seleção dos tipos de informações que irão compor o PU, os SRCs devem escolher como estas informações serão representadas e utilizadas pelo SR.

A representação da informação lateral em SRCs, especificamente os SRs baseados em ontologias (SROs) e SRs baseados em Grafos de Conhecimento (SRGCs), é o objeto de estudo desta tese. As seções a seguir fundamentam SRs baseados em multiatributo, SROs e SRGCs.

3.2.1 SR baseado em multiatributo

Usuários frequentemente consideram mais de um aspecto quando escolhem itens (ADOMAVICIUS; KWON, 2007; MANOUSELIS; COSTOPOULOU, 2007). Portanto, as informações laterais sobre itens geralmente descrevem atributos relacionados a diferentes aspectos dos itens. Por exemplo, durante a escolha de um filme, os usuários podem considerar importantes aspectos como os gêneros, diretores e atores do filme (KO; SON; KO, 2015).

Os SRs baseados em multiatributo geralmente constroem modelos de PU na forma de uma Matriz de Interações Usuário-Atributo, representando usuários como vetores de atributos (ou vetores de palavras-chave). A Tabela 4 exemplifica esta matriz, considerando como atributo os gêneros de filmes e valores das células representando as quantidades de itens com os quais os usuários interagem que apresentam cada valor de atributo. Estes modelos de PU permitem a utilização de técnicas de FBC ou, quando combinados a Matriz de Interação Usuário-Item (Seção 3.1.1), de FH.

Um exemplo de uso desta matriz usuário-atributo é a técnica de FH proposta por Fernandes, Sacenti e Willrich (2017), que teve a participação do autor desta tese. Esta técnica foi inspirada na FC clássica exemplificada na Seção 3.1.3, com uma variação na determinação dos vizinhos próximos. Nesta proposta, a vizinhança de um usuário não é determinada pela similaridade entre valores das avaliações dos usuários, mas sim pela similaridade da frequência de ocorrência de atributos diretamente relacionados aos itens. Para tal, a Matriz de Interações Usuário-Atributo é adotada para representar as informações laterais sobre os itens e armazenar

Tabela 4 – Matriz de interações usuário-atributo

atributo/usuário	u53	u835	u986	u5238
Action	2	2	1	1
Adventure	2	2	2	1
Animmation	1	1	1	1
Sci-Fi	4	3	3	2
Thriller	1	2	1	2
War	2	2	1	1

Fonte: criada pelo autor.

para cada usuário a frequência de ocorrências de itens que possuam determinado atributo.

Em Fernandes, Sacenti e Willrich (2017), a similaridade entre dois usuários, representados por vetores de atributos nesta proposta, é calculada utilizando a correlação de Pearson, de modo análogo ao exemplo da Seção 3.1.3. Esta proposta permite o uso das técnicas de filtragem *Top-N* ou de filtragem por limiar para determinar as vizinhanças dos usuários. Finalmente, a etapa de inferência de utilidade desta proposta adota o mesmo cálculo de utilidade apresentado pela Equação 3.3, utilizando uma Matriz de Avaliações Usuário-Item.

No trabalho de Fernandes, Sacenti e Willrich (2017), demonstrou-se que houve uma redução do erro de predição do grau de utilidade dos itens, mesmo em cenários de grande esparsidade de dados de avaliação. Isto porque a Matriz de Interações Usuário-Atributo não apresenta dados faltantes e eleva o nível de abstração da representação do usuário, sendo mais fácil de formar vizinhança entre usuários. Deste modo, esta técnica permite mitigar o problema da esparsidade de dados. Porém, a facilidade em gerar vizinhanças pode prejudicar a qualidade da recomendação, pois a similaridade baseada em multiatributo pode não ser suficiente para representar as preferências dos usuários ao considerar usuários próximos apenas sob a perspectiva dos atributos modelados pelo PU.

3.2.2 SR baseado em ontologia

Há mais de uma década, SRs baseados em Ontologia (SROs) vem sendo considerados uma tendência emergente para a área sistemas de recomendação (MIDDLETON; ROURE; SHADBOLT, 2009; RODRÍGUEZ-GARCÍA et al., 2015; BAHRAMIAN; ABBASPOUR; CLARAMUNT, 2017). A representação semântica do modelo de PU por meio de ontologias (apresentada na Seção 2.1) possibilita ao SR utilizar um modelo computacional de preferências de usuário mais rico e detalhado que o de técnicas clássicas de recomendação (BAHRAMIAN; ABBASPOUR; CLARAMUNT, 2017). Nesta tese, chamamos esta representação semântica de Perfil Ontológico de Usuário (POU). Os SROs constroem os modelos de POU de diferentes maneiras: como uma Matriz de Interações Usuário-Conceitos (usuários representados por vetores de conhecimento) (GOWAN, 2003; GAUCH et al., 2007), hierarquias de conceito (DAOUD et al., 2007), ou instâncias anotadas de uma ontologia de referência (POUs – em inglês,

ontological user profile) (SIEG; MOBASHER; BURKE, 2007).

Conforme os diferentes tipos de ontologias (Seção 2.1.1), as ontologias são usadas no SRO para representar algum dos seguintes conhecimentos:

- **Conhecimento acerca dos itens**: SROs que se baseiam no conteúdo dos itens podem representar este conhecimento através de uma ou mais ontologias de domínio relacionado item. Por exemplo, caso os itens a recomendar sejam filmes ou músicas, a ontologia de domínio *Movie Ontology*[1] e *Music Ontology*[2] pode ser consideradas.

- **Conhecimento acerca dos usuários**: SROs que se baseiam na informação demográfica dos usuários podem representar este conhecimento através de ontologias de perfil de usuário (KATIFORI et al., 2007). Tais ontologias permitem expressar conceitos relacionados a diversos tipos de informações, como as demográficas, de relacionamentos sociais, de contexto de usuários, interações dos usuários sobre os itens, e outros. Diferentes SROs expressam este conhecimento em diferentes níveis e granularidades. Alguns trabalhos, como Sieg, Mobasher e Burke (2007), representam conhecimento acerca dos usuários na forma de uma "visão" da ontologia de referência onde os conceitos são anotados por um valor numérico indicando o grau de interesse do usuário no conceito. Outros trabalhos, como Silva (2015) e Salles e Willrich (2015), adotam ontologias permitindo especificar as interações dos usuários com os itens, além de combinar conhecimentos acerca de itens e usuários.

- **Conhecimento acerca da recomendação**: SROs que utilizam ontologias para descrever os resultados de recomendação, como a ontologia *The Recommendation Ontology*[3]. Este tipo de ontologia permite padronizar o consumo de itens recomendados externamente no sistema alvo ou internamente em sistemas de recomendação híbrida.

- **Conhecimento acerca do processo de recomendação**: Neste nível de representação, é possível adotar uma ontologia de tarefa provendo suporte ao processo de recomendação. As ontologias de tarefa fornecem um conjunto de condições, por meio do qual descrevem genericamente como resolver um tipo de problema (MIZOGUCHI; VANWELKENHUYSEN; IKEDA, 1995). Silva (2015) declara que ontologias de domínio não têm capacidade de suportar o processo de recomendação de maneira isolada. Então, este trabalho propõe uma ontologia de tarefa de recomendação que permite descrever como um processo é executado através de uma sucessão de métodos ordenados e seus parâmetros de entrada.

O conhecimento bem-descrito acerca dos usuários e itens pode ser compartilhado e reutilizado por outros SROs, uma clássica vantagem de ontologias. Além disso, a adoção de ontologias mais abstratas também permite atribuir aos SROs menor dependência ao domínio

[1] Acesso: http://www.movieontology.org:80/2010/01/movieontology.owl, em: 14/06/2018.
[2] Acesso: http://purl.org/ontology/mo/, em: 22/11/2021.
[3] Acesso: http://purl.org/ontology/rec/core##, em: 22/11/2021

do item, tornando-os assim, mais adaptáveis a diferentes domínios. Por exemplo, Garcia et al. (2010), Pan et al. (2010), Mendoza et al. (2015) e Moreno et al. (2016) para o domínio de filmes; Sieg, Mobasher e Burke (2010), Liao et al. (2010) e Chen, Kuo e Liao (2015) para livros; e Kim (2013), Wardhana e I. (2013) e Rodríguez-García et al. (2015) para música. A recomendação de itens de forma independentemente do domínio em que este esteja inserido é vista como um desafio (RICCI; ROKACH; SHAPIRA, 2011).

Além de suportar a construção de modelos de POU, as ontologias também influenciam nas técnicas de recomendação. Existem pelo menos duas estratégias para gerar recomendações a partir de um modelo de POU:

1. Construir os POUs (SIEG; MOBASHER; BURKE, 2010) e posteriormente transformar estes modelos em PUs que suportem técnicas de recomendação clássicas (CANTADOR; BELLOGIN; CASTELLS, 2008);

2. Aplicar algoritmos de aprendizado e motores de inferência diretamente aos POUs (GARCIA et al., 2010; AGUILAR; VALDIVIEZO-DÍAZ; RIOFRIO, 2016).

Enquanto que primeira estratégia permite reusar técnicas de recomendação consolidadas e difundidas pela literatura (SCHELTER; OWEN, 2012; SEMINARIO; WILSON, 2012; WALUNJ; SADAFALE, 2013; BOKDE; GIRASE; MUKHOPADHYAY, 2015), a segunda estratégia tem como desafio o problema da escalabilidade devido à necessidade de busca e exploração de grandes coleções de documentos distribuídos não-estruturados (GAUCH et al., 2007).

Um exemplo de SRO que adota a primeira estratégia é o proposto por este autor em Sacenti, Willrich e Fileto (2018), e descrito no Capítulo 5. Neste trabalho, o modelo de POU é construído com base em um arcabouço conceitual que utiliza ontologias de diferentes níveis de abstração. Depois, os POUs são transformados em uma Matriz de Interação Usuário-Entidade (usuários representados por vetores de entidades), que é análoga à Matriz de Interações Usuário-Atributo (Seção 3.2.1). Este trabalho adotou a mesma técnica de recomendação proposta em Fernandes, Sacenti e Willrich (2017), também apresentada pela Seção 3.2.1.

Outro exemplo de SRO é o proposto em Santos (2019), que adota a segunda estratégia para gerar recomendações. Este trabalho captura o histórico de entrada em estabelecimentos (*check-in*) oriundas de uma mídia social para construir o modelo de POU na forma de uma Base de Conhecimento (BC). Então, este trabalho aplica uma técnica de FH de três etapas. A primeira etapa é a de obtenção de dados em uma BC. A segunda etapa é uma FBC realizada com o auxílio de um motor de inferência que utiliza a informação e as regras de restrição e de associação da BC para predizer estabelecimentos de interesse do usuário. A terceira etapa realiza uma FC com base nos usuários similares (vizinhos próximos) conectados pela rede social do usuário alvo da recomendação (amigos) extraída da mídia social. Os estabelecimentos visitados por vizinhos próximos são confrontados com os estabelecimentos obtidos pela segunda etapa, depois são ranqueados e ordenados em uma lista de recomendação.

Algumas propostas de SROs (PASSANT; HEITMANN; HAYES, 2009; NOIA et al., 2012; OSTUNI et al., 2013; BELLINI et al., 2017; ANGELIS et al., 2017) consideram informações laterais em dados conectados (DCs – apresentado na Seção 2.1.4) ou dados abertos conectados (DACs). As principais vantagens de reusar DC em SROs são (NOIA; CANTADOR; OSTUNI, 2014):

- **Conhecimento multidomínio:** disponibilidade de grande quantidade de DCs sobre diferentes domínios (p.ex., locais geográficos, músicas, filmes, arte, pessoas, fatos, conhecimentos gerais).
- **Acesso padronizado aos dados:** facilidade na recuperação de DCs devido a descrição formal, bem-definida e a sua publicação na *Web*.
- **Análise semântica:** disponibilidade de informações laterais complementares capazes de enriquecer o modelo de POU, por exemplo com mais relacionamentos indiretos ao item (p.ex., no domínio de filmes, o local de nascimento do diretor de um filme).

O reuso de DCs ou DACs geralmente adiciona ao processo de recomendação de SRO duas novas etapas (NOIA; CANTADOR; OSTUNI, 2014; TOMEO et al., 2016; VAGLIANO; MONTI; MORISIO, 2017):

1. Enriquecimento semântico, que conecta um item do sistema alvo da recomendação a um recurso dos DCs por meio de anotações semânticas (Seção 2.3).
2. Seleção de informações laterais nos DCs, que identifica as declarações RDF sobre o recurso anotado que não são relevantes para o processo de recomendação e adiciona estas declarações ao modelo de POU.

A seleção de características (informação lateral, atributos) é um processo de selecionar os atributos mais relevantes de um conjunto de dados para um modelo de predição baseado em conteúdo (RAGONE et al., 2017). Selecionar as k características mais relevantes para a tarefa de recomendação é equivalente a descobrir quais informações em uma coleção de DCs são relevantes para a recomendação e quais são apenas ruídos (MUSTO et al., 2015).

A seleção manual de informações laterais em DCs é uma abordagem ingênua, custosa e dependente de domínio, pois exige que um especialista conheça o domínio dos itens a recomendar e as informações laterais disponíveis nos DCs. Enquanto a seleção automática de informações laterais ainda apresenta oportunidades de pesquisa (MUSTO et al., 2015; RAGONE et al., 2017).

3.2.3 SR baseado em GC

Os SRs baseados em GC (SRGCs) constroem o modelo de POU na forma de uma estrutura conectada de usuários, itens e entidades, representada por Grafos de Conhecimento

(GCs – Seção 2.2). Os SRGCs geram recomendações a partir dos GCs usando três estratégias principais (Guo et al., 2020):

1. Métodos baseados em *embedding* de GC (Seção 2.4) que geram recomendações com base na representações latentes dos usuários, itens e entidades representados pelo GC (ZHANG et al., 2016; ZHANG et al., 2018; PIAO; BRESLIN, 2018; CAO et al., 2019).

2. Métodos baseados em caminhos que exploram a estrutura do GC para gerar recomendações (YU et al., 2013; SUN et al., 2018).

3. Métodos unificados que se beneficiam de ambas as estratégias anteriores (WANG et al., 2018; WANG et al., 2019).

Os SRGCs baseados em *embedding* transformam o GC em um modelo de *embedding* de GC (em inglês, KG *Embedding* – KGE) (WANG et al., 2017). A ideia geral destes métodos é aprender representações vetoriais latentes de usuários, itens, entidades e propriedades (arestas) do GC, junto com uma função que mapeia pares de usuário-item em uma pontuação de preferência (etapa de inferência de utilidade do item). Além da tarefa de recomendação, os *embeddings* de GC permitem predizer novas conexões (em inglês, *link prediction*), também chamada de tarefa de complementação do GC (em inglês, *KG completion*). Quando combinadas, estas tarefas podem potencialmente melhorar a qualidade da recomendação.

Tradicionalmente, os SRGCs baseados em caminho exploram os GCs como uma rede de informações heterogêneas (em inglês, *Heterogeneous Information Network* – HIN) (Guo et al., 2020). Esses SRGCs calculam a similaridade entre entidades com base em meta-caminhos.

Os métodos unificados são uma tendência de pesquisa recente que combina métodos baseados em *embeddings* e métodos baseados em caminhos. Os SRGCs unificados apoiam-se em algoritmos de propagação em *embeddings* que exploram a representação semântica de entidades e propriedades (arestas) do GC. Um dos primeiros trabalhos a introduzir a propagação em *embeddings* em SRGCs foi RippleNet (WANG et al., 2018).

3.3 MÉTRICAS DE AVALIAÇÃO DE SR

A avaliação de SRs em termos de eficácia e eficiência do processo de recomendação é uma etapa essencial que auxilia a busca pela melhoria da experiência do usuário no sistema alvo da recomendação. A eficácia neste contexto está relacionada à qualidade das recomendações geradas, e eficiência, por sua vez, está relacionado ao custo computacional e, principalmente, ao tempo necessário para treinar o modelo de PU. Existem diversas métricas que avaliam diferentes aspectos da recomendação produzida por um SR: erro, acurácia, diversidade, serendipidade, novidade, custo computacional, entre outras. Nesta seção, apresentamos os modos de realização de avaliações em SR e as métricas adotadas pelas abordagens propostas nesta tese.

A avaliação de um SR pode ser realizada em modo *offline* ou *online*. O modo *offline* separa as interações já capturadas pelo SR em conjuntos de treinamento (contendo interações

que serão utilizadas no treinamento do modelo de PU) e de teste (contendo interações reservadas pela avaliação para comparar com as recomendações geradas). O modo *online* realiza testes A/B massivos (p.ex., dando a alguns usuários a versão A de um SR e para outros a versão B) que comparam a satisfação do usuário (via formulários de pesquisa) ou a rentabilidade do sistema alvo da recomendação. Os experimentos descritos nesta tese adotam a avaliação *offline*.

A primeira abordagem proposta nesta tese (descrita no Capítulo 5) avalia SROs que predizem a avaliação (*rating*) de itens. Esta abordagem adota duas métricas de erro já consolidadas e que já foram avaliadas em Seminario e Wilson (2012): raiz do erro quadrático médio (em inglês, *Root Mean Square Error* – RMSE), e o erro absoluto médio (em inglês, *Mean Absolute Error* – MAE). A métrica RMSE representa a raiz do quadrado da diferença entre o valor da avaliação de um item, que é fornecido pelo usuário e que está reservado no conjunto de teste, e o valor predito pelo SR da avaliação para aquele item (Equação 3.4).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=0}^{n}(r_i - p_i)^2} \tag{3.4}$$

Enquanto que a métrica MAE representa a média da diferença absoluta entre a avaliação do usuário e o valor predito da avaliação (Equação 3.5).

$$MAE = \frac{1}{n}\sum_{i=0}^{n}|r_i - p_i| \tag{3.5}$$

A segunda abordagem (descrita no Capítulo 6) avalia SRGCs que predizem a utilidade de itens. As métricas de erro adotadas pela primeira abordagem não são adequadas para avaliar estes SRGCs, pois avaliam o erro de predição da avaliação do usuário. Nesta situação, as métricas de acurácia são as mais adequadas. Portanto, a segunda abordagem adota métricas de acurácia e métricas de custo computacional. As métricas de acurácia adotadas por esta abordagem são a precisão em N (p@N), cobertura em N (em inglês, *recall* – r@N), ganho cumulativo com desconto normalizado em N (em inglês, *Normalized Discounted Cumulative Gain at N* – nDCG@N) e média da precisão média em N (em inglês, *Mean Average Precision* – mAP@N).

A métrica p@N é a precisão considerando apenas os N primeiros itens recomendados a cada usuário (Equação 3.6). Para cada usuário, é calculada a fração dos primeiros N itens recomendados que são relevantes para aquele usuário (t_{rr}/t_{rec}), onde t_{rr} é o total de itens recomendados que são relevantes em N e t_{rec} é o total de itens recomendados em N. A precisão em N é a média das frações (t_{rr}/t_{rec}) de todos os usuários. A precisão também pode ser expressa como uma razão entre recomendações verdadeiro positivas vp e falso positivas fp.

$$p@N = \frac{t_{rr}}{t_{rec}} = \frac{vp}{vp + fp} \tag{3.6}$$

A métrica r@N é a cobertura (revocação) considerando apenas os N primeiros itens recomendados a cada usuário (Equação 3.7). Para cada usuário, é calculada a fração de itens relevantes que estão entre os primeiros N itens recomendados (t_{rr}/t_{rel}), onde t_{rr} é o total de itens

recomendados que são relevantes em N e t_{rel} é o total de itens relevantes. A cobertura em N é a média das frações (t_{rr}/t_{rel}) de todos os usuários. A cobertura também pode ser expressa como uma razão entre recomendações verdadeiro positivas vp e falso negativas fn.

$$r@N = \frac{t_{rr}}{t_{rel}} = \frac{vp}{vp+fn} \tag{3.7}$$

A métrica nDCG@N é o ganho cumulativo com desconto normalizado considerando apenas os N primeiros itens recomendados a cada usuário (Equação 3.8). Para cada usuário, é calculada a soma das relevâncias (rel_i) dos primeiros N itens recomendados penalizada pelo logaritmo do respectivo ranqueamento (posição na lista), também chamada de $DCG@N$. Essa soma é normalizada pelo DCG@N ideal ($IDCG@N$), ou seja, considerando todos os itens relevantes para um usuário em uma ordem monotonicamente decrescente. O nDCG@N é a média desta soma normalizada considerando todos os usuários.

$$nDCG@N = \frac{DCG@N}{IDCG@N} \tag{3.8}$$
$$DCG@N = \sum_{i=1}^{N} \frac{rel_i}{log_2(i+1)}$$

A métrica mAP@N é a média da precisão média considerando os N primeiros itens recomendados a cada usuário (Equação 3.9). Para cada usuário, é calculada a soma do inverso multiplicativo da posição (ranqueamento) de cada item relevante presente nos primeiros N itens recomendados ($P(k) \times rel(k)$). Esta soma é dividida pelo número total de itens relevantes (t_{rel}). O mAP@N é a média deste valor ($AP@N$) considerando todos os usuários.

$$mAP@N = \frac{\sum_{q=1}^{N} AP(q)}{N} \tag{3.9}$$
$$AP@N = \frac{\sum_{k=1}^{N} P(k) \times rel(k)}{t_{rel}}$$

3.4 CONSIDERAÇÕES FINAIS

Este capítulo teve por objetivo apresentar a fundamentação teórica sobre Sistemas de Recomendação (SRs) e técnicas classicas de filtragem, SRs baseados em conhecimento (SRCs) considerando multiatributos, ontologias e GCs, e as métricas de avaliação de SRs que foram adotadas por esta tese. A seguir, o Capítulo 4 apresenta os trabalhos relacionados a esta tese.

4 TRABALHOS RELACIONADOS

Os Sistemas de Recomendação baseados em Conhecimento (SRCs) apoiam-se no conhecimento explícito do domínio dos itens e dos usuários para determinar como os itens satisfazem as necessidades do usuário (BURKE, 2007). Os SRCs incorporam ao modelo de perfil de usuário informações laterais sobre usuários (p.ex., informações demográficas) e itens (p.ex., no domínio do filmes, pode-se citar atores, diretores e gêneros), oriundas do sistema alvo de recomendação ou de fontes externas como dados conectados (DCs – p.ex., DBpedia, Freebase, LinkedMDb), dados estruturados (p.ex., IMDb, Rotten Tomatoes) e mídias sociais (p.ex., Facebook Graph, Twitter). Os SRCs se mostraram eficazes em mitigar o problema da esparsidade (isto é, o problema da falta de dados) de interações usuário-item.

Apesar da adição de informações laterais geralmente produzir recomendações com melhor qualidade, seu uso implica em um aumento expressivo da complexidade do treinamento do SRC devido ao aumento no volume e diversidade dos dados considerados. Neste sentido, para o uso efetivo de SRCs em cenários reais, é de extrema importância que se considere a efetividade ao se definir uma técnica que explore a informação lateral no processo de recomendação. Para ser efetivo, um SRC necessita ser eficaz e eficiente ao mesmo tempo, ou estabelecer equilíbrio entre estes requisitos. Nesta tese, a efetividade de um SRC consiste em prover recomendações com qualidade (com eficácia) de forma otimizada, de maneira mais rápida ou com menor montante de recursos (com eficiência).

A eficiência dos SRs é geralmente omitida pelos trabalhos científicos, sendo que poucos trabalhos apresentam uma análise relacionando a eficiência e a eficácia (PAUN, 2020). Neste sentido, a análise do impacto da representação da informação lateral na eficiência e eficácia dos SRCs, bem como a definição de técnicas que visam aumentar suas efetividades, são temas relevantes de pesquisa, dada a sua importância para o uso de SRCs em cenários reais e a escassez de trabalhos na literatura que abordam este tema.

De forma a identificar os impactos da forma de representação das informações laterais em SRCs é necessário comparar diferentes formas de representação de conhecimento. Esta tese considera a representação do conhecimento em SRCs na forma de ontologias, nos chamados SRs baseados em ontologias (SROs) e na forma de grafos de conhecimentos, nos chamados SRs baseados em Grafos de Conhecimento (SRGCs). Esta tese visa investigar os impactos destas representações tanto na eficácia quanto na eficiência do sistema, e propor soluções para melhorar a eficiência dos SROs e SRGC. Para isto, esta tese propõe duas abordagens para mitigar o problema do alto custo de treinamento de modelos de recomendação baseados em informações laterais. A primeira abordagem reduz a complexidade da tarefa de recomendação via a conversão da representação do conhecimento representado em ontologia em uma matriz de preferência. Esta matriz é considerada em um um SR baseado em filtragem híbrida. A segunda abordagem visa mitigar o custo computacional através da redução do volume da informação lateral, por meio de uma técnica de sumarização aplicada a SRGCs que combina *embeddings* com a clusterização de nodos *K-Means*.

Este capítulo apresenta trabalhos relacionados a estas estas duas abordagens para mitigar o problema da eficiência de SRCs. Para a primeira abordagem, este capítulo apresenta o estado da arte de SRs baseados em taxonomias, SROs e SRs baseados em dados conectados (DCs). Para a segunda abordagem, este capítulo apresenta o estado da arte de SRGCs e técnicas de SG.

4.1 SRS BASEADOS EM TAXONOMIAS, ONTOLOGIAS E DADOS CONECTADOS

Esta seção apresenta os trabalhos relacionados à primeira abordagem proposta para mitigar os problemas de eficiência em SRCs. Estes trabalhos foram identificados por meio de duas pesquisas bibliográficas visando cumprir a Etapa 1 do método de pesquisa adotado nesta tese (Seção 1.4). A primeira trata-se de uma pesquisa exploratória sobre SROs que buscou identificar como ontologias foram utilizadas por estes sistemas. A segunda trata-se de uma pesquisa descritiva sobre SRs que constroem Perfis de Usuário (PUs) baseados em taxonomias, ontologias e DCs.

A primeira pesquisa adotou o processo de revisão sistemática proposto em Kitchenham e Charters (2007) para atender a pergunta de pesquisa "Sistemas de recomendação baseados em ontologias são empregados em quais domínios de aplicação?". Esta pesquisa selecionou 190 trabalhos por meio de duas buscas em 8 motores de busca, sendo utilizado na primeira busca as palavras-chave *recommendation*, *recommender*, *recommended*, *ontology*, *linked data* e *semantic web* e na segunda busca as palavras-chave *recommender system*, *recommendation system* e *ontology*. Os trabalhos foram classificados em 20 categorias de acordo com o o domínio de aplicação dos SRs propostos, sendo as categorias mais predominantes a de SRO sobre lugar (26 trabalhos), sobre multimídia (26 trabalhos), e sobre notícias e páginas Web (21 trabalhos). Esta pesquisa teve o objetivo de introduzir o autor desta tese na área de SROs.

A segunda pesquisa selecionou 193 trabalhos por meio de 2 buscas no motor de buscas *Google Scholar*, sendo utilizado na primeira busca as palavras-chave *recommender* e *ontology* e na segunda busca as palavras-chave *recommender* e *linked data*. Os trabalhos foram ordenados cronologicamente e foram separados em 8 categorias de acordo com a existência das palavras-chave *linked data*, *ontology* e *graph* no corpo de texto dos trabalhos. Estas categorias serviram para estabelecer uma ordem prioritária para a revisão. Os trabalhos mais relevantes encontrados por esta pesquisa foram classificados em três classes não disjuntas: (I) SR baseado em taxonomia, (II) SRO; (III) SR baseado em DCs. As Seções 4.1.1, 4.1.2 e 4.1.3 apresentam alguns trabalhos para cada uma destas classes de SROs, respectivamente. Depois, é apresentada uma análise comparativa entre a primeira abordagem desta tese e os trabalhos relacionados mais relevantes.

4.1.1 SR baseado em taxonomia

Os autores de (GAUCH et al., 2007) identificaram que a maioria dos pesquisadores que usam ontologias ou DCs para construir o modelo de PU adotam a representação de vetores de conceitos ponderados (como a Matriz de Interações Usuário-Entidade ilustrada na Seção 3.2.2). Esta representação é similar à representação baseada em vetores de palavras-chave (como a Matriz de Interações Usuário-Atributo ilustrada na Seção 3.2.1). Por exemplo, em Ostuni et al. (2014), Oramas et al. (2017) os autores definem um mapeamento do modelo baseado em grafo de DCs para vetores de conceitos ponderados.

Entretanto, esta representação assume que os conceitos sejam independentes entre si, o que limita a expressividade do modelo de PU e prejudica a eficácia da recomendação (ALSHAIKH; UCHYIGIT; EVANS, 2017). Uma representação de PU mais rica é aquela que representa estes vetores de conceito ou palavras-chave na forma de taxonomia (p.ex., árvore de conceitos ou hierarquia de entidades) (ZIEGLER; LAUSEN; SCHMIDT-THIEME, 2004; DAOUD et al., 2007; SIEG; MOBASHER; BURKE, 2010). Estas taxonomias permitem explorar as relações de ordenamento parcial entre as entidades de uma ontologia ou coleção de DCs (CHANDRASEKARAN et al., 2008).

Por exemplo, os autores de Lakkaraju, Gauch e Speretta (2008) adotam uma árvore de conceitos ponderada cujos pesos que estão associados aos nodos-folha da árvore são usados para atualizar os pesos associados aos nodos superiores. Em Alshaikh, Uchyigit e Evans (2017), os autores adotam uma árvore de conceitos dinâmica que se adapta a mudança de preferências e necessidades do usuário. Em Bahramian, Abbaspour e Claramunt (2017), é definida uma taxonomia de pontos de interesse ponderada para representar o POU, que é atualizado utilizando uma técnica de *spreading activation* (COLLINS; LOFTUS, 1975; ANDERSON, 1983).

No processo de seleção de itens, os usuários levam em consideração diversas características relacionadas aos itens (ADOMAVICIUS; KWON, 2007; MANOUSELIS; COSTOPOULOU, 2007). Por exemplo, para escolha de um filme, o usuário pode considerar diversas características, como gênero do filme, atores, ano de lançamento. Portanto, idealmente os SRs baseados em taxonomias deveria considerar mais de uma taxonomia ao mesmo tempo, para considerar diferentes características. Entretanto, a pesquisa bibliográfica não encontrou trabalhos que representem o PU considerando mais de uma taxonomia.

Esta tese analisa variações desta representação considerando diferentes organizações de características de itens, em termos de aspectos e hierarquias. Por exemplo, a data de lançamento e o gênero são aspectos de um filme e cujos valores (indivíduos) podem estar organizados segundo hierarquias (p.ex., quinquênio-década-tridécada para datas de lançamento e a ontologia MovieOntology para gêneros). Na primeira abordagem da proposta, estas variações são definidas manualmente por meio de um arcabouço conceitual para construção de SROs.

4.1.2 SR baseado em ontologia

SROs foram introduzidos na literatura por Middleton, Roure e Shadbolt (2004). Deste então, muitos trabalhos (MOBASHER; JIN; ZHOU, 2004; CANTADOR; BELLOGIN; CASTELLS, 2008; SEMERARO et al., 2009; ANAND; KEARNEY; SHAPCOTT, 2007) adotaram representações semânticas sobre o conteúdo de itens baseadas em ontologias. Por exemplo, o PU adotado em Mobasher, Jin e Zhou (2004) é construído na forma de uma matriz de usuários por atributos semânticos que representam características diretamente relacionadas ao item que são descritas por uma ontologia.

Além de mecanismos de representação de conhecimento, as ontologias permitem adicionar à técnica de recomendação algoritmos de aprendizado e motores de inferência (AGUILAR; VALDIVIEZO-DÍAZ; RIOFRIO, 2016). Alguns trabalhos adotam regras de associação para descrever e explicar o comportamento dos usuários (EYHARABIDE; AMANDI, 2012; KUCHAŘ; KLIEGR, 2017; PRIMO; VICARI; BERNARDI, 2012). Conforme já descrito pela Seção 3.2.2, em Santos (2019) o autor propõe um SRO sobre estabelecimentos que utiliza motor de inferência e filtragem colaborativa para gerar recomendações.

Uma das vantagens das ontologias é o reuso de parte do conhecimento na especificação formal de conhecimento em outros domínios e aplicações. Como visto na Seção 2.1.1, por meio da existência de diferentes níveis (tipos) de ontologias, é possível criar um arcabouço de ontologias reusáveis e adaptáveis a diferentes classes de problemas (CALERO; RUIZ; PIATTINI, 2006). Esta vantagem das ontologias também é explorada em SRs. Por exemplo, Silva (2015) e Salles (2017) que exploram esta característica das ontologias para a produção de SRs facilmente adaptáveis a diferentes domínios dos itens a recomendar. Esta tese adota o termo independência de domínio para se referenciar a capacidade de reuso ou adaptação de um SRO para sistemas que recomendam itens de diferentes domínios (p.ex., filmes, músicas, pessoas).

Esta pesquisa bibliográfica identificou que a maior parte dos SROs propostos são definidos considerando um único domínio de itens (SRs para filmes, livros, músicas, produtos, estabelecimentos). Grande parte dos trabalhos ditos independentes de domínio não apresentam soluções baseadas em semântica para uma fácil adaptação ao domínio do item. Em termos práticos, a independência de domínio deve ser atingida em SRCs explorando o propriedade de reuso das ontologias. Para tal, um SRO deveria definir uma ontologia de base para o perfil de usuário que pode ser reusada em diferentes domínios, bem como uma ontologia de tarefa (Seção 2.1.1) que descreve o processo de recomendação baseada em ontologia. Existem poucos trabalhos que definam uma ontologia de perfil de usuário ou uma ontologia de tarefa que descreva processo de recomendação. A ontologia de tarefa de recomendação permite descrever o SRO com conceitos relacionados ao domínio dos itens, aos tipos de ações, aos critérios de recomendação, às técnicas de construção de perfil e às técnicas de filtragem.

Em Salles (2017) os autores propõem uma ontologia de referência para SRs, chamada Ontologia de Recomendação (RecOnt). A RecOnt define conceitos de alto nível em SRs, como usuários da recomendação, itens a recomendar, e os fatores de interesses. Fatores de interesses

são características dos itens relevantes a recomendação. Em Salles (2017), a ontologia foi utilizada basicamente como uma forma de descrição formal do contexto de um repositório digital, relacionando conceitos de uma ontologia de domínio (Seção 2.1.1) e conceitos definidos na tarefa de recomendação. Em Salles (2017), os autores propõem o mapeamento de ontologias em grafos, visando adotar técnicas de processamento de grafos para gerar as recomendações. Nesta proposta, a RecOnt somente considera ações de acesso a itens, portanto é limitada a captura implícita do perfil de usuário. Além disso, a RecOnt contempla apenas fatores de interesse representados por indivíduos na ontologia de domínio.

