Universidade Federal do Paraná

Marina Assako Hoshiba Pimentel

Busca e Ranqueamento de Recursos Educacionais com suporte de Agrupamento de *Tags*

Curitiba PR
2017

Marina Assako Hoshiba Pimentel

Busca e Ranqueamento de Recursos Educacionais com suporte de Agrupamento de *Tags*

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Informática no Programa de Pós-Graduação em Informática, setor de Ciências Exatas, da Universidade Federal do Paraná.
Área de concentração: *Ciência da Computação*.

Orientador: Marcos Didonet Del Fabro.

Curitiba PR
2017

P644b Pimentel, Marina Assako Hoshiba
Busca e ranqueamento de recursos educacionais com suporte de agrupamento de Tags / Marina Assako Hoshiba Pimentel. – Curitiba, 2017.
73 f. : il. color. ; 30 cm.

Dissertação - Universidade Federal do Paraná, Setor de Ciências Exatas, Programa de Pós-Graduação em Informática, 2017.

Orientador: Marcos Didonet Del Fabro.

1. Recursos educacionais. 2. Agrupamento de Tags. 3. Folksonomia. I. Universidade Federal do Paraná. II. Didonet Del Fabro, Marcos. III. Título.

CDD: 004.68

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO PARANÁ
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
Setor CIÊNCIAS EXATAS
Programa de Pós-Graduação INFORMÁTICA

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em INFORMÁTICA da Universidade Federal do Paraná foram convocados para realizar a arguição da dissertação de Mestrado de **MARINA ASSAKO HOSHIBA PIMENTEL** intitulada: **Busca e Ranqueamento de Recursos Educacionais com suporte de Agrupamento de Tags**, após terem inquirido a aluna e realizado a avaliação do trabalho, são de parecer pela sua aprovação no rito de defesa.

A outorga do título de mestre está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

Curitiba, 14 de Agosto de 2017.

MARCOS DIDONET DEL FABRO
Presidente da Banca Examinadora (UFPR)

CLODIS BOSCARIOLI
Avaliador Externo (UNIOESTE)

ROBERTO PEREIRA
Avaliador Interno (UFPR)

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO PARANÁ
Setor de Ciências Exatas
Coordenação de Pós-Graduação em Informática

Aos meus queridos pais Paulo e Clotilde pelo exemplo de vida.

Agradecimentos

A Deus por ter permitido a realização de mais uma grande conquista.

Ao meu orientador Professor Dr. Marcos Didonet Del Fabro, pelo tempo e atenção dedicados e, sobretudo, por compartilhar seu profundo conhecimento.

Ao meu amado Mozart, meus queridos filhos Caio, Shinji e Akemi pelo apoio, pela paciência e incentivo durante essa jornada.

Aos meus pais e irmãos que incondicionalmente apoiam, incentivam e acreditam em minhas decisões.

À toda a equipe C3SL, especialmente aos colegas do projeto Portalmec, com quem compartilho essa conquista.

Enfim, agradeço a todos àqueles que, desde o início, confiaram no meu esforço, e contribuíram para a realização deste trabalho.

Resumo

A busca e recuperação de recursos educacionais em repositórios digitais tem sido uma tarefa árdua, principalmente devido às implementações dos algoritmos de busca baseados em busca sintática. Apesar dos sistemas de busca serem bastante utilizados, pesquisas relatam que é desafiador para os professores a busca e seleção dos recursos disponíveis nos repositórios digitais, pois muitos conteúdos irrelevantes são retornados. Caso não seja aplicado um método de classificação adequado, o usuário terá dificuldades para encontrar resultados consistentes e relevantes para a sua busca. O presente trabalho tem como objetivo propor um modelo de busca e ranqueamento de recursos educacionais em repositórios digitais com suporte do agrupamento de *tags*. A formação dos agrupamentos baseia-se nas medidas de coocorrências entre *tags*. Com o suporte destes agrupamentos é possível realizar a busca de recursos educacionais por meio das *tags* correlacionadas ao termo original de busca. O peso dos recursos encontrados é dado pelo somatório dos pesos das respectivas *tags* do agrupamento atribuídas aos recursos. Este conjunto de resultados é somado ao conjunto de resultados encontrados via motor de busca. Os recursos educacionais dos dois conjuntos passam por uma espécie de normalização de pesos para possibilitar a união dos resultados e um novo ranqueamento é calculado, reclassificando os recursos educacionais, impulsionando e destacando os resultados considerados relevantes em relação ao termo original de busca bem como às *tags* correlacionadas. O modelo proposto foi instanciado utilizando-se a infraestrutura de um portal existente, mostrando assim a sua viabilidade. Pela expansão dos termos de busca com as *tags* correlacionadas, o modelo proposto encontra resultados distintos que antes não eram encontrados apenas pelo uso de um motor de busca, ampliando e diversificando os resultados. A avaliação dos resultados dos experimentos é feita de forma empírica.

Palavras-chave: recursos educacionais, agrupamento de tags, busca, ranqueamento, folksonomia.

Abstract

The search and retrieval of educational resources in digital repositories has been an arduous task, mainly due to the implementation of the search algorithms based on syntactic search. Despite search engines are widely used, researches show how challenging it is for teachers to search and retrieve resources available in digital repositories, where many irrelevant content are returned. If an appropriate classification method is not applied, it will be difficult to the users to find consistent and relevant results for their search. The present work aims to propose a model for educational resources searching process in digital repositories supported by tag clustering. The clusters are calculated based on tags co-occurrences measure. With the support of these tag clustering structure, it is possible to search educational resources related to the original search term as well as with correlated tags. The ranking weight of the found resources are given by the sum of the respective correlated tag weights assigned to the resources. Another result set is given by the search engine. The educational resources of the two groups undergo a kind of weight normalization to enable the union of the results. A new ranking is calculated, boosting and highlighting the results considered relevant in relation to the original search term as well as to the correlated tags. The proposed model was instantiated using the infrastructure of an existing portal, showing this way its viability. The expansion of the search terms with the correlated tags enables distinct results to be found through the proposed model, that previously were not found only by the search engine. The evaluation of all experiments is done empirically.

Keywords: educational resources, tag clustering, seaching, ranking, folksonomy.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Acrônimos

API	*Application Programming Interface*
GN	Girvan e Newman
MEV	Modelo de Espaço Vetorial
RE	Recursos Educacionais
RI	Recuperação da Informação
REST	*Representational State Transfer*
TF-IDF	*Term Frequency-Inverted Document Frequency*

Capítulo 1

Introdução

Vivemos numa era de abundância de informações, disponíveis principalmente por meio da internet. Os repositórios digitais com seu conjunto de documentos organizados e disponibilizados eletronicamente também fazem parte dessa fonte de informações Lagoze et al. (2006). Porém, o grande volume traz implicações no processo de organização, representação e gerenciamento de toda essa variedade de conteúdos. O formato e a quantidade de informações sobre esses conteúdos acabam impactando diretamente na recuperação dos mesmos, não sendo tarefa trivial categorizá-los de forma adequada para possibilitar a recuperação de informação relevante para quem a busca nesse ambiente digital Aguiar et al. (2014).