Em Silva (2015) os autores propõem: (i) uma ontologia de tarefa de recomendação que descreve os métodos, ordenamento e parâmetros do processo de recomendação; e (ii) uma ontologia de domínio sobre SRs, que descreve conceitos como usuários, itens, categorias, similaridade e preferências de usuários. Nesta proposta, a adaptação a um domínio do item requer a especificação de indivíduos que representam as categorias dos itens, bem como relações de hierarquia entre estes indivíduos. Por exemplo, no domínio de filmes, as categorias podem ser os diferentes gêneros de filmes. Porém, em Silva (2015) não esclarece como gerar categorias e hierarquias (aparentemente especificadas de modo manual), como considerar mais de um aspecto do item, como hierarquias de categoria são consideradas pela técnica de recomendação, nem como informações laterais indiretamente relacionadas ao item podem ser levadas em consideração (p.ex., local de nascimento do diretor de um filme). Por exemplo, considerando o domínio de filmes, a proposta não esclarece como combinar, além do gênero, outras informações do item, como data de lançamento, principais atores, diretores, dentre muitos outros.

4.1.3 SR baseado em dados conectados

A utilização de DCs em SRs foi introduzida na literatura por Passant, Heitmann e Hayes (2009) e Heitmann e Hayes (2010). Alguns dos trabalhos encontrados apresentam uma revisão sistemática recente sobre SR baseados em DCs (MARTÍNEZ, 2017; VAGLIANO, 2017).

Os autores de Passant (2010) propõem um SR baseado em DCs que utiliza entidades do *DBpedia* [1] (LEHMANN et al., 2015) para gerar recomendações de músicas. Em Bellini et al. (2017), uma rede neural é modelada utilizando propriedades *dc:subject*[2] e *rdf:type*[3] do *DBpedia*. Os autores de Angelis et al. (2017) propõe um SR baseado na filtragem híbrida que utiliza técnicas baseadas em confiança, filtragem colaborativa e consultas semânticas em um grafo social enriquecido por entidades do DBpedia, GeoNames[4] e Europeana[5].

A quantidade de dados conectados é potencialmente imensa, e estes dados ligados diretamente ou indiretamente aos itens a recomendar abrangem diferentes características. Como

[1] Acesso: http://wiki.dbpedia.org/, em: 22/11/2021.
[2] O prefixo *dc* refere-se a http://purl.org/dc/terms/
[3] O prefixo *rdf* refere-se a http://www.w3.org/1999/02/22-rdf-syntax-ns#
[4] Acesso: http://www.geonames.org/, em: 22/11/2021.
[5] Acesso: http://europeana.eu/portal/, em: 22/11/2021

já apresentado, nem todas as características são relevantes para o processo de recomendação, se tornando dados irrelevantes. Por isso, os SRs devem selecionar um conjunto de características que serão levadas em consideração no processo de recomendação. A seleção manual destas características é processo árduo e difícil. Desta forma, é importante que este processo de seleção de características seja automatizado. No domínio de SRs, a seleção automática de características de DCs conectados foi introduzida na literatura por Musto et al. (2015) que comparou sete técnicas diferentes de seleção automática. Além destes, Ragone et al. (2017) propõe uma técnica de de seleção de características baseada em sumarização de esquema.

A primeira abordagem investigada, apresentada no Capítulo 5, propõe um arcabouço conceitual para a construção de modelos de PU de SROs. Este arcabouço permite o reuso de informações laterais coletadas em fontes externas ao SRO, como os DCs.

4.1.4 Análise Comparativa

Esta seção apresentam uma análise comparativa entre a primeira abordagem desta tese e os trabalhos relacionados mais relevantes. A tabela 5 compara os trabalhos considerando:

- **Técnica**: indica a técnica utilizada para mitigar o problema do volume de dados e, consequentemente, está relacionada à eficiência da solução. As técnicas levantadas foram a conversão da representação do conhecimento em vetores de conceitos ponderados, a seleção automática de um subconjunto de características, a representação de conhecimento em uma forma simplificada e mais escalável como a taxonomia de lugares ou de produtos, a conversão em grafos e a conversão em matriz de preferências.

- **Representação de aspectos do item**: indica se a representação de conhecimento adotada descreve os aspectos do item (p.ex., gênero, diretor ou ator do filme) considerados pela técnica explicitamente ou não.

- **Quantidade de taxonomias de item**: indica quantas taxonomias (hierarquias de conceitos ou entidades) são adotadas pela representação de conhecimento.

- **Independência de domínio**: indica se o SR projetado é de fácil adaptação a diferentes domínios.

Em Ostuni et al. (2014), os autores mitigam o problema do volume de dados e eficiência convertendo DCs em vetores de conceitos ponderados. O autores de Musto et al. (2015) reduzem o volume de dados por meio da seleção automática de características. Em Ziegler, Lausen e Schmidt-Thieme (2004) e Bahramian, Abbaspour e Claramunt (2017), os autores adotam uma representação de conhecimento simplificada na forma de taxonomia. Em Salles e Willrich (2015), os autores convertem a ontologia em grafo. Os SROs levantados em geral não exploram completamente as ontologias e o conhecimento explicitado é limitado, descrevendo apenas uma

hierarquia de conceitos (taxonomia) de item e ignorando outros aspectos relacionados diretamente ao item (p.ex., no domínio de filmes: duração, data de lançamento, ator e diretor) ou indiretamente (p.ex., local de nascimento do ator principal).

A primeira abordagem investigada, apresentada no Capítulo 5 e publicada em Sacenti, Willrich e Fileto (2018), propõe (i) um arcabouço conceitual para a construção de modelos de perfil ontológico de usuário (POU) e SROs, (ii) uma técnica de conversão de ontologia em matriz de preferências, (iii) uma técnica de filtragem híbrida baseada em matriz de preferências. O arcabouço conceitual proposto estende o desenvolvido por (SALLES, 2017; SILVA, 2015), permite a representação explícita de aspectos e taxonomias, e permite o reuso da ontologia de tarefa com ontologias de diferentes domínios, garantindo assim independência de domínio ao SRO construído. Além disso, este arcabouço permite o reuso de informações laterais coletadas em fontes externas ao SRO, como os DCs. A técnica de conversão de ontologia em matriz de preferências permite reduzir a complexidade da representação do conhecimento e assim, mitigar o problema do custo computacional do treinamento do SRO. A técnica de filtragem híbrida proposta é mais eficiente em comparação com técnicas de inferência, predição de *links* ou estruturais (p.ex., *embedding, random walk* ou baseada em grafos). Ainda, o arcabouço conceitual permite a substituição da técnica proposta pela de outros autores, havendo uma conversão adequada para o formato de entrada requerido.

4.2 SRS BASEADOS EM GRAFOS DE CONHECIMENTO

Esta seção apresenta os trabalhos relacionados à segunda abordagem para mitigar o problema do alto custo de treinamento de modelos de recomendação baseados em grafos de conhecimento. Estes trabalhos foram identificados por meio de uma pesquisa bibliográfica vi-

Tabela 5 – Tabela comparativa de trabalhos relacionados à primeira abordagem

Trabalho	Técnica	Aspecto	Taxonomia	Independência
Ostuni et al. (2014)	conversão de DCs em vetores de conceitos ponderados	implícito	nenhuma	dependente
Musto et al. (2015)	seleção automática de características	implícito	nenhuma	dependente
Bahramian, Abbaspour e Claramunt (2017)	representação em taxonomia de lugares	implícito	única	dependente
Ziegler, Lausen e Schmidt-Thieme (2004)	representação em taxonomia de produtos	implícito	única	dependente
Salles e Willrich (2015)	conversão de ontologia em grafo	explícito	múltiplas	independente
Sacenti, Willrich e Fileto (2018)	**conversão de ontologia em matriz de preferências**	**explícito**	**múltiplas**	**independente**

sando cumprir a Etapa 7 do método de pesquisa desta tese (Seção 1.4). Esta pesquisa descritiva identificou o estado da arte de SRGCs e trabalhos relacionados a redução de volume de dados, em especial a SG. Esta pesquisa identificou 3 tipos de SRGCs: (i) baseados em *embedding* de GC, (ii) baseados em caminho e (iii) unificados. Os trabalhos mais relevantes encontrados por esta pesquisa são apresentados a seguir. Depois, é apresentada uma análise comparativa entre a segunda abordagem desta tese e os trabalhos mais relevantes.

4.2.1 SRGCs baseados em *embedding*

Conforme apresentado na Seção 3.2.3, os SRGCs baseados em *embedding* transformam o GC em um modelo de *embedding* de GC (em inglês, KG *Embedding* – KGE) (WANG et al., 2017). Conforme apresentado na Seção 2.4, o *embedding* de GC é um modelo de representação latente e de baixa dimensionalidade que relaciona entidades (nodos) e propriedades (tipos de arestas) de um GC a vetores algébricos.

Alguns SRGCs baseados em *embedding* consideram que o GC que representa apenas informações laterais sobre os itens, ignorando as interações do usuário em sua representação. Por exemplo, os autores em Zhang et al. (2016) propõe CKE que captura informações laterais de diferentes tipos: atributos e relacionamentos representados em um GC, descrições textuais (p.ex., sinopse de um filme) e representações visuais (p.ex., pôster de um filme). Este SRGC gera recomendações utilizando uma abordagem baseada na filtragem colaborativa (FC). Enquanto as informações representadas pelo GC são codificadas pelo modelo de *embedding* TransR, a descrição textual e o a representação visual são codificados por uma técnica de AutoEncoder. Os modelos gerados são combinados com a representação da Matriz de Interação Usuário-Item para aprimorar os recomendação gerada pela técnica de FC.

Outros SRGCs baseados em *embedding* exploram GCs que representam itens, usuários, interações e informações laterais. Por exemplo, o método de FC com Grafos de Conhecimento (CFKG) proposto por Zhang *et al.* (ZHANG et al., 2018) aplica o modelo de *embedding* TransE a um GC que representa as interações do usuário e informações laterais (p.ex., revisão sobre um item, marca de um item, categorias de item e itens adquiridos em uma mesma compra) e soluciona o problema da recomendação com predição de declarações (em inglês, *link prediction*).

Além disso, alguns SRGCs baseados em *embedding* exploram o compartilhamento de conhecimento entre a tarefa de recomendação e a tarefa de complementação do GC (em inglês, *KG completion*). Por exemplo, Cao *et al.* (CAO et al., 2019) propôs KTUP que implementa ambas as tarefas em dois módulos distintos. O módulo de recomendação utiliza um modelo de distância de tradução (Seção 2.4) proposto neste trabalho, chamado de TUP, que modela a correção de vetores de *embedding* que representam pares usuário-item observados e não observados. O módulo de complementação do GC utiliza o modelo de *embedding* TransH. Além disso, os autores deste trabalho adotam uma abordagem de filtragem de entidades infrequentes que re-

duz o número de declarações no conjunto de dados de treinamento. O SRGC CoFM (PIAO; BRESLIN, 2018), por sua vez, treina conjuntamente uma máquina de fatoração (em inglês, *Factorization machine* - FM) e o modelo TransE compartilhando parâmetros de regularização de itens e entidades alinhados.

4.2.2 SRGCs baseados em caminho

Os SRGCs baseados em caminho utilizam técnicas de recomendação baseada a exploração da estrutura de conexões do GC. Tradicionalmente, os primeiros SRGCs desta classe representam o GCs como uma rede de informação heterogênea (em inglês, *Heterogeneous Information Network* – HIN) (Guo et al., 2020). Esses SRGCs calculam a similaridade entre entidades com base nos caminhos entre os nodos que representam estas entidades no GC. HeteRec (YU et al., 2013) e RKGE (SUN et al., 2018) são exemplos desta classe de SRGC.

HeteRec explora a similaridade de meta-caminhos entre nodos da HIN para enriquecer a Matriz de Interação Usuário-Item, construindo vetores latentes refinados representando usuários e itens. Já o RKGE extrai automaticamente listas de caminhos entre usuários e itens, sem definir meta-caminhos manualmente. Especificamente, o RKGE primeiro enumera os caminhos de item-para-usuário, que são conectados por uma sequência de propriedades heterogêneas e que são restritos a um determinado comprimento. Em seguida, RKGE aplica uma rede neural recorrente para codificar todos os caminhos de item-para-usuário em um único vetor que representa a relação semântica entre aquele usuário e aquele item. Finalmente, esse vetor é utilizado na etapa de inferência de utilidade do item para o usuário (ou preferência do usuário pelo item).

4.2.3 SRGCs unificados

Os métodos unificados combinam os métodos baseados em *embedding* com métodos baseados em caminho. Um dos primeiros trabalhos a introduzir a propagação de *embedding* no grafo de itens foi RippleNet (WANG et al., 2018).

Primeiro, RippleNet modela as entidades do GC utilizando *embeddings* pré-treinados. Em seguida, este SRGC amostra conjuntos de *multi-hop ripples* do GC, que são caminhos de usuário para entidades vizinhas, que são restritos a determinado comprimento. Esses caminhos são usados para refinar os *embeddings* pré-treinados com base na similaridade entre as entidades inicial e final de cada caminho. Desta forma, RippleNet propaga a preferência do usuário ao longo dos caminhos do GC para itens que ainda não interagidos.

Finalmente, Wang *et al.* (WANG et al., 2019) propôs o KGAT, que explora o mecanismo de propagação no GC. KGAT refina o modelo de *embedding* de entidade gerado pelo TransR, levando em consideração as entidades vizinhas *multi-hop* do GC e modelando diretamente as relações de ordem superior entre usuários e itens.

4.2.4 Redução do volume de dados e sumarização de GC

O volume e a qualidade de dados proporcionam desafios a vários sistemas que empregam algoritmos de aprendizado de máquina (AMs – em inglês, *machine learning*), incluindo muitos SRs. Em particular, o grande volume de dados necessário para atingir altos níveis de eficácia podem ter um impacto considerável nos custos computacionais (PAUN, 2020). Enquanto isso, dados irrelevantes e ruidosos podem degradar o desempenho (SMYTH; KEANE, 1995). Assim, muitas pesquisas foram realizadas para mitigar esses problemas, resultando em várias abordagens para reduzir o tamanho dos dados de entrada e o ruído antes de empregar os algoritmos de AM. Nesta seção, focamos a discussão na poda de instâncias (*instance prunning*), edição baseada em casos (*case-based editing*), redução de dimensionalidade e abordagens de Sumarização de Grafo (SG). Os primeiros três tipos de abordagens foram propostos em trabalhos seminais que influenciaram muitos trabalhos nas últimas décadas e já se mostraram eficazes. No entanto, mais pesquisas ainda são necessárias para determinar o potencial das abordagens de SG para melhorar o AM ao reduzir e compactar adequadamente, por exemplo, declarações de um GC que podem ser úteis como informações laterais.

A poda de instâncias (ou seleção de protótipo) (WILSON, 1972; WILSON; MARTINEZ, 1997; GARCIA et al., 2012) remove dados anotados para aliviar custos computacionais e melhorar os resultados de algoritmos de AM, como classificação baseada em k-vizinhos mais próximos (Cunningham; Delany, 2020). Da mesma forma, a edição baseada em casos (também conhecida como manutenção baseada em casos, em inglês, *Case-Base Maintenance* – CBM) (LEAKE; WILSON, 1998; ARSHADI; JURISICA, 2004) tenta aumentar a eficiência e a eficácia do raciocínio baseado em casos (em inglês, *Case-Based Reasoning*) reduzindo a base de casos. Essas abordagens de redução de dados podem ser categorizadas como preservação de competência ou aprimoramento de competência (NAKHJIRI; SALAMó; SàNCHEZ-MARRè, 2020). A preservação da competência remove dados redundantes que não contribuem para a qualidade dos resultados, enquanto o aprimoramento de competência remove dados ruidosos e corrompidos. Essas abordagens têm o potencial de diminuir o espaço de armazenamento e o tempo de processamento, além de melhorar a eficácia. O vizinho mais próximo editado (em inglês, *Edited Nearest Neighbor* – ENN) (WILSON, 1972; WILSON; MARTINEZ, 1997) foi o primeiro método formal de aprimoramento de competência a remover instâncias do conjunto de treinamento quando as suas classes preditas não concordam com as de seus k-vizinhos mais próximos. As abordagens CBM que objetivam a preservação de competência baseiam-se em um conjunto de propriedades dos casos (por exemplo, os conjuntos de acessibilidade e cobertura introduzidos por (SMYTH, 1998)) para identificar quais casos devem ser selecionados para compor a base de conhecimento (isto é, o conjunto editado) usada como entrada de um motor de raciocínio. Uma revisão recente das extensões e variações de ENN e CBM pode ser encontrada em (NAKHJIRI; SALAMó; SàNCHEZ-MARRè, 2020).

Técnicas de Redução de Dimensão (em inglês, *Dimension Reducing Techniques* – DRTs) (VLACHOS et al., 2002) convertem vetores de características de alta dimensionalidade

em uma codificação de dimensão inferior. Vários trabalhos revisam DRT para AM (PELUFFO-ORDÓÑEZ; LEE; VERLEYSEN, 2014; REDDY et al., 2020; SORZANO; VARGAS; MONTANO, 2014; AYESHA; HANIF; TALIB, 2020). Um DRT pode se basear na da seleção de características e/ou transformação de características (MAATEN; POSTMA; HERIK, 2009). O primeiro reduz a dimensionalidade selecionando um subconjunto de variáveis (dimensões) que são relevantes para um problema específico. O segundo transforma os vetores originais em vetores mais compactos, nos quais características redundantes e irrelevantes são removidas, mas as propriedades relevantes são preservadas. De acordo com (AYESHA; HANIF; TALIB, 2020), usar um DRT pode ser uma maneira eficiente de reduzir as variáveis de entrada de algoritmos de AM. No entanto, escolher um DRT adequado para uma situação particular pode ser bastante complicado, pois há uma grande variedade de DRTs, com objetivos e restrições variados, para tipos distintos de dados.

O uso de Sumarização de Grafos (GCs) (LIU et al., 2018) é outra alternativa para reduzir entradas de AM, quando essas entradas incluem grafos, como (extratos de) GCs representando informações laterais. As técnicas de SG podem ser complementares a outras abordagens de redução de dados e mais apropriado para certas circunstâncias. A SG pode levar em consideração aspectos como topologia de grafos e similaridade de nodos/arestas. Além disso, a SG pode ser menos dependente do método de AM, pois pode considerar ou não o treinamento do modelo de AM ao sumarizar o grafo.

Liu *et al.* (LIU et al., 2018) fornece uma taxonomia de métodos de SG com base em seus tipos de entrada e técnicas empregadas. Čebirić *et al.* (ČEBIRIĆ et al., 2019) fornece uma taxonomia dos métodos de sumarização RDF existentes. Alguns métodos de SG consideram apenas a conectividade do GC representado como um grafo não rotulado ou uma matriz de adjacência. Por exemplo, Fiorucci *et al.* (FIORUCCI; PELOSIN; PELILLO, 2020) propõe um método de SG baseado no Lema de Regularidade de Szemerédi para separar informações estruturais de ruído em grafos volumosos. Outros métodos de SG adotam técnicas baseadas em representação latente para obter vetores latentes de uma dimensionalidade inferior. Por exemplo, DeepLENS (LIU; CHENG; QU, 2020) explora *word embeddings* para codificar triplas e gerar um sumário ideal selecionando os k-triplas mais salientes.

Esta tese investiga o potencial da SG para condensar informações laterais representadas em GCs potencialmente grandes. O objetivo é reduzir os custos de armazenamento e processamento, minimizando a perda de informações e preservando padrões e/ou propriedades relevantes para os SRGCs. Esta tese propõe um método de SG que, como alguns métodos anteriores, adota *embedding* de GC (ComplEx em nossa implementação atual). No entanto, diferentemente de outros trabalhos, este método agrupa nodos do GC referentes a entidades que são similares no espaço do *embedding* e aglutina (funde) os nodos participantes de cada agrupamento em um único supernodo. O agrupamento de nodos pode ser feito em uma única-visão (GC inteiro) ou em múltiplas visões (vários subgrafos separados referindo-se a aspectos distintos do GC, como gêneros de filmes, atores e diretores). O método de *clustering* k-Means é empregado com valores variados de k para agrupar nodos nas estratégias única-visão e multi-

visão, gerando sumários alternativos. A seguir, esta seção discute trabalhos relacionados sobre SG, *clustering* multi-visão e suas aplicações para SR em geral e SRGC.

Na última década, várias tecnologias exploraram a estratégia multi-visão em GC. O algoritmo de agrupamento multi-visão foi introduzido por (BICKEL; SCHEFFER, 2004) e desde então tem sido explorado para alavancar a diversidade, precisão e robustez de partições do GC. Em tais métodos, os grafos que representam diferentes pontos de vista do mesmo assunto são primeiro fundidos e depois agrupados. Outras abordagens tentam encontrar uma maneira de maximizar a qualidade do *clustering* em cada visão, levando em consideração a consistência do *cluster* em diferentes visões (Yang; Wang, 2018). Por exemplo, os autores de Hussain, Mushtaq e Halim (2014) propõem um algoritmo de agrupamento de multi-visão para documentos que aplica um método de agrupamento de única-visão a cada visão e gera três matrizes de similaridade baseadas em partições agrupadas. Em seguida, ele agrega essas matrizes para obter uma matriz de similaridade unificada para o agrupamento final. Xue *et al.* (XUE et al., 2015) propôs um método de fusão multi-visão consciente de grupos para agrupar imagens, que adota pesos diferentes para caracterizar a similaridade entre grupos distintos. Nie *et al.* (NIE; CAI; LI, 2017) propõe um modelo de aprendizagem multi-visão com vizinhos adaptativos, que é avaliado com conjuntos de dados de imagens. Este método realiza a classificação semissupervisionada e o aprendizado e agrupamento de variedades locais (*local manifold learning*), simultaneamente, e define automaticamente um coeficiente de peso por visão.

Embora a SG recentemente tenha se tornado popular na visualização de grafos e na otimização de consultas, seu uso em é relativamente novo e inexplorado. Wu *et al.* (WU et al., 2015) propõe o GCCR, um SR baseado em mídia confiável (*media trust-aware RS*) para recomendar fontes de notícias a usuários de redes sociais heterogêneas. O GCCR emprega SG e *clustering* baseado em conteúdo que é baseado em K-SNAP (ZHANG; TIAN; PATEL, 2010) para particionar uma coleção de usuários em diferentes grupos de interesse.

Guo *et al.* (GUO; ZHANG; YORKE-SMITH, 2015) empregou um algoritmo de agrupamento de multi-visão que considera taxas de usuários e relações de confiança social para melhorar a precisão e cobertura de SRs. Primeiro, os *clusters* são gerados usando o algoritmo *K-medoids*. Então, o modelo *Support Vector Regression* é adotado para prever itens para usuários que pertencem a mais de um *cluster*. Yu *et al.* (Yu et al., 2018) propõe um modelo de filtragem colaborativa de agrupamento de múltiplos conteúdos (MCCCF) para SRs. Este método aplica um agrupamento multi-visão na recomendação tradicional baseada em conteúdo para descobrir a relevância semântica latente entre itens e/ou usuários. Em seguida, este método usa o algoritmo kNN para explorar sua similaridade. Finalmente, a matriz de similaridade pode ser aplicada aos métodos tradicionais de CF.

Esta pesquisa bibliográfica não encontrou nenhum estudo que propõe o uso de algoritmos de SG como uma etapa de pré-processamento de dados para qualquer SRGC, nem que analise os impactos da SG no tempo de treinamento e na eficácia dos modelos de SRGC. Para realizar esta análise, propomos um método de SG baseado em latência que é simples, mas eficaz, e que combina *embedding* baseado em semântica latente (ComplEx) e *clustering* (K-Means)

em abordagens de única-visão e multi-visão.

4.2.5 Análise comparativa

Esta seção apresentam uma análise comparativa entre a segunda abordagem desta tese e os trabalhos relacionados a redução do volume de dados. A tabela 5 compara os trabalhos considerando:

- **Domínio de aplicação**: indica o escopo de aplicação da técnica proposta. Os domínio levantados foram Sistema de Recomendação (SR), Grafo de Conhecimento (GC), grafos em geral, classificação de padrões (*nearest neighbor algorithm*), CBR, clusterização.
- **Método de redução do volume de dados**: indica o método utilizado para mitigar o problema do volume de dados e, consequentemente, o problema de eficiência da solução. Os métodos levantados foram poda de instância, edição de base de casos, redução de dimensionalidade, Sumarização de Grafos (SG) e clusterização.
- **Técnica proposta ou adotada**: indica a técnica utilizada para implementar o método. As técnicas levantadas foram remoção de instâncias, remoção de casos, redução por *embedding*, sumarização por regularidade, particionamento de usuários, sumarização de grafo e redução por clusterização.
- **Tipo do dado de entrada**: indica o formato de entrada da técnica. Os tipos de dado levantados são anotações, base de casos, vetores de característica, grafo não rotulado e GC rotulado.
- **Representação de visões**: indica a estratégia adotada para gerar visões. Única representa a não geração de visões e reduz o conjunto de dados de entrada como um todo, enquanto que multi representa a geração de múltiplas visões que separa o conjunto de dados de entrada e reduz o volume de dados de modo estratificado.

Em Wilson e Martinez (1997), os autores adotam poda de instâncias para reduzir o volume de dados e melhorar a eficiência do processamento. Os autores de Leake e Wilson (1998) realizam a edição da base de casos. Em Vlachos et al. (2002) e Liu, Cheng e Qu (2020), os autores adotam técnicas baseadas em *embedding*. Em Fiorucci, Pelosin e Pelillo (2020), os autores propõem uma técnica de sumarização de grafos não-rotulados baseada no lema da regularidade. Os autores de Bickel e Scheffer (2004) desenvolvem e estudam algoritmos de clusterização multi-visão hierárquicos baseados em particionamento e aglomeração. Em Wu et al. (2015), os autores propõem uma técnica de particionamento de usuários baseada na sumarização de grafo e clusterização de conteúdo aplicada a SRs. Poucos trabalhos (RAGONE et al., 2017; WU et al., 2015) aplicam técnicas de sumarização de grafo em tarefas específicas de um SR e nenhum estudo encontrado propõem abordagens de sumarização para melhorar a eficiência e obter ganhos na eficácia da recomendação em SRGCs.

A segunda abordagem, apresentada no Capítulo 6 e publicada em Sacenti, Fileto e Willrich (2021), visa mitigar o custo computacional através da redução do volume da informação lateral, por meio de uma técnica de SG aplicada a SRGCs que combina *embeddings* com a clusterização de nodos *K-Means*. Esta técnica é aplicada a GC rotulados e permite a adoção de duas estratégias para a geração de visões: a única-visão e a multi-visão. Além disso, a segunda abordagem propõe uma metodologia de avaliação de eficiência e eficácia.

4.3 CONSIDERAÇÕES FINAIS

Este capítulo apresentou os trabalhos relacionados às duas abordagens propostas por esta tese que visam mitigar o problema do custo computacional causado pelo uso de informação lateral em SRCs. O capítulo descreve o método de busca adotado em cada pesquisa bibliográfica, as categorias de trabalhos levantados e, finalmente, compara os trabalhos mais relevantes com as abordagens propostas. A seguir, o Capítulo 5 apresenta a primeira abordagem da proposta desta tese (SACENTI; WILLRICH; FILETO, 2018) para mitigar o custo computacional da informação lateral por meio da conversão de ontologias em matrizes de preferência.

Tabela 6 – Tabela comparativa de trabalhos relacionados à segunda abordagem

Trabalho	Domínio	Método	Técnica	Tipo	Visões
Wilson e Martinez (1997)	classificação de padrões	poda de instâncias	remoção de instâncias	anotações	única
Leake e Wilson (1998)	CBR	edição de base de casos	remoção de casos	base de casos	única
Vlachos et al. (2002)	grafos em geral	redução de dimensionali.	*embedding*	vetores de caraterística	única
Fiorucci, Pelosin e Pelillo (2020)	grafos em geral	SG	regularidade	grafo não rotulado	única
Liu, Cheng e Qu (2020)	GC	SG	*embedding*	GC rotulado	única
Bickel e Scheffer (2004)	clusterização	clusterização	multi-visão hierárquica	documentos de texto	multi
Wu et al. (2015)	SR	clusterização e SG	particionam. de usuários	GC rotulado	única
Sacenti, Fileto e Willrich (2021)	SR	**SG**	***embedding* e *clustering***	**GC rotulado**	**multi**

5 UM SRH BASEADO EM ONTOLOGIAS MULTI-HIERÁRQUICAS

Como apresentado no capítulo 3, SRs que usam informações laterais sobre usuários e itens (p.ex., no domínio do filmes, atores, diretores e gêneros) se mostraram eficazes para mitigar o problema da esparsidade de interações usuário-item em Sistemas de Recomendação (SRs). O objetivo desta tese é investigar os impactos da representação e sumarização do conhecimento em sistemas de recomendação, tanto a nível de eficiência e eficácia, bem como propor abordagens para mitigar o problema de eficiência de SR que exploram informações laterais. Este capítulo apresenta a primeira abordagem da proposta desta tese, que investiga os impactos da representação de conhecimento em um SR baseado em ontologia (SROs) que utiliza uma técnica híbrida de recomendação. Esta abordagem mitiga o problema da eficiência de SRs baseados em ontologias (SROs) através da conversão da ontologia em uma representação na forma de matrizes de valores. Esta conversão evita o uso de técnicas de treinamento de alto custo computacional quando a ontologia é utilizada diretamente pelo SRO. Trata-se de uma técnica de formação de vizinhança baseada nas características de itens (FERNANDES; SACENTI; WILLRICH, 2017) combinada com um arcabouço conceitual para desenvolvimento e aperfeiçoamento de SROs (SACENTI; WILLRICH; FILETO, 2018).

Esta primeira abordagem explora algumas das vantagens de ontologias na área da recomendação. A primeira vantagem explorada é a possibilidade de criar ontologias de diferentes níveis de abstração identificando o conhecimento de domínio do item para tornar o SRO mais independente do domínio do item a recomendar. Nesta tese, a independência de domínio é a capacidade de reuso ou fácil adaptação de um SR a um determinado domínio de itens a recomendar (p.ex., filmes, músicas, pessoas).

A segunda vantagem explorada é a facilidade de compartilhamento de conhecimento bem-formalizado por ontologias ou dados conectados (DCs) publicados na Web que descrevam a informação lateral tanto sobre usuários e suas preferências, quanto sobre os itens a recomendar. Em geral, dados sobre usuários são obtidos pelo SRO no próprio sistema alvo da recomendação, enquanto que os dados sobre os itens são geralmente coletados de fontes de dados externas, por exemplo pelo reuso de ontologias e dados conectados.

Uma terceira vantagem é que as ontologias permitem descrever de forma formal e interpretável por máquina os mais diversos aspectos dos usuários (p.ex., informações demográficas, como idade, sexo e ocupação) e itens (no domínio de filme, pode-se citar gêneros, atores, diretores e data de lançamento). Esta descrição forma inclui a possibilidade de descrever relações hierárquicas, onde a informação lateral sobre determinado aspecto pode ser organizada na forma de hierarquias (taxonomias).

Como apresentado na Seção 4.1.4, os SROs em geral não exploram completamente as ontologias e o conhecimento explicitado é limitado, descrevendo apenas uma hierarquia de conceitos (taxonomia) de item e ignorando outros aspectos relacionados diretamente ao item (p.ex., no domínio de filmes: duração, data de lançamento, ator e diretor) ou indiretamente (p.ex., local de nascimento do ator principal).

Para demonstrar os impactos do uso de ontologias em SR, que explore todas as vantagens citadas anteriormente, este capítulo propõe um arcabouço conceitual (um conjunto de conceitos, ferramentas e métodos para desmembrar um problema específico) para construção de SROs que representa a informação lateral considerando diversos aspectos de item e múltiplas hierarquias de entidades, e também explora o uso de ontologias de diferentes níveis de abstração para promover a independência de domínio. Além disso, a solução proposta reusa o conhecimento sobre itens disponível em fontes de dados conectados.

Para facilidade de referência, o arcabouço proposto é denotado por ORBS (anagrama para a sigla SRbO: Sistema de Recomendação baseado em Ontologia). A palavra da língua inglesa *orbs* traduz-se para o termo orbes, que significa[1] corpo esférico ou circular, bola, esfera, globo; mundo, universo; corpo celeste; ou área de atividade. Além disso, o formato circular de um orbe remete a representação gráfica de entidades em ontologias (ilustrada na Seção 2.1) e pelo contexto deste termo no misticismo (bola de cristal), como a ferramenta de leitura de sorte, sugestão de destino e predição do futuro.

Este capítulo está organizado em três seções. A primeira, a Seção 5.1, apresenta uma visão geral de ORBS e é seguida por subseções que detalham cada uma das etapas de ORBS, incluindo a separação da descrição do conhecimento nas ontologias de domínio, de perfil de usuário e aplicação, a determinação de preferências do usuário e o mapeamento da ontologia de aplicação para o modelo de dados do algoritmo de recomendação. De acordo com o método de pesquisa desta tese (Seção 1.4), essa seção apresenta o resultado das Etapas 2 e 3. A segunda, Seção 5.2, apresenta os experimentos e resultados obtidos com um protótipo prova-de-conceito implementando algumas funcionalidades de ORBS. Essa seção apresenta o resultado das Etapas 4 e 5 do método de pesquisa desta tese. Finalmente, a terceira seção apresenta as considerações finais sobre a primeira abordagem da proposta desta tese.

5.1 ORBS: UM ARCABOUÇO CONCEITUAL PARA A CONSTRUÇÃO DE SRO

A Figura 4 oferece uma visão geral do arcabouço conceitual ORBS, ilustrando os principais atores, componentes e artefatos envolvidos em diferentes etapas do processo da recomendação. Nesta figura, os itens numerados (de I a VII) se referem às etapas do processo de recomendação. Na esquerda da Figura 4 estão os atores do mundo real, os usuários que interagem com os itens a recomendar. Esta proposta considera que os itens servem de informação base para anotações semânticas que descrevem os diferentes aspectos por meio de entidades de ontologias. Estas anotações semânticas são geradas a partir a informação lateral adquirida no sistema alvo da recomendação ou em fontes de dados conectados. O lado direito da Figura 4 ilustra três artefatos que são ontologias de diferentes níveis de abstração que permitem expressar a informação lateral necessária para o processo de recomendação. Finalmente, no centro da Figura 4 está o próprio ORBS.

[1] Disponível em: https://www.priberam.pt/dlpo/orbes, acesso em: 22/11/2021.

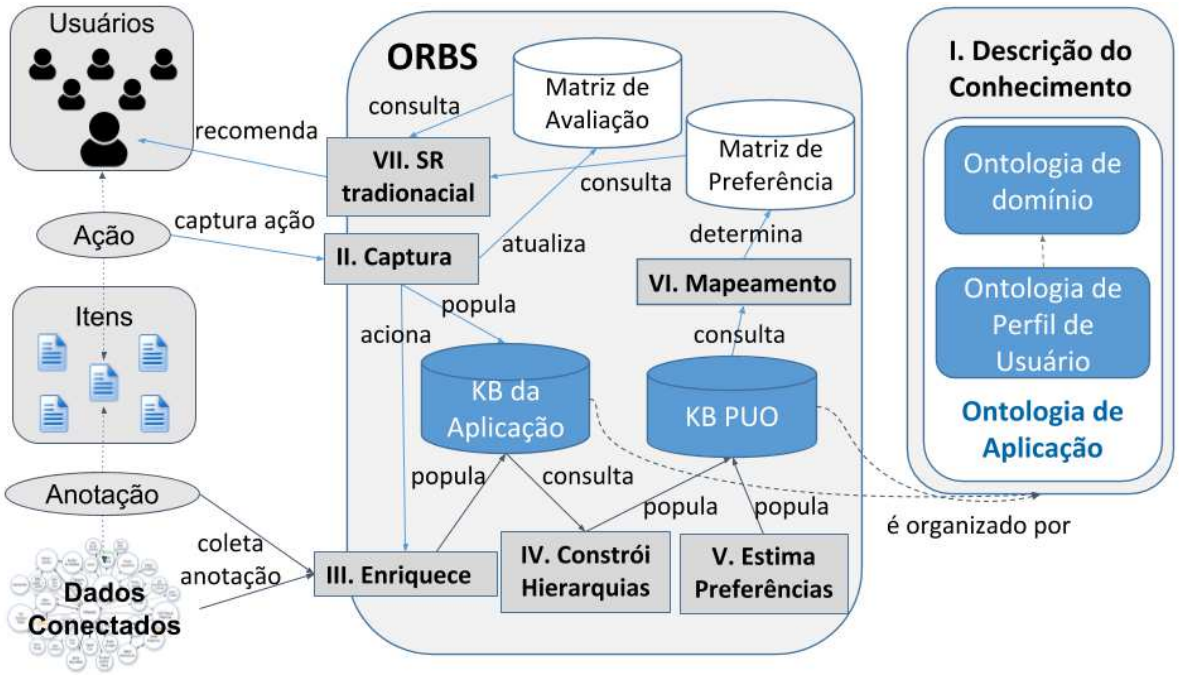

Figura 4 – Visão geral preliminar da proposta.

Fonte: criado pelo autor.

Para a representar o Perfil Ontológico de Usuário (POU), ORBS tem como entrada as interações usuário-item, as anotações semânticas e as ontologias. ORBS tem como premissa que as preferências de um usuário sobre os itens podem ser determinadas, em parte, pela observação da informação lateral relacionada aos itens que o usuário interage. Neste capítulo, o aspecto (propriedade, característica) do item é chamado de *fator de interesse* (SALLES, 2017). O *fator de interesse* é representado por uma propriedade que relaciona itens de interações usuário-item com entidades anotadas semanticamente a estes itens e que descrevam um mesmo aspecto do filme, como formalmente descrito pela Definição 7.

Definição 7. Seja $u \in U$ o conjunto de usuários, $i \in I$ o conjunto de itens, $int \in Int$ um conjunto de interações usuário-item relacionando os conjuntos U e I, $e \in E$ o conjunto de entidades relacionadas a informação lateral sobre os itens em I, e $as \in AS$ o conjunto de anotações semânticas na forma de $as = \langle i, p, e \rangle$ relacionando I e E por propriedades $p \in P$. Um *fator de interesse* $f \in F$ é uma propriedade p que representa um determinado aspecto de item, conectando itens $i \in I$ a entidades $e \in E$ relacionadas a este aspecto e onde o conjunto de fatores de interesse é um subconjunto do conjunto de propriedades $F \subset P$.

O *fator de interesse* está relacionado a uma ou mais entidades que, por sua vez, podem servir como um critério para a ordenação dos itens quando organizadas em uma hierarquia de entidades. Neste sentido, a informação lateral sobre itens no ORBS é representa como um conjunto de *hierarquias de fator de interesse*, como a formalmente descrita pela Definição 8.

Definição 8. Seja I o conjunto de itens, E o conjunto de entidades relacionadas a informação lateral sobre os itens em I, f um *fator de interesse* relacionando I a $e \in E_f$, onde $E_f \subset E$. Uma *hierarquia de fator de interesse* H_f é uma hierarquia de entidades H cujas entidades pertencem a E_f.

Para promover maior independência de domínio ao SRO, os conceitos de *fator de interesse* e *hierarquia de fator de interesse* devem ser descritos de maneira genérica na ontologia de ORBS, separado do conhecimento específico do domínio do item. Para tal, é descrita a ontologia de perfil de usuário que isola conceitos de alto nível da ontologia de domínio. Quando combinadas em uma ontologia de aplicação, definem o POU do SRO construído por ORBS.

Como ilustrado na Figura 4, as etapas do processo de recomendação de ORBS são apresentadas a seguir:

- **I. Descrição do Conhecimento**: trata-se de uma etapa de adaptação do ORBS ao domínio do item a recomendar. Esta etapa utiliza ontologias de três diferentes níveis de abstração para descrever o conhecimento requerido pelo processo de recomendação: ontologia de domínio, ontologia de perfil de usuário e ontologia de aplicação. A ontologia de domínio descreve o vocabulário relacionado a um domínio de item (p.ex., filmes, músicas, vagas de emprego). A ontologia de perfil de usuário descreve o vocabulário relacionado ao modelo do POU, independente do domínio de item. A ontologia de aplicação descreve o vocabulário relacionado a um SRO específico, anotando conceitos e instâncias de ontologias de domínio com os conceitos e instâncias da ontologia de perfil de usuário, definindo *fatores de interesse* e *hierarquias de fator de interesse*.

- **II. Captura de Dados**: trata-se da captura de interações usuário-item. Cada interação relevante ao processo de recomendação será mantida na Base de Conhecimento (BC) sobre usuários e itens, chamada BC da Aplicação. Caso a interação seja uma avaliação sobre o item, esta interação também implicará na atualização de uma Matriz de Avaliação Usuário-Item.

- **III. Enriquecimento Semântico**: nesta etapa, a cada nova captura de interação, ou então em tempos regulares, a BC da Aplicação é enriquecida a partir da coleta ou geração de anotações semânticas sobre os itens oriundas do sistema alvo de recomendação ou de fontes de dados conectados.

- **IV. Construção das *hierarquias de fator de interesse***: esta etapa é responsável pela construção semiautomática de *hierarquias de fator de interesse* que representam a informação lateral do SRO.

- **V. Determinação das preferências**: nesta etapa, as entidades de *hierarquias de fator de interesse* são ponderadas (anotadas) com valores numéricos representando o grau de preferência de um determinado usuário por cada entidade, construindo os POUs.

- **VI. Determinação da Matriz de Preferência**: na corrente versão de ORBS, trata-se da conversão dos POUs em vetores de entidades ponderadas que compõem a Matriz de Preferência que serve como entrada para o SR proposto por Fernandes, Sacenti e Willrich (2017).
- **VII. Sistema de Recomendação (SR) tradicional**: na corrente versão de ORBS, trata-se do SR proposto por Fernandes, Sacenti e Willrich (2017), não-semântico, que gera recomendações com base nas matrizes de Preferência e de Avaliação Usuário-Item.

Como visto na descrição das etapas acima, primeiramente são determinadas as preferências dos usuários na forma de POUs que, então, são convertidas para Perfis de Usuário (PUs) não-semânticos no formato requerido por um SR. A conversão da ontologia em PU não-semântico é a forma adotada nesta abordagem para mitigar os problemas de eficiência dos SROs. Esta proposta se baseia na hipótese que a determinação da recomendação diretamente a partir da BC não é escalável (GAUCH et al., 2007). Por isto, é proposto o uso da Matriz de Preferência gerada a partir da BC de POUs, ao invés de utilizar diretamente a BC de POUs para produzir recomendações. O SR adotado na corrente versão de ORBS aplica uma técnica de recomendação híbrida, considerando as interações usuário-itens representadas na Matriz de Avaliação Usuário-Item e as preferências dos usuários pelo conteúdo do item (entidades de *hierarquias de fator de interesse*) representado pela Matriz de Preferência. Então, ambas as matrizes são processadas por uma técnica de filtragem colaborativa baseada em usuário. Note que as etapas VI e VII podem ser modificadas para adaptar ORBS a outras técnicas de recomendação.

5.1.1 Descrição do Conhecimento

Dado o requisito de que ORBS promova a independência de domínio de SROs e, inspirado nas propostas de Silva (2015) e Salles (2017), a descrição de conhecimento é realizada com base em ontologias com três diferentes níveis de abstração: ontologia de domínio, ontologia de perfil de usuário e ontologia de aplicação[2].

5.1.1.1 Ontologia de Domínio

Como visto na Seção 2.1.2, uma ontologia de domínio descreve o vocabulário relacionado a um domínio (p.ex., filmes, músicas, vagas de emprego), ao especializar conceitos pertencentes a uma ontologia de alto nível (GUARINO et al., 1998).

No contexto de ORBS, o conhecimento sobre um determinado item pode ser explicitado com base em uma ontologia do domínio relacionado ao item. Por exemplo, caso os itens a recomendar sejam músicas, a ontologia de domínio *Music Ontology*[3] poderia ser adotada como

[2] Acesso: https://github.com/juarezsacenti/ORBS/tree/master/src/resources/main/importedOntologies, em: 22/11/2021.

[3] Acesso: http://purl.org/ontology/mo/, em: 22/11/2021.

ontologia de domínio. Outro exemplo é a ontologia de domínio *Movie Ontology*[4] (MO), usada para exemplificar as etapas do processo de recomendação de ORBS. Esta ontologia disponibiliza um vocabulário controlado de conceitos relacionados ao domínio de filmes. A Figura 5 apresenta os principais conceitos da MO, que inclui: *dbo:Film*[5] (filme), *mo:Genre*[6] (gênero), *dbo:director* (diretor), *dbo:Actor* (ator), *mo:Award* (premiação) e *dbo:Language* (língua).

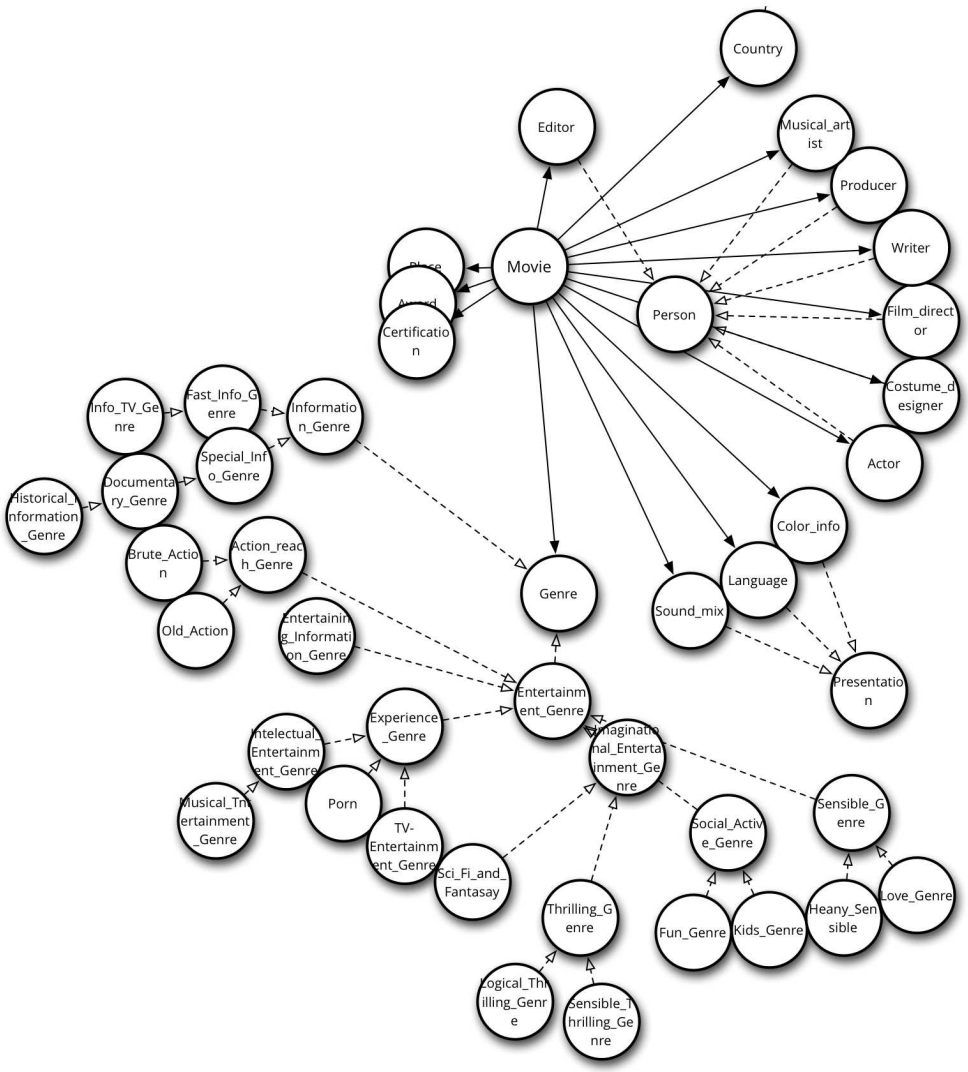

Figura 5 – Fragmento da Movie Ontology.

Fonte: Corte da figura disponível em http://movieontology.org/documentation/, acessada em: 14/06/2018.

ORBS utiliza ontologias de domínio para descrever os *fatores de interesse* e *hierarquias de fator de interesse* que influenciam no grau de relevância de um item para um determinado usuário e que devem compor o POU.

[4] Acesso: http://www.movieontology.org:80/2010/01/movieontology.owl, em: 14/06/2018.
[5] O prefixo *dbo* refere-se a URL: http://dbpedia.org/ontology/
[6] O prefixo *mo* refere-se a URL: http://www.movieontology.org/2009/10/01/movieontology.owl#

5.1.1.2 Ontologia de Perfil de Usuário

A ontologia de perfil de usuário descreve entidades e propriedades genéricas usadas para representar (ou explicar) o POU, sendo uma ontologia independente do domínio de item (KATIFORI et al., 2007). Na classificação de Guarino et al. (1998), esta ontologia pode ser considerada parcialmente de alto nível, visto que é independente de domínio, porém dependente do problema de recomendação.

Esta ontologia genérica representa o conhecimento acerca do usuário, em particular as interações de usuários com os itens. Além disso, esta ontologia permite expressar quais *fatores de interesse* relacionados aos itens influenciam na determinação de relevância de um item, bem como as *hierarquias de fator de interesse*.

A ontologia de perfil de usuário adotada por ORBS tem como ponto de partida a ontologia RecOnt, proposta por Salles (2017). Como descrito na Seção 4.1.2, esta ontologia define conceitos genéricos, como *Audiência*, *Item*, *Fator de Interesse*, e propriedades como *acessou*, para indicar que um usuário acessou um item, e *temFI*, que relaciona um item ao seu *Fator de Interesse*. Como indicado na Seção 4.1.2, a RecOnt possui algumas limitações (p.ex., a declaração de *fatores de interesse* não é baseada em propriedades de objeto e/ou propriedades de dados), o que não permite seu reuso nesta tese.

Para construção da ontologia de PU para SROs, foi adotada a metodologia proposta por Noy e McGuinness (2001). A Figura 6 mostra um fragmento preliminar da ontologia de PU. Os conceitos *Usuario*, *Acao*, *Item* e as propriedades de objeto *fazInteracao*, *temDestino* descrevem como as interações usuário-item são representadas na BC de Aplicação de ORBS. Os conceitos *Categoria* (entidade de uma hierarquia) e *FatorDeInteresse* e as propriedades *rdfs:subClassOf* e *ehDoFator* descrevem *hierarquias de fator de interesse* ordenando itens a respeito de diferentes *fatores de interesse* do usuário. O conceito *Interesse* representa a influência de um *fator de interesse* na preferência de um usuário. O peso da influência de um *fator de interesse* na preferência de um usuário pode ser especificado manualmente, capturado explicitamente do usuário, ou computacionalmente inferido (não abordado nesta tese).

5.1.1.3 Ontologia de Aplicação

É importante observar que nem todo conhecimento especificado a partir da ontologia de domínio do item é relevante para o processo de recomendação. Por exemplo, conceitos como *dbo:Language* da MO (Seção 5.1.1.1) pode apresentar pouca variação de valores para o conjunto de dados da aplicação (CANTADOR; BELLOGIN; CASTELLS, 2008), de modo a apresentar pouca influência na decisão de uma pessoa gostar ou não de um filme.

No contexto de ORBS, uma ontologia de aplicação anota entidades e propriedades da ontologia de domínio utilizando o vocabulário definido pela ontologia de perfil de usuário. Em outros termos, esta ontologia permite expressar quais entidades e propriedades da ontologia de domínio são utilizados para definir o POU.

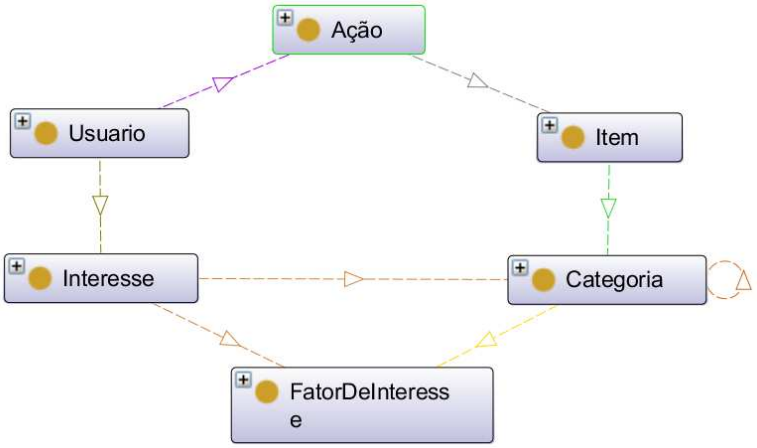

Figura 6 – Fragmento da ontologia RecOnt2.

Fonte: adaptado pelo autor de Silva (2015) e Salles (2017).

A Figura 7 exemplifica com um fragmento da ontologia de aplicação de ORBS. Esta figura contém elipses que representam entidades, retângulos que representam valores de propriedades de dados e arestas direcionais rotuladas que representam declarações de propriedades de dados e objetos.

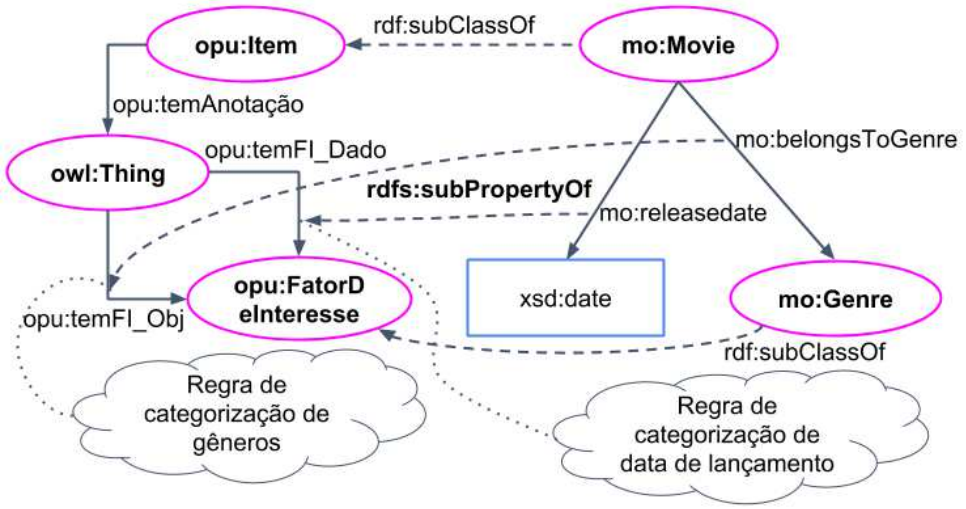

Figura 7 – Fragmento da ontologia de aplicação de ORBS

Fonte: criado pelo autor.

Nesta figura, as entidades rotuladas com o prefixo *opu*[7] em negrito representam os conceitos da ontologia de PU. Estes conceitos anotam os conceitos de Movie Ontology *mo:Movie* e *mo:Genre*. Além dos conceitos, as propriedades *mo:releasedate* e *mo:belongsToGenre* são anotadas como subpropriedade da propriedade *opu:temFI*, indicando quais *fatores de interesse*

[7] Prefixo da Ontologia de Perfil de Usuário

devem ser considerados pelo SRO. As regras de categorização serão detalhadas na seção 5.1.4, e definem como construir *hierarquias de fator de interesse*.

5.1.2 Captura e Representação Semântica dos Dados

Nesta proposta, a captura de dados é feita em modo híbrido, considerando tanto dados de avaliações explícitas dos itens fornecidas pelo usuário, quanto dados sobre os itens que sofreram qualquer tipo de interação implícita por parte do usuário (p.ex., visualização, compra, critério de busca utilizado).

Todos os dados capturados por ORBS são representados na BC da Aplicação, na forma de indivíduos dos conceitos *opu:Usuario*, *opu:Acao* e *opu:Item*, ou de subpropriedades das propriedades de objeto *opu:fazInteracao*, *opu:temDestino* da ontologia de perfil de usuário, conforme o exemplo da Subseção 5.1.1.3.

No exemplo ilustrado pela Figura 8 discorre sobre o usuário de identificação número 53, suas interações com três filmes distintos. Este usuário, representado pelo indivíduo (elipse azul) *u53*, interagiu por meio das interações representadas pelos indivíduos (elipses pretas) *a1*, *a2* e *a3* com os filmes *Matrix, The (1999)* e *Rambo: First Blood Part II (1985)* e *Lord of the Rings, The (1978)*, avaliando-os com as notas 5, 3 e 5 respectivamente.

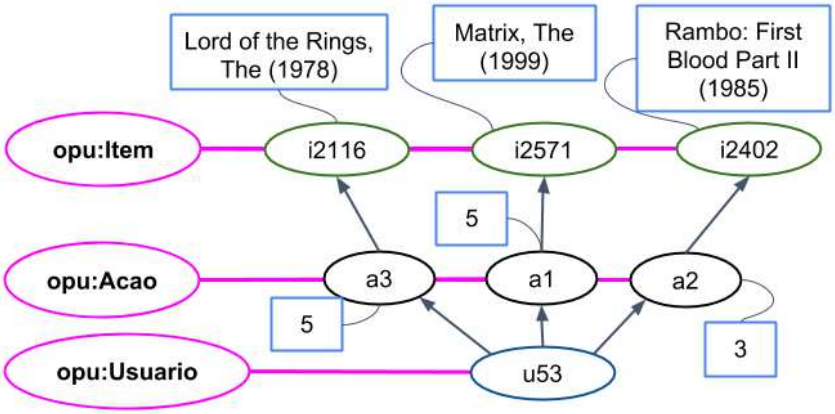

Figura 8 – Fragmento da ontologia de aplicação de ORBS - Etapa de captura

Fonte: criado pelo autor.

As arestas indicam a existência de uma declaração RDF. Por exemplo, as 5 triplas: *{u53, rdf:type, opu:Usuario}*, *{a1, rdf:type, opu:Acao}*, *{i2402, rdf:type, opu:Item}*, *{u53, opu:fazInteracao, a2}*, *{a2, opu:temDestino, i2402}*; representam declarações RDF que descrevem uma interação entre o usuário *u53* e o item *i2402*. A interação pode representar uma avaliação numérica por meio de uma declaração *{a2, opu:valorAvaliacaoNumerica, 3}* indicando que a nota 3 foi dada ao item *i2402* pelo usuário *u53*.

5.1.3 Enriquecimento semântico da base de conhecimento

A etapa de enriquecimento semântico da BC gera ou coleta anotações sobre itens do sistema alvo e de fontes externas, como dados conectados. Esta geração ou coleta ocorre apenas quando uma interação usuário-item tem destino a um novo item ainda não representado na BC de Aplicação, ou quando esta BC é atualizada. As anotações semânticas seguem o vocabulário estabelecido pela ontologia de aplicação.

Por exemplo, anotações de item como a data de lançamento e o gênero de um filme podem ser coletadas do sistema alvo de anotação e descritas na base de conhecimento com as propriedades de MO: *mo:releasedate* e *mo:belongsToGenre*, segundo exemplificado na Seção 5.1.1.3. No exemplo ilustrado pela Figura 9, os indivíduos (elipses verdes) *mo:Thriller*, *mo:Sci-Fi*, *mo:Action* e *mo:War* representam entidades conectadas aos itens *i2571* e *i2402* por meio de anotações semânticas com a propriedade *mo:belongsToGenre*. Já as declarações: *{i2402, mo:BelongsToGenre, mo:Action}* e *{i2402, mo:releasedate,"1985"}* representam anotações semânticas que conectam o item *i2402* ao gênero *Action* e a data de lançamento *1985*.

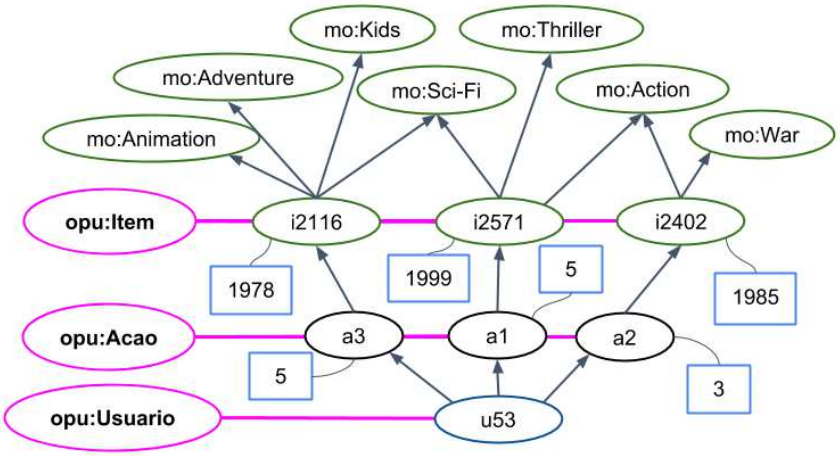

Figura 9 – Fragmento da ontologia de aplicação de ORBS - Etapa de enriquecimento

Fonte: criado pelo autor.

Anotações do item coletadas de fontes externas exigem uma conexão entre os itens do sistema alvo e as entidades desta fonte. Em dados conectados, estas conexões também podem ser expressas por meio de anotações semânticas utilizando propriedades como *owl:sameAs*[8].

5.1.4 Construção de hierarquias

ORBS representa POUs como *hierarquias de fator de interesse*, i.e. hierarquias de entidades que estão relacionadas a um *fator de interesse* comum e que permitem ordenar itens. Uma hierarquia de entidade permite representar com maior precisão quais entidades são mais

[8] Descrição disponível em: https://www.w3.org/TR/owl-ref/#sameAs-def, acesso em: 22/11/2021

similares (estão mais próximas). Consequentemente, a representação do POU por meio de *hierarquias de fator de interesse* tem potencial para influenciar na eficácia de SRs (Pergunta 2).

Entidades relacionadas a aspectos de itens e descritas por ontologias de domínio podem ser pré-processadas durante a etapa de enriquecimento e organizadas em hierarquias de entidades, relacionado estas entidades com a entidade *opu:Categoria* da ontologia de perfil de usuário. Deste modo, as *hierarquias de fator de interesse* podem ser descritas manualmente pelo gerente de ORBS, como a hierarquia de data de lançamento de filmes descrita mais adiante na Seção 5.2.2.1, ou ser obtidas a partir do reuso de ontologias de domínio, como a hierarquia de gênero de MO.

Por exemplo, considere o *fator de interesse mo:belongsToGenre* que conecta itens a entidades do tipo *mo:Genre* (gêneros). A Figura 10 apresenta a hierarquia que ordena as entidades deste *fator de interesse*. Como exemplificado na Seção 5.1.1.3, as 6 triplas: *{mo:Action, rdf:type, mo:Brute_Action}*, *{War, rdf:type, mo:Brute_Action}*, *{mo:Brute_Action, rdfs:subClassOf, mo:Action_Reach}*, *{mo:Action_Reach, rdfs:subClassOf, mo:Entertainment}*, *{mo:Entertainment, rdfs:subClassOf, mo:Genre}*, *{Animation, ehDoFator, mo:Genre}*; representam parte da hierarquia de gêneros de *Movie Ontology* (MO).

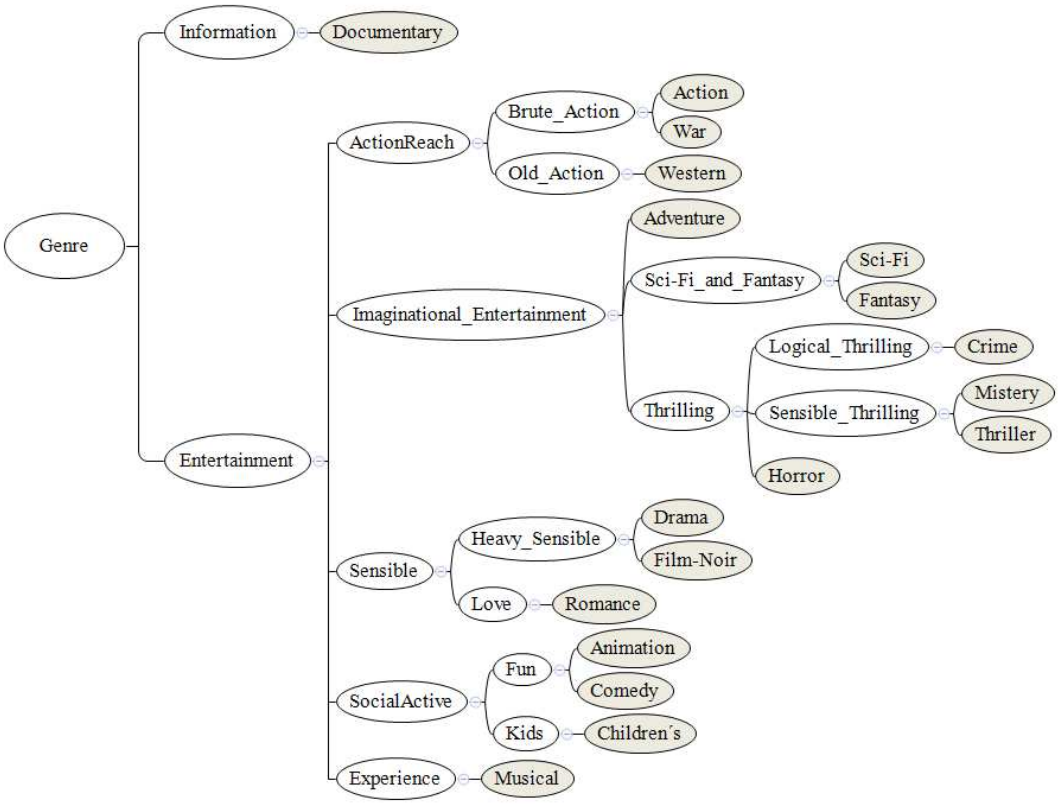

Figura 10 – Hierarquia do fator de interesse gênero

Fonte: (FERNANDES; SACENTI; WILLRICH, 2017).

As *hierarquias de fator de interesse* podem necessitar mais de uma propriedade para conectar entidades a outras entidades e aos itens. Por exemplo, a hierarquia de gêneros neces-

sita das propriedades: *mo:belongsToGenre*, que conecta itens a gêneros; *rdf:type*, que conecta gêneros a conceitos de gêneros abstratos (p.ex., *Brute Action*); e *rdfs:subClassOf*, que conecta conceito de gêneros abstratos a conceitos de gêneros mais abstratos (p.ex., *Action Reach*).

O mapeamento de propriedades da ontologia de domínio para *hierarquias de fator de interesse* já foi realizado em trabalhos anteriores, especificamente na etapa de construção de hierarquias de Sacenti et al. (2015), que tem como entrada uma lista de propriedades que orienta a exploração da ontologia, iniciando pela anotação semântica de um item. Esta lista é aqui chamada de regra de categorização, já referenciada na Seção 5.1.1.3.

A Figura 11 apresenta a BC de Aplicação após as etapas de captura, de enriquecimento e da construção de hierarquias. Os indivíduos (elipses verdes) *mo:Thriller*, *mo:Sci-Fi*, *mo:Action* e *mo:War*, e os conceitos (elipses roxas) *mo:Sensible_Thrilling*, *mo:Thrilling*, *mo:Sci-Fi_and_Fantasy*, *mo:Imaginational_Entertainment*, *mo:Brute_Action*, *mo:Action_-Reach*, *mo:Entertainment* e *mo:Genre* representam os gêneros da ontologia de domínio MO. A influência do *fator de interesse* gênero na preferência do usuário *u53* é representada pelo indivíduo *int1*, cujo peso de influência atribuído é 2. Como dito anteriormente, o peso da influência é especificado manualmente na corrente versão de ORBS e a inferência deste peso não é abordada nesta tese. Embora não ilustrado nesta figura, o indivíduo *int1* também está relacionado ao usuário *u53* pela propriedade *opu:temInteresse*. As 6 declarações: *{mo:Action, rdf:type, mo:Brute_Action}*, *{War, rdf:type, mo:Brute_Action}*, *{mo:Brute_Action, rdfs:subClassOf, mo:-Action_Reach}*, *{mo:Action_Reach, rdfs:subClassOf, mo:Entertainment}*, *{mo:Entertainment, rdfs:subClassOf, mo:Genre}*, *{Animation, ehDoFator, mo:Genre}*; representam parte da hierarquia de gêneros de MO.