Na esfera da Educação, a tarefa de buscar e selecionar recursos educacionais (RE) relevantes em repositórios digitais tem sido tarefa desgastante e árdua para os professores. Grandes repositórios podem conter dezenas de milhares de objetos de aprendizagem diferentes, tornando difícil a tarefa de encontrar objetos de interesse dos Santos et al. (2015). Estudos como os de Silverstein et al. (1999); de Souza et al. (2008); Costa et al. (2013); Coelho (2009), mostram que os serviços de busca implementados nesses repositórios ainda estão longe de atender as necessidades do usuário, pois tem limitações que fazem com que poucos resultados significativos sejam retornados. Entre as limitações podemos destacar problemas como a busca simplesmente sintática e buscas baseadas somente na análise dos metadados[1] dos RE. Além disso, se o resultado da busca não é bem ranqueado o problema se agrava ainda mais, pois segundo as pesquisas, os usuários não costumam analisar mais do que os dez primeiros resultados obtidos.

O presente trabalho está inserido no contexto do desenvolvimento de um portal (Portalmec[2]) com elementos de redes sociais, especializado no compartilhamento de RE. O público alvo são professores, educadores, profissionais da educação e a população brasileira de modo geral. Este trabalho explora abordagens que possam melhorar o processo de busca e ranqueamento de recursos educacionais, bem como apresentar a sua implementação e experimentos realizados para testar a viabilidade e eficiência do modelo proposto.

1.1 Motivação

Após análise de mais de um bilhão de consultas, Silverstein et al. (1999) constataram que (i) os usuários geralmente realizam buscas curtas, com poucos termos; (ii) os usuários normalmente não modificam suas buscas e principalmente (iii) os usuários geralmente não consideram mais do que os 10 primeiros resultados. Nesta análise fica evidente que uma

[1] Metadados são dados sobre dados Baca (2008)

[2] https://portalmec.c3sl.ufpr.br/#/home

ampliação de termos correlacionados é bem vinda, visto que o usuário fornece pouquíssimos termos e além disso não pretende mudar sua consulta inicial, e uma classificação eficiente dos recursos mais relevantes é crucial para que o usuário receba resultados consistentes na busca por recursos digitais.

A recuperação de RE em repositórios digitais consiste geralmente em uma tarefa árdua, principalmente devido às implementações dos algoritmos de busca baseados somente em metadados ou palavras-chave, que são comuns nestes repositórios. Estas técnicas limitam ainda mais o processo de busca sintática de Souza et al. (2008). Em Costa et al. (2013) também pode-se constatar como é desafiador para os professores a busca e seleção dos diversos RE disponíveis nos repositórios digitais. O estudo mostra que apesar de os sistemas de busca serem bastante utilizados, conteúdos irrelevantes são retornados para os professores.

Nos estudos realizados por Coelho (2009); Coelho et al. (2012), foi possível verificar que as máquinas de busca e os repositórios digitais existentes apresentam dificuldades para a recuperação de RE. Dentre as dificuldades podem-se citar longas listas de resultados, poucos resultados relevantes e muitas vezes mal ranqueados. O estudo exploratório reforça a necessidade de criação de um mecanismo apropriado para recuperação de RE que se valha de outros recursos para facilitar a pesquisa, como o uso das *tags*. Vale aqui ressaltar que o agrupamento de *tags* tem sido explorado para melhorar os serviços de busca, navegação e recomendação utilizados na internet Gemmell et al. (2008); Shepitsen et al. (2008); Rafailidis e Daras (2013); Liu e Niu (2014).

Para suplantar o gargalo de aquisição de conhecimento que era considerado um sério problema para os sistemas baseados em conhecimento Hotho et al. (2006a), Thomas Vander Wal introduziu o termo folksonomia, que é o resultado da atribuição livre e pessoal de etiquetas (*tags*) aos objetos Peters (2009), técnica cada vez mais comum nos sistemas de compartilhamento social de recursos. Esses sistemas tem-se ampliado nos últimos anos e seu sucesso baseia-se no fato de que quase nenhum conhecimento específico é necessário para que o usuário participe. Soma-se ainda o fato de que seus usuários podem usufruir de benefícios imediatos sem muita sobrecarga. Tais sistemas permitem que os próprios usuários enviem seus recursos, atribuindo-lhes palavras-chave arbitrárias, mais conhecidas como *tags* ou anotações.

Pelos motivos expostos nesta seção, conclui-se que o processo de busca por RE ainda é uma área onde melhorias podem ser exploradas. Alguns dos principais portais de objetos educacionais no Brasil são: Banco Internacional de Objetos Educacionais[3], Portal do Professor[4], TV Escola[5], Domínio Público[6], Escola digital[7]. Dentre esses diversos portais voltados ao compartilhamento de RE disponíveis no Brasil, boa parte possui limitações nos seus serviços de busca e recuperação de RE. A motivação deste trabalho é buscar soluções para atacar os principais problemas citados e propor um modelo de busca de RE em repositórios de RE.

1.2 Objetivos

O objetivo principal deste trabalho é propor um modelo de busca de RE em repositórios digitais que combata a restrição da busca sintática somente pelo termo de busca, bem como aplicar um bom ranqueamento para destacar recursos relevantes à busca realizada. Para isso

[3]http://objetoseducacionais2.mec.gov.br
[4]http://portaldoprofessor.mec.gov.br/index.html
[5]http://tvescola.mec.gov.br/tve/home
[6]http://www.dominiopublico.gov.br/pesquisa/PesquisaObraForm.jsp
[7]http://escoladigital.org.br

propõe-se um modelo de busca que combina a utilização de um motor de busca tradicional[8] com um processo de busca baseado em agrupamento de *tags*. A ideia é complementar RE encontrados por meio de agrupamento de *tags* aos RE encontrados via motor de busca, gerando como resultado final um conjunto de RE mais amplo, com mais RE relevantes e melhor ranqueados.

No modelo idealizado, o uso das *tags* terá papel fundamental para enriquecer os resultados retornados no processo de busca por RE. Para tirar o maior proveito das informações contidas no universo de *tags* também utilizaremos a técnica de agrupamento (*clustering*). Agrupamento é a técnica de reunir um conjunto de objetos em subconjuntos ou *clusters* que sejam coerentes internamente Manning et al. (2008). Para que seja possível a formação desses agrupamentos de *tags*, é necessário identificar uma medida de similaridade entre as *tags*, pois é por esta medida que se pode agrupar elementos similares. Com o uso das *tags* e dos agrupamentos pretende-se:

- minimizar o problema da busca sintática ou da busca limitada a palavras-chave, por meio da ampliação semântica que será feita por meio das *tags* similares ao termo de busca, recuperadas do agrupamento de *tags*,
- enriquecer os resultados da busca com RE encontrados a partir dos termos similares, combatendo assim o problema de obter poucos resultados relevantes, e
- melhorar o ranqueamento de RE, de forma que será possível dar relevância e destacar não somente os RE relacionados com os termos originais, mas também com as *tags* similares.

Para atingir os objetivos desta pesquisa as principais etapas devem ser vencidas:

1. identificação e aplicação de um fator de similaridade entre *tags* para permitir o cálculo da medida de similaridade entre elas.
2. análise e uso de uma ferramenta que realize o agrupamento de *tags* baseado em suas medidas de similaridade.
3. proposta de um processo de busca de RE baseado em agrupamento de *tags*.
4. proposta de uma nova forma de ranqueamento de RE, recalculando a relevância dos RE para poder classificar o conjunto de resultados obtidos via motor de busca e via agrupamento de *tags*.