5.1.5 Determinação de Preferências dos Usuários

O POU de um usuário é representado por um segmento da ontologia de aplicação, tendo como ponto central o indivíduo representando o usuário e todas as entidades diretamente e indiretamente relacionadas a este usuário (interações, itens, *fatores de interesse*, *hierarquias de fator de interesse*). A etapa de determinação de preferências determina, para cada uma das entidades relacionadas a item do POU, a preferência do usuário por itens conectados àquela entidade.

Em trabalho anterior (SACENTI et al., 2015), é proposta uma técnica de customização de hierarquias de conceitos com base na frequência de uso destes conceitos em anotações semânticas. Nesta tese, esta técnica será adaptada para contar o número de interações usuário-item relacionadas diretamente ou indiretamente a entidades conectadas aos itens e ordenadas por hierarquias. Note que, além desta medida, diferentes técnicas podem ser consideradas para estimar a preferência do usuário por entidades conectadas aos itens, como *spreading activation* e *random walk*.

Esta etapa, considerando apenas a técnica de contagem de interações em hierarquias, calcula para cada *fator de interesse* a frequência de interações de um usuário para cada entidade

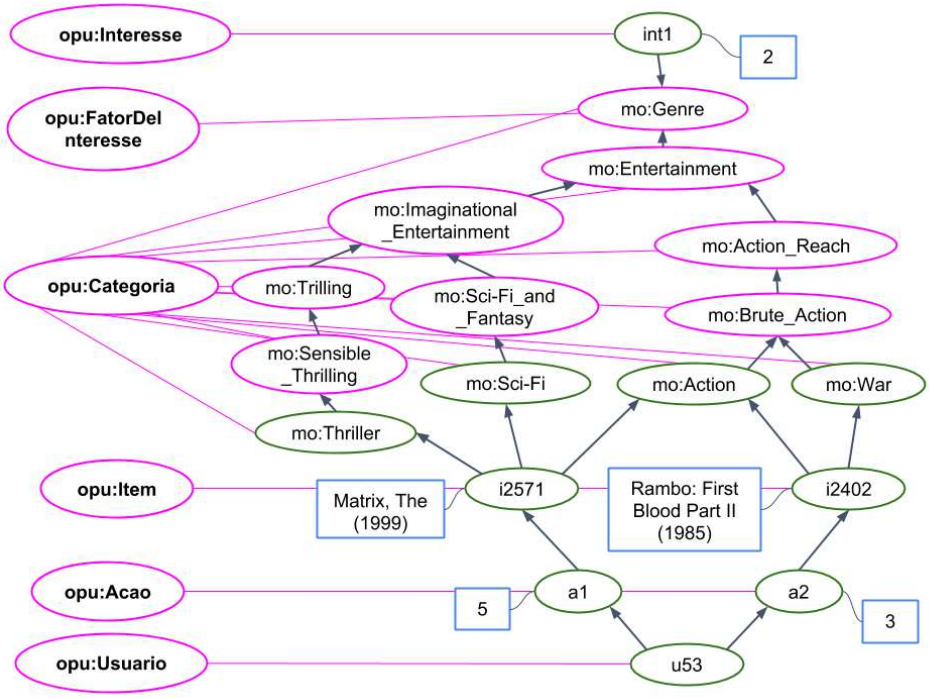

Figura 11 – Fragmento da ontologia de aplicação de ORBS - Etapa de construção de hierarquia

Fonte: criado pelo autor.

pertencente a hierarquia deste fator. O item de uma interação pode estar conectado a diferentes entidades de uma hierarquia. Por exemplo, na Figura 11 o item *i2116* (*"The Lord Of The Rings, 1978"*) está conectado às entidades *Adventure*, *Animation*, *Children's*, *Sci-Fi* da hierarquia de gênero.

Dado um conjunto de interações usuário-item $int \in Int$ na forma $int = \langle u, i, v \rangle$ (Definição 6), um conjunto de anotações semânticas $as \in AS$ na forma $as = \langle i, p, e \rangle$ (Definição 5) conectado itens a entidades e uma hierarquia de entidades $H(E,D)$ (Definição 3) tal que $e \in E$, podemos classificar a relação entre a informação base da anotação (a interação usuário-item ou o item) e uma entidade da hierarquia como direta (Definição 9) e indireta (Definição 10).

Definição 9. Uma interação usuário-item $int = \langle u, i, v \rangle$ e uma entidade da hierarquia $e \in H$ possuem uma relação direta quando existir uma anotação semântica $as = \langle i, p, e \rangle$, i.e. quando a entidade e é uma anotação semântica do item i.

Definição 10. Uma interação usuário-item $int = \langle u, i, v \rangle$ e uma entidade e_i da hierarquia $H(E,D)$, tal que $e_i \in E$, possuem uma relação indireta se e somente se existir uma relação direta $\langle i, p, e_d \rangle$, tal que $e_d \in E$, e uma cadeia de declarações RDF $d \in D$ em H que relacione e_d e e_i.

Por exemplo, na Figura 11 o item *i2116* (*"The Lord Of The Rings, 1978"*) apresenta relações diretas com apenas 4 entidades da hierarquia de gênero. Entretanto, o item *i2116* possui relações indiretas com 7 outras entidades, sendo uma delas *Imaginational_Entertainment* através das cadeias de declarações RDF: *(Sci-Fi, rdf:type, Sci-Fi_and_Fantasy)* e *(Sci-Fi_and_Fan-*

tasy,rdfs:subClassOf, Imaginational_Entertainment), ou apenas *(Adventure, rdf:type, Imaginational_Entertainment)*.

Diferentes técnicas de contagem de interações em hierarquias podem ser adotadas: por exemplo, contagem de interações específicas (avaliação de estrelas, *like-dislike*, acesso), contagem do valor das avaliações, contagens de anotações diretas ou indiretas, contagens considerando diferentes técnicas de propagação (*spread activation*) de valores entre entidades diretamente relacionadas. Neste trabalho, para cada entidade relacionada ao item é contado o número de interações usuário-item (p.ex., do tipo avaliação ou acesso) cujo item possui relação direta ou indireta com aquela entidade, desconsiderando o valor de avaliações, conforme descrito pela Definição 11.

Definição 11. Seja Int_u um conjunto de interações usuário-item de um usuário u, $i \in I$ o conjunto de itens de Int_u, $as \in AS$ o conjunto de anotações semânticas na forma de $as = \langle i, p, e \rangle$ conectando itens a entidades, e uma hierarquia de entidades $H(E,D)$ que organiza as entidades de AS. A frequência de uso $freq(e,u,H)$ de uma entidade $e \in E$ com respeito a H é o número de interações usuário-item distintos em $int \in Int_u$ tal que exista $as \in AS$ conectando o item alvo da interação de u a e (conexão direta) ou que exista uma ou mais $d \in D$ em H que, combinadas com as e int, conecte u a e (conexão indireta ou caminho).

Por exemplo, a Figura 12 exemplifica a contagem da frequência de uso de entidades de um subgrafo da hierarquia de gênero para o usuário *u53*. Os nodos do subgrafo da hierarquia de gênero são rotulados pelo nome da entidade seguido pelo número de itens relacionado direta ou indiretamente com aquela entidade. As arestas entre nodos brancos representam a propriedade *rdfs:subClassOf* e entre nodos branco e cinza, a propriedade *rdf:type*. As arestas pontilhadas entre *u53* e nodos são as relações diretas entre os itens de interações de *u53* e entidades da hierarquia de gênero.

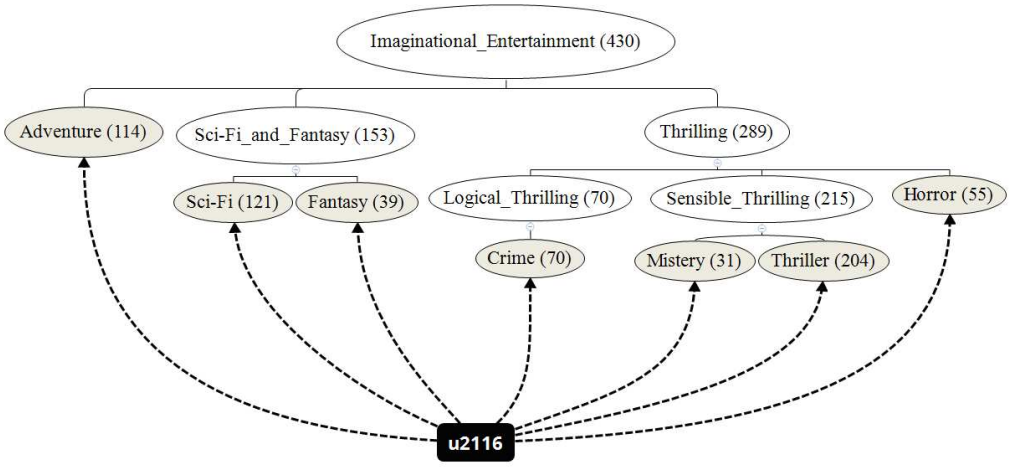

Figura 12 – Exemplo de contagem de interações de usuário em fragmento da hierarquia de gênero

Fonte: criado pelo autor.

Na Figura 12, a entidade *Sci-Fi_and_Fantasy* está direta ou indiretamente relacionada a 153 interações, enquanto *Sci-Fi* está a 121 e *Fantasy* a 39. A entidade *Sci-Fi_and_Fantasy* está relacionada a um número de interações inferior à soma de interações de suas subcategorias *Sci-Fi* e *Fantasy* pois um item pode relacionar-se com mais de uma entidade. Do mesmo modo, caso houvesse interações usuário-item diretamente relacionadas a *Sci-Fi_and_Fantasy*, ela poderia estar ponderada com a um número superior à soma de interações de suas subcategorias.

5.1.6 Mapeamento de POUs para Matriz de Preferência

Como já citado, o grande volume e diversidade de informações que forma o POU geralmente provocam a redução da eficiência de SROs. Um modo de mitigar este problema é converter o POU em estruturas de dados mais eficientes e escaláveis, como as de técnicas clássicas de recomendação. Na etapa de mapeamento ocorre a transformação de POUs descritos na ontologia de aplicação para PU descritos de acordo com o modelo de dados empregado pelo SR adotado por ORBS.

Nesta seção, definimos como *hierarquias de fatores de interesse* são mapeadas para vetores de entidades ponderadas, considerando os pesos da influência dos fatores de interesse na preferência de um usuário manualmente descritos pelo gerente de ORBS na ontologia de aplicação. Estes vetores compõem a Matriz de Preferência que serve como entrada para o SR proposto por Fernandes, Sacenti e Willrich (2017), adotado pela corrente versão de ORBS.

Definição 12. Seja F um conjunto de *fatores de interesse*, $peso(u, f)$ o peso do interesse de u pelo *fator* $f \in F$, H_f uma hierarquia do fator de interesse f, e $freq(e, u, H_f)$ a frequência de ocorrência de uma entidade e em interações do usuário u conforme uma hierarquia H_f. O vetor de conceitos ponderados V_u de um usuário u tem dimensão equivalente a soma do número de entidades H_f para cada *fator de interesse* $f \in F$, e $v_{u,e} \in V_u$ equivale a frequência de uso $freq(c, H_{u,f})$ multiplicada pelo peso do interesse $peso(u, f)$, representado pela Equação 5.1.

$$v_{u,e} = peso(u, f) \times freq(e, u, H_{u,f}) \tag{5.1}$$

Por exemplo, considere novamente a Figura 12. O vetor de entidades ponderadas do usuário $u2116$ ordenado pelas entidades: *Adventure*, *Sci-Fi*, *Fantasy*, *Crime*, *Mistery*, *Thriller*, *Horror*, *Sci-Fi_and_Fantasy*, *Logical_Thrilling*, *Sensible_Thrilling*, *Thrilling*, *Imaginational-_Entertainment*, e considerando peso do interesse de $u2116$ pelo *fator de interesse gênero* igual a 1, é descrito por $V_{u2116} = \{114, 121, 39, 70, 31, 204, 55, 153, 70, 215, 289, 430\}$.

5.1.7 Motor de Recomendação

Na corrente versão de ORBS, a Matriz de Preferência é utilizada para determinar os vizinhos próximos dos usuários, que são aqueles que possuem preferências similares em termos

da informação lateral sobre os itens já interagidos pelo usuário. Para tal, é adotado o SR proposto por Fernandes, Sacenti e Willrich (2017), não-semântico, que gera recomendações com base nas matrizes de Preferência e de Avaliação Usuário-Item. Note que a Matriz de Preferência considera que a preferência é determinada pela existência de interações usuário-item ao invés da Matriz de Avaliação que considera valores de avaliações explícitas. Como visto anteriormente, o objetivo aqui é mitigar o problema da esparsidade na determinação dos vizinhos próximos e, de modo geral, a Matriz de Preferência é menos esparsa que a Matriz de Avaliação.

A determinação da vizinhança proposta por Fernandes, Sacenti e Willrich (2017) é similar à técnica de recomendação multiatributo exemplificada na Seção 3.2.1. Após a determinação da vizinhança, o SR realiza as demais etapas da FC baseada em usuário para determinação da utilidade dos itens a recomendar normalmente, utilizando a Matriz de Avaliação, conforme o exemplo da Seção 3.1.3.

Conforme indicado no escopo desta tese (Seção 1.3), não há intenção de propor um novo SR, mas sim estudar a representação de conhecimento nos SRs existentes e aprimorar seus modelos de PUs. Para tal, ORBS emprega as etapas anteriores para construir POUs e mapeá-los em PUs na forma de vetores de entidades ponderadas. Estes PUs são constantemente atualizados revisitando as etapas anteriores de modo iterativo. Deste modo, a corrente versão de ORBS adota um SR já existente para realizar a etapa descrita por esta seção. Note que as etapas VI e VII podem ser modificadas para adaptar ORBS a outras técnicas de recomendação existentes.

5.2 AVALIAÇÕES EXPERIMENTAIS

Para avaliar a representação de conhecimento sobre informação lateral na forma de ontologias, e também avaliar a factibilidade de ORBS, foi desenvolvido um protótipo prova-de-conceito implementando algumas das funcionalidades propostas pelo arcabouço ORBS. Este protótipo permitiu a realização de avaliações experimentais sobre o domínio de filmes, que utilizaram o conjunto de dados MovieLens[9] (HARPER; KONSTAN, 2015). A qualidade das recomendações geradas por este protótipo foi comparada com a de um SR clássico e outro SR baseado em multiatributo proposto por Fernandes, Sacenti e Willrich (2017), que não considera *hierarquias de fator de interesse*.

5.2.1 Planejamento Experimental

Esta tese avalia ORBS por meio de uma análise comparativa da qualidade de recomendação obtida com os SRs clássico, multiatributo e com o ORBS. A Figura 13 apresenta o modelo geral adotado para o experimento descrito nesta seção. Este modelo define as entradas, saídas, fatores de experimento controláveis e não-controláveis dos SRs sujeitos ao experimento. Os componentes deste modelo são descritos a seguir:

[9] Acesso: http://movielens.org/, em: 22/11/2021.

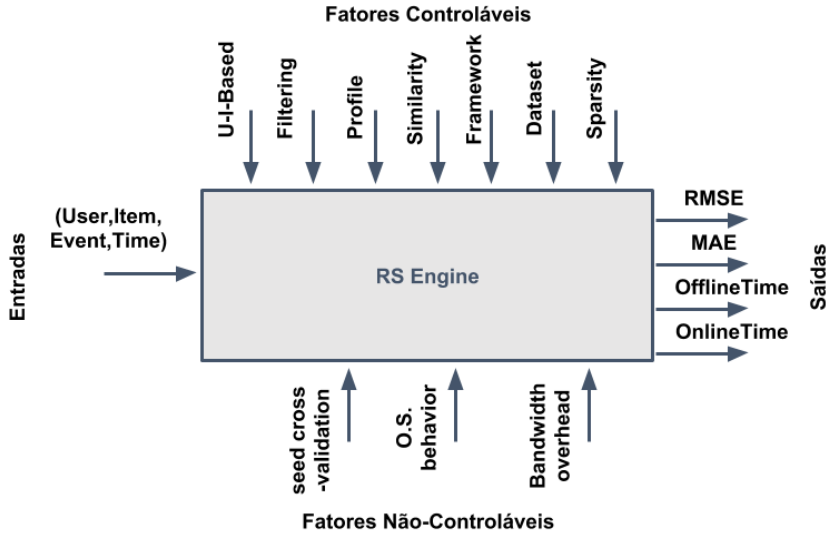

Figura 13 – Modelo geral de experimentos em SR

Fonte: baseada na figura de modelo geral de um processo (MONTGOMERY, 2017).

- **Entradas:** tratam-se das interações usuário-item $int \in Int$ observadas no sistema alvo no formato $int = \langle u, i, v \rangle$, onde $v \in V$ é uma tupla de metadados descrevendo informações sobre a interação como o tipo (*TipoDaInteracao* - p.ex., compra, acesso ou avaliação) e a intensidade (p.ex., valor da avaliação, preço da compra).

- **Fatores de experimento controláveis:** tratam-se das configurações controláveis do experimento. São considerados os fatores referentes a qual SR é avaliado, quais *fatores de interesse* ou *hierarquias de fator de interesse* são considerados pelo PU (*Profile*) e é simulada uma variação de esparsidade nos dados de entrada.

- **Fatores de experimento não controláveis:** tratam-se de fenômenos não controlados pelo experimento que podem influenciar nos seus resultados. São considerados o fator referente a variação do comportamento do sistema no escalonamento de processos (*O.S. behavior*).

- **Saídas:** tratam-se das medidas coletadas após a execução de cada experimento. São consideradas as métricas da raiz do erro quadrático médio (em inglês, *Root-Mean-Square Error* - RMSE) e do erro absoluto médio (em inglês, *Mean Absolute Error* - MAE).

Recapitulando alguns conceitos sobre planejamento e análise de experimentos (MONTGOMERY, 2017), um fator de experimento possui um ou mais níveis (valores). Combinações de níveis de fatores de experimento são chamados de tratamento. Um experimento impõe um tratamento a um grupo de objetos ou indivíduos para observar resultados (saídas). Os resultados do tratamento de uma execução de experimento são chamados de amostra. Nas próximas seções, serão detalhados os conjuntos de dados de entrada, os tratamentos e as saídas dos experimentos iniciais.

5.2.1.1 Dados de entrada

Os dados de entrada do experimento descrito nesta seção são as interações usuário-item do conjunto de dados MovieLens 1M[10] (ML1M), que inclui:

- **Dados de usuários**: ao todo, o ML1M especifica 6040 usuários, sendo que para cada usuário são oferecidos um conjunto de informações, como gênero, idade, ocupação e endereço. Estas informações sobre usuários não foram consideradas neste experimento.
- **Dados sobre filmes**: o ML1M considera 3952 filmes, e são oferecidas informações sobre os filmes como ano de lançamento do filme (que está concatenado ao título do filme) e os gêneros de cada filme, dentre outras informações que não são consideradas neste experimento.
- **Dados de avaliações**: o ML1M contém 1.000.209 avaliações explícitas variando em uma escala inteira $[1;5]$.

Este conjunto de interações usuário-item foi particionado em um conjunto de treinamento e outro de avaliação de acordo com a proporção 7:3, respectivamente.

5.2.1.2 Tratamentos

Nos experimentos iniciais, foram selecionados 4 cenários experimentais definidos pelos fatores controláveis do experimento: SR e PU. Estes cenários consideram 3 tipos de SR: *Classic*, que é a abordagem clássica de SR baseado em Filtragem Colaborativa (FC); *MA*, que determina os vizinhos próximo com base em um PU na forma de uma matriz multiatributo conforme a proposta de Fernandes, Sacenti e Willrich (2017); e *ORBS*, a proposta descrita neste capítulo. Todos os SRs avaliados utilizam a FC baseada em usuário (Seção 3.1.3), com a seguinte configuração: correlação de Pearson (Equação 3.1 da Seção 3.1.3) para determinar a similaridade entre usuários; uso da técnica *k-Nearest Neighbor* (kNN) com $k = 100$ para determinar a vizinhança de usuários; e uso da função de utilidade clássica da FC considerando a Matriz de Avaliação.

Além do SR avaliado, a principal diferença entre os cenários experimentais é o tipo de PU considerado para determinar a vizinhança dos usuários. Os cenários considerados por este experimento são:

- **Classic:** SR baseado em FC clássica que considera apenas avaliações usuário-item representadas por uma Matriz de Avaliação;
- **MA-Genre:** que, além da Matriz de Avaliação, considera apenas anotações semânticas conectando filmes a gêneros (18 entidades de gênero equivalentes aos indivíduos da hierarquia de gênero de MO);

[10] Acesso: http://grouplens.org/datasets/movielens/1m/, em: 22/11/2021.

- **ORBS-Genre:** que, além da Matriz de Avaliação, considera um POU definido por ORBS representando apenas uma *hierarquia de fator de interesse* sobre os gêneros dos filmes;
- **ORBS-Date:** que, além da Matriz de Avaliação, considera um POU definido por ORBS representando apenas uma *hierarquia de fator de interesse* sobre as datas de lançamento dos filmes.
- **ORBS-G+D:** que, além da Matriz de Avaliação, considera um POU definido por ORBS representando ambas as *hierarquias de fator de interesse* sobre gêneros e datas de lançamento.

Além disso, este experimento adiciona um terceiro fator controlável que é a variação da esparsidade dos dados de entrada simulada durante o pré-processamento de dados. A coleção de dados ML1M foi condicionada a 2 níveis de esparsidade (*Sparsity*) distintos: 0% de esparsidade, onde 100% das interações usuário-item são consideradas como avaliações; e 75% de esparsidade, onde 25% das interações de ML1M são consideradas como avaliações e 75% são consideradas interações de acesso. Isto adiciona aos dados de entrada interações implícitas, que representam o caso em que o usuário assiste o filme, mas não se dispõe a avaliá-lo.

5.2.1.3 Medidas de saída

Este experimento coleta duas métricas de erro: a métrica da raiz do erro quadrático médio (em inglês, *Root Mean Square Error* – RMSE), e a métrica do erro absoluto médio (em inglês, *Mean Absolute Error* – MAE). Estas métricas e suas equações foram apresentadas na Seção 6.3.5. A coleta de métricas de acurácia (p.ex., precisão e cobertura) ou diversidade, assim como captura do tempo de treinamento ou de predição, não foram escopo deste experimento.

5.2.2 Implementação dos SRs avaliados

A implementação dos SRs avaliados em cada cenário experimental (*Classic*, *MA*, *ORBS*) adotou a linguagem JavaSE-1.8 e o *framework* Apache Mahout[11], versão *mahout-mr:0.12.2*, para executar as tarefas de determinação de vizinhança (baseado na FC), ranqueamento dos itens recomendados e avaliação do erro (métricas RMSE e MAE). A implementação deste experimento foi publicada em https://github.com/juarezsacenti/ORBS, viabilizando futura reprodução.

Enquanto o SR *Classic* adotado é o SR baseado em FC disponibilizado em Apache Mahout. A implementação de *MA* estende a do SR *Classic*, alterando a técnica de determinação de vizinhança para considerar o modelo de PU multiatributo, conforme empregado em Fernandes, Sacenti e Willrich (2017). As implementações do SR *ORBS* e do protótipo de prova-de-conceito do arcabouço conceitual de mesmo nome são detalhadas a seguir.

[11] Acesso: https://mahout.apache.org/, em: 22/11/2021.

A arquitetura do protótipo de ORBS foi inspirada nos componentes D-A-S-E dos motores de recomendação de do *framework* PredictionIO[12](CHAN et al., 2013):

Fonte e preparador de Dados (D): tratam-se de dois componentes. O primeiro, fonte de dados (em inglês, *Data Source*), recebe as interações usuário-item observadas no sistema alvo e as formatam em dados de treinamento. O segundo, preparador de dados (em inglês, *Preparator*), transforma dados de treinamento em dados preparados no formato requerido pelo algoritmo de recomendação.

Algoritmo(s) (A): trata-se do componente algoritmo (em inglês, *Algorithm*) responsável por aplicar a técnica de recomendação para treinar/atualizar um modelo e predizer recomendações.

Serviço (S): trata-se do componente responsável por receber consultas de recomendação, agregar e formatar os resultados obtidos por um ou mais componentes de algoritmo em uma lista de itens recomendados no formato requisitado pela consulta.

Avaliador (E*): trata-se do componente avaliador (em inglês, *Evaluation Metrics**) responsável por avaliar o motor de recomendação, comparar algoritmos e analisar parâmetros de configuração.

A Figura 14 apresenta a arquitetura proposta para o protótipo. Conforme ilustrado por esta figura, as entradas consideradas são: (i) as iterações usuário-item do sistema alvo em formato RDF; (ii) anotações de itens, interações ou usuário no formato RDF; e (iii) a ontologia de aplicação definida pelo gerente do ORBS que combina a ontologia de perfil de usuário com uma ou mais ontologias de domínio e é criada durante a Etapa I de ORBS (Seção 5.1.1).

A arquitetura de ORBS é composta pelos mesmos componentes D-A-S-E que assimilam as etapas de ORBS. O componente *DataSource* realiza as etapas: *I. Descrição*, que cria as ontologias de ORBS (descrita pela Seção 5.1.1); e *II. Captura*, que recebe notificações de interações efetuadas no sistema alvo e armazena na base de conhecimento *BC* (Seção 5.1.2). O componente *Preparator* realiza as etapas: *III.Enriquecimento*, que atualiza as informações laterais descritas pela ontologia de domínio sobre itens em *BC* (Seção 5.1.3); *IV. Hierarquia*, que constrói *hierarquias de fatores de interesse* (Seção 5.1.4); *V. Preferências*, que calcula a frequência de interações de um usuário para cada entidade das *hierarquias de fator de interesse* (Seção 5.1.5); e *VI. Mapeamento*, que transforma *hierarquias de fator de interesse* em vetores de entidades ponderadas (Seção 5.1.6). O componente *Algorithm* realiza a etapa *IV. Motor de Recomendação*, que treina um modelo de recomendação colaborativa baseada em conceitos ponderados e prediz recomendações (Seção 5.1.7). Por fim, o componente *Evaluation Metrics* realiza a etapa de coleta das medidas de saída analisadas pelo experimento: *MAE* e *RMSE*.

O protótipo de ORBS não considera enriquecimento semântico via fontes externas como dados conectados. Além disso, POUs são convertidos em vetores de entidades ponderadas

[12] Acesso: https://predictionio.apache.org/, em: 22/11/2021.

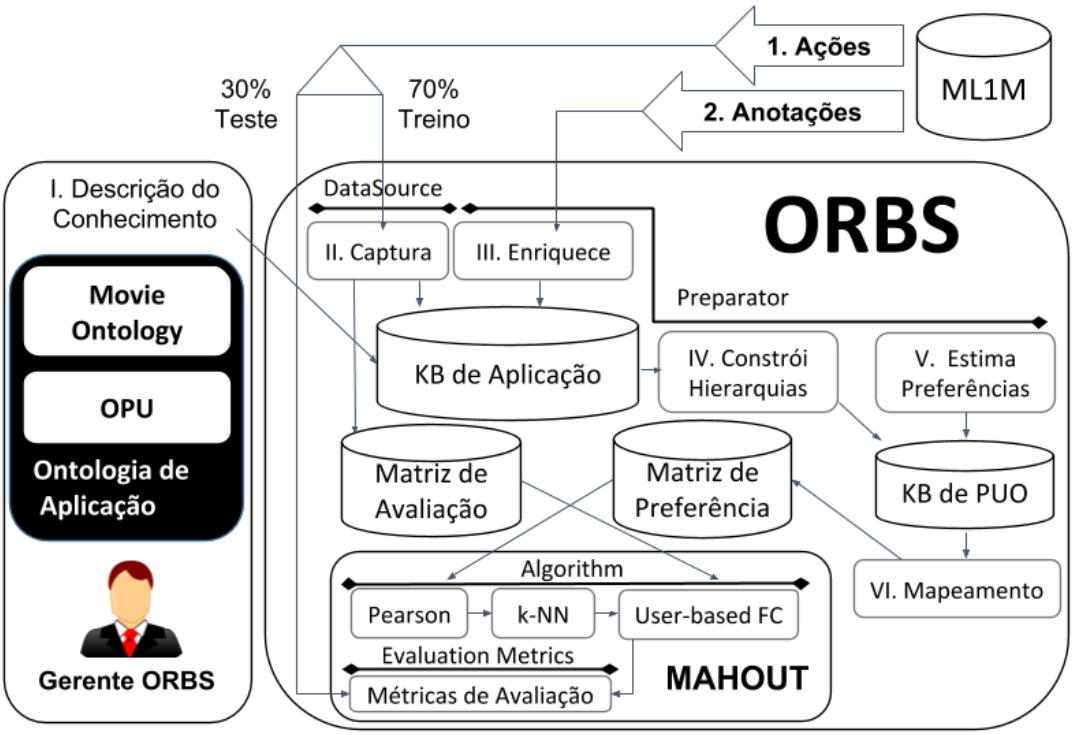

Figura 14 – Arquitetura do protótipo prova-de-conceito de ORBS

Fonte: criado pelo autor.

que compõem uma única Matriz de Preferência que combina todas as *hierarquias de fator de interesse*. As próximas seções descrevem detalhes de implementação de algumas das etapas de ORBS.

5.2.2.1 Descrição do Conhecimento

O protótipo de ORBS adotou uma versão preliminar de ontologia de perfil de usuário, chamada RecOnt2 e ilustrada pela Figura 6 (Seção 5.1.1.2). Além de RecOnt2, este protótipo adotou duas ontologias de domínio sobre filmes: *Movie Ontology* (MO) (Seção 5.1.1.1) e uma ontologia com uma hierarquia de datas de lançamento (Seção 5.1.4). A ontologia de aplicação deste protótipo é similar àquela ilustrada pelas Figuras 7, 8, 9 e 11. As ontologias RecOnt2, de hierarquia de data de lançamento e de aplicação foram desenvolvidas utilizando a ferramenta Protégé 5.2[13]. Todas as ontologias são armazenadas na BC de Aplicação implementado pelo protótipo de ORBS utilizando o *framework* Apache Jena[14].

Como dito anteriormente, a hierarquia de datas de lançamento de filmes foi desenvolvida manualmente. Esta ontologia agrupa o tempo em quinquênios, décadas e a cada três décadas (*Tridecada*). Na Figura 15, o nome de um intervalo indica sua duração e ano de início

[13] Acesso: https://protege.stanford.edu/, em: 22/11/2021.
[14] Acesso: https://jena.apache.org/, em: 22/11/2021.

(p.ex., *Quinquenio2000* é o intervalo de 5 anos $[2000; 2005[$ que começa em 2000 e termina antes de 2005).

Figura 15 – Hierarquia do fator de interesse data de lançamento

Fonte: criado pelo autor.

5.2.2.2 Pré-processamento de dados de entrada

Esta implementação das etapas de captura de dados e enriquecimento da BC de Aplicação reusaram dados disponíveis no conjunto de dados ML1M, sem realizar a coleta em um sistema alvo da recomendação ou em fontes de dados externas. Deste modo, as interações usuário-item e as anotações semânticas sobre itens (ano de lançamento e gêneros) foram extraídas de ML1M e pré-processadas utilizando a ferramenta Pentaho Data Integration[15]. Esta ferramenta converteu o formato destes dados em declarações RDF que foram armazenadas no

[15] Acesso: https://www.hitachivantara.com/go/pentaho.html, em: 22/11/2021.

BC de Aplicação. As etapas *II. Captura* e *III. Enriquecimento* de ORBS (Seções 5.1.2 e 5.1.3) são simuladas para importar os dados pré-processados.

Durante o pré-processamento, os 18 gêneros de ML1M foram manualmente mapeados para indivíduos de gênero de MO. Além dos dados pré-processados, a Matriz de Avaliação é obtida através do pré-processamento de dados de avaliações de ML1M. Essa matriz é importada no protótipo diretamente pelo componente *Algorithm* durante a etapa *VII. Motor de Recomendação*.

5.2.2.3 Hierarquias de fator de interesse

No protótipo de ORBS, as *hierarquias de fator de interesse* são construídas por meio do mapeamento de declarações RDF de propriedades existentes na ontologia de aplicação para uma única propriedade que representa a ordenação estrutural da hierarquia. Assim, ontologia de aplicação deste protótipo não prevê a descrição das regras de categorização mais complexas. As regras de categorização foram informadas para este protótipo na forma de listas de propriedades.

Por exemplo, a Figura 16 apresenta o objeto JSON[16] com os parâmetros da ferramenta *Hierarchy Builder* do *framework Semantic Dimensions* (SeDim), originado no método proposto por Sacenti et al. (2015), para criação da *hierarquia de fator de interesse gênero*.

```
{
  "hierarchyProperty": "mysro:genreHierarchy",
  "mappingProperties": [
    {
      "property": "mo:belongsToGenre",
      "isTransitive": false
    },
    {
      "property": "rdf:type",
      "isTransitive": false
    },
    {
      "property": "rdfs:subClassOf",
      "isTransitive": true
    }
  ]
}
```

Figura 16 – Objeto JSON descrevendo lista de propriedades para o mapeamento

Fonte: criado pelo autor.

O objeto JSON é composto de um atributo *hierarchyProperty* que define a propriedade que representa a hierarquia, e de uma lista de objetos JSON *mappingProperties* que descreve quais propriedades serão mapeadas para a propriedade indicada por *hierarchyProperty*.

[16] Acesso: https://www.json.org/, em: 22/11/2021.

Por exemplo, para este objeto JSON, *Hierarchy Builder* inicia o mapeamento de propriedades nos indivíduos de item da BC de Aplicação. Caso exista uma declaração RDF relacionando um indivíduo de item a um indivíduo de gênero pela propriedade *mo:belongsToGenre*, *Hierarchy Builder* mapeia esta declaração substituindo a propriedade *mo:belongsToGenre* pela *mysro:genreHierarchy* e processa o indivíduo de gênero (caso ainda não a tenha processado). Por sua vez, caso exista uma declaração relacionando o indivíduo de gênero a um conceito de gênero pela propriedade *rdf:type*, *Hierarchy Builder* mapeia a declaração processa o conceito de gênero. O atributo *isTransitive* indica quando busca por uma propriedade deve ser repetida em entidades sucessoras.