1.3 Contribuição

Dado o número reduzido de portais brasileiros que permitem o acesso e compartilhamento de RE e constatadas as dificuldades para se encontrar RE relevantes pelos serviços de busca disponíveis nesses poucos portais de Souza et al. (2008); Coelho (2009); Costa et al. (2013), o presente trabalho ao agregar as informações das *tags* correlacionadas amplia o conjunto de resultados relacionados ao termo buscado, auxiliando os usuários a encontrarem RE relevantes.

Conforme constatado na pesquisa de Silverstein et al. (1999), os usuários realizam suas buscas a partir de poucos termos de entrada. Por isso consideramos pertinente realizar a ampliação dos termos de busca com os termos correlacionados, sem a intervenção do usuário. Essa medida aumenta a quantidade de material relevante ao assunto que o usuário esteja buscando. Além disso, pelo fato de outros termos similares e correlacionados também serem levados

[8]Neste trabalho foi utilizado Elasticsearch, um motor de busca de código aberto.

em consideração na busca, ameniza-se o problema da busca sintática, permitindo recuperar resultados que na maioria das vezes diferem e que não seriam retornados pelos motores de busca tradicionais.

O fato de considerar *tags* similares para o processo de busca de RE amplia os resultados obtidos, enriquecendo o conjunto total que é formado juntamente com os resultados obtidos pelo motor de busca. A probabilidade de aumentar a quantidade de resultados relevantes retornados ao usuário são desta forma ampliadas. Ampliar o conjunto total sem uma boa classificação não é suficiente para satisfazer a necessidade do usuário que busca por recursos que tenham relevância ao tema pesquisado. Portanto, é crucial a realização de um bom ranqueamento, de forma a apresentar nas primeiras posições os RE que tem a melhor pontuação perante o conjunto de termos utilizados para a busca.

O ranqueamento idealizado neste trabalho também se beneficia com o uso das *tags* similares, pois é possível aumentar o número de RE que merecem ser destacados. Visto que as *tags* foram agrupadas por terem sido consideradas similares, logo, os RE encontrados por meio delas possivelmente são relevantes ao tema que o usuário busca. Por isso, o ranqueamento destaca não somente os RE encontrados por meio do termo original de busca (que naturalmente já é feito pelo motor de busca), mas também dá impulsão e destaque aos RE relevantes encontrados pelo agrupamento de *tags*. A ideia desse novo ranqueamento é diminuir ainda mais a relevância dos RE recuperados pelo motor de busca que já foram considerados de pouca relevância em relação aos termos da busca.

1.4 Organização do documento

Este trabalho está estruturado da seguinte forma: no Capítulo 2 é apresentado o referencial teórico necessário para o desenvolvimento da pesquisa; no Capítulo 3 citam-se trabalhos relacionados; no Capítulo 4 descreve-se o modelo proposto para o processo de busca por RE por meio do agrupamento de *tags*, bem como a abordagem de ranqueamento que mescla resultados do motor de busca e do agrupamento de *tags*; no Capítulo 5 são detalhados os experimentos, bem como a análise dos resultados e, finalmente, o Capítulo 6, traz a conclusão e perspectivas deste trabalho.

Capítulo 2

Fundamentos teóricos

Este capítulo visa estabelecer a fundamentação teórica necessária ao acompanhamento deste trabalho, detalhando os principais termos e conceitos utilizados, visando o entendimento do processo de busca e ranqueamento de recursos em repositórios digitais.

2.1 Etiquetagem e folksonomia

A etiquetagem no mundo real é largamente utilizada para rotular, localizar produtos e serviços. Esta mesma ideia foi levada para o mundo virtual por meio da Web 2.0. A etiquetagem (do termo inglês, *tagging*) é uma forma de indexação, em que os próprios usuários da informação classificam os documentos pela atribuição de termos descritivos, também chamados *tags*, palavras-chave ou etiquetas. A organização de conteúdo por meio da etiquetagem permite futura navegação, filtragem ou busca através das etiquetas Ribeiro et al. (2013).

A promessa da etiquetagem colaborativa é que, ao explorar o universo das *tags*, pode-se descobrir informações úteis que não seriam encontradas com os motores de busca tradicionais. Existem duas maneiras de explorar o universo das *tags*: no processo de busca e refinamento; ou usando algum tipo de visualização deste universo como a nuvem de *tags* Begelman et al. (2006).

O processo de etiquetagem deu origem ao termo folksonomia, que resulta da junção das palavras *folk* (= povo; é definida por pessoas e para pessoas) + *taxonomy* (= taxonomia: termo de origem grega que significa "a ciência de classificar as coisas"), neologismo criado em 2004 por Thomas Vander Wal Peters (2009).

Folksonomia é portanto, o resultado da atribuição livre e pessoal de etiquetas a informações ou objetos num ambiente social compartilhado, visando a sua posterior recuperação Isotani et al. (2009).

A folksonomia pode ser representada como uma tupla $F = (U, T, R, C)$, onde: U, T e R são conjuntos finitos, cujos elementos são os **usuários** (pessoas que associam as *tags* aos recursos), ***tags*** (etiquetas usadas pelas pessoas para anotar os recursos) e **recursos** (documentos que recebem a atribuição das *tags*) respectivamente, e C é a **relação ternária** entre esses elementos, ou seja, $C \subseteq U \times T \times R$, cujos elementos são chamados atribuição de *tags* Hotho et al. (2006a). Esses conceitos são ilustrados na Tabela 2.1 que representa uma pequena amostra das informações contidas em uma folksonomia F.

Tabela 2.1: Representação do conjunto de informações de uma Folksonomia

Usuário (U)	**Tag (T)**	**Recurso (R)**	**Relação ternária (C)**
João	energia elétrica, usina elétrica	Usina de Itaipú.	{(João, Usina de Itaipú, energia elétrica), (João, Usina de Itaipú, usina elétrica)}
Maria	colônia, escravos	Escravidão no Brasil.	{(Maria, Escravidão no Brasil, colônia), (Maria, Escravidão do Brasil, escravos)}

A folksonomia acaba gerando um tipo de classificação social ou categorização colaborativa dos recursos, pois permite uma categorização livre em linguagem natural, não sendo adotadas regras ou políticas de indexação e nem o controle de vocabulários Catarino e Baptista (2007). Deste modo, pode-se dizer que qualquer usuário poderia contribuir, sem treinamento prévio, para a formação de uma folksonomia. Este processo torna-se muito útil quando não há ou há um número insuficiente de profissionais bibliotecários, ou especialistas no assunto, para realizar a classificação de grandes quantidades de documentos, o que é comum na Web ou em repositórios digitais. Desta forma, a etiquetagem colaborativa tem ganhado popularidade na Web, utilizando-se da folksonomia para a atribuição de termos para representação do conhecimento Golder e Huberman (2006).

2.2 Recuperação da informação

Cunhada em 1951 por Calvin Mooers, a recuperação da informação trata dos aspectos intelectuais da descrição da informação e sua especificação para busca, e também de qualquer sistema, técnicas ou máquinas usados para realizar esta operação Mooers (1951).