Conforme dito anteriormente, o protótipo adotou duas *hierarquias de fator de interesse*: de *gênero*, ilustrada pela Figura 10 (Seção 5.1.1.1), e de *data de lançamento*, ilustrada pela Figura 15 (Seção 5.2.2.1).

5.2.2.4 Preferências

Nestes experimentos iniciais, a contagem de interações de usuário sobre itens para cada categoria e subcategoria das *hierarquias de fator de interesse* é realizada de acordo com a Definição 11 de frequência de uso de categorias (Seção 5.1.5).

5.2.2.5 Mapeamento

No protótipo de ORBS, a etapa *VI. Mapeamento* utiliza a Equação 5.1 (Seção 5.1.6) para transformar hierarquias de *fatores de interesse* em vetores de entidades ponderadas. Em cada cenário experimental, para todos os usuários, os pesos de interesse pelos *fatores de interesse gênero* e *data de lançamento* foram configurados em 1. Conforme dito anteriormente, as *hierarquias de fatores de interesse* são mapeadas para uma única Matriz de Preferência.

5.2.3 Análise de Resultados

As Figuras 17 e 18 apresentam os valores das métricas de avaliação de erro RMSE e MAE para os casos de estudo *Classic*, *MA-Genre*, *ORBS-Genre*, *ORBS-Date*, *ORBS-G+D* para os datasets com 100% e 25% de dados (níveis de esparsidade 0% e 75% respectivamente).

Os resultados obtidos permitem discutir a Pergunta de Pesquisa 1: *A representação de informação lateral por meio de características relacionadas a múltiplos aspectos resulta na melhoria da eficácia dos resultados de SRs?* Esta pergunta já foi parcialmente respondida por Soares e Viana (2017), que conclui que a combinação de diferentes informações (*fatores de interesse*) na formação do PU obtém performance mais estável que em casos de teste com PU que consideram *fatores de interesse* individualmente. O experimento aqui proposto reafirma esta conclusão, pois os cenários experimentais *ORBS-Genre* e *ORBS-Date* apresentam medidas de erro superiores às de *ORBS-G+D*.

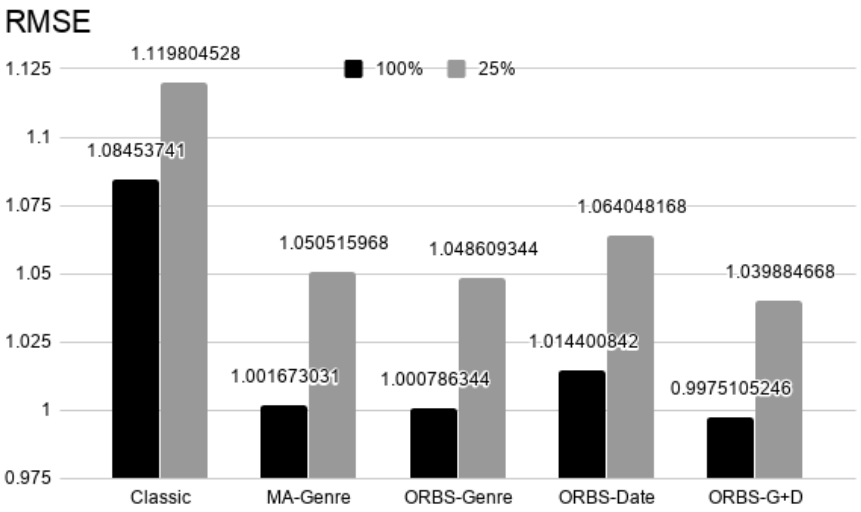

Figura 17 – Resultados da métrica RMSE

Fonte: criado pelo autor.

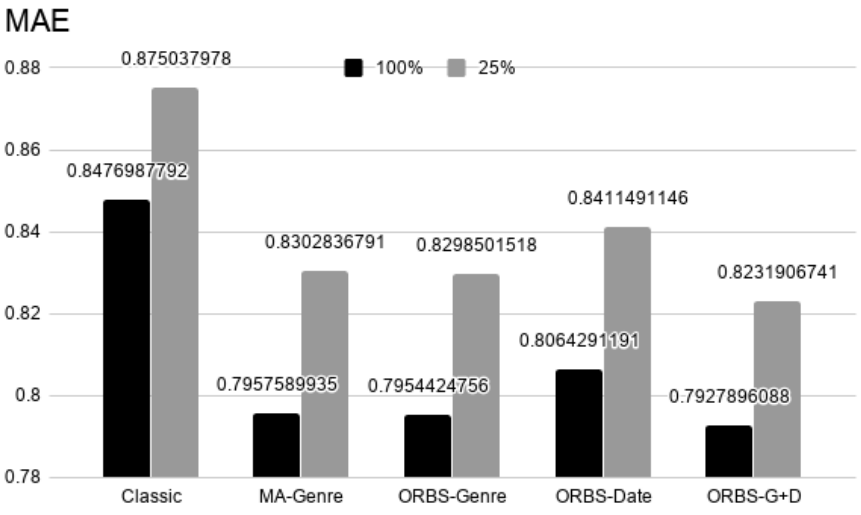

Figura 18 – Resultados da métrica MAE

Fonte: criado pelo autor.

O cenário experimental *MA-Genre* apresentou resultados levemente superiores a *ORBS-Genre*, atendendo a Pergunta de Pesquisa 2: *A representação de informação lateral por meio de características organizadas em taxonomias ou ontologias hierárquicas resulta na melhoria da eficácia dos resultados de SRs?* Deste modo, há indícios de que a composição do POU com *hierarquias de fatores de interesse* pode aumentar da qualidade de recomendações. Além disso, em trabalho anterior (FERNANDES; SACENTI; WILLRICH, 2017) foi observada uma variação muito pequena no RMSE para diferentes níveis de granularidade da hierarquia de gênero de filmes (grupos de 18, 13 e 7 gêneros), demonstrando que o tamanho e complexidade de hierarquias também influencia na qualidade das recomendações.

Assim como em Soares e Viana (2017) e Fernandes, Sacenti e Willrich (2017), PU baseados em metadados (*MA-Genre*, *ORBS-Genre*, *ORBS-Date* e *ORBS-G+D*) obtiveram medidas de erro inferiores a PU baseados em avaliações (*Classic*). Além disso, a diferença das medidas de erro entre tratamentos que mantiveram 25% das avaliações é superior a diferença entre tratamentos que mantiveram 100% das avaliações. Por exemplo, a diferença entre *ORBS-G+D* e *ORBS-Genre* é maior entre os tratamentos com 25%. Isto é um indício de que o benefício de representar a informação lateral como múltiplas *hierarquias de fator de interesse*, em termos de redução de erro de SRs, é maior quando as interações usuário-item são mais esparsas.

5.3 CONSIDERAÇÕES FINAIS

O protótipo de ORBS e os resultados dos experimentos com ontologias se demonstraram promissores durante a investigação sobre a representação de conhecimento, mostrando que o SRO proposto (neste experimento representado por *ORBS-G+D*) tem potencial para melhorar a qualidade das recomendações.

Entretanto, várias oportunidades de pesquisa a respeito de ORBS e do experimento realizado não foram exploradas neste estudo. O aprofundamento das etapas de ORBS, como o aprimoramento da ontologia de perfil de usuário, estudo do enriquecimento semântico de interações usuário-item, a construção e especificação de *hierarquias de fator de interesse*, a avaliação de técnicas alternativas de contagem de interações de um dado usuário em uma hierarquia e alternativas de uso do POU em modelos de recomendação, são alguns exemplos. Além disso, é indicada a necessidade de novos experimentos considerando conjuntos de dados de outros domínios de item, como músicas, livros ou vagas de emprego e avaliando outros SRs como o baseado na Filtragem Baseada em Conteúdo (FBC), SROs e SRGCs.

Uma das principais limitações desta abordagem é que a técnica de recomendação adotada nos experimentos (FC baseada em usuário e em matriz de preferência) é uma técnica simples que não acompanha o estado-da-arte em SRs baseados em conhecimento (SROs e SRGCs), que exploram algoritmos baseados em fatoração de matriz e aprendizado de máquina.

Estes resultados motivaram novos rumos para esta tese que culminaram na segunda abordagem da proposta acerca da sumarização de grafos de conhecimento e seus impactos em SRGCs baseados em *embeddings*. A seguir, o Capítulo 6 descreve esta nova abordagem, o

método de sumarização adotado, o método de avaliação adotado, o planejamento experimental e os resultados obtidos.

6 SUMARIZAÇÃO DE GC PARA SISTEMA DE RECOMENDAÇÃO

De modo geral, SRs baseados em GC (SRGCs) exigem alto custo computacional para gerar recomendações, por causa do grande volume de dados sobre interações usuário-item e informação lateral representada em Grafos de Conhecimento (GCs). A Sumarização de Grafo (SG) permite reduzir o volume do GC, e além disso, tem potencial para eliminar ruídos e dados irrelevantes para a recomendação. Métodos de SG têm sido aplicados na visualização de grafos e na otimização de consultas. Porém, estes métodos podem não ser adequados para SRGCs, pois a sumarização é dependente do domínio de aplicação. Esta tese investiga o uso de SG visando gerar versões sumarizadas do GC, potencialmente aumentando a eficácia, sem grandes impactos na eficiência.

Este capítulo descreve a proposta de um novo método de sumarização de GC, chamado de Sumarização KGE-K-Means, e investiga seu uso como uma etapa de pré-processamento para SRGCs. Este método é aplicado a SRGCs, especificamente no pré-processamento do GC usado para representar informação lateral sobre os itens a serem recomendados. A premissa deste método é que o sumário de GC (sGC) pode condensar a informação lateral e acelerar o treinamento do modelo de recomendação, mantendo a representação de preferências de usuários necessária para gerar recomendações de qualidade. A figura 19 ilustra uma visão geral de um SRGC que adota o método proposto. Esta figura apresenta: o Sistema Alvo da Recomendação onde são capturadas as informações sobre usuários, itens e interações; as fontes externas onde são capturas informações laterais sobre itens e/ou usuários; e o SRGC composto pelas tarefas de treinamento de modelo e recomendação. As etapas de preparação e atualização dos dados de entrada capturam as informações e constrói o GC (original, antes da sumarização) e a base de interações usuário-item. O método proposto, Sumarização KGE-K-Means, está situado como uma das etapas de pré-processamento de dados deste SRGC.

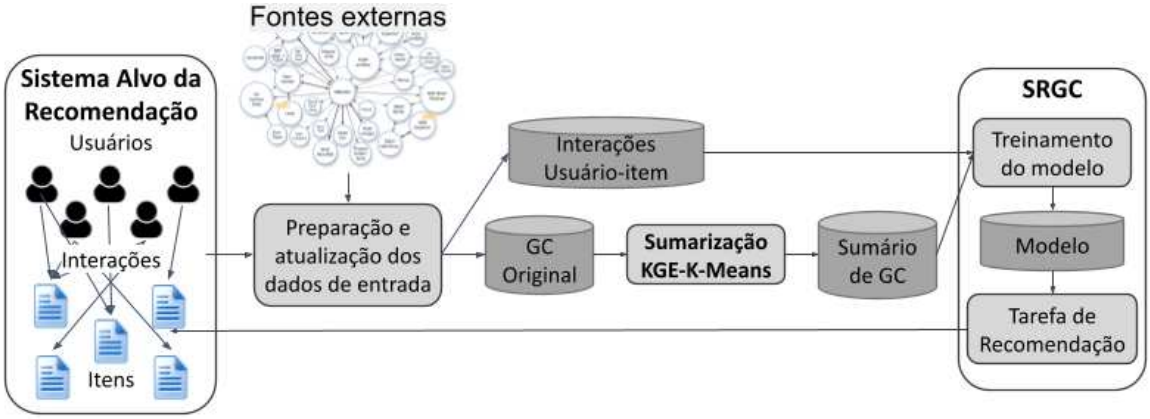

Figura 19 – Sumarização KGE-K-Means como etapa de pré-processamento de um SR baseado em GC

Fonte: criado pelo autor.

A Seção 6.1 descreve KGE-K-Means, que é uma técnica de sumarização baseada no agrupamento de nodos semanticamente similares em supernodos. Para determinar a similari-

dade entre nodos é adotada a representação vetorial em um espaço de *embeddings* de grafo, em duas estratégias alternativas: a única-visão e a multi-visão. A Seção 6.2 descreve como SRGCs podem usar KGE-K-Means para gerar sumários de GC (sGCs) alternativos, com base em ambas estratégias e com taxas de sumarização crescentes, para acelerar o treinamento de modelos de recomendação baseados em GC. Além disso, descrevemos um método de avaliação, chamado KG-Summ-Rec, para avaliar os impactos da Sumarização KGE-K-Means na eficiência e eficácia de diferentes SRGCs. Finalmente, a Seção 6.3 relata os experimentos com a sumarização de GC usando conjuntos de dados sobre filmes e a Seção 6.4 apresenta os resultados experimentais obtidos.

6.1 SUMARIZAÇÃO KGE-K-MEANS

A Figura 20 ilustra o método de Sumarização KGE-K-Means proposto nesta tese. Considere como entrada do método KGE-K-Means o *GC Original* que representa a informação lateral (entidades e propriedades) relacionada aos itens a serem recomendados. Esta informação pode ser oriunda de fontes de dados estruturados e de domínio específico (por exemplo, IMDb) e/ou dados conectados e de domínio genérico (por exemplo, DBpedia). KGE-K-Means sumariza apenas entidades que representam a informação lateral. Usuários e itens não são sujeitos a sumarização para manter a independência entre o pré-processamento do GC e o processo de recomendação do SRGC.

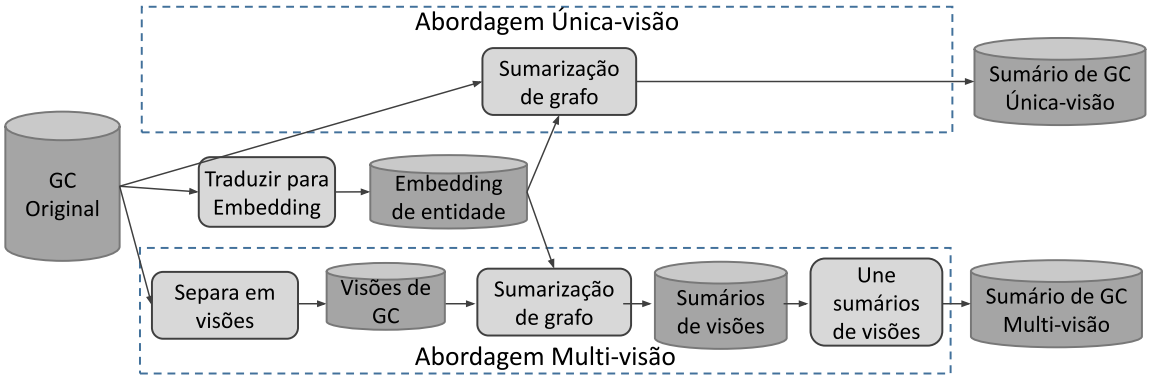

Figura 20 – Sumarização KGE-K-Means com as estratégias única-visão e multi-visão

Fonte: criado pelo autor. Publicado pela primeira vez em Journal of Intelligent Information Systems, 2021, pela Springer Nature.

Primeiro, *GC Original* é traduzido para *Embedding de entidade*, aplicando modelos como ComplEx (TROUILLON et al., 2016). A partir dos vetores das entidades sujeitas a sumarização, é calculada a similaridade ou proximidade semântica entre estas entidades. Depois, a *Sumarização do grafo* baseada no algoritmo de clusterização K-Means agrupa as entidades mais similares em *k clusters* (grupos). Então, KGE-K-Means aglutina as entidades de um mesmo *cluster* em um único supernodo representado por uma nova URI no sumário de GC (*sGC*). As declarações RDF (fatos) de/para entidades aglutinadas são substituídas por declarações de/para

o respectivo supernodo. Declarações duplicadas geradas por estas alterações são removidas, contribuindo para reduzir o tamanho do GC. Por exemplo, durante a sumarização de um GC do domínio de filme, as entidades que representam os atores *Anatoli Davydov*, *Julianne Moore* e *Sylvester Stallone* são aglutinadas em um mesmo supernodo devido à similaridade semântica entre os nodos do grafo representando estes atores. Então, caso estes três atores atuem num mesmo filme (por exemplo, *Assassins* lançado em 1995), o *sGC* terá uma única aresta entre o nodo deste filme e o supernodo que representa este conjunto de atores, enquanto o GC não sumarizado (*GC Original*) declara três triplas RDF, cada uma relacionando este filme a um dos atores.

A descrição da Sumarização KGE-K-Means adota o termo *aglutinar* para descrever a transformação de nodos representando entidades do GC em um supernodo, que é representado por uma nova URI no sumário de GC. Diferente do termo *agrupar* que significa *reunir em grupo, formar grupo com* e é usado para indicar o efeito causado por um algoritmo de *clustering*, o termo *aglutinar* significa *aderir, ligar fortemente* e é usado para indicar o efeito causado pela sumarização (i.e., após a substituição de declarações). Portanto, *aglutinar* foi adotada para diferenciar estes efeitos, indicando que houve uma combinação (fusão) de nodos em um supernodo, generalizando as entidades em um conceito de nível superior.

A Sumarização KGE-K-Means permite taxas de sumarização distintas, variando o parâmetro k do algoritmo de agrupamento K-Means. O número de supernodos é equivalente ao número de *clusters* encontrados. A taxa de sumarização é definida como $\alpha = 1 - n/N$, onde N é o número de entidades no *GC Original* e n é o número de entidades e supernodos no *GC sumarizado*. Portanto, quanto maior essa proporção, maior será o número de entidades agrupadas em supernodos.

Conforme ilustrado na Figura 20, KGE-K-Means suporta duas estratégias, única-visão e multi-visão, descritas a seguir.

6.1.1 Estratégia de única-visão

A *estratégia de única-visão* aplica a sumarização a todo o *GC Original* e gera um *Sumário de GC Única-visão*. Portanto, esta estratégia permite aglutinar em um mesmo supernodo entidades de tipos variados (por exemplo, atores, diretores, gêneros). Por exemplo, um supernodo pode aglutinar o ator *Sylvester Stallone* e o gênero *Ação*.

O algoritmo da Sumarização KGE-K-Means única-visão (Algoritmo 1) tem como entradas o *GC Original* (*gc*), a lista de URIs dos *itens* e a *taxa* de sumarização, e como saída o *Sumário de GC Única-visão*. Primeiro, o *gc* é traduzido para *embedding* (linha 2). Depois, as entidades são agrupadas como um todo (linhas 3 e 4), aglutinadas em supernodos e as suas declarações são substituídas e adicionadas ao sumário do GC (linha 7 a 12). Então, as declarações duplicadas do *Sumário de GC Única-visão* são removidas (linha 15).

O algoritmo de agrupamento de KGE-K-Means (Algoritmo 2) mostra em mais detalhes como o algoritmo K-Means é utilizado no método proposto. Esta função tem como entradas

Algoritmo 1 Sumarização KGE-K-Means única-visão

Entradas gc: Grafo de Conhecimento (GC) original
$itens$: lista de URIs de itens do GC
$taxa$: taxa de sumarização
Saída $sumárioDeGC\_ÚnicaVisão$: conjunto de triplas sumarizadas do GC

1: **função** KGEKMEANS_ÚNICAVISÃO($gc, itens, taxa$)
2: $modelo \leftarrow TraduzParaEmbedding(gc)$ ▷ KGE, p.ex.: ComplEx
3: $entidades \leftarrow RecuperaEntidades(gc, itens)$
4: $clusters \leftarrow Agrupamento(entidades, modelo, taxa)$ ▷ Algoritmo 2
5: $sumárioDeGC \leftarrow gc$
6: **para cada** $c \in clusters$ **faça**
7: **se** c contém mais de uma entidade **então**
8: $uriDoCluster \leftarrow DefineURI(gc, c)$
9: $entidadesDoCluster \leftarrow RecuperaEntidades(c)$
10: **para cada** $e \in entidadesDoCluster$ **faça**
11: $sumárioDeGC \leftarrow SubstituiEntidadesNoGC(sumárioDeGC, e, uriDoCluster)$
12: **fim para**
13: **fim se**
14: **fim para**
15: $sumárioDeGC\_ÚnicaVisão \leftarrow RemoveDuplicatas(sumárioDeGC)$
16: **devolve** $sumárioDeGC\_ÚnicaVisão$
17: **fim função**

Algoritmo 2 Agrupamento de KGE-K-Means

Entradas $entidades$: lista de URIs das entidades do GC sujeitas ao agrupamento
$modelo$: modelo de *embedding* do GC
$taxa$: taxa de sumarização
Saída $clusters$: mapeamento cluster-entidades

1: **função** AGRUPAMENTO($entidades, modelo, taxa$)
2: $vetores \leftarrow RecuperaVetores(entidades, modelo)$
3: $k \leftarrow \lceil |entidades| * (1 - taxa) \rceil$
4: $clusters \leftarrow KMeans(vetores, k)$ ▷ K-Means
5: **devolve** $clusters$
6: **fim função**

as *entidades* a serem agrupadas, o *modelo de embedding* contendo a representação vetorial destas entidades (linha 2) e a *taxa* de sumarização, e como saída o mapeamento *cluster*-entidades. O número de *clusters k* é calculado (linha 3) como o teto da multiplicação da quantidade de *entidades* pelo complemento da *taxa* de sumarização (taxa de retenção). O mapeamento é obtido utilizando o algoritmo K-Means (linha 4), com base na proximidade dos vetores (similaridade semântica).

6.1.2 Estratégia de multi-visão

A *estratégia multi-visão* é proposta para avaliar uma sumarização mais restritiva. Nesta estratégia, ilustrada na parte inferior da Figura 20, a tarefa *Separação em visões* divide o *GC Original* em um conjunto de *visões de GC*. Cada visão representa um aspecto (faceta, subcon-

junto) da informação lateral sobre os itens a serem recomendados. Exemplos de aspectos para descrever filmes são *Gênero*, *Ator* e *Diretor*. Na implementação atual, KGE-K-Means cria uma visão de GC v_i escolhendo primeiro uma propriedade r_i e, em seguida, coletando declarações $\langle e_h, r_i, e_t \rangle \subseteq N \times R \times N$, onde e_h ou e_t é uma entidade que se refere a um item a ser recomendado (p.ex., filme), e a propriedade r_i restringe a outra entidade da declaração a uma classe de interesse. Por exemplo, as relações *hasGenre*, *hasActor* e *hasDirector*, restringem a classe da entidade relacionada ao filme as classes *Genre*, *Actor* e *Director*, respectivamente, permitindo a geração de três visões de GC distintas.

Depois de criar as *visões de GC*, a tarefa de *Sumarização de grafo* sumariza cada visão independentemente, produzindo um sumário para cada visão. Observe que, diferente da *estratégia única-visão*, os supernodos gerados aqui só podem aglutinar entidades que pertençam à uma mesma visão de GC (na implementação atual, que pertençam a uma mesma classe). Finalmente, a tarefa **União de sumários de visões** reconecta os sumários de visões em um único sumário de GC, chamado *Multi-view sKG*. Isso é feito considerando as declarações RDF (fatos) do GC. Assim como na *estratégia única-visão*, as declarações sobre entidades aglutinadas são substituídas por declarações sobre os respectivos supernodos e as declarações duplicadas foram removidas.

O algoritmo da Sumarização KGE-K-Means multi-visão (Algoritmo 3) possui entradas e saídas análogas às da Sumarização KGE-K-Means única-visão (Algoritmo 1). As *propriedades* do *gc* são utilizadas para separá-lo em visões (linha 5), as entidades de cada visão são agrupadas e aglutinadas independentemente (linha 6 e 7), suas declarações são substituídas (linhas 9 a 16) e, por fim, as visões são reconectadas (linha 17). Então, as declarações duplicadas do *Sumário de GC Multi-visão* são removidas (linha 20). Teoricamente, a função de remoção de duplicatas da estratégia multi-visão é diferente da usada na estratégia única-visão, pois pode considerar casos em que o mesmo conjunto de entidades pode ser representado por supernodos distintos e de diferentes visões. Uma função de remoção de duplicatas que trate estes casos permite reduzir mais o volume do GC, porém pode causar impacto negativo na qualidade. Esta estratégia não foi adotada em nossos experimentos, sendo indicada para trabalhos futuros.

Para demonstrar que a replicação de uma entidade em múltiplas visões não causa aumento no número de declarações e de entidades no *Sumário de GC Multi-visão*, considere o seguinte cenário: Suponha que uma entidade *e1* descreve itens através de 20 propriedades em um GC a ser sumarizado. Suponha que a estratégia multi-visão é escolhida, separando *gc* em 20 visões, uma para cada propriedade do GC. Segundo o Algoritmo 3, a entidade *e1* é aglutinada nos supernodos $sn1, sn2, sn3, ..., sn_i, ..., sn20$, gerados pela sumarização de cada visão, respectivamente. Além disso, suponha que cada um destes supernodos contém $e1$ e, pelo menos, uma única outra entidade $e2, ..., e_k, ..., e21$ qualquer. Finalmente, suponha que todas as entidades e_k descrevem apenas um item através de uma única declaração com a respectiva propriedade.

No dado cenário, é possível calcular quantas declarações são sumarizadas. Note que $e1$ e e_k são entidades relacionadas a itens, pois são sumarizados, portanto não são itens. No *GC original*, temos 20 declarações conectando a entidade $e1$ a algum item por meio de cada uma das

Algoritmo 3 Sumarização KGE-K-Means multi-visão

Entradas gc: Grafo de Conhecimento (GC) original
$itens$: URI de itens do GC
$taxa$: taxa de sumarização
Saída $sumárioDeGC\_MultiVisão$: conjunto de triplas sumarizadas do GC

função KGEKMEANS_MULTIVISÃO($gc, itens, taxa$)
  $modelo \leftarrow TraduzParaEmbedding(gc)$ ▷ KGE, p.ex.: ComplEx
  $sumário \leftarrow \varnothing$
  $propriedades \leftarrow RecuperaPropriedades(gc, itens)$
  para cada $p \in propriedades$ **faça**
    $visãoDeGC \leftarrow RecuperaTriplasComP(gc, itens, p)$ ▷ Separação em visões
    $entidades \leftarrow RecuperaEntidades(visãoDeGC, itens)$
    $clustersComP \leftarrow Agrupamento(entidades, modelo, taxa)$ ▷ Algoritmo 2
    $sumárioDeV \leftarrow visãoDeGC$
    para cada $c \in clustersComP$ **faça**
      se c contém mais de uma entidade **então**
        $uriDoCluster \leftarrow DefineURI(gc, c)$
        $entidadesDeC \leftarrow RecuperaEntidades(c)$
        para cada $e \in entidadesDeC$ **faça**
          $sumárioDeV \leftarrow SubstituiEntidadeNaVisão(sumárioDeV, e, uriDoCluster)$
        fim para
      fim se
    fim para
    $sumário$ adiciona $sumárioDeV$ ▷ União de sumários de visões
  fim para
  $sumárioDeGC\_MultiVisão \leftarrow RemoveDuplicatas(sumário)$
  devolve $sumárioDeGC\_MultiVisão$
fim função

20 declarações. Adicionamos mais 20 declarações sobre cada entidade em $e2, ..., e_k, ..., e21$. Um total de 40 declarações e 21 entidades. Supondo o caso em que cada uma destas 40 declarações conecte as entidades $e1, ..., e_k, ..., e21$ a 40 itens diferentes, a tarefa de *Separação em visões* gera 20 visões com 2 declarações cada (sobre $e1$ e e_k e 2 itens distintos). Depois, a tarefa de *Sumarização de grafo* gera 20 visões com 2 declarações cada (sobre sn_i conectado a cada um dos 2 itens em determinada visão). Finalmente, a tarefa de *União de sumários de visões* gera um *Sumário de GC Multi-visão* com 40 declarações (conectando 20 supernodos sn_i a 40 itens distintos). Neste caso, a sumarização manteve o número de declarações (40) e reduziu o número de entidades de 21 para 20. Deste modo, a replicação de uma entidade em múltiplas visões implica, no pior caso, num número de declarações constante e na diminuição do número de entidades do GC.

6.2 AVALIAÇÃO KG-SUMM-REC

Devido à falta de uma metodologia consolidada para avaliar a eficiência e a eficácia dos modelos de aprendizado de máquina para SRs (PAUN, 2020), esta tese propõe um método simplificado para avaliar os impactos da Sumarização KGE-K-Means na eficiência e a eficácia

de diferentes SRGCs, chamado de KG-Summ-Rec. A Figura 21 ilustra o método de avaliação KG-Summ-Rec que também envolve duas tarefas principais para um SRGC baseado em *embeddings*: *treinamento do modelo* e *tarefa de recomendação*. O *treinamento do modelo* aprende uma representação vetorial a partir das interações usuário-item e das declarações do GC capaz de predizer interações futuras. Note que apenas um GC é necessário para treinar o modelo de recomendação, conforme ilustrado na figura (representado pelo símbolo de disjunção exclusiva $\oplus$). A *tarefa de recomendação* classifica e ordena os itens com base em sua relevância (alinhamento com as preferências do usuário alvo da recomendação), calculando-a com o modelo de *embeddings* aprendido sobre usuários, itens e entidades, para então retornar os *N* itens mais relevantes.

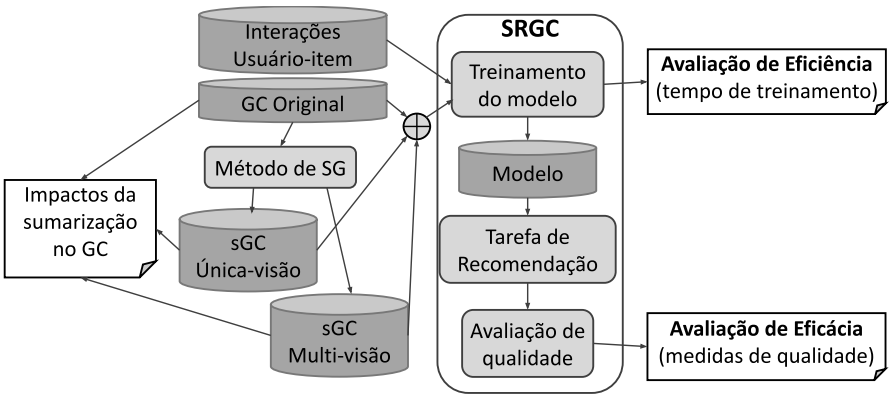

Figura 21 – Avaliação dos impactos da Sumarização KGE-K-Means no SR baseado em GC

Fonte: criado pelo autor. Publicado pela primeira vez em Journal of Intelligent Information Systems, 2021, pela Springer Nature.

Um dos objetivos desta tese é avaliar os impactos da Sumarização KGE-K-Means na redução do volume do GC e principalmente na eficiência e na eficácia dos SRGCs. O método KG-Summ-Rec avalia o impacto da sumarização no GC considerando medidas estatísticas sobre o número de declarações e entidades do GC. Para *Avaliação de Eficiência*, o método KG-Summ-Rec adota o tempo de treinamento do modelo de recomendação como a métrica de eficiência, pois esta tarefa é a mais cara do SRGC baseado em *embedding* e a mais impactada pela sumarização. Entretanto, note que a sumarização afeta a eficiência do processo de recomendação como um todo e não apenas o treinamento do modelo. Para *Avaliação de eficácia*, o método KG-Summ-Rec considera as métricas de qualidade amplamente adotadas para avaliar a eficácia de SR, incluindo precisão (p@N), cobertura (r@N), ganho cumulativo com desconto normalizado (nDCG@N) e média da precisão média (mAP@N). Portanto, ao avaliar a qualidade da recomendação (resultado final), KG-Summ-Rec avalia o impacto da sumarização na eficácia de todo o processo de recomendação.

6.3 EXPERIMENTOS COM SUMARIZAÇÃO DE GRAFOS

Esta seção descreve os experimentos de um estudo comparativo conduzido nesta tese utilizando o método de avaliação KG-Summ-Rec (Seção 6.2) para avaliar o impacto da Sumarização KGE-K-Means (Seção 6.1) em SRGCs baseados em *embedding*. Primeiro, esta seção descreve a visão geral dos experimentos, que é seguido pela descrição dos conjuntos de dados, dos métodos de pré-processamento, dos recomendadores, das métricas de avaliação, dos cenários experimentais e dos detalhes de implementação.

6.3.1 Visão Geral

A Figura 22 descreve as entradas, a sequência de tarefas realizadas e as saídas destes experimentos. Dois GCs descrevendo dados do domínio de filme são usados como entradas: *GC de Sun* (SUN et al., 2018) e *GC de Cao* (CAO et al., 2019). As entradas são sujeitas a três pré-processamentos distintos: a Sumarização *KGE-K-Means*, que gera os sumários de GC (*sGCs*); a *Filtragem de entidades*, que gera os GCs filtrados (*fGCs*); e a combinação da sumarização seguida pela filtragem, que gera os sumários de GC filtrados (*sfGCs*). Os GCs gerados no pré-processamento são usados, um de cada vez, no treinamento de SRGCs, conforme ilustrado na figura (representado pelo símbolo de disjunção exclusiva $\oplus$).

A tarefa de *Avaliação de SRGCs* pela Figura 22 é detalhada na Figura 21. Nesta tarefa, os *impactos da sumarização no GC*, assim como o tempo de treinamento e medidas de qualidade dos modelos de recomendação, são coletados considerando os GCs originais de Sun e Cao e seus derivados (*sKGs*, *fKGs* e *sfKGs*), com taxas de sumarização crescentes. As seções a seguir fornecem mais detalhes sobre a implementação, conjuntos de dados, métodos de SRGC, parâmetros e métricas de avaliação usados nos experimentos.

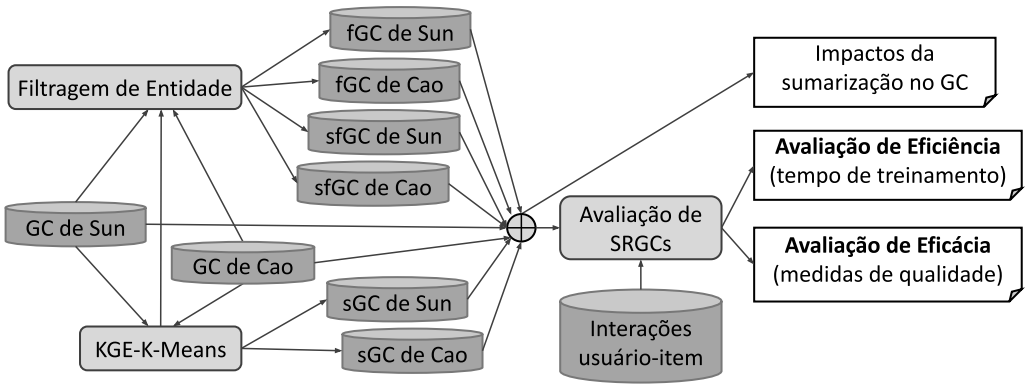

Figura 22 – Sequência de tarefas realizadas nos conjuntos de dados pré-processados

Fonte: criado pelo autor. Publicado pela primeira vez em Journal of Intelligent Information Systems, 2021, pela Springer Nature.

6.3.2 Conjunto de dados

As *interações usuário-item* e os GCs de *Sun* e de *Cao* usados como entradas dos experimentos foram obtidos de dois conjuntos de dados do domínio do filme:

- **Conjunto de dados de Sun (SUN et al., 2018)** [1]: contém um subconjunto das avaliações usuário-item do conjunto de dados MovieLens 1M (HARPER; KONSTAN, 2015), com 99.975 avaliações (1 a 5 estrelas) feitas por 943 usuários para 1.675 filmes. O número médio de avaliações por usuário é 106 e a esparsidade de avaliações é 93,6%. As informações laterais conectadas a este conjunto de filmes, representadas por *GC de Sun* na Figura 22, são oriundas do IMDb. Este GC contém 12.311 declarações RDF na forma sujeito-propriedade-objeto com 3 propriedades distintas: *temAtor*, *temDiretor* e *temGênero*. Sujeitos e objetos podem ser filmes ou 5.124 outras entidades (incluindo 3.947 atores, 1.153 diretores e 24 gêneros). A esparsidade do *GC de Sun* é 99,9%.

- **Conjunto de dados de Cao (CAO et al., 2019)** [2]: contém um subconjunto das avaliações usuário-item de MovieLens 1M, com 998.539 avaliações feitas por 6.040 usuários para 3.260 filmes. O número médio de avaliações por usuário é 165 e a esparsidade de avaliações é 94,9%. As informações laterais conectadas a este conjunto de filmes, representadas por *GC de Cao* na Figura 22, são oriundas do conjunto de dados LODRecSys [3] que enriquece os filmes de MovieLens 1M com declarações e entidades da DBpedia. Este GC contém 428.777 declarações que descrevem 20 propriedades (por exemplo, *cinematografia, produtores, história, série, baseado em, estrelado, diretor, prêmio*) e 11.443 entidades, com esparsidade de GC em 99,9%. Portanto, *GC de Cao* contém um número maior de entidades do que o *GC de Sun*.

A Tabela 7 fornece estatísticas do conteúdo desses dois conjuntos de dados. Observe que o *GC de Cao* possui volume maior que o *GC de Sun*. Ambos reusam avaliações de um conjunto de dados bem conhecido do domínio de recomendação de filmes, o MovieLens 1M. A esparsidade de avaliações usuário-item é calculada como $n/u*i$, onde n é o número de avaliações, u é o número de usuários e i é o número de itens. A esparsidade de GC, por sua vez, é calculada como $n/p*e^2$, onde n é o número de declarações, p é o número de propriedades e e é o número de entidades representando informação lateral diretamente relacionada a entidades que representam filmes.

[1] Disponível em: https://github.com/sunzhuntu/Recurrent-Knowledge-Graph-Embedding, acesso em: 22/11/2021.
[2] Disponível em: https://github.com/TaoMiner/joint-kg-recommender
[3] Aval.: https://github.com/sisinflab/LODrecsys-datasets/tree/master/Movielens1M

6.3.3 Pré-processamentos

Como já apresentado, esta tese investiga métodos de pré-processamento visando a redução do custo computacional em SRGCs. Conforme ilustrado na Figura 22, os seguintes métodos de pré-processamento do GC são avaliados:

- ***Filtragem de entidades***: trata-se de uma técnica simples para reduzir o tamanho do GC via a remoção de entidades infrequentes. Este método de poda rudimentar é adotado por alguns SRGCs, incluindo (SUN et al., 2018) e (CAO et al., 2019). Nestes experimentos, como em (CAO et al., 2019), as entidades são consideradas infrequentes quando aparecem em no máximo 10 declarações.

- ***KGE-K-Means***: é o método de Sumarização KGE-K-Means proposto nesta tese (Seção 6.1). Nestes experimentos, o modelo de *embedding* ComplEx (TROUILLON et al., 2016) é treinado usando os seguintes parâmetros: dimensionalidade do *embedding* k em 150, número de amostras negativas *eta* em 20, número de épocas em 150, número de lotes em 100, otimizador *Adam* com taxa de aprendizagem em $0,1$, função de perda baseada em vizinhança de perda negativa (nll) multiclasse e regularizador L2 com lambda em 10^{-4}. No algoritmo de agrupamento K-Means, o número de *clusters* k a serem encontrados é determinado de acordo com a taxa de sumarização escolhida em cada experimento. O número de vezes que o algoritmo k-means é executado com diferentes centróide posicionados aleatoriamente é 50 e o número máximo de interações é 500.

- ***KGE-K-Means + Filtragem de entidades***: é a combinação de dois métodos anteriores, explorando tanto a aglutinação de entidades semelhantes em supernodos quanto a eliminação de entidades raras. Note que primeiro é realizada a sumarização e depois é aplicada a filtragem no sumário de GC (sGC) obtido.

A geração de amostras negativas para o ComplEx foi aleatória considerando todas as entidades e considerando-as como sujeito ou objeto das triplas geradas. Indica-se a identificação de técnicas de geração de amostras negativas mais adequadas em trabalhos futuros.

Tabela 7 – Estatísticas dos *GCs originais* de *Sun* e *Cao*

		GC de Sun	*GC de Cao*
Interações Usuário-Item	# Usuários	943	6,040
	# Itens	1.675	3.260
	# Avaliações	99.975	998.539
	Avg. Avaliação/Usuário	106	165
	Esparsidade	93,6%	94,9%
Informação Lateral	# Entidades	5.124	11.443
	# Propriedades	3	20
	# Declarações	12.311	428.777
	Esparsidade de GC	99,984%	99,983%

6.3.4 Recomendadores

Estes experimentos avaliam os seguintes SRGCs baseados em *embedding*, treinados com o *GC Original* e suas versões derivadas (fGCs, sGCs ou sfGCs):

- ***CFKG* (ZHANG et al., 2018)**: trata-se de um SRGC que aplica o modelo de *embedding* TransE (BORDES et al., 2013) em um grafo unificado com usuários, itens, entidades e propriedades.

- ***CKE* (ZHANG et al., 2016)**: trata-se de um SRGC que combina vários *embeddings* de itens de diferentes fontes, incluindo TransR (LIN et al., 2015) no GC.

- ***CoFM* (PIAO; BRESLIN, 2018)**: trata-se de um SRGC que aprende conjuntamente a representação do conhecimento, utilizando TransE, e o modelo de recomendação, utilizando máquina de fatoração (em inglês, *Factorization Machine* - FM).

- ***KTUP* (CAO et al., 2019)**: trata-se de um SRGC que aprende conjuntamente a representação de conhecimento, utilizando TransH, e o modelo de recomendação, utilizando o TUP (CAO et al., 2019).

As configurações de hiperparâmetro[4] foram definidos empiricamente para cada método, partindo da lista incompleta de configurações para MovieLens 1M disponibilizada em (CAO et al., 2019). Estas configurações consideram as características de ambos os conjuntos de dados (*Cao* e *Sun*) usados nestes experimentos e a necessidade de estabelecer condições igualitárias para realizar o estudo comparativo do tempo de treinamento (ou seja, restringindo o número de épocas de treinamento).

Para CFKG, CKE e CoFM, foram usados os *embeddings* pré-treinados BPRMF (método de fatoração de matriz) e TransE, enquanto KTUP usa TUP e TransH. O número de preferências do TUP (equivalente ao número de propriedades) é definido em 3 para os *GCs de Sun* e 20 para os *GCs de Cao*. Para os métodos CFKG, CKE, CoFM e KTUP, é configurado o hiperparâmetro de aprendizagem conjunta (*joint ratio*) λ em $0,5$. No BPRMF e no TUP, foi usado o coeficiente de regularização L_2 em 10^{-5} e o otimizador Adagrad. Nos outros métodos, foram adotados L_2 em 0 e o otimizador Adam. A taxa de aprendizagem é 0,001 para TransE e TransH, e 0,005 para os métodos restantes. Todos os métodos usam um número de lotes de 256 e dimensionalidade do *embedding* em 100. O número máximo de épocas e o uso da estratégia de parada antecipada são definidos para cada cenário experimental, como será descrito na Seção 6.3.6.

[4] A lista completa de parâmetros está disponível em github.com/juarezsacenti/kg-summ-rec

6.3.5 Métricas de avaliação

Conforme apresentado na Seção 6.2, o KG-Summ-Rec define um método para avaliar os impactos da sumarização em termos de redução do tamanho do GC, eficiência do modelo de recomendação e eficácia dos SRGC. Nestes experimentos, o tempo de treinamento em segundos foi adotado como métrica de eficiência.

Além disso, foram adotadas as seguintes métricas de acurácia: a precisão em *N* (p@N), cobertura em *N* (em inglês, *recall* – r@N), ganho cumulativo com desconto normalizado em *N* (em inglês, *Normalized Discounted Cumulative Gain at N* – nDCG@N) e média da precisão média em *N* (em inglês, *Mean Average Precision – mapatn*). Estas métricas e suas equações foram apresentadas na Seção 6.3.5.

6.3.6 Cenários experimentais

Como já apresentado, o objetivo deste estudo é o de avaliar os impactos da sumarização no custo de treinamento dos modelos de recomendação e na eficácia da recomendação. Para tanto, dois cenários experimentais são definidos. O primeiro cenário visa avaliar os impactos do aumento da taxa de sumarização no tempo de treinamento do modelo e na eficácia da recomendação. Neste cenário, os SRGCs escolhidos são treinados com um número fixo de épocas para permitir uma comparação justa de seus tempos de treinamento usando sumários de GC distintos. Este cenário experimental adota o *conjunto de dados de Sun* e a validação cruzada *5-fold* com particionamento aleatório seguindo as proporções 6:2:2 para treinamento, validação e teste, respectivamente. Neste primeiro cenário experimental, restringimos o treinamento de TransE e TransH a 1000 épocas e o treinamento de CFKG, CKE, CoFM e KTUP a 500 épocas. Esses parâmetros foram escolhidos para evitar sobre-ajuste (em inglês, *overfitting*), agilizar o processamento e fornecer condições semelhantes de treinamento. Além disso, como o BPRMF e o TUP são modelos de recomendação de itens que consideram apenas as avaliações usuário-item, portanto não são afetados pela sumarização e por isso foram treinados uma única vez para cada fold usando a estratégia de parada antecipada. Por exemplo, para o *fold-0* o modelo BPRMF foi treinado uma única vez e reusado como modelo pré-treinado para os recomendadores CFKG, CKE e CoFM que são treinados com por sua vez com o GC original ou um sGC.

No segundo cenário experimental, avaliamos os impactos na eficácia dos SRGCs baseados em *embedding* usando uma estratégia de parada antecipada para treinamento, que atinge um número maior de épocas do que no primeiro cenário. A estratégia de parada antecipada interrompe o treinamento antes que o desempenho pare de melhorar, permitindo comparar a eficácia ótima dos SRGCs. Portanto, neste cenário, o número de épocas em cada treinamento é variável. Este cenário usa os dois conjuntos de dados do domínio de filmes (*Cao* e *Sun*) com validação *hold-out*, separando aleatoriamente cada conjunto em subconjuntos de treinamento, validação e teste na proporção 7:1:2.

6.3.7 Implementação

A implementação do método de sumarização KGE-K-Means usa Python 3.7, métodos de tradução para *embedding* da biblioteca Ampligraph (COSTABELLO et al., 2019) 1.3.2 e o método de agrupamento K-Means do módulo Python Scikit-learn 0.23.2, que utiliza a distância euclidiana (Equação 2.1 apresentada no Capítulo 2). Os SRGCs empregados foram CFKG, CKE, CoFM e KTUP cujas implementações foram obtidas na página do Projeto de Cao[5]. Os modelos de recomendação foram adaptados para adotar métricas suportadas pelo *framework* CaseRecommender (COSTA et al., 2018) 1.1.0. As métricas de avaliação de CaseRecommender foram adotadas para padronizar esta tarefa, visando a comparação de SRGCs implementados por outros autores além dos implementados por Cao et al. (2019) em trabalhos futuros.

Os experimentos foram executados em uma CPU Intel (R) Xeon (R) E5-2640 v4 @ 2,40 GHz com 10 núcleos físicos (HT habilitado) e dois nós NUMA (20 núcleos físicos + HT). Este servidor possui 128 GB de RAM disponíveis e um NVIDIA Tesla K40c.

[5] Disponível em: https://github.com/TaoMiner/joint-kg-recommender, acesso em: 22/11/2021.

6.4 RESULTADOS EXPERIMENTAIS

Esta seção apresenta os resultados dos experimentos descritos na seção anterior, que avaliam os impactos da sumarização de Grafos de Conhecimento (GCs) na eficiência e na eficácia de Sistemas de Recomendação baseados em GCs (SRGCs). A Seção 6.4.1 avalia a redução do volume do GC após a sumarização, apresentando e comparando várias estatísticas dos GCs originais e sumários de GCs (sGCs). Depois, as Seções 6.4.2 e 6.4.3 avaliam os resultados relativos aos impactos da sumarização no tempo de treinamento (em segundos) do modelo de SRGCs e na qualidade da recomendação (em termos de precisão, cobertura, nDCG e mAP). Finalmente, a Seção 6.5 discute os resultados encontrados. Os resultados apresentados na Seção 6.4.2 são obtidos através de validação cruzada *5-fold* e os demais resultados, através da validação *hold-out*.

6.4.1 Impacto da sumarização na redução do grafo de conhecimento

As tabelas 8 e 9 apresentam os impactos da Sumarização KGE-K-Means no *GC de Sun* usando as estratégias de única-visão e multi-visão, respectivamente. Ambas as tabelas fornecem uma análise quantitativa desses GCs e os resultados obtidos com taxas de sumarização α de 25%, 50% e 75%. Nestas tabelas, o primeiro e o segundo grupos apresentam os números de entidades e de declarações que descrevem a informação lateral de filmes, sem contabilizar os números de filmes, de usuários e de interações usuário-item, antes e depois da sumarização. A última linha dessas tabelas apresenta a esparsidade estimada para cada versão do GC.

Como pode ser visto na Tabela 8, na estratégia de única-visão, o número de entidades após a sumarização está diretamente relacionado a taxa de sumarização. No entanto, a taxa de sumarização em cada tipo de entidade (visões de Gênero, Diretor e Ator) é diferente da taxa de sumarização global. Observa-se nesta estratégia que, quanto maior o número de entidades (indivíduos) de uma classe, maior será a redução de volume destas entidades. Por sua vez, a estratégia de multi-visão (Tabela 9) reduz uniformemente o número de entidades.

Nos sGCs sumarizados pela estratégia única-visão, o número de declarações foi reduzido em 89,4%, 80% e 76,7%, respectivamente com a taxa de sumarização em 25%, 50% e 75%. Já quando usada a estratégia multi-visão, o número de declarações foi reduzido em 90%, 79,3% e 70,6%, respectivamente com a taxa de sumarização em 25%, 50% e 75%. Devido ao fato que a KGE-K-Means é uma sumarização baseada em supernodos, a taxa de sumarização está relacionado a redução do número de nodos (que são aglutinados em supernodos). O impacto da sumarização sobre o número de declarações é dependente o número de declarações duplicadas que são removidas durante o processo de sumarização.

Nas taxas de sumarização em 50% e 75%, a redução do número de declarações foi maior na estratégia de multi-visão. Isto porque a estratégia de única-visão não aglutinou *Gêneros* em supernodos devido a baixa diversidade de entidades deste tipo e a baixa proximidade destas entidades com as demais no espaço vetorial modelado pelo *embedding*. Isto impediu a

Tabela 8 – Impactos da Sumarização KGE-K-Means Única-visão no *GC de Sun*

Métricas	GC original	Taxas de sumarização de KGE-K-Means α=25%	α=50%	α=75%
# Gêneros	24	24 (100%)	24 (100%)	24 (100%)
# Diretores	1.153	1.104 (95,8%)	911 (79%)	211 (18,3%)
# Atores	3.947	2.715 (68,8%)	1.639 (41,5%)	1.107 (28%)
# Entidades	5.124	3.843 (75%)	2.562 (50%)	1.281 (25%)
# *hasGenre*	3.974	3.974 (100%)	3.974 (100%)	3.974 (100%)
# *hasDirector*	1.800	1.746 (97%)	1.672 (92,9%)	1.664 (92,4%)
# *hasActor*	6.537	5.289 (80,9%)	4.207 (64,4%)	3.806 (58,2%)
# Declarações	12.311	11.009 (89,4%)	9.853 (80%)	9.444 (76,7%)
Esparsidade de GC	99,984%	99,975%	99,950%	99,808%

Tabela 9 – Impactos da Sumarização KGE-K-Means Multi-visão no *GC de Sun*

Métricas	GC original	Taxas de sumarização de KGE-K-Means α=25%	α=50%	α=75%
# Gêneros	24	18 (75%)	12 (50%)	6 (25%)
# Diretores	1.153	865 (75%)	577 (50%)	289 (25%)
# Atores	3.947	2.961 (75%)	1.974 (50%)	987 (25%)
# Entities	5.124	3.844 (75%)	2.563 (50%)	1.282 (25%)
# *hasGenre*	3.974	3.873 (97,5%)	3.588 (90,3%)	3.271 (82,3%)
# *hasDirector*	1.800	1.673 (92,9%)	1.669 (92,7%)	1.667 (92,6%)
# *hasActor*	6.537	5.542 (84,8%)	4.503 (68,9%)	3.754 (57,4%)
# Declarações	12.311	11.088 (90%)	9.760 (79,3%)	8.692 (70,6%)
Esparsidade de GC	99,984%	99,975%	99,950%	99,824%

redução de 3.974 declarações do tipo *temGênero*. A estratégia multi-visão permite distribuir a taxa de sumarização igualmente em cada visão, independente do número de entidades e da proximidade vetorial. Isto proporcionou o aumento no número de declarações duplicatas após a aglutinação de entidades.

Em particular, as declarações com a propriedade *temGênero* apresentam maior coocorrência de entidades, o que potencializa a geração de declarações duplicatas. Já as declarações com a propriedade *temDiretor* apresentam menor coocorrência, sendo das propriedade do conjunto de dados a mais próxima de uma relação *1-1*, visto que 1.675 filmes estão conectados a 1.153 diretores por 1.800 declarações (poucos filmes tem mais de um diretor). As declarações da propriedade *temAtor* apresentam maior coocorrência pois os filmes apresentam em geral 3 atores ou mais e atores geralmente atuam em mais de um filme.

Finalmente, a Sumarização KGE-K-Means resultou em uma pequena redução da esparsidade do GC. Esta redução da esparsidade pode ser justificada, pelo menos em parte, pelo fato de que a sumarização proposta reduz o número de entidades e agrupando-as em supernodos, reduzindo o espaço de declarações formadas por pares de entidades. Considerando a esparsidade do GC como $n/r * e^2$ (descrito em 6.3.2), a redução do número de declarações n

(causada pela remoção de declarações duplicatas após a substituição de entidades por super-nodos) foi inferior ao impacto da redução do número de entidades *e*, ocasionando a redução observada na esparsidade de GC.

Conforme apresentado na Seção 6.3.3, a Sumarização KGE-K-Means pode ser combinada com a *Filtragem de Entidade* infrequente para aumentar a taxa de sumarização final. A Figura 23 apresenta:

- (i) as taxas de poda para entidades e declarações, obtidas com o conjunto de dados de *Sun* ao usar apenas a *Filtragem de Entidade* (fGC), onde foi removido declarações de entidades infrequentes (entidades que aparecem em 10 declarações ou menos). Como visto na Figura 23a, esta filtragem podou 74,7% das entidades e, em consequência, 65,6% das declarações foram removidas de fGC.
- (ii) as taxas de sumarização, ao usar a Sumarização KGE-K-Means (sGCs). A Figura 23b apresenta as taxas de sumarização de entidades e declarações obtidas apenas por KGE-K-Means (sGC) nas estratégias de única-visão e multi-visão com valores diferentes para α (25%, 50% e 75%).
- (iii) a taxa de redução do grafo ao combinar os dois métodos de pré-processamento de GC (sumarização e filtragem de entidades, sfGCs), ilustrado pela Figura 23b.

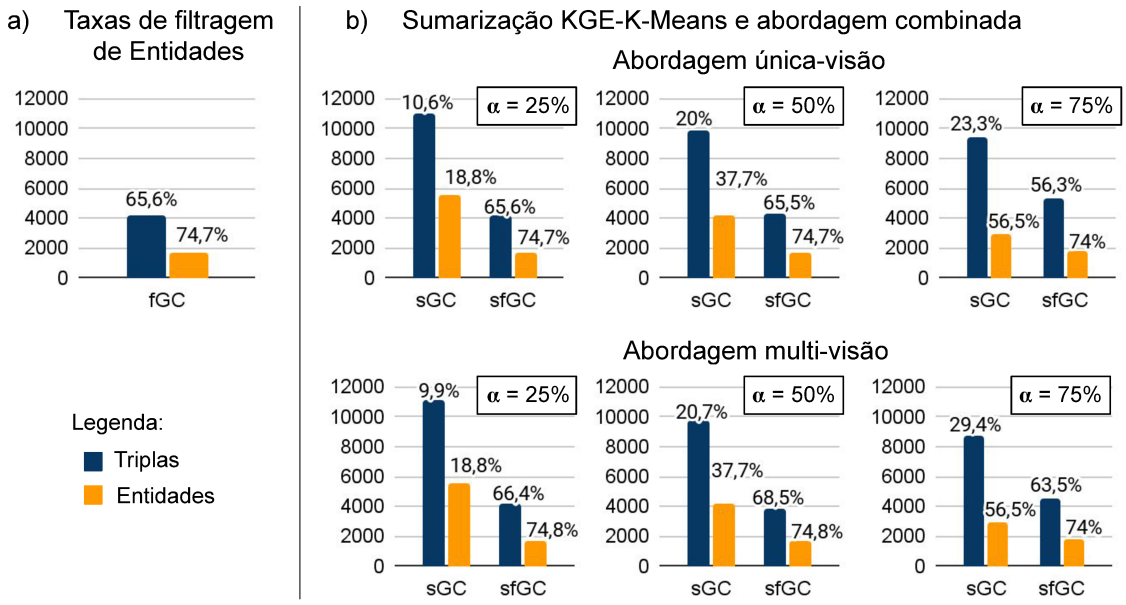

Figura 23 – Taxas de poda e sumarização da filtragem de entidade e sumarização KGE-K-Means

Fonte: criado pelo autor. Publicado pela primeira vez em Journal of Intelligent Information Systems, 2021, pela Springer Nature.

Como pode ser visto na Figura 23, as taxas de sumarização observadas nos sGCs são menores que as taxas de poda de fGC. A taxa de sumarização de entidades é menor que α porque KGE-K-Means não sumariza filmes. Uma observação importante é que a redução de

volume do GC obtida com o pré-processamento combinado é um pouco menor do que a obtida apenas usando a *Filtragem de Entidade*. Portanto, KGE-K-Means melhora a redução de volume do GC quando é combinada a *Filtragem de Entidade*.

6.4.2 Custos de treinamento pela eficácia da recomendação

Conforme descrito na Seção 6.3.6, o primeiro cenário experimental analisa os impactos da Sumarização KGE-K-Means nos custos de treinamento e na eficácia da recomendação nos SRGCs selecionados por meio de experimentos utilizando o conjunto de dados de *Sun*. A Tabela 10 apresenta os resultados desses experimentos em termos do tempo de treinamento em segundos e das métricas de eficácia de SR descritas na Seção 6.3.5. Estes resultados referem-se à validação cruzada *5-fold*.

Procurando oferecer subsídios para responder a Pergunta da Pesquisa 3, foi utilizado o teste não paramétrico de Wilcoxon (1945) para avaliar a significância estatística das variações de precisão apresentadas na Tabela 10. Primeiro, foi testada a aceitação da hipótese nula, de que não há diferença entre a precisão de modelos treinados com sGCs e GCs original. Nenhum dos sGCs rejeitou a hipótese nula em um nível de confiança de 5%, indicando que não houve diferença significativa de precisão entre os modelos. No entanto, alguns resultados do valor-p foram próximos a 0,05 (0,0625 de valor-p) no teste, indicando que a rejeição da hipótese pode ter sido causada pela pequena quantidade de repetições do experimento (5-*folds*). Além disso, foi testado se as medidas de precisão de modelos treinados com GCs original são maiores do que as de modelos treinados com sGCs. Neste teste, apenas as medidas de CKE usando a estratégia de sumarização de única-visão com $\alpha = 75$ rejeitaram a hipótese nula em um nível de confiança de 5%, apresentando um valor-p de 0,03125, indicando que este modelo teve um resultado significativamente maior de precisão comparado aos outros modelos. Finalmente, também foi testado se os modelos treinados com GC original têm precisão inferior a dos modelos treinados com sGCs. Neste teste, apenas os métodos CFKG usando a estratégia de multi-visão com $\alpha = 25$ e CoFM usando a estratégia de única-visão com $\alpha = 75$ rejeitaram a hipótese nula em um nível de confiança de 5% (0,03125 de valor-p), indicando que estes modelos obtiveram precisão menor que a dos demais modelos.

A Figura 24 permite comparar a precisão média dos 10 primeiros itens recomendados (p@10) com o tempo de treinamento (segundos) de cada SRGC analisado. O tempo de treinamento (linhas tracejadas) é claramente reduzido com o aumento da taxa de sumarização. Essa tendência foi observada em todos os SRGCs. O menor impacto na eficiência do treinamento foi observado em CKE com multi-visão e $\alpha = 25\%$: uma redução média de 0,25% e desvio padrão de 30,68s. O maior impacto foi em CoFM com multi-visão e $\alpha = 75\%$: 5,85% de redução média e desvio padrão de 10,31s. Como pode ser visto na Figura 24, a taxa de sumarização tem um baixo impacto na precisão (linhas sólidas) de todos os SRGCs. O mesmo também foi observado em outras métricas de eficácia.

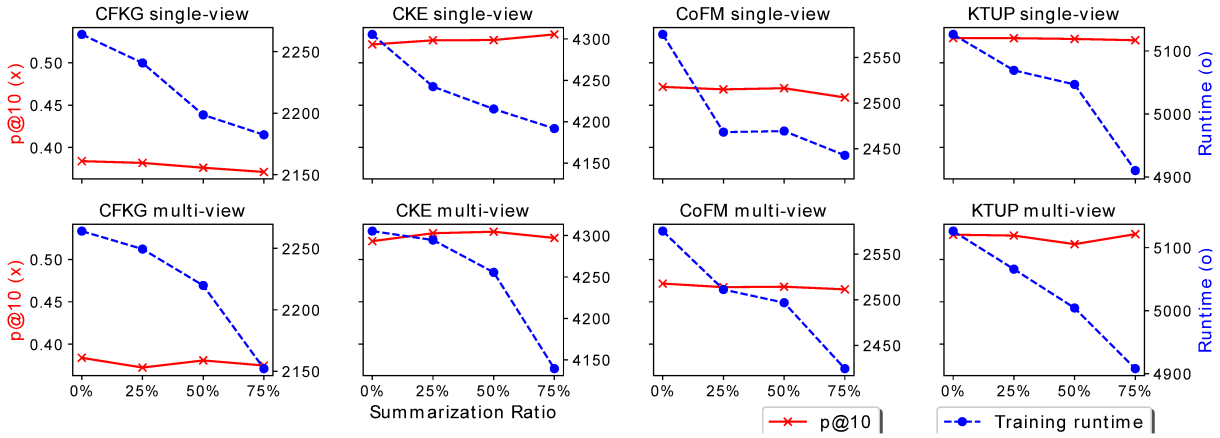

Figura 24 – Avaliação de eficiência-eficácia

Fonte: criado pelo autor. Publicado pela primeira vez em Journal of Intelligent Information Systems, 2021, pela Springer Nature.

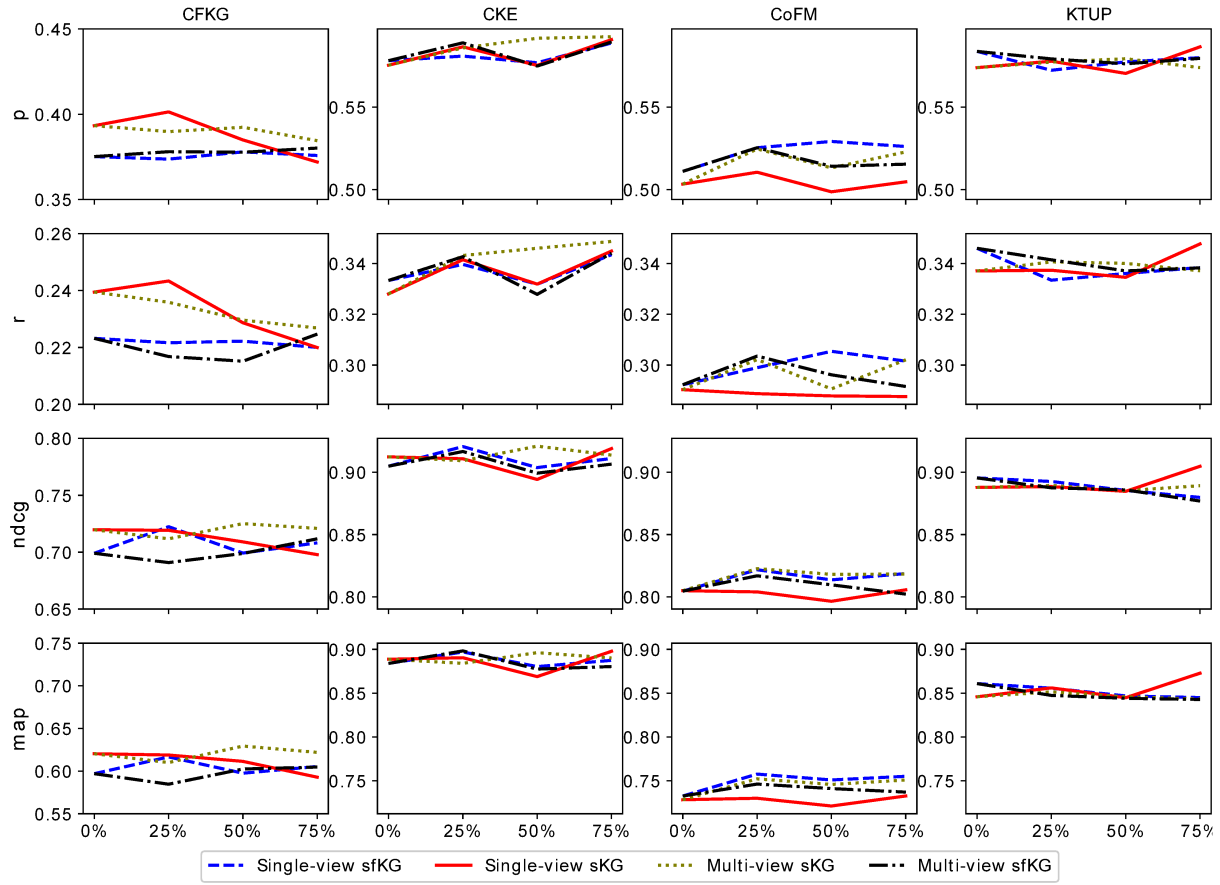

Figura 25 – Resultados de qualidade de recomendação de SRs baseado em GC treinados com sGCs e sfGCs de *Sun*

Fonte: criado pelo autor. Publicado pela primeira vez em Journal of Intelligent Information Systems, 2021, pela Springer Nature.

Tabela 10 – Avaliação eficiência-eficácia

SRGC	Estratégia SG	α	p@10	r@10	ndcg@10	map@10	tempo (s)
CFKG	GC original	-	**0,3839**	**0,2343**	0,7033	0,6011	2264
	Única-visão	25%	0,3818	0,2299	0,7040	**0,6017**	2240,8
		50%	0,3761	0,2277	0,7008	0,5976	2198,6
		75%	0,3711	0,2248	0,6960	0,5912	2182,4
	Multi-visão	25%	0,3724	0,2248	0,6894	0,5836	2249,4
		50%	0,3808	0,2292	**0,7053**	0,6010	2219,8
		75%	0,3748	0,2278	0,6989	0,5958	**2152,2**
CKE	GC original	-	0,5216	0,2987	0,8487	0,8026	4305,4
	Única-visão	25%	0,5265	0,3028	0,8539	0,8094	4242,4
		50%	0,5268	0,3040	0,8513	0,8058	4215,4
		75%	**0,5334**	0,3095	**0,8573**	0,8110	4191,8
	Multi-visão	25%	0,5309	0,3068	0,8537	0,8097	4294,6
		50%	0,5327	**0,3096**	0,8565	**0,8114**	4255,6
		75%	0,5254	0,3015	0,8465	0,8015	**4139,4**
CoFM	GC original	-	**0,4715**	**0,2713**	**0,7709**	**0,6917**	2575,2
	Única-visão	25%	0,4686	0,2705	0,7666	0,6868	2468,2
		50%	0,4700	0,2705	0,7700	0,6889	2469,4
		75%	0,4591	0,2625	0,7642	0,6809	2442,8
	Multi-visão	25%	0,4673	0,2678	0,7644	0,6847	2511,2
		50%	0,4678	0,2701	0,7665	0,6862	2496,8
		75%	0,4647	0,2657	0,7639	0,6826	**2424,6**
KTUP	GC original	-	0,5292	0,3069	0,8442	0,7916	5126,2
	Única-visão	25%	0,5290	0,3079	0,8454	0,7938	5069,2
		50%	0,5281	0,3056	0,8418	0,7886	5046,8
		75%	0,5266	0,3037	0,8470	0,7934	4910,2
	Multi-visão	25%	0,5282	0,3083	0,8411	0,7881	5065,8
		50%	0,5180	0,3000	0,8285	0,7746	5004,2
		75%	**0,5299**	**0,3086**	**0,8478**	**0,7966**	**4908**

6.4.3 Eficácia da recomendação com treinamento ótimo

O segundo cenário experimental avalia o impacto da sumarização sobre a eficácia de SRGCs analisados usando uma estratégia de parada antecipada durante a fase de treinamento dos *embeddings*. Esta estratégia permite atingir uma eficácia de SRGCs próxima da ótima. A Figura 25 apresenta as medidas de eficácia dos SRGCs treinados com o GC de *Sun* (GC original ou sGC em $\alpha = 0$), com o GC pré-processado por *Filtragem de Entidade* infrequente (fGC ou sfGC em $\alpha = 0$), com os GCs sumarizados pelo KGE-K-Means (sKGs) e pré-processados pelos métodos combinados (sfKGs). O eixo *X* refere-se à taxa de sumarização de GC (α), que varia de 0% a 75%.