Recuperar informação é encontrar material (normalmente documentos) de natureza não estruturada (normalmente texto) dentro de uma ampla coleção (normalmente armazenada em computadores) e que satisfaça a necessidade da informação Manning et al. (2008). Na visão computacional, o problema consiste principalmente na construção de índices eficientes, processamento de buscas com alto desempenho, desenvolvimento de algoritmos que criem classificações e que recupere o melhor conjunto de resposta para a busca Baeza-Yates e Ribeiro-Neto (1999).

Portanto, um sistema de RI interroga, por meio de uma consulta (*query*) e recebe como resposta um conjunto de documentos classificados de acordo com algum relacionamento, chamado relevância, entre os documentos e a consulta Goffman (1964).

2.2.1 Cálculo da relevância

Dada uma busca, entende-se como relevância a melhor resposta encontrada com base na sua distribuição de probabilidade Goffman (1964). Em repositórios com amplo número de documentos, o resultado de uma busca pode retornar uma quantidade de documentos que pode facilmente exceder a capacidade humana de filtrá-los, sendo essencial que um motor de busca classifique e ordene os documentos pelas suas pontuações.

Buscas que permitem a digitação de texto livre, sem usar nenhum tipo de operador (como os booleanos), são populares na web e tratam a consulta como um conjunto de palavras.

Por isso um mecanismo de pontuação aceitável seria calcular a pontuação como sendo o somatório dos pesos dos termos que coincidam com os termos da busca Manning et al. (2008).

Um dos algoritmos mais conhecidos para cálculo de relevância (*ranking*) é o PageRank Brin e Page (1998), adotado pelo motor de busca Google[1] que calcula a importância de uma

[1]https://www.google.com

página baseado na quantidade e na qualidade dos *links* que apontam para ela. Assumindo que a página Pa tem as páginas $P1...Pn$ que apontam para ela; e que o parâmetro d é um fator de amortecimento que pode ser ajustado entre 0 e 1, normalmente estabelecido para 0,85; e que $C(Pa)$ é definido como o número de links que saem da página Pa, o PageRank de uma página Pa é dado pela Equação (2.1):

$$PR(Pa) = (1 - d) + d \times ((\frac{PR(P_1)}{C(P_1)}) + ... + (\frac{PR(T_n)}{C(T_n)})) \tag{2.1}$$

A medida de PageRank forma uma distribuição de probabilidade em páginas Web, então a soma de todos os PageRanks das páginas da Web será igual a um (1). O fator de ajustamento d pode ser simplificado como a probabilidade do usuário não clicar nos links que o documento Pa aponta e sim resolver iniciar a navegação em uma outra página qualquer, de forma aleatória.

A Figura 2.1 mostra um exemplo ilustrativo da distribuição dos pesos nas páginas web calculado por PageRank. O nó B tem um valor de PageRank mais elevado do que o nó C, apesar de ter bem menos ligações do que o nó C. Isso deve-se ao fato do nó B receber ligação vinda de um nó importante que é o nó A.

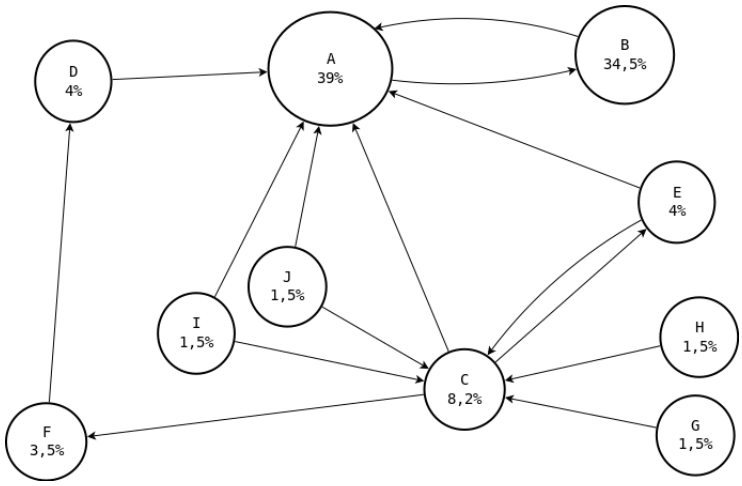

Figura 2.1: Exemplo ilustrado do algoritmo PageRank

Outro esquema de atribuição de peso é conhecido como **frequência de termos** e denota-se como $TF_{t,d}$, sendo os índices o termo e o documento nesta ordem. Para cada termo no documento é atribuído um peso, que depende do número de ocorrências do termo no documento. A ideia é calcular a pontuação relacionando o termo de pesquisa t e um documento d, baseado no peso de t em d. A abordagem mais simples é atribuir ao peso o número de ocorrências do termo t no documento d Manning et al. (2008). Quanto mais frequente, mais relevante.

TF como definido, apresenta um problema crítico: todos os termos são considerados igualmente importantes. De fato, alguns termos tem pouco ou nenhum poder discriminatório que possa determinar relevância. Para atenuar esse problema, adota-se a **frequência de documentos** (DF) e denota-se como DF_t, definido como o número de documentos na coleção que contém o termo t. Para graduar o peso do termo usando a medida dada por DF, define-se o **inverso da frequência nos documentos** (IDF), dado pela Equação (2.2), sendo N o número de documentos da coleção. O IDF de um termo raro é alto, enquanto o IDF de um termo frequente provavelmente será baixo Manning et al. (2008).

$$IDF_t = \log \frac{N}{DF_t}. \tag{2.2}$$

Combinando freqüência de termo e inverso da frequência nos documentos pode-se produzir um peso composto para cada termo em cada documento. O esquema de pontuação conhecido como **TF-IDF** atribui para o termo t um peso no documento d dado por

$$TF{-}IDF_{t,d} = TF_{t,d} \times IDF_t. \tag{2.3}$$

O peso atribuído por $TF{-}IDF$ para o termo t no documento d tem valor:

1. alto, quando t ocorre muitas vezes em um pequeno número de documentos (dando-lhes alto poder discriminatório);

2. baixo, quando o termo ocorre poucas vezes em um documento, ou ocorre em muitos documentos (indicando baixa relevância);

3. baixo, quando o termo ocorre praticamente em todos os documentos

A Tabela 2.2 representa um exemplo de cálculo do TF-IDF para quatro termos (*carro, automóvel, seguro, melhor*) em três documentos (*d1, d2 e d3*) numa coleção composta de 806.791 documentos.

A coluna **DF** denota o número de documentos na coleção em que cada termo ocorre. Desta forma pode-se calcular o inverso da frequência nos documentos (Equação (2.2)) representado na coluna **IDF**. A frequência dos termos em cada documento é representado nas colunas **TF**. Pode-se calcular então o peso dado por TF-IDF (Equação (2.3)) para cada termo em cada um dos documentos como mostra a tabela. Por exemplo, o termo *carro* tem um peso igual a 44,55 para o documento $d1$; 6,6 para o documento $d2$ e 39,6 para o documento $d3$.

Tabela 2.2: Exemplo de cálculo do TF-IDF

			TF			**TF-IDF**		
Termo	**DF**	**IDF**	**d1**	**d2**	**d3**	**d1**	**d2**	**d3**
carro	18.165	1,65	27	4	24	44,55	6,6	39,6
automóvel	6.723	2,08	3	33	0	6,24	68,64	0
seguro	19.241	1,62	0	33	29	0	53,46	46,98
melhor	25.235	1,5	14	0	17	21	0	25,5

O **Modelo de Espaço Vetorial** (MEV) (do inglês *Vector Space Model*) Salton et al. (1975) é um modelo algébrico que representa documentos texto como vetores de identificadores. Considerando um espaço de documentos constituído por documentos D_i, cada um identificado por seus termos t_j indexados, com seus respectivos pesos de acordo com sua relevância. Cada documento D_i é representado por um vetor t-dimensional $D_i = (d_{i1}, d_{i2}, ..., d_{it})$, d_{ij} representando o peso do j-ésimo termo, que pode ser calculado utilizando-se o esquema de pontuação **TF-IDF**. Dado o vetor de índices para dois documentos, é possível computar o coeficiente de similaridade entre ambos, o que reflete o grau de similaridade dos termos e pesos correspondentes.

A similaridade entre dois documentos no espaço vetorial pode ser calculada utilizando-se a **similaridade cosseno** Manning et al. (2008). Dados dois documentos $d1$ e $d2$, a similaridade cosseno dos seus respectivos vetores $\vec{V}(d_1)$ e $\vec{V}(d_2)$ é calculada como:

$$sim(d1, d2) = \frac{V(\vec{d}1).V(\vec{d}2)}{\left|V(\vec{d}1)\right|\left|V(\vec{d}2)\right|}. \tag{2.4}$$

O numerador representa o produto interno dos vetores $V(\vec{d}1)$ e $V(\vec{d}2)$, enquanto o denominador é o produto de seus comprimentos euclidianos. O produto interno $\vec{x}.\vec{y}$ de dois vetores é definido como $\sum_{i=1}^{M} x_i y_i$. Sendo $\vec{V}(d)$ o vetor do documento d, com M componentes $\vec{V}_1(d) \ldots \vec{V}_M(d)$. O comprimento euclidiano é calculado por $\sqrt{\sum_{i=1}^{M} \vec{V}_i^2(d)}$. A normalização do comprimento dos vetores $\vec{V}(d1)$ e $\vec{V}(d2)$ para vetores unitários é dado por $\vec{v}(d1) = \vec{V}(d1)/\left|\vec{V}(d1)\right|$ e $\vec{v}(d2) = \vec{V}(d2)/\left|\vec{V}(d2)\right|$.

Considerando-se os dados da Tabela 2.3, o valor do comprimento euclidiano para $d1$, $d2$ e $d3$ seria 30,56; 46,84 e 41,30 respectivamente, e os valores euclidianos normalizados para os três documentos são dados na coluna **Comprimento euclidiano normalizado de TF**.

A Equação (2.4) pode então ser redefinida como:

$$sim(d1, d2) = \vec{v}(d1).\vec{v}(d2) \tag{2.5}$$

Portanto, a Equação (2.5) pode ser vista como o produto interno da versão normalizada dos dois vetores de documento. Esta medida é o cosseno do ângulo θ entre os dois vetores, como mostra a Figura 2.2.

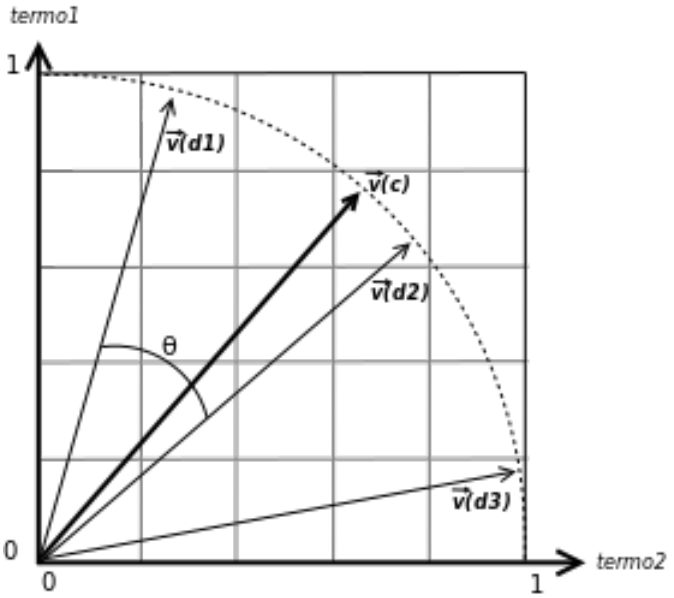

Figura 2.2: Ilustração da similaridade cosseno ($sim(d1, d2) = cos\theta$)

Por meio do MEV é possível comparar uma busca ($\vec{v}(c)$) contra um documento d_i, ambos representados como vetores, conforme Figura 2.2. O documento mais similar à c será aquele que apresentar o maior produto interno ($sim(c, d_i)$). Quando a atribuição de termos para dois vetores é idêntica, o ângulo formado (θ) será zero, produzindo assim a máxima medida de similaridade. Consequentemente, pode-se usar a similaridade cosseno entre o vetor da consulta e o vetor do documento como pontuação do documento para aquela consulta, dado por $score(c, d) = sim(c, d)$. As pontuações resultantes podem ser utilizadas para classificar os documentos para uma determinada consulta Manning et al. (2008). No exemplo da Figura 2.2, considere que o ângulo formado entre os vetores $\vec{v}(c)$ e $\vec{v}(d1)$, $\vec{v}(d2)$ e $\vec{v}(d3)$ seja 25°, 10° e 40° respectivamente. Desta forma, para a consulta dada por c, serão classificados $d2$, $d1$ e $d3$ nesta ordem, sendo $d2$ o de maior relevância para a consulta.

Tabela 2.3: Comprimento euclidiano normalizado

	TF			**Comprimento euclidiano normalizado de TF**		
Termo	**d1**	**d2**	**d3**	**d1**	**d2**	**d3**
carro	27	4	24	0,88	0,09	0,58
automóvel	3	33	0	0,1	0,71	0
seguro	0	33	29	0	0,71	0,7
melhor	14	0	17	0,46	0	0,41

Diferente do MEV, o **modelo Booleano** é um modelo de recuperação da informação no qual podemos realizar consultas na forma de expressão booleana formada pelos termos, ou seja, os termos são combinados por operadores como *AND* (e), *OR* (ou) e *NOT* (negação). Neste modelo cada documento é considerado somente como um conjunto de palavras Manning et al. (2008). Por exemplo, na consulta: *full AND text AND search AND (elasticsearch OR lucene)*, somente documentos que contém todos os termos *full*, *text*, *search* e ainda um dos termos *elasticsearch* ou *lucene* são considerados documentos que satisfazem a consulta.

2.2.2 Motor de busca Elasticsearch

Apesar de existirem outros motores de busca como *Xapian* Xapian (2000), Apache Solr Apache (2006) e Indri Lemur (2000), para o presente trabalho consideramos suficiente explanar sobre o motor de busca Elasticsearch Elastic (2015), por ser considerado um motor de busca robusto, rápido e fácil de ser utilizado Gormley e Tong (2015). Serão descritos os aspectos relevantes para o entendimento deste trabalho, principalmente sobre a forma de cálculo de relevância adotada por esta ferramenta.

Elasticsearch é um servidor de busca de código aberto baseado no Apache Lucene McCandless et al. (2010). Lucene é uma biblioteca de mecanismos de busca avançada, de alto desempenho e totalmente equipada Białecki et al. (2012). Elasticsearch é codificado em Java e utiliza Lucene internamente para toda a sua indexação e busca, visando facilitar a busca de texto completo, escondendo as complexidades de Lucene atrás de uma API RESTful Gormley e Tong (2015).