Como pode ser observado na Figura 25, o uso da Sumarização KGE-K-Means não

impacta significativamente a eficácia da recomendação, confirmando o que foi observado no cenário experimental anterior (Seção 6.4.2). A eficácia pode ser melhor analisada na Figura 26, que permite comparar as medidas de SRGCs, com a exceção de CFKG que foi oculto devido ao seu baixo desempenho, treinados com o GC original e suas versões derivadas. CKE usando a estratégia de multi-visão e α = 75% sem *Filtragem de Entidade* (sKG-mv-75) foi o melhor SRGC em relação às métricas de precisão e cobertura. O melhor resultado de nDCG foi alcançado pelo CKE sKG-mv-50 e o melhor mAP por CKE sfKG-mv-25 (com *Filtragem de Entidade* infrequente).

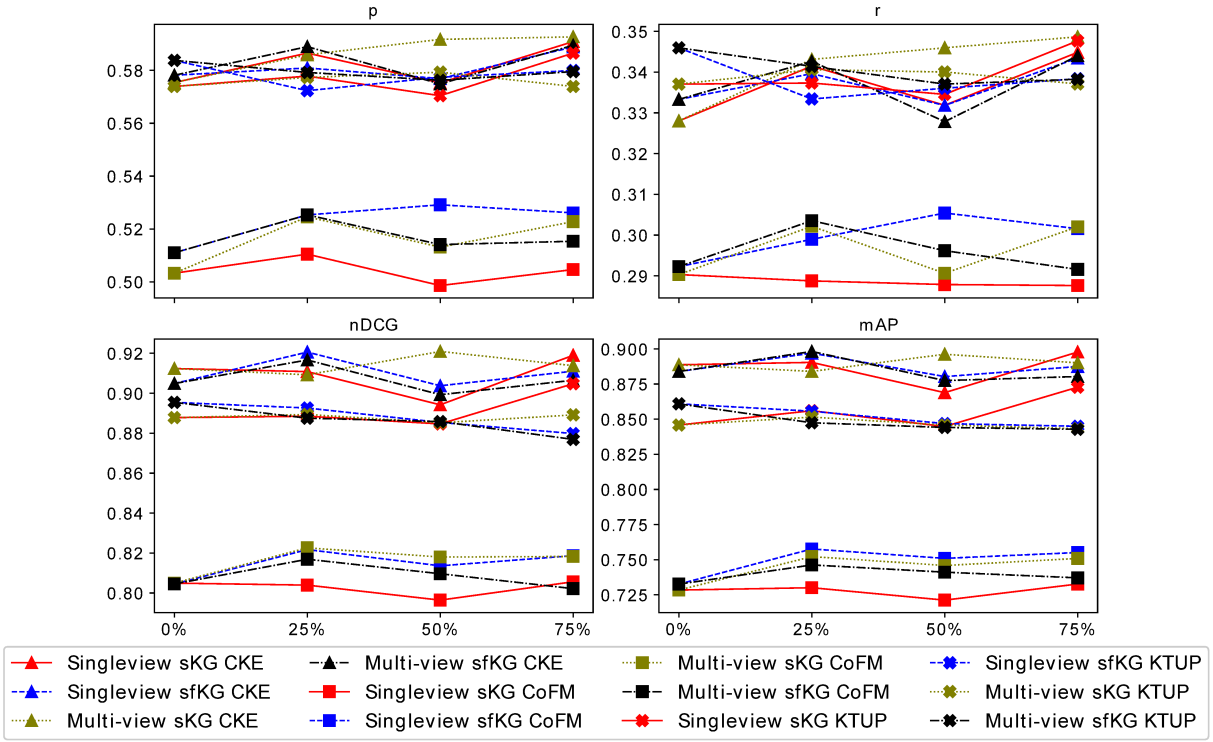

Figura 26 – Resultados de qualidade de recomendação para sGCs e sfGCs de Sun (sem resultados de CFKG)

Fonte: criado pelo autor. Publicado pela primeira vez em Journal of Intelligent Information Systems, 2021, pela Springer Nature.

Finalmente, os impactos da sumarização do KGE-K-Means no tempo de treinamento e na eficácia da recomendação também foi avaliado usando o conjunto de dados de *Cao*. A estratégia de única-visão de 25% sem filtragem (sv-sKG-25) alcançou uma redução no tempo de treinamento de 12,82% para CFKG e 22,71% para CoFM. No entanto, a eficácia da recomendação com sGCs foi um pouco pior do que a obtida com o *GC original*, para todas as métricas consideradas. Mais experimentos são necessários para encontrar configurações de sumarização que melhorem a eficácia da recomendação.

6.5 CONSIDERAÇÕES FINAIS

O método proposto, Sumarização KGE-K-Means, mostrou-se útil para reduzir o tempo de treinamento, sem impactos significativos na eficácia de SRGCs baseados em *embedding*. Uma vez que a informação lateral representada no GC é geralmente mais estável do que as interações usuário-item, o GC precisa ser sumarizado com menos frequência do que o retreino do modelo de recomendação deve ser feito. Em outras palavras, o mesmo sGC pode ser reutilizado para treinar o modelo de um SRGC repetidamente, adicionando apenas as novas interações usuário-item. Além disso, considerando o conjunto de dados de *Sun*, a aceleração do tempo de treinamento obtido usando um sGC apenas uma única vez é cerca de metade do tempo gasto por nosso método para sumarizar o GC. Consequentemente, o tempo geral gasto para manter o modelo de recomendação atualizado pode ser reduzido significativamente.

Os experimentos com o GC de *Sun* apresentam resultados positivos, mesmo apesar do tamanho relativamente pequeno desse conjunto de dados. Como se deveria esperar, os experimentos com o GC de *Cao* resultaram em ganhos maiores devido à maior quantidade de informação lateral.

A estratégia de multi-visão permitiu mais agilidade no treinamento do que a estratégia de única-visão, para a maioria dos SRGCs e, principalmente, para uma taxa de sumarização de 75%. Por outro lado, não houve mudança significativa na eficácia da recomendação usando estratégias de sumarização distintas (única-visão e multi-visão). Finalmente os modelos CKE e KTUP foram os que alcançaram os melhores resultados de qualidade de recomendação para ambas as estratégias de sumarização e taxas de sumarização distintas, incluindo sem sumarização (*baseline*).

A seguir, o Capítulo 7 apresenta a conclusão desta tese, enumerando as limitações de cada abordagem para mitigar o problema do alto custo de treinamento de modelos de recomendação baseados em informações laterais, as contribuições desta tese e indicando trabalhos futuros.

7 CONCLUSÃO

Sistemas de recomendação baseados em conhecimento (SRCs) têm explorado informações laterais para mitigar o problema da esparsidade em interações usuário-item e, assim, melhorar a qualidade das recomendações. Entretanto, a adição de informações laterais acarreta uma maior complexidade ao treinamento do SRC, devido principalmente ao aumento no volume e na diversidade dos dados considerados pelo treinamento. Neste sentido, para o uso efetivo de SRC em sistemas reais, é importante a realização de estudos visando analisar, além da eficácia (qualidade da recomendação), a eficiência do sistema em termos de custo computacional ou tempo de resposta. Maior parte dos trabalhos na área realizam análise de eficácia dos SRCs, mas maior parte negligencia sua eficiência. Portanto, análises sobre as diferentes representações de conhecimento em sistemas de recomendação baseados em ontologias (SROs) e em grafos de conhecimento (SRGCs), especialmente a análise do impacto destas representações na qualidade da recomendação e no custo computacional, oferecem relevantes oportunidades de pesquisa.

Esta tese teve como objetivo investigar os impactos em termos de eficácia e eficiência das formas de representação de conhecimento em SRCs, além de propor e avaliar abordagens para mitigar os problemas de eficiência em SRCs. Principalmente buscou-se definir meios de melhorar a eficiência do processo de treinamento do modelo de preferências do SRCs a partir da redução do tempo deste treinamento. Esta tese avaliou a seguinte representação de conhecimento baseada em ontologias e grafos de conhecimento (GCs), que é utilizada como entrada do treinamento do modelo de preferência de SRCs e tem como meta melhorar a eficácia e/ou a eficiência da recomendação. Esta tese avaliou a seguinte hipótese: *Múltiplos aspectos, hierarquias e visões para representar e organizar a informação lateral e aliados à sumarização podem melhorar a eficiência do treinamento e a eficácia de sistemas de recomendação baseados em ontologias e grafos de conhecimento*.

Esta tese avaliou a representação das informações laterais em SRs considerando diferentes organizações de características de itens (em termos de aspectos, hierarquias e visões) e diferentes técnicas de sumarização. Esta tese investigou duas abordagens visando melhorar a eficiência de SRCs a partir da redução do tempo de treinamento do modelo. Uma delas, focada em SROs, visa converter o modelo de preferência de usuário (estruturados por ontologias ou grafos de conhecimento) em representações menos complexas e mais eficientes, que permitem o uso de técnicas de recomendação de baixo custo computacional, como as baseadas em matrizes e vetores. A segunda abordagem é focada nos SRGCs, que visa reduzir o volume e a diversidade da informação lateral por meio de técnicas de redução de dimensionalidade, seleção de características e sumarização.

Considerando a primeira abordagem, esta tese propôs uma Ontologia de Perfil de Usuário e um arcabouço conceitual para SROs, chamado de ORBS (acrônimo para SR Baseado em Ontologia), que suporta: (i) a construção de perfis ontológicos de usuário (POUs) usando informação lateral sobre itens, ontologias e coleções de dados conectados, (ii) a construção de

hierarquias de fatores de interesse, (iii) a determinação de vizinhança considerando hierarquias de fatores de interesse, (iv) a análise de erro de predição e comparação entre diferentes SRs. Este arcabouço permite criar e analisar um SRO híbrido que determina a vizinhança de um usuário com base em informações laterais e interações usuário-item explícitas e implícitas, mas que determina o ranqueamento das recomendações utilizando apenas interações explícitas. A Ontologia de Perfil de Usuário tem o potencial para promover a fácil adaptação do SRO em relação ao domínio do item a ser recomendado, embora esta funcionalidade deva ser validada em avaliações futuras que estão além do escopo desta tese.

Norteada pela segunda abordagem, esta tese investigou o uso de Sumarização de Grafo (SG) como uma etapa de pré-processamento de SRGCs. Com este intuito, esta tese propôs um método de SG aplicado à tarefa de recomendação, denominado Sumarização KGE-K-Means. Este método combina a técnica de *embedding* com a técnica de agrupamento de nodos de GCs, e apresenta duas abordagens distintas: (i) a única-visão, que sumariza o GC como um todo, e (ii) a multi-visão, que sumariza subgrafos do GC de forma independente. Além disso, esta tese propôs um método, denominado KG-Summ-Rec, para avaliar o impacto de técnicas de sumarização no pré-processamento de SRGCs em termos de eficácia e eficiência.

7.1 RESULTADOS EXPERIMENTAIS OBTIDOS E LIMITAÇÕES

O primeiro experimento realizado nesta tese endereçou as perguntas de pesquisa 1, "A representação de informação lateral por meio de características relacionadas aos múltiplos aspectos do item resulta na melhoria da eficácia dos resultados de SRs?", e 2, "A representação de informação lateral por meio de características organizadas em taxonomias ou ontologias hierárquicas resulta na melhoria da eficácia dos resultados de SRs?". Para tal, foi adotado o arcabouço ORBS para criar SROs treinados com diferentes conjuntos de informação lateral: *Classic*, que não considera informação lateral; *MA-Genre* que considera como informação lateral apenas o aspecto gênero de filme não organizado por hierarquia; *ORBS-Genre*, *ORBS-Date* e *ORBS-G+D*, que consideram como informação lateral os aspectos de gênero, data de lançamento e ambos, respectivamente, organizados por hierarquias. Estes SROs foram treinados utilizando dados do MovieLens 1M[1] (HARPER; KONSTAN, 2015) enriquecidos com a ontologia Movie Ontology[2], segundo os critérios de validação *hold-out*. A implementação deste experimento foi publicada em https://github.com/juarezsacenti/ORBS, viabilizando futura reprodução.

Neste experimento, *ORBS-G+D* resultou um erro inferior ao de *ORBS-Genre* e *ORBS-Date*. Esta constatação trouxe indícios para responder a pergunta da pesquisa 1, onde constatou-se que o uso de múltiplos aspectos como informação lateral melhora a eficácia da recomendação. Este resultado está de acordo com Soares e Viana (2017), onde os autores concluíram que a combinação de diferentes informações (aspectos) na formação do PU obtém performance mais estável que em casos de teste com PU que consideram um único aspecto. Além disso,

[1] Acesso: http://movielens.org/, em: 22/11/2021.
[2] Acesso: http://www.movieontology.org:80/2010/01/movieontology.owl, em: 14/06/2018.

ORBS-Genre teve um erro inferior ao de *MA-Genre*, oferecendo indícios para responder a pergunta da pesquisa 2, de que o uso de hierarquia na representação da informação lateral melhora a eficácia da recomendação. Este resultado entra em acordo com Fernandes, Sacenti e Willrich (2017), que demonstraram que o tamanho e complexidade de hierarquias também influencia na qualidade das recomendações.

Entretanto, estes indícios não são suficientes para responder de forma categórica as perguntas de pesquisa 1 e 2, devido às seguintes limitações dos experimentos realizados nesta primeira abordagem: (i) a baixa confiabilidade dos resultados obtidos (dado que a forma de validação adotada foi o *hold-out*); (ii) o uso de um único conjunto de dados (MovieLens 1M) e de um único domínio (filme); e (iii) a baixa diversidade de aspectos e hierarquias (gêneros e data de lançamento).

A abordagem da conversão da representação para mitigar o custo computacional da informação lateral permite a utilização de técnicas baseadas em matrizes, que são mais exploradas pela literatura e, portanto, mais acessíveis e escaláveis. Esta abordagem demonstrou melhora de eficiência do SR. O tempo de resposta do treinamento desta abordagem é na ordem de minutos, enquanto o uso abordagens que realizam treinamentos com base em grafos de conhecimento levam horas. De modo geral, a aplicabilidade de ORBS se estende aos cenários de recomendação de itens que apresentem disponibilidade e diversidade de informação lateral, em termos de aspectos e hierarquias.

A segunda abordagem para melhora da eficiência em SRGCs visou encontrar respostas para as perguntas de pesquisa 3, “A sumarização de informação lateral resulta na melhoria da eficiência do treinamento sem prejuízo na eficácia dos resultados de SRs?”, e 4, “A abordagem de sumarização multi-visão resulta na melhoria da eficiência do treinamento e/ou eficácia dos resultados de SRs, quando comparada com a abordagem de única-visão?”. Nesta abordagem, para melhoria da eficiência de SRGCs, adotou-se um método de sumarização de grafo KGE-K-Means como etapa de pré-processamento para reduzir o volume dos dados usados no treinamento de diferentes SRGCs: CFKG, CKE, CoFM e KTUP. O experimento adotou diferentes taxas de sumarização: 0%, 25%, 50% e 75%. Estes SRGCs foram treinados utilizando dois conjuntos de dados distintos: o de Cao (CAO et al., 2019), que combina MovieLens 1M com informações laterais do IMDb; e o de Sun (SUN et al., 2018), que combina MovieLens 1M com informações laterais do DBpedia. O experimento adotou dois cenários diferentes: hold-out com parada antecipada do treinamento e 5-fold com épocas fixas no treinamento. Além disso, o experimento comparou as abordagens única-visão e multi-visão da sumarização KGE-K-Means com a filtragem de entidades infrequentes e considerou quatro métricas de eficácia (precisão, cobertura, ganho cumulativo com desconto normalizado e média da precisão média, nos N primeiros resultados) e uma de eficiência (tempo de treinamento em segundos). A implementação deste experimento foi publicada em https://github.com/juarezsacenti/kg-summ-rec, viabilizando futura reprodução.

Neste experimento, taxas de sumarização maiores produziram treinamento significativamente mais eficientes que taxas menores, sem mudanças significativas (ou tendência clara

de ganhos ou perdas) em termos de eficácia da recomendação. Este é um indício para responder a pergunta de pesquisa 3, pois constatou-se que a sumarização de informação lateral melhora a eficiência do treinamento sem prejuízo na eficácia dos resultados. Além disso, a abordagem multi-visão resultou em uma maior redução do tempo de treinamento quando comparada a única-visão, principalmente para a taxa de sumarização em 75%. Este é um indício para responder a pergunta de pesquisa 4, onde foi possível constatar que a abordagem de sumarização multi-visão melhora a eficiência do SR quando comparada à abordagem única-visão. Embora o uso da informação lateral melhore a eficácia da recomendação, a representação deste conhecimento necessita ser beneficiada por técnicas de redução do volume como sumarização ou seleção de características. Portanto, é necessária uma primeira etapa que reúne informações laterais diversas e uma segunda etapa que descobre quais informações laterais são uteis para o processo da recomendação.

Destaca-se aqui que os experimentos realizados nesta segunda abordagem apresentam algumas limitações: (i) a baixa replicação do experimento (validação *5-folds*, 2 conjuntos de dados); (ii) o uso de dados de um único domínio (filme); (iii) a baixa diversidade de técnicas de sumarização avaliadas (filtragem de entidades e KGE-K-Means).Uma dificuldade encontrada neste experimento foi o elevado tempo de treinamento de SRGCs baseados em *embedding* e a necessidade de uma máquina robusta para executar os experimentos. Outra dificuldade encontrada foi a complexidade das técnicas de SRGCs baseadas em *embedding* e a dificuldade de ajuste fino dos parâmetros e dos hiperparâmetros necessários.

De modo geral, a aplicabilidade de KGE-K-Means se estende aos cenários de recomendação de itens que apresentem disponibilidade e diversidade de informação lateral, em termos de visões de GCs. O treinamento dos SRGCs baseados em *embedding* avaliados durou horas para conjuntos de dados relativamente pequenos (com cerca de milhares de avaliações e informações laterais). Outros desafios em aberto desta abordagem são o viés adicionado pelo método de clusterização K-Means e o problema da explicabilidade das técnicas de *embedding*.

7.2 CONTRIBUIÇÕES DA TESE

Os seguintes trabalhos foram publicados no escopo desta tese:

1. Fernandes, B. B.; **Sacenti, J. A. P.**; Willrich, R.. *"Using implicit feedback for neighbors selection: Alleviating the sparsity problem in collaborative recommendation systems"*. Em: Proceedings of the 23rd Brazilian Symposium on Multimedia and the Web, Webmedia'17, Gramado, Brazil. NY, USA: ACM, 2017. p. 341–348. ISBN 978-1-4503-5096-9 (FERNANDES; SACENTI; WILLRICH, 2017).

2. **Sacenti, J. A. P.**; Willrich, R.; Fileto, R.. *"Hybrid recommender system based on multi-hierarchical ontologies"*. Em: Proceedings of the 24th Brazilian Symposium on Multimedia and the Web, WebMedia'18, Salvador, Brazil. NY, USA: ACM, 2018. p. 149–156. ISBN 978-1-4503-5867-5 (SACENTI; WILLRICH; FILETO, 2018).

3. **Sacenti, J. A. P.**; Willrich, R.; Fileto, R.. *"Knowledge graph summarization impacts on movie recommendations"*. Journal of Intelligent Information Systems, Springer, NY, USA, 2021 (SACENTI; FILETO; WILLRICH, 2021).

As principais contribuições desta tese, bem como as publicações referentes a cada contribuição, são enumeradas a seguir:

1. Uma ontologia de tarefa, chamada de Ontologia de Perfil de Usuário (OPU), que tem o potencial para promover a independência aos SROs (SACENTI; WILLRICH; FILETO, 2018).

2. ORBS[3], um arcabouço conceitual que suporta a construção de perfis ontológicos de usuário (POUs) e a análise de predição de erro SROs (SACENTI; WILLRICH; FILETO, 2018).

3. Um método de determinação de vizinhança de usuário baseada em informações laterais e interações usuário-item explícitas e implícitas (FERNANDES; SACENTI; WILLRICH, 2017).

4. KGE-K-Means, um método de SG que combina *embeddings* baseados em semântica latente (ComplEx) e agrupamento de nós (K-Means) nas abordagens de única-visão e multi-visão (SACENTI; FILETO; WILLRICH, 2021);

5. KG-Summ-Rec[4], um processo de recomendação que explora estratégias alternativas de SG como uma etapa de pré-processamento de SRGCs (SACENTI; FILETO; WILLRICH, 2021);

6. Uma avaliação do impacto do método SG proposto ao gerar sGCs com taxas de sumarização crescentes no tempo de treinamento do modelo do SR e na eficácia da recomendação (SACENTI; FILETO; WILLRICH, 2021).

Além disso, o autor realizou apresentações sobre esta tese em dois eventos:

1. **Sacenti, J. A. P.**; Willrich, R.; Fileto, R.. *"Hybrid recommender system based on multi-hierarchical ontologies"*. Em: Proceedings of the 24th Brazilian Symposium on Multimedia and the Web, WebMedia'18, Salvador, Brazil. NY, USA: ACM, 2018. p. 149–156. ISBN 978-1-4503-5867-5 (SACENTI; WILLRICH; FILETO, 2018).

2. **Sacenti, J. A. P.**; Willrich, R.; Fileto, R.. *"Exploiting Knowledge Graphs Structure and Embeddings on Knowledge-aware Recommender Systems"*. Em: The 1st ACM Latin American School on Recommender Systems, LARS 2019. Fortaleza, Brazil, 2019.

[3] Acesso: https://github.com/juarezsacenti/ORBS, em: 22-11-2021.
[4] Acesso: https://github.com/juarezsacenti/kg-summ-rec, em: 22-11-2021.

7.3 TRABALHOS FUTUROS

Como dito anteriormente, as diferentes representações de conhecimento em sistemas de recomendação e os impactos destas representações na qualidade e custo computacional da recomendação oferecem relevantes oportunidades de pesquisa. A informação lateral tem potencial para melhorar a eficácia de SRs, justificando seu aumento no custo computacional. A organização e sumarização da informação lateral permitem reduzir este custo, beneficiando o SR como um todo.

As limitações observadas na seção anterior implicam na necessidade da realização de novos experimentos que ampliem o escopo desta tese. A seguir, são enumeradas as sugestões para trabalhos futuros:

1. A investigação do potencial de ORBS em proporcionar aos SRs a independência de domínio do item, avaliando o desacoplamento e a facilidade de adaptação do SR a outros conjuntos de dados. OBRS foi avaliado apenas no domínio de filmes. É indicado como trabalho futuro a avaliação dos métodos propostos considerando outros domínios de item além de filmes (p.ex., livros como no conjunto de dados *amazon-books*, músicas como em *last-fm*, estabelecimentos ou pontos de interesse como em *yelp*).

2. A automatização da etapa de descrição do conhecimento de ORBS, atualmente efetuada manualmente pelo gerente do ORBS. Além disso, atualmente ORBS não possui uma estratégia para a declaração e geração automática de hierarquias complexas.

3. Esta tese propôs uma técnica de sumarização baseada em Complex e K-Means. Como trabalho futuro pode-se investigar quais técnicas de *embedding* são mais adequadas para capturar similaridade a ser considerada na sumarização de GCs com informações laterais úteis a SRGCs. Além disso, investigar outras técnicas de agrupamento alternativas ao K-Means.

4. A investigação sobre a geração de amostras negativas mais adequada para a técnica de *embedding* usada pela sumarização.

5. A investigação de uma abordagem multi-visão heterogênea que permite aplicar diferentes técnicas de sumarização em visões distintas.

6. A determinação de critérios apropriados para lidar com entidades que aparecem em mais de uma faceta (p.ex., ator e diretor) na sumarização multi-visão.

7. A determinação de uma técnica de remoção de supernodos duplicados (que aglutinam conjuntos de entidades equivalentes) para a sumarização multi-visão.

8. A investigação dos possíveis papéis das informações laterais sumarizadas para o problema da explicabilidade dos modelos de recomendação.

Atualmente, o autor desta tese planeja uma pesquisa subsequente para expandir os experimentos com novas abordagens de sumarização de GCs visando melhorar o desempenho da redução de ruído e a qualidade da recomendação (trabalho futuro 3). Além disso, planeja-se analisar uma abordagem alternativa de multi-visão, na qual diferentes métodos SG podem ser aplicados a cada visão do grafo (trabalho futuro 5).

REFERÊNCIAS

ADOMAVICIUS, G.; KWON, Y. New recommendation techniques for multicriteria rating systems. **IEEE Intelligent Systems**, IEEE, Manhattan, New York, v. 22, n. 3, 2007.

AGGARWAL, C. C. et al. **Recommender systems**. Berlin, Heidelberg: Springer, 2016. v. 1.

AGUILAR, J.; VALDIVIEZO-DÍAZ, P.; RIOFRIO, G. A general framework for intelligent recommender systems. **Applied Computing and Informatics**, Elsevier, 2016.

ALI, S. M. et al. Topic and sentiment aware microblog summarization for twitter. **J Intell Inf Syst**, Springer, Berlin, Heidelberg, v. 54, n. 1, p. 129–156, 2020.

ALSHAIKH, M. A.; UCHYIGIT, G.; EVANS, R. A research paper recommender system using a dynamic normalized tree of concepts model for user modelling. In: IEEE. **Research Challenges in Information Science (RCIS), 2017 11th International Conference on**. Manhattan, New York, 2017. p. 200–210.

ANAND, S. S.; KEARNEY, P.; SHAPCOTT, M. Generating semantically enriched user profiles for web personalization. **ACM Transactions on Internet Technology (TOIT)**, ACM, NY, USA, v. 7, n. 4, p. 22, 2007.

ANDERSON, J. R. A spreading activation theory of memory. **Journal of verbal learning and verbal behavior**, Elsevier, v. 22, n. 3, p. 261–295, 1983.

ANGELIS, A. D. et al. A social cultural recommender based on linked open data. In: **Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization**. New York, NY, USA: ACM, 2017. (UMAP '17), p. 329–332. ISBN 978-1-4503-5067-9. Disponível em: http://doi.acm.org/10.1145/3099023.3099092.

ARSHADI, N.; JURISICA, I. Maintaining case-based reasoning systems: A machine learning approach. In: **Adv in Case-Based Reason**. Berlin, Heidelberg: Springer, 2004. p. 17–31. ISBN 978-3-540-28631-8.

AYESHA, S.; HANIF, M. K.; TALIB, R. Overview and comparative study of dimensionality reduction techniques for high dimensional data. **Inf Fusion**, v. 59, p. 44–58, 2020. ISSN 1566-2535.

BAHRAMIAN, Z.; ABBASPOUR, R. A.; CLARAMUNT, C. A context-aware tourism recommender system based on a spreading activation method. **International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences**, v. 42, 2017.

BELLINI, V. et al. Auto-encoding user ratings via knowledge graphs in recommendation scenarios. In: **Proceedings of the 2Nd Workshop on Deep Learning for Recommender Systems**. New York, NY, USA: ACM, 2017. (DLRS 2017), p. 60–66. ISBN 978-1-4503-5353-3. Disponível em: http://doi.acm.org/10.1145/3125486.3125496.

BERNERS-LEE, T.; HENDLER, J.; LASSILA, O. The semantic web. **Scientific american**, JSTOR, v. 284, n. 5, p. 34–43, 2001.

BERNERS-LEE, T. et al. **Linked Data - Design Issues**. 1998. http://www.w3.org/DesignIssues/LinkedData.html.

BERNERS-LEE, T. et al. **Semantic web road map**. 1998.

BICKEL, S.; SCHEFFER, T. Multi-view clustering. In: IEEE. **ICDM**. Manhattan, New York, 2004. v. 4, p. 19–26.

BIZER, C.; HEATH, T.; BERNERS-LEE, T. Linked data-the story so far. **International journal on semantic web and information systems**, v. 5, n. 3, p. 1–22, 2009.

BOBADILLA, J. et al. Recommender systems survey. **Knowledge-based systems**, Elsevier, v. 46, p. 109–132, 2013.

BOBADILLA, J.; SERRADILLA, F. The effect of sparsity on collaborative filtering metrics. In: AUSTRALIAN COMPUTER SOCIETY, INC. **Proceedings of the Twentieth Australasian Conference on Australasian Database-Volume 92**. Darlinghurst, Australia, 2009. p. 9–18.

BOKDE, D. kumar; GIRASE, S.; MUKHOPADHYAY, D. An item-based collaborative filtering using dimensionality reduction techniques on mahout framework. In: AFRICAN JOURNALS ONLINE. **4th Post Graduate Conference for Information Technology (iPGCon-2015), Sangamner, Published in International Journal of Engineering Science and Technology (SEST)**. Lagos, Nigeria, 2015. p. 2394–0905.

BOLLACKER, K. et al. Freebase: a collaboratively created graph database for structuring human knowledge. In: ACM. **Proc ACM SIGMOD Int Conf Manag of Data**. Broadway, New York, 2008.

BORDES, A. et al. Translating embeddings for modeling multi-relational data. In: **Adv in Neural Inf Processing Syst**. Red Hook, New York: Curran Associates, Inc., 2013. p. 2787–2795. Disponível em: https://proceedings.neurips.cc/paper/2013/file/1cecc7a77928ca8133fa24680a88d2f9-Paper.pdf.

BURKE, R. Hybrid web recommender systems. In: BRUSILOVSKY, P.; KOBSA, A.; NEJDL, W. (Ed.). **The Adaptive Web: Methods and Strategies of Web Personalization**. Berlin, Heidelberg: Springer, 2007. p. 377–408. ISBN 978-3-540-72079-9. Disponível em: https://doi.org/10.1007/978-3-540-72079-9_12.

CALERO, C.; RUIZ, F.; PIATTINI, M. **Ontologies for software engineering and software technology**. Berlin, Heidelberg: Springer Science & Business Media, 2006.

CANTADOR, I.; BELLOGIN, A.; CASTELLS, P. A multilayer ontology-based hybrid recommendation model. **AI Commun.**, IOS Press, Amsterdam, The Netherlands, The Netherlands, v. 21, n. 2-3, p. 203–210, abr. 2008. ISSN 0921-7126. Disponível em: http://dl.acm.org/citation.cfm?id=1460172.1460184.

CAO, Y. et al. Unifying knowledge graph learning and recommendation: Towards a better understanding of user preferences. In: **The World Wide Web Conference**. New York, NY, USA: Association for Computing Machinery, 2019. (WWW '19), p. 151–161. ISBN 9781450366748. Disponível em: https://doi.org/10.1145/3308558.3313705.

CAZELLA, S. C.; NUNES, M. A. S.; REATEGUI, E. B. A ciência da opinião: Estado da arte em sistemas de recomendação. In: **XXX Congresso da Sociedade Brasileira de Computação**. Berlin, Heidelberg: Springer, 2010.

ČEBIRIĆ, Š. et al. Summarizing semantic graphs: a survey. **The VLDB J**, v. 28, n. 3, p. 295–327, Jun 2019. ISSN 0949-877X.

CHAN, S. et al. Predictionio: a distributed machine learning server for practical software development. In: ACM. **Proceedings of the 22nd ACM international conference on Information & Knowledge Management**. NY, USA, 2013. p. 2493–2496.

CHANDRASEKARAN, K. et al. Concept-based document recommendations for citeseer authors. In: SPRINGER. **International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems**. Berlin, Heidelberg, 2008. p. 83–92.

CHEN, L.-C.; KUO, P.-J.; LIAO, I.-E. Ontology-based Library Recommender System Using MapReduce. **Cluster Computing**, Kluwer Academic Publishers, Hingham, MA, USA, v. 18, n. 1, p. 113–121, 2015. ISSN 1386-7857. Disponível em: http://dx.doi.org/10.1007/s10586-013-0342-z.

CHEN, Y. et al. Solving the sparsity problem in recommender systems using association retrieval. **Journal of computers**, v. 6, n. 9, p. 1896–1902, 2011.

CHERNEV, A.; BöCKENHOLT, U.; GOODMAN, J. Choice overload: A conceptual review and meta-analysis. **Journal of Consumer Psychology**, v. 25, n. 2, p. 333 – 358, 2015. ISSN 1057-7408.

COLLINS, A. M.; LOFTUS, E. F. A spreading-activation theory of semantic processing. **Psychological review**, American Psychological Association, v. 82, n. 6, p. 407, 1975.

COSTA, A. D. et al. Case recommender: A flexible and extensible python framework for recommender systems. In: **Proc 12th ACM Conf Recomm Syst**. NY, USA: ACM, 2018. (RecSys '18), p. 494–495. ISBN 978-1-4503-5901-6.

COSTABELLO, L. et al. **AmpliGraph: a Library for Representation Learning on Knowledge Graphs**. 2019.

CREMONESI, P.; EPIFANIA, F.; GARZOTTO, F. User profiling vs. accuracy in recommender system user experience. In: ACM. **Proceedings of the International Working Conference on Advanced Visual Interfaces**. NY, USA, 2012. p. 717–720.

Cunningham, P.; Delany, S. J. k-Nearest Neighbour Classifiers: 2nd Edition (with Python examples). **arXiv e-prints**, p. arXiv:2004.04523, abr. 2020.

DAOUD, M. et al. Learning implicit user interests using ontology and search history for personalization. In: SPRINGER. **International Conference on Web Information Systems Engineering**. Berlin, Heidelberg, 2007. p. 325–336.

DESROSIERS, C.; KARYPIS, G. A comprehensive survey of neighborhood-based recommendation methods. In: **Recommender systems handbook**. Berlin, Heidelberg: Springer, 2011. p. 107–144.

EHRLINGER, L.; WÖSS, W. Towards a definition of knowledge graphs. In: **SEMANTiCS (Posters, Demos, SuCCESS)**. Pennsylvania, USA: Citeseer, 2016.