Elasticsearch usa o modelo Booleano para encontrar documentos, e uma fórmula conhecida como função prática de pontuação (*practical scoring function*) para calcular a relevância, empregando conceitos como TF-IDF e modelo de espaço vetorial, e ainda adiciona recursos como fator de coordenação, normalização de comprimento de campo e impulso no termo ou na cláusula de consulta.

A normalização no comprimento do campo diz que quanto mais curto o campo, maior o peso. Quanto mais longo for, menos provável que os termos no campo sejam relevantes. Um termo que aparece em um campo curto como de título tem peso maior do que o mesmo termo que aparece em um campo de conteúdo longo. A normalização do comprimento do campo *norm* é o inverso da raiz quadrada do número de termos no campo $numTerms$ dada por:

$$norm(d) = \frac{1}{\sqrt{numTerms}} \tag{2.6}$$

O aspecto mais significante para influenciar o cálculo de relevância é a impulsão em tempo de busca (*query-time boosting*). Impulsionar um campo significa torná-lo mais importante

que outros campos. O parâmetro *boost* pode ser usado para aumentar a importância de uma cláusula da consulta em tempo de execução da busca. O valor padrão e neutro de *boost* é igual à 1. No exemplo da Figura 2.3 consta uma consulta pelos termos *quick brown fox* em que a cláusula de consulta *title* recebe um valor de impulsão igual à 2 (linha 10), dobrando a importância de *title* em relação à cláusula *content*.

```
GET /_search
{
    "query": {
        "bool": {
            "should": [
                {
                    "match": {
                        "title": {
                            "query": "quick brown fox",
                            "boost": 2
                        }
                    }
                },
                {
                    "match": {
                        "content": "quick brown fox"
                    }
                }
            ]
        }
    }
}
```

Figura 2.3: Exemplo de consulta Elasticsearch

Outro recurso que Elasticsearch fornece é o parâmetro *explain*, que quando requisitado, retorna a explicação de como o resultado da consulta foi calculado. A Figura 2.4 mostra a explicação abreviada do resultado para uma busca pelo termo *fox*, em que (1) mostra a pontuação final para o termo *fox* no campo *text* do documento de *ID* 0; (2) frequência de termos (tf), ou seja, o termo *fox* aparece somente uma vez no campo *text* no documento de *ID* 0; (3) inverso da frequência nos documentos (idf) de *fox* no campo *text* de todos os documentos do índice pesquisado; (4) fator de normalização de comprimento de campo para este campo.

```
weight(text:fox in 0) [PerFieldSimilarity]:  0.15342641   (1)
result of:
    fieldWeight in 0                         0.15342641
    product of:
        tf(freq=1.0), with freq of 1:        1.0          (2)
        idf(docFreq=1, maxDocs=1):           0.30685282   (3)
        fieldNorm(doc=0):                    0.5          (4)
```

Figura 2.4: Exemplo de retorno de uma consulta Elasticsearch

2.3 Algoritmos de Agrupamento de Dados

Os algoritmos de agrupamento de dados (*clustering*) reúnem um conjunto de objetos em subconjuntos ou *clusters*, que sejam coerentes internamente, mas claramente diferentes uns dos outros. Desta maneira, os objetos de um mesmo *cluster* devem ser o mais parecidos possíveis, e objetos de um *cluster* devem ser tão diferentes quanto possível dos documentos em outros *clusters* Manning et al. (2008). No agrupamento, é a distribuição e composição dos dados que determinarão a sua associação a um determinado grupo (*cluster*) Manning et al. (2008).

Diversos algoritmos de agrupamento tem sido propostos na literatura Newman e Girvan (2004); Kaufman e Rousseeuw (2009); Jain et al. (1999); Markines et al. (2009). Algoritmos particionais e hierárquicos são os dois tipos de algoritmos de agrupamento mais estudados e são amplamente utilizados devido à relativa simplicidade e facilidade de implementação comparados a outros algoritmos de agrupamento Aggarwal e Reddy (2013).

2.3.1 Algoritmos particionais

Algoritmos de agrupamento particionais visam descobrir os agrupamentos presentes nos dados pela otimização de uma função objetivo específica e iterativamente melhoram a qualidade das partições. Estes algoritmos precisam do fornecimento de um conjunto de amostras ou grupos iniciais que são então melhorados iterativamente Aggarwal e Reddy (2013).

Estes algoritmos operam em duas etapas: Na primeira, determinam-se k representantes para os K agrupamentos que se deseja encontrar, de forma a minimizar a função objetivo. Na segunda, cada objeto é atribuído ao agrupamento cujo representante estiver mais próximo Jain et al. (1999).

K-Means é um dos algoritmos particionais mais estudados e mais amplamente utilizados Aggarwal e Reddy (2013). O algoritmo clássico começa escolhendo os k pontos representativos como os centróides iniciais. Cada ponto é associado ao agrupamento cujo centróide está mais próximo baseado numa medida de proximidade escolhida. Uma vez que os agrupamentos são formados, os centróides de cada agrupamento são atualizados. O algoritmo repete esses dois passos iterativamente até que os centróides não sofram alterações ou nenhuma outra alternativa no critério de convergência seja encontrada.

Uma representação do algoritmo pode ser vista na Figura 2.5, que mostra como os centróides são alterados após duas iterações do algoritmo: (a) dados iniciais; (b) membros dos agrupamentos após primeira iteração; (c) membros dos agrupamentos após segunda iteração. A desvantagem desta abordagem está no fato da necessidade de se conhecer previamente o número de agrupamentos definido por k.

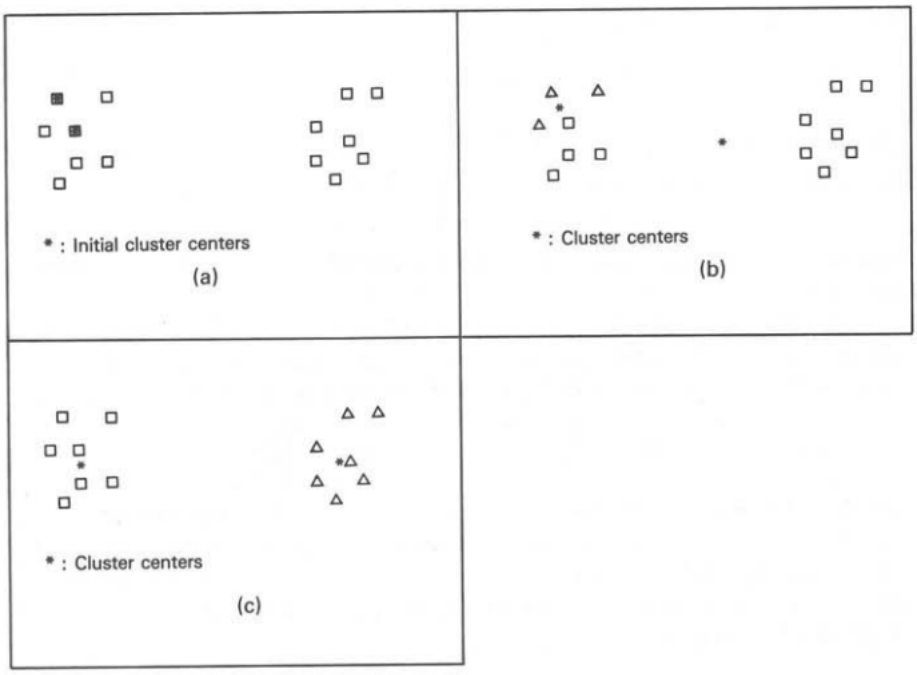

Figura 2.5: Ilustração do algoritmo *K-means*

fonte: Jain e Dubes (1988)

2.3.2 Algoritmos hierárquicos

Algoritmos de agrupamento hierárquicos abordam o problema do agrupamento pelo desenvolvimento de uma estrutura de dados baseada em árvore binária também conhecida como dendograma como na Figura 2.6. Uma vez construído o dendograma, pode-se automaticamente escolher o número certo de agrupamentos dividindo a árvore em diferentes níveis, obtendo desta forma diferentes soluções de agrupamento para o mesmo conjunto de dados sem a necessidade de reexecutar o algoritmo de agrupamento novamente.

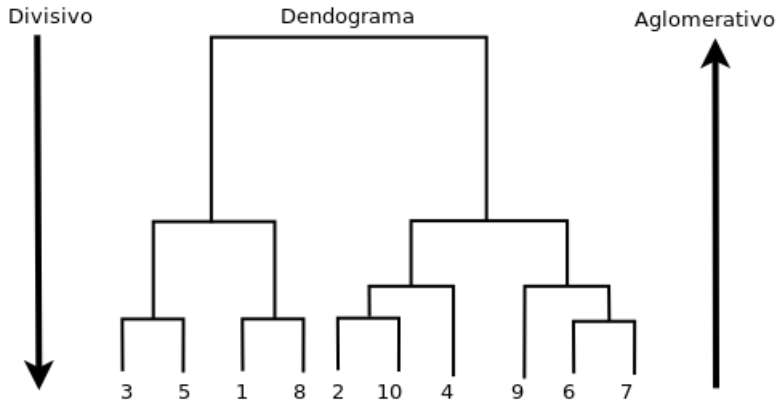

Figura 2.6: Dendograma, ilustração do algoritmo hierárquico

Algoritmos de agrupamento hierárquico baseiam-se em conceitos de similaridade entre os vértices e são classificados em aglomerativo e divisivo Jain e Dubes (1988). Na abordagem aglomerativa cada vértice da rede é considerado uma comunidade unitária. Em seguida, arestas são iterativamente adicionadas ao grafo para unir os subgrafos até que todos os vértices pertençam

a apenas um grafo. Já a abordagem divisiva inicia com apenas um grafo contendo todos os vértices e procede dividindo este grafo em subgrafos cada vez menores, até que cada vértice seja um grafo isolado ou até que se alcance algum critério de parada como, por exemplo, o número de subgrafos desejados. A vantagem deste tipo de algoritmo é sua flexibilidade em relação ao nível de granularidade dos agrupamentos que se deseja obter, os quais podem ser facilmente analisados pela representação gráfica proporcionada pelo dendograma.

2.3.3 Detecção de comunidades

Comunidades são módulos fortemente intra-conectados que geralmente correspondem a unidades funcionais importantes. As grandes redes contêm informações abundantes sobre a organização de um sistema. O desafio é extrair informações úteis por trás da estrutura de inúmeros nós e links. Ferramentas para simplificar e destacar estruturas importantes em redes são essenciais para a compreensão de sua organização. Essas ferramentas são chamadas de métodos de detecção de comunidades e são projetadas para identificar as comunidades presentes na rede Bohlin et al. (2014).

Um problema fundamental relacionado à detecção de comunidades é como definir a melhor divisão da rede em comunidades, visto que em redes reais geralmente não há informação disponível sobre o número e tamanho das comunidades existentes. Pesquisas exploram a estrutura de rede ou grafo que o conjunto de elementos e seus relacionamentos formam, para então realizar a detecção de comunidades ou agrupamentos Girvan e Newman (2002); Rosvall e Bergstrom (2008).

No Capítulo 3 são abordadas algumas ferramentas disponíveis para detecção de comunidades.

2.4 Considerações do capítulo

Este capítulo apresentou termos, conceitos e ferramentas importantes para o entendimento do processo de recuperação da informação, com enfoque em técnicas utilizadas para busca e ranqueamento (ou classificação) de objetos. Além disso conceitos como folksonomia e agrupamento de dados foram abordados para fundamentar os conhecimentos necessários para o entendimento da proposta de busca e ranqueamento abordada neste trabalho.

Capítulo 3

Trabalhos correlatos

Neste capítulo são apresentados os trabalhos relacionados ao tema da proposta. Foram avaliados trabalhos que tratam sobre a recuperação de informação em repositórios de recursos educacionais, que abordam principalmente as dificuldades apresentadas pelos serviços de busca e que também apontam as vantagens trazidas pela adoção da folksonomia e do uso de etiquetas ou *tags* para anotar os RE.

Também fez parte da pesquisa avaliar trabalhos que abordam técnicas de agrupamento e aqueles que relatam sobre a adoção do agrupamento de *tags* para ser utilizada no processo de recuperação da informação. Para fechar o embasamento da proposta, foi necessário conhecer alguns trabalhos que tragam conceitos e técnicas empregadas para realizar o cálculo de relevância e o ranqueamento de recursos digitais.

3.1 Problemas na recuperação de recursos educacionais

de Souza et al. (2008) destacam como é árdua a tarefa de recuperação de RE em repositórios digitais. Para os autores, a recuperação de informação contida em grandes repositórios, quando realizada baseando-se em estratégias de busca sintática, encontra um limitante natural, resultante dos próprios mecanismos de sinonímia existentes em todas as linguagens naturais. Os autores propõem uma arquitetura e a construção de uma ferramenta de recuperação semântica de objetos de aprendizagem em repositórios, utilizando tesauros de uso genérico, para expansão de buscas levando-se em consideração aspectos semânticos. A dificuldade neste tipo de abordagem utilizando tesauro para tratamento semântico na busca, conforme relatado pelos próprios autores, é a escassez de tesauros e repositórios em língua portuguesa.

Em Costa et al. (2013) e Pontes et al. (2014) pode-se constatar como é desafiador para os professores a busca e seleção dos diversos RE disponíveis nos repositórios digitais. Os estudos mostram que apesar de os sistemas de busca serem bastante utilizados, muitos conteúdos irrelevantes são retornados para os professores. As pesquisas tratam sobre a filtragem de informação com aplicações na recomendação de recursos digitais educacionais para auxiliar os professores na tarefa de encontrar RE relevantes às suas necessidades.

Grandes repositórios podem conter dezenas de milhares de objetos de aprendizagem diferentes, tornando difícil a tarefa de encontrar objetos de interesse dos Santos et al. (2015). Os autores realizam experimentos para comparar dois tipos de pré-processamentos baseados em clusterização para a posterior utilização no processo de recomendação de objetos de aprendizagem. A primeira abordagem utiliza a clusterização de objetos de aprendizagem pela similaridade de suas descrições e títulos, enquanto a segunda abordagem realiza a clusterização de usuários

com base nas categorias em que estão cadastrados. Os resultados obtidos demonstraram que a clusterização de objetos de aprendizagem aprimora a qualidade das recomendações.