EYHARABIDE, V.; AMANDI, A. Ontology-based user profile learning. **Applied Intelligence**, Springer, Berlin, Heidelberg, v. 36, n. 4, p. 857–869, 2012.

FERNANDES, B. B.; SACENTI, J. A. P.; WILLRICH, R. Using implicit feedback for neighbors selection: Alleviating the sparsity problem in collaborative recommendation systems. In: **Proc 23rd Braz Symp Multimed and the Web, Webmedia 2017, Gramado, Brazil**. NY, USA: ACM, 2017. p. 341–348. ISBN 978-1-4503-5096-9.

FIORUCCI, M.; PELOSIN, F.; PELILLO, M. Separating structure from noise in large graphs using the regularity lemma. **Pattern Recognition**, v. 98, p. 107070, 2020.

FREITAS, F. L. G. de. Ontologias e a web semântica. **Jornada de Mini-Cursos em Inteligência Artificial, SBC**, v. 8, 2003.

GARCIA, M. N. M. et al. Semantic based web mining for recommender systems. In: **Advances in Intelligent and Soft Computing**. Berlin, Heidelberg: Springer-Verlag, 2010. v. 79, p. 17–25. ISSN 18675662. Disponível em: http://dx.doi.org/10.1007/978-3-642-14883-5_3.

GARCIA, S. et al. Prototype selection for nearest neighbor classification: Taxonomy and empirical study. **IEEE Trans Pattern Anal and Mach Intell**, IEEE, Manhattan, New York, v. 34, n. 3, p. 417–435, 2012.

GAUCH, S. et al. User profiles for personalized information access. **The adaptive web**, Springer, Berlin, Heidelberg, p. 54–89, 2007.

GOWAN, J. P. M. **A multiple model approach to personalised information access**. Tese (Doutorado) — Citeseer, 2003.

GUARINO, N. et al. Formal ontology and information systems. In: IOS PRESS. **Proceedings of FOIS**. Amsterdam, Netherlands, 1998. v. 98, n. 1998, p. 81–97.

GUO, G.; ZHANG, J.; YORKE-SMITH, N. Leveraging multiviews of trust and similarity to enhance clustering-based recommender systems. **Knowl-Based Syst**, v. 74, p. 14 – 27, 2015. ISSN 0950-7051.

Guo, Q. et al. A survey on knowledge graph-based recommender systems. **IEEE Trans Knowl and Data Eng**, IEEE, Manhattan, New York, p. 1–1, 2020.

HANANI, U.; SHAPIRA, B.; SHOVAL, P. Information filtering: Overview of issues, research and systems. **User modeling and user-adapted interaction**, Kluwer Academic Publishers, v. 11, n. 3, p. 203–259, 2001.

HARPER, F. M.; KONSTAN, J. A. The movielens datasets: History and context. **ACM Trans Interact Intell Syst (TIIS)**, ACM, NY, USA, v. 5, n. 4, p. 1–19, 2015.

HASSANZADEH, O.; CONSENS, M. P. Linked movie data base. In: BIZER, C. et al. (Ed.). **Proceedings of the WWW2009 Workshop on Linked Data on the Web, LDOW**. CEUR-WS.org, 2009. (CEUR Workshop Proceedings, v. 538). Disponível em: http://ceur-ws.org/Vol-538.

HEITMANN, B.; HAYES, C. Using linked data to build open, collaborative recommender systems. In: AAAI PRESS. **AAAI spring symposium: linked data meets artificial intelligence**. 2010. v. 2010. Disponível em: https://www.aaai.org/ocs/index.php/SSS/SSS10/paper/download/1067/1452.

HUSSAIN, S. F.; MUSHTAQ, M.; HALIM, Z. Multi-view document clustering via ensemble method. **J Intell Inf Syst**, Springer, Berlin, Heidelberg, v. 43, n. 1, p. 81–99, 2014.

IAQUINTA, L. et al. Introducing serendipity in a content-based recommender system. In: IEEE. **Hybrid Intelligent Systems, 2008. HIS'08. Eighth International Conference on**. Manhattan, New York, 2008. p. 168–173.

JOUDREY, D. N.; TAYLOR, A. G.; MILLER, D. P. **Introduction to cataloging and classification**. Littleton, CO: Libraries unlimited, 2015.

JURAFSKY, D.; MARTIN, J. H. **Speech and Language Processing (2nd Edition)**. USA: Prentice-Hall, Inc., 2009. ISBN 0131873210.

KADIMA, H.; MALEK, M. Toward ontology-based personalization of a recommender system in social network. In: IEEE. **Soft Computing and Pattern Recognition (SoCPaR), 2010 International Conference of**. Manhattan, New York, 2010. p. 119–122.

KANOJE, S.; GIRASE, S.; MUKHOPADHYAY, D. User profiling trends, techniques and applications. v. 1, p. 2348–4853, 11 2014.

KATIFORI, A. et al. Creating an ontology for the user profile: Method and applications. In: ROLLAND, C.; PASTOR, O.; CAVARERO, J. (Ed.). **Proceedings of the First International Conference on Research Challenges in Information Science, RCIS**. Ouarzazate, Morocco, 2007. p. 407–412.

KIM, H. H. A Semantically Enhanced Tag-based Music Recommendation Using Emotion Ontology. In: **Proceedings of the 5th Asian Conference on Intelligent Information and Database Systems - Volume Part II**. Berlin, Heidelberg: Springer-Verlag, 2013. (ACIIDS'13), p. 119–128. ISBN 978-3-642-36542-3. Disponível em: http://dx.doi.org/10.1007/978-3-642-36543-0_13.

KITCHENHAM, B. A.; CHARTERS, S. **Guidelines for performing Systematic Literature Reviews in Software Engineering**. UK, 2007. Disponível em: https://www.elsevier.com/__data/promis_misc/525444systematicreviewsguide.pdf.

KO, H.-G.; SON, J.-S.; KO, I.-Y. Multi-aspect collaborative filtering based on linked data for personalized recommendation. In: ACM. **Proceedings of the 24th International Conference on World Wide Web**. NY, USA, 2015. p. 49–50.

KUCHAŘ, J.; KLIEGR, T. Inbeat: Javascript recommender system supporting sensor input and linked data. **Knowledge-Based Systems**, Elsevier, v. 135, p. 40–43, 2017.

LAKKARAJU, P.; GAUCH, S.; SPERETTA, M. Document similarity based on concept tree distance. In: ACM. **Proceedings of the nineteenth ACM conference on Hypertext and hypermedia**. NY, USA, 2008. p. 127–132.

LEAKE, D. B.; WILSON, D. C. Categorizing case-base maintenance: Dimensions and directions. In: **Adv in Case-Based Reason**. Berlin, Heidelberg: Springer, 1998. p. 196–207. ISBN 978-3-540-49797-4.

LEHMANN, J. et al. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. **Semant web**, IOS Press, v. 6, n. 2, p. 167–195, 2015.

LIAO, I.-E. et al. A library recommender system based on a personal ontology model and collaborative filtering technique for English collections. **The Electronic Library**, Emerald Group Publishing Limited, v. 28, n. 3, p. 386–400, jun 2010. ISSN 0264-0473. Disponível em: http://www.emeraldinsight.com/doi/abs/10.1108/02640471011051972.

LIN, Y. et al. Learning entity and relation embeddings for knowledge graph completion. **29th AAAI Conf Artif Intell**, p. 2181–2187, 2015.

LIU, Q.; CHENG, G.; QU, Y. **DeepLENS: Deep Learning for Entity Summarization**. 2020.

LIU, Y. et al. Graph summarization methods and applications: A survey. **ACM Comput. Surv.**, ACM, NY, USA, v. 51, n. 3, jun. 2018. ISSN 0360-0300.

MAATEN, L. V. D.; POSTMA, E.; HERIK, J. Van den. Dimensionality reduction: a comparative review. **J Mach Learn Res**, v. 10, p. 66–71, 2009.

MANOLA, F. et al. Rdf primer. **W3C recommendation**, Citeseer, v. 10, n. 1-107, p. 6, 2004.

MANOUSELIS, N.; COSTOPOULOU, C. Experimental analysis of design choices in multiattribute utility collaborative filtering. **International Journal of Pattern Recognition and Artificial Intelligence**, World Scientific, v. 21, n. 02, p. 311–331, 2007.

MARTÍNEZ, C. N. F. **Recommender Systems based on Linked Data**. Tese (Doutorado) — Politecnico di Torino, 2017.

MASINTER, L.; BERNERS-LEE, T.; FIELDING, R. T. Uniform resource identifier (uri): Generic syntax. 2005.

MENDOZA, L. O. Colombo et al. RecomMetz: A context-aware knowledge-based mobile recommender system for movie showtimes. **Expert Systems with Applications**, v. 42, n. 3, p. 1202–1222, 2015. ISSN 0957-4174. Disponível em: http://www.sciencedirect.com/science/article/pii/S0957417414005557.

MESAS, R. M.; BELLOGÍN, A. Exploiting recommendation confidence in decision-aware recommender systems. **J Intell Inf Syst**, Springer, Berlin, Heidelberg, v. 54, n. 1, p. 45–78, 2020.

MIDDLETON, S. E. **Capturing knowledge of user preferences with recommender systems**. Tese (Doutorado) — university of Southampton, 2003.

MIDDLETON, S. E.; ROURE, D. D.; SHADBOLT, N. R. Ontology-based recommender systems. In: ____. **Handbook on Ontologies**. Berlin, Heidelberg: Springer, 2004. p. 477–498. ISBN 978-3-540-24750-0. Disponível em: https://doi.org/10.1007/978-3-540-24750-0_24.

MIDDLETON, S. E.; ROURE, D. D.; SHADBOLT, N. R. Ontology-based recommender systems. In: **Handbook on ontologies**. Berlin, Heidelberg: Springer, 2009. p. 779–796.

MIKOLOV, T. et al. **Efficient Estimation of Word Representations in Vector Space**. 2013.

MIKOLOV, T. et al. **Distributed Representations of Words and Phrases and their Compositionality**. 2013.

MIRIZZI, R. et al. Movie recommendation with dbpedia. In: AMATI, G.; CARPINETO, C.; SEMERARO, G. (Ed.). **Proceedings of the Italian Information Retrieval Workshop , IIR**. CEUR-WS.org, 2012. (CEUR Workshop Proceedings, v. 835), p. 101–112. Disponível em: http://ceur-ws.org/Vol-835/paper12.pdf.

MIZOGUCHI, R.; VANWELKENHUYSEN, J.; IKEDA, M. Task ontology for reuse of problem solving knowledge. **Towards Very Large Knowledge Bases: Knowledge Building & Knowledge Sharing**, IOS press Amsterdam, v. 46, p. 59, 1995.

MOBASHER, B.; JIN, X.; ZHOU, Y. Semantically enhanced collaborative filtering on the web. In: **Web Mining: From Web to Semantic Web**. Berlin, Heidelberg: Springer, 2004. p. 57–76.

MONTGOMERY, D. C. **Design and analysis of experiments**. New Jersey, USA: John wiley & sons, 2017.

MORENO, M. N. et al. Web mining based framework for solving usual problems in recommender systems. A case study for movies' recommendation. **Neurocomputing**, v. 176, p. 72–80, 2016. ISSN 0925-2312. Disponível em: http://www.sciencedirect.com/science/article/pii/S0925231215005457.

Owl 2 web ontology language. structural specification and functional-style syntax (second edition). 2012. Disponível em: http://www.w3.org/TR/owl2-syntax/.

MUSTO, C. et al. Automatic selection of linked open data features in graph-based recommender systems. In: BOGERS, T.; KOOLEN, M. (Ed.). **Proceedings of the 2nd Workshop on New Trends on Content-Based Recommender Systems co-located with 9th ACM Conference on Recommender Systems, CBRecSys**. CEUR-WS.org, 2015. (CEUR Workshop Proceedings, v. 1448), p. 10–13. Disponível em: http://ceur-ws.org/Vol-1448/paper3.pdf.

MUSTO, C. et al. Semantics-aware recommender systems exploiting linked open data and graph-based features. **Knowledge-Based Systems**, Elsevier, v. 136, p. 1–14, 2017.

NAKHJIRI, N.; SALAMó, M.; SàNCHEZ-MARRè, M. Reputation-based maintenance in case-based reasoning. **Know.-Based Syst.**, Elsevier Science Publishers B. V., NLD, v. 193, n. C, abr. 2020. ISSN 0950-7051.

NIE, F.; CAI, G.; LI, X. Multi-view clustering and semi-supervised classification with adaptive neighbours. In: AAAI PRESS. **Proc 35th AAAI Conf Artif Intell**. 2017. (AAAI'17), p. 2408–2414. Disponível em: https://ojs.aaai.org/index.php/AAAI/article/view/10909.

NOIA, T. D.; CANTADOR, I.; OSTUNI, V. C. Linked open data-enabled recommender systems: Eswc 2014 challenge on book recommendation. In: SPRINGER. **Semantic Web Evaluation Challenge**. Berlin, Heidelberg, 2014. p. 129–143.

NOIA, T. D. et al. Linked open data to support content-based recommender systems. In: ACM. **Proceedings of the 8th International Conference on Semantic Systems**. NY, USA, 2012. p. 1–8.

NOY, N. F.; MCGUINNESS, D. L. **Ontology Development 101: A Guide to Creating Your First Ontology**. California, USA, 2001. Disponível em: http://www.ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness-abstract.html.

ORAMAS, S. et al. Sound and music recommendation with knowledge graphs. **ACM Transactions on Intelligent Systems and Technology (TIST)**, ACM, NY, USA, v. 8, n. 2, p. 21, 2017.

OSTUNI, V. C. et al. A linked data recommender system using a neighborhood-based graph kernel. In: SPRINGER. **International Conference on Electronic Commerce and Web Technologies**. Berlin, Heidelberg, 2014. p. 89–100.

OSTUNI, V. C. et al. Top-n recommendations from implicit feedback leveraging linked open data. In: ACM. **Proceedings of the 7th ACM conference on Recommender systems**. NY, USA, 2013. p. 85–92.

OWEN, S.; OWEN, S. Mahout in action. Manning Shelter Island, NY, 2012.

PAN, P.-Y. et al. The development of an ontology-based adaptive personalized recommender system. In: IEEE. **Electronics and Information Engineering (ICEIE), 2010 International Conference On**. Manhattan, New York, 2010. v. 1, p. V1–76.

PASSANT, A. dbrec: music recommendations using dbpedia. In: SPRINGER. **International Semantic Web Conference**. Berlin, Heidelberg, 2010. p. 209–224.

PASSANT, A.; HEITMANN, B.; HAYES, C. Using linked data to build recommender systems. **RecSys '9, New-York, NY USA**, Citeseer, 2009.

PAUN, I. Efficiency-effectiveness trade-offs in recommendation systems. In: **14th ACM Conf Recomm Syst**. NY, USA: ACM, 2020. (RecSys '20), p. 770–775. ISBN 9781450375832.

PELUFFO-ORDÓÑEZ, D. H.; LEE, J. A.; VERLEYSEN, M. Recent methods for dimensionality reduction: A brief comparative analysis. In: **22th Eur Symp Artif Neural Netw, ESANN 2014, Bruges, Belgium, April 23-25, 2014**. I6DOC, 2014. Disponível em: http://www.elen.ucl.ac.be/Proceedings/esann/esannpdf/es2014-170.pdf.

PESKA, L.; VOJTAS, P. Enhancing recommender system with linked open data. In: SPRINGER. **International Conference on Flexible Query Answering Systems**. Berlin, Heidelberg, 2013. p. 483–494.

PIAO, G.; BRESLIN, J. G. Transfer learning for item recommendations and knowledge graph completion in item related domains via a co-factorization model. In: SPRINGER. **Eur Semant Web Conf**. Berlin, Heidelberg, 2018. p. 496–511.

PIETERSE, V.; KOURIE, D. Lists, taxonomies, lattices, thesauri and ontologies: Paving a pathway through a terminological jungle. v. 41, p. 217–229, 01 2014.

PRIMO, T. T.; VICARI, R. M.; BERNARDI, K. S. User profiles and learning objects as ontology individuals to allow reasoning and interoperability in recommender systems. In: IEEE. **Global Engineering Education Conference (EDUCON), 2012 IEEE**. Manhattan, New York, 2012. p. 1–9.

RAGONE, A. et al. Schema-summarization in linked-data-based feature selection for recommender systems. In: **Proc Symp Appl Comput**. NY, USA: ACM, 2017. (SAC '17), p. 330–335. ISBN 9781450344869. Disponível em: https://doi.org/10.1145/3019612.3019837.

REDDY, G. T. et al. Analysis of dimensionality reduction techniques on big data. **IEEE Access**, IEEE, Manhattan, New York, v. 8, p. 54776–54788, 2020.

RICCI, F.; ROKACH, L.; SHAPIRA, B. Introduction to recommender systems handbook. In: **Recommender systems handbook**. Berlin, Heidelberg: Springer, 2011. p. 1–35.

RICCI, F.; ROKACH, L.; SHAPIRA, B. Recommender systems: introduction and challenges. In: **Recommender systems handbook**. Berlin, Heidelberg: Springer, 2015. p. 1–34.

RODRÍGUEZ-GARCÍA, M. Á. et al. Ontology-based music recommender system. **Distributed Computing and Artificial Intelligence, 12th International Conference**, Springer, Cham, The Capital Region of Denmark, 2015. Disponível em: http://link.springer.com/chapter/10.1007/978-3-319-19638-1%7B_%7D5.

RODRíGUEZ-GARCíA, M. A. et al. Blinddate recommender: A context-aware ontology-based dating recommendation platform. **J of Inf Sci**, v. 45, n. 5, p. 573–591, 2019.

ROZEMBERCZKI, B. et al. GEMSEC: Graph Embedding with Self Clustering. In: ACM. **Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2019**. NY, USA, 2019. p. 65–72.

SACENTI, J. A. et al. Automatically tailoring semantics-enabled dimensions for movement data warehouses. In: SPRINGER. **International Conference on Big Data Analytics and Knowledge Discovery**. Berlin, Heidelberg, 2015. p. 205–216.

SACENTI, J. A. P. **Adaptação de Hierarquias de Dados Conectados para Análise de Informação**. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, Centro Tecnológico. Programa de Pós-Graduação em Ciência da Computação, 2016.

SACENTI, J. A. P.; FILETO, R.; WILLRICH, R. Knowledge graph summarization impacts on movie recommendations. **J Intell Inf Syst**, Springer, NY, USA, 2021.

SACENTI, J. A. P.; WILLRICH, R.; FILETO, R. Hybrid recommender system based on multi-hierarchical ontologies. In: **Proc 24th Braz Symp Multimed and the Web, WebMedia 2018, Salvador, Brazil**. NY, USA: ACM, 2018. p. 149–156. ISBN 978-1-4503-5867-5.

SALLES, A. **Serviço web de recomendação baseado em ontologias e grafos para repositórios digitais**. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, Centro Tecnológico. Programa de Pós-Graduação em Ciência da Computação, 2017.

SALLES, A.; WILLRICH, R. Recommending Web Service Based on Ontologies for Digital Repositories. In: **Proceedings of the 21st Brazilian Symposium on Multimedia and the Web**. NY, USA: ACM, 2015. (WebMedia '15), p. 65–72. ISBN 978-1-4503-3959-9.

SANTOS, M. P. dos. **Recomendação contextual de estabelecimentos baseado em ontologia**. Dissertação (Bacharel) — Universidade Federal de Santa Catarina, 2019.

SCHELTER, S.; OWEN, S. Collaborative filtering with apache mahout. **Proc. of ACM RecSys Challenge**, ACM, NY, USA, 2012.

SEMERARO, G. et al. Knowledge infusion into content-based recommender systems. In: ACM. **Proceedings of the third ACM conference on Recommender systems**. NY, USA, 2009. p. 301–304.

SEMINARIO, C. E.; WILSON, D. C. Case study evaluation of mahout as a recommender platform. In: AMATRIAIN, X. et al. (Ed.). **Proceedings of the Workshop on Recommendation Utility Evaluation: Beyond RMSE (RUE 2012) at the 6th ACM International Conference on Recommender Systems (RecSys 2012)**. CEUR-WS.org, 2012. (CEUR Workshop Proceedings, v. 910), p. 45–50. Disponível em: http://ceur-ws.org/Vol-910/paper10.pdf.

SENECAL, S.; NANTEL, J. The influence of online product recommendations on consumers online choices. **Journal of retailing**, Elsevier, v. 80, n. 2, p. 159–169, 2004.

SHERIDAN, P. et al. An ontology-based recommender system with an application to the star trek television franchise. **Future Internet**, v. 11, n. 9, 2019. ISSN 1999-5903. Disponível em: https://www.mdpi.com/1999-5903/11/9/182.

SHOKEEN, J.; RANA, C. Social recommender systems: techniques, domains, metrics, datasets and future scope. **J Intell Inf Syst**, p. 633–667, 2020.

SIEG, A.; MOBASHER, B.; BURKE, R. Ontological user profiles for representing context in web search. In: IEEE. **Web Intelligence and Intelligent Agent Technology Workshops, 2007 IEEE/WIC/ACM International Conferences on**. Manhattan, New York, 2007. p. 91–94.

SIEG, A.; MOBASHER, B.; BURKE, R. Improving the Effectiveness of Collaborative Recommendation with Ontology-based User Profiles. In: **Proceedings of the 1st International Workshop on Information Heterogeneity and Fusion in Recommender Systems**. New York, NY, USA: ACM, 2010. (HetRec '10), p. 39–46. ISBN 978-1-4503-0407-8. Disponível em: http://doi.acm.org/10.1145/1869446.1869452.

SILVA, T. N. **Um modelo baseado em ontologia para suporte a tarefa intensiva em conhecimento de recomendação**. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, Centro Tecnológico. Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento, 2015.

SMYTH, B. Case-base maintenance. In: **Tasks and Methods in Applied Artif Intell**. Berlin, Heidelberg: Springer, 1998. p. 507–516. ISBN 978-3-540-69350-5.

SMYTH, B.; KEANE, M. T. Remembering to forget. In: IJCAI. **Proc 14th IJCAI**. Montreal, Canada, 1995.

SOARES, M.; VIANA, P. The semantics of movie metadata: enhancing user profiling for hybrid recommendation. In: SPRINGER. **World Conference on Information Systems and Technologies**. Berlin, Heidelberg, 2017. p. 328–338.

SORZANO, C. O. S.; VARGAS, J.; MONTANO, A. P. **A survey of dimensionality reduction techniques**. 2014.

SPERETTA, M.; GAUCH, S. Personalized search based on user search histories. In: IEEE. **Web Intelligence, 2005. Proceedings. The 2005 IEEE/WIC/ACM International Conference on**. Manhattan, New York, 2005. p. 622–628.

STUDER, R.; BENJAMINS, V. R.; FENSEL, D. Knowledge engineering: principles and methods. **Data & knowledge engineering**, Elsevier, v. 25, n. 1-2, p. 161–197, 1998.

SUN, Z. et al. Recurrent knowledge graph embedding for effective recommendation. In: **Proc 12th ACM Conf Recomm Syst**. NY, USA: ACM, 2018. (RecSys '18), p. 297–305. ISBN 9781450359016.

SYDOW, M.; PIKUŁA, M.; SCHENKEL, R. The notion of diversity in graphical entity summarisation on semantic knowledge graphs. **J Intell Inf Syst**, Springer, Berlin, Heidelberg, v. 41, n. 2, p. 109–149, 2013.

TOMEO, P. et al. Exploiting linked open data in cold-start recommendations with positive-only feedback. In: ACM. **Proceedings of the 4th Spanish Conference on Information Retrieval**. NY, USA, 2016. p. 11.

TROUILLON, T. et al. Complex embeddings for simple link prediction. In: **Proc 33rd Int Conf Mach Learn - Volume 48**. JMLR.org, 2016. (ICML'16), p. 2071–2080. Disponível em: http://proceedings.mlr.press/v48/trouillon16.pdf.

VAGLIANO, I. **Content Recommendation Through Linked Data**. Tese (Doutorado) — Politecnico di Torino, 2017.

VAGLIANO, I.; MONTI, D. M.; MORISIO, M. Semrevrec: a recommender system based on user reviews and linked data. In: CEUR-WS.ORG. **Proceedings of the Poster Track of the 11th ACM Conference on Recommender Systems (RecSys)**. 2017. Disponível em: http://ceur-ws.org/Vol-1905/recsys2017_poster10.pdf.

VLACHOS, M. et al. Non-linear dimensionality reduction techniques for classification and visualization. In: **Proc 8th ACM SIGKDD Int Conf Knowl Discov and Data Min**. NY, USA: ACM, 2002. (KDD '02), p. 645–651. ISBN 158113567X.

WALUNJ, S. G.; SADAFALE, K. An online recommendation system for e-commerce based on apache mahout framework. In: ACM. **Proceedings of the 2013 annual conference on Computers and people research**. NY, USA, 2013. p. 153–158.

WANG, H. et al. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In: ACM. **Proc 27th ACM Int Conf Inf and Knowl Manag**. New York, USA, 2018. p. 417–426.

WANG, Q. et al. Knowledge graph embedding: A survey of approaches and applications. **IEEE Trans Knowl and Data Eng**, IEEE, Manhattan, New York, v. 29, n. 12, p. 2724–2743, 2017.

WANG, X. et al. KGAT: knowledge graph attention network for recommendation. In: ACM. **Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD**. New York, USA, 2019. p. 950–958.

WANG, Z. et al. Knowledge graph embedding by translating on hyperplanes. In: AAAI PRESS. **AAAI**. 2014. v. 14, p. 1112–1119. Disponível em: https://ojs.aaai.org/index.php/AAAI/article/view/8870/8729.

WARDHANA, A. T. A.; I., H. T. N. Combining foaf and music ontology for music concerts recommendation on facebook application. In: IEEE. **New Media Studies (CoNMedia), 2013 Conference on**. Manhattan, New York, 2013. p. 1–5.

WAZLAWICK, R. **Metodologia de Pesquisa para Ciência da Computação**. Elsevier Brasil, 2015. ISBN 9788535277838. Disponível em: https://books.google.com.br/books?id=BZioBQAAQBAJ.

WILCOXON, F. Individual comparisons by ranking methods. **Biom Bull**, [International Biometric Society, Wiley], v. 1, n. 6, p. 80–83, 1945. ISSN 00994987.

WILSON, D. L. Asymptotic properties of nearest neighbor rules using edited data. **IEEE Trans Syst, Man, and Cybern**, IEEE, Manhattan, New York, SMC-2, n. 3, p. 408–421, 1972.

WILSON, D. R.; MARTINEZ, T. R. Instance pruning techniques. In: **Mach Learn: Proc 14th Int Conf ICML'97**. Burlington, USA: Morgan Kaufmann Publishers, 1997. p. 404–411.

WU, J. et al. Trust-aware media recommendation in heterogeneous social networks. **World Wide Web**, v. 18, n. 1, p. 139–157, Jan 2015. ISSN 1573-1413.

XIA, W. et al. Effective collaborative filtering approaches based on missing data imputation. In: IEEE. **INC, IMS and IDC, 2009. NCM'09. Fifth International Joint Conference on**. Manhattan, New York, 2009. p. 534–537.

XUE, Z. et al. Gomes: A group-aware multi-view fusion approach towards real-world image clustering. In: IEEE. **2015 IEEE Int Conf Multimed and Expo (ICME)**. Manhattan, New York, 2015. p. 1–6.

YANG, B. et al. **Embedding Entities and Relations for Learning and Inference in Knowledge Bases**. 2015.

Yang, Y.; Wang, H. Multi-view clustering: A survey. **Big Data Min and Anal**, v. 1, n. 2, p. 83–107, June 2018. ISSN 2096-0654.

YU, H. et al. Tag recommendation method in folksonomy based on user tagging status. **J Intell Inf Syst**, Springer, Berlin, Heidelberg, v. 50, n. 3, p. 479–500, 2018.

Yu, H. et al. Web items recommendation based on multi-view clustering. In: IEEE. **2018 IEEE 42nd Annual Comput Softw and Appl Conf (COMPSAC)**. Manhattan, New York, 2018. v. 01, p. 420–425.

YU, X. et al. Recommendation in heterogeneous information networks with implicit user feedback. In: **Proc 7th ACM Conf Recomm Syst**. NY, USA: ACM, 2013. (RecSys '13), p. 347–350. ISBN 9781450324090.

ZHANG, F. et al. Collaborative knowledge base embedding for recomm systems. In: **Proc 22nd ACM SIGKDD Int Conf Knowl Discov and Data Min**. NY, USA: ACM, 2016. p. 353–362.

ZHANG, N.; TIAN, Y.; PATEL, J. M. Discovery-driven graph summarization. In: IEEE. **2010 IEEE 26th Int Conf Data Eng (ICDE 2010)**. Manhattan, New York, 2010. p. 880–891.

ZHANG, Y. et al. Learning over knowledge-base embeddings for recommendation. **In SIGIR**, 2018.

ZHAO, Y.; SHEN, B. Empirical study of user preferences based on rating data of movies. **PloS one**, Public Library of Science, v. 11, n. 1, p. e0146541, 2016.

ZHENG, X. et al. A tourism destination recommender system using users' sentiment and temporal dynamics. **J Intell Inf Syst**, Springer, Berlin, Heidelberg, v. 51, n. 3, p. 557–578, 2018.

ZIEGLER, C.-N.; LAUSEN, G.; SCHMIDT-THIEME, L. Taxonomy-driven computation of product recommendations. In: ACM. **Proceedings of the thirteenth ACM international conference on Information and knowledge management**. NY, USA, 2004. p. 406–415.

GLOSSÁRIO

A | C | D | E | F | G | H | I | M | N | O | P | R | S | U | V

A

aresta componente de um grafo que interliga seus nodos. 148

C

conceito conceito, categoria ou classe de uma ontologia. 148

D

dado conectado um dos principais objetivos destes dados conectados (DC) é extender a *Web* tradicional, onde documentos (páginas e sítios) são interconectados por hiperligações, para uma *Web* com dados que possam estar diretamente conectados. 28, 29, 148

dado estruturado são aqueles que possuem estruturas bem definidas, rígidas, pensadas antes da própria existência do dado que será carregado naquela estrutura (p.ex., dados armazenados em banco de dados relacional ou em planilhas) . 148

declaração RDF tripla sujeito-predicado-objeto que compõe uma ontologia ou um grafo de conhecimento. 148

E

eficiência uma medida para identificar se a quantidade correta de recursos foi usada para entrega de um processo, serviço ou atividade. 148

eficácia uma medida para identificar se os objetivos de um processo, serviço ou atividade foram atingidos. 50, 148

entidade conceito ou indivíduo de uma ontologia. 148

esparsidade em SRs, se refere a falta de dados usados para prever as preferências dos usuários, do tipo: usuários com poucas ou nenhuma interação (e.g. ratings), ou itens tiveram poucas ou nenhuma interação. Em GCs, se refere a falta de triplas descrevendo e interligando entidades. 27, 49, 53, 148

F

fator de interesse aspecto ou característica de itens a recomendar que é de interesse de usuários do sistema alvo da recomendação (p.ex., no domínio defilme: gêneros, atores, diretores, quando foi lançado). 148

G

Grafo de Conhecimento (GC) é um grafo rotulado cujos nós denotam entidades de diferentes tipos (p.e., Usuário, Filme, Gênero, Diretor, Ator) e as arestas denotam relações entre eles (p.e., tipo, avalia, assiste, temGênero, temDiretor, temAtor). 28, 56, 107, 148

H

hierarquia de entidade hierarquia que organiza entidades de uma ontologia por meio de relações de ordenamento parcial. 148

hierarquia de fator de interesse hierarquia que organiza entidades relacionadas a um fator de interesse por meio de relações de ordenamento parcial. 148

I

indivíduo indivíduo ou instância de uma ontologia. 148

informação lateral no contexto de SRs, seriam informações adicionais ou secundárias, não geradas pelo próprio sistema (e.g., sistema de filmes sob demanda, de compras), sobre os itens e usuários. Essas informações contém características sobre usuários e itens, bem como novas relações entre eles. 28, 107, 148

interação usuário-item interações dos usuários com os itens do sistema alvo da recomendação. 148

item item do sistema alvo da recomendação. 148

M

modelo de preferência contém estimativas dos interesses dos usuários e apoiam a tarefa de recomendação (p.ex., PU, PUO e *embeddings* de GC. 27, 148, 151

multi-visão abordagem de SG que aplica a sumarização separadamente em cada visão. 148

N

nodo componente do grafo que é conectado a outros nodos por arestas. 148

O

objeto objeto de uma declaração RDF. 148

ontologia uma especificação explícita e formal de uma conceitualização compartilhada. 28, 56, 148

P

Perfil de Usuário (PU) é uma construção estruturada, ou um modelo, contendo dados (ou informações) diretamente ou indiretamente relacionadas às preferências do usuário, ao seu comportamento e ao seu contexto. 27, 148

Perfil Ontológico de Usuário (POU) PU descrito por meio de uma ontologia . 30, 35, 57, 148

predicado predicado de uma declaração RDF. 148

R

restrição restrição de uma ontologia. 148

S

Sistema de Recomendação (SR) sistema que apoiam-se em modelos de preferência para ajudar os usuários a encontrar e escolher recursos, digitais ou não (p.ex., itens de consumo, outros usuários, lugares e anotações). Sistemas que aplicam técnicas e ferramentas de software para fornecer recomendações de itens a um usuário que sejam na maior parte de interesse pessoal deste usuário.. 27, 148

SR baseado em Conhecimento (SRC) SR que se apoiam no conhecimento explícito do domínio dos itens e dos usuários paradeterminar como os itens satisfazem as necessidades do usuário . 27, 148

SR baseado em GC (SRGC) SR que explora o GC como um modelo consistido de usuários, itens, interações e informação lateral. 60, 107, 148

SR baseado em Ontologia (SRO) SR que incorporam informação lateral por meio de ontologias . 57, 148

sujeito sujeito de uma declaração RDF. 148

Sumarização de Grafo (SG) são algoritmos que podem transformar grafos em representações mais compactas, preservando propriedades que são úteis para a aplicação ou domínio. 31, 107, 148

sumário resumo de um conjunto de dados (p.ex., grafo de conhecimento). 148

supernodo aglutinação de nodos de um grafo. 148

U

única-visão abordagem de SG que aplica a sumarização a todo o GC (i.e., considerando-o uma única visão) . 148

usuário usuário do sistema alvo da recomendação. 148

V