3.2 Uso da folksonomia para auxiliar na busca por recursos

No trabalho de Sinclair e Cardew-Hall (2008) são conduzidos experimentos nos quais os participantes respondem várias questões utilizando ou a pesquisa tradicional, em que o usuário digita o termo na barra de busca, ou navegando pela nuvem de *tags*. O objetivo principal dos autores era avaliar se realmente a nuvem de *tags* auxilia o usuário no processo de busca. Os resultados dos experimentos mostraram que os participantes preferem utilizar a busca tradicional quando necessitam de informação específica, porém preferem a busca pela nuvem de *tags* quando a necessidade de informação é mais genérica. O estudo aponta que a nuvem de *tags* facilita ao usuário a visualização do resumo de todo conteúdo de um repositório e que a navegação pelas *tags* é considerada um facilitador principalmente para buscas em língua estrangeira. De um modo geral, os resultados apontam que o uso da nuvem de *tags* realmente agrega valor no processo de busca.

Já Morrison (2008) realizou experimentos de recuperação de informação com 33 participantes a fim de comparar os resultados dos sites de marcação social usando folksonomia com as buscas feitas por meio dos tradicionais motores de busca ou por diretórios de tópicos. Os participantes também julgaram a relevância dos resultados apresentados nas buscas. As principais constatações mostram que os motores de busca ainda apresentam maior precisão e revocação [1], mas os resultados foram considerados mais relevantes quando estes aparecem em ambos, tanto nos motores de busca quanto nas folksonomias, em comparação com os resultados retornados somente pelos motores de busca.

Um mecanismo para recuperação de objetos de aprendizagem baseado em serviço de diretório que integra metadados utilizados nos principais repositórios brasileiros e recursos de anotação social foi proposto por Patrocinio e Ishitani (2009). Os autores concluem que a associação desses recursos enriquece as possibilidades de navegação e busca por objetos com a utilização de etiquetas, auxiliando na integração de repositórios heterogêneos, interoperabilidade, compartilhamento do conhecimento e aumento da disponibilidade de objetos de aprendizagem. Os autores apenas destacam a importância do uso das etiquetas, mas não exploram a formação de agrupamentos que pode ser obtida pela similaridade entre as etiquetas. Os agrupamentos poderiam ampliar ainda mais as possibilidades de busca por serem capazes de mapear as etiquetas correlacionadas no conjunto.

Nas pesquisas realizadas por Coelho (2009), foi possível verificar que tanto as máquinas de busca quanto os repositórios digitais pesquisados, apresentam resultados pouco satisfatórios na recuperação de RE. Para os autores, o melhor desempenho foi dado aos serviços que permitem a participação do usuário para anotação de recursos, sendo utilizado o serviço de *bookmarking* conhecido como *del.icio.us*[2]. O estudo exploratório reforça a necessidade de criação de um mecanismo apropriado para recuperação de RE que se valha de outros recursos para facilitar a pesquisa, como por exemplo empregar recursos de anotação dos RE.

Segundo Li et al. (2016), o acesso à semântica do conteúdo visual foi melhorado ao adicionar novas *tags* relevantes, refinando as existentes e utilizando-as na recuperação dos recursos. O artigo apresenta uma pesquisa sobre trabalhos de atribuição, refinamento e

[1] Precisão é a fração de documentos retornados que são relevantes. Revocação é a fração de documentos relevantes que são retornados Manning et al. (2008).

[2] http://del.icio.us

recuperação de *tags* em imagens, ilustrando conexões e diferenças entre os muitos métodos e suas aplicabilidades, ajudando o público interessado a escolher um método existente ou a elaborar um método próprio com os dados da pesquisa em mãos. Com base na principal observação de que todos os trabalhos dependem do aprendizado da relevância da *tag* como ingrediente base, os trabalhos, que variam em termos de metodologias e tarefas específicas, foram avaliados com um protocolo experimental desenvolvido para compará-los diretamente com o estado da arte. Um conjunto selecionado de onze trabalhos representativos para atribuição, refinamento e/ou recuperação de *tags* foram implementados e avaliados, apresentando os melhores desempenhos em cada tarefa específica. Por exemplo, a recuperação de imagens utilizando a relevância de *tag* aprendida produz resultados mais precisos em comparação com a recuperação de imagens usando tags originais. Para atribuição e recuperação de *tags*, métodos que exploram *tags* juntamente com a mídia de imagem por meio de aprendizagem baseada em instância assumem a posição de liderança.

Saoud e Kechid (2016) definem um perfil social de usuário baseado na estrutura de folksonomia e, usando esse perfil definem uma nova abordagem para realizar a expansão de consultas e personalizar e melhorar o processo de recuperação de informação distribuída. Os resultados do trabalho mostram que a integração do perfil social no processo de seleção da fonte e no processo de mesclagem de resultados melhora as métricas de relevância nos sistemas de recuperação de informação distribuídos.

3.3 Agrupamento de *tags*

No presente trabalho não será implementado algoritmo de agrupamento de dados, porém considerada a importância do assunto neste trabalho, nesta seção abordaremos alguns dos trabalhos existentes nesta área. Para a implementação do modelo proposto no trabalho será utilizada uma ferramenta de detecção de comunidades que será acoplada à infraestrutura do Portalmec.

As grandes redes formadas por inúmeros nós e seus relacionamentos contêm informações abundantes sobre a organização de um sistema. O desafio é extrair informações úteis nessa estrutura. Ferramentas poderosas para simplificar e destacar estruturas importantes em redes são essenciais para a compreensão de sua organização. Essas ferramentas são conhecidas como técnicas de agrupamento ou métodos de detecção de comunidades e são projetadas para identificar módulos fortemente intra-conectados Lancichinetti e Fortunato (2009b).

Apesar de não serem especificamente relacionados ao contexto de repositórios de RE, trabalhos semelhantes existem no que tange a agrupamento de *tags* por uma medida de similaridade para serem utilizadas no processo de busca. O agrupamento de *tags* tem sido explorado para melhorar os serviços de busca, navegação e recomendação utilizados na internet Gemmell et al. (2008); Shepitsen et al. (2008). As *tags* similares ou correlacionadas que são identificadas pelos agrupamentos ajudam o usuário no processo de busca Begelman et al. (2006).

Sun et al. (2009) propõem um *framework* de agrupamento chamado RankClus, que parte do pressuposto de que classificar objetos sem considerar a quais grupos eles pertencem muitas vezes leva a resultados estúpidos. RankClus integra agrupamento com classificação, o qual gera classificação condicional em relação aos clusters para melhorar a qualidade do *ranking* e usa classificação condicional para gerar novos atributos de medida para melhorar o agrupamento. Como resultado, a qualidade do agrupamento e classificação são mutuamente aprimoradas, o que significa que os agrupamentos tornam-se mais precisos e a classificação fica mais significativa. Além disso, os resultados de agrupamento com classificação podem fornecer visões mais informativas sobre os dados.

Hassan-Montero e Herrero-Solana (2006) propõem uma abordagem onde a formação e apresentação da nuvem de *tags* é baseada em agrupamento de *tags*. Os agrupamentos são formados por meio do algoritmo de agrupamento *K-means*, baseado na medida de coocorrência relativa, ou coeficiente de Jaccard, que é dado pela divisão do número de recursos onde as *tags* coocorrem pelo número de recursos onde as *tags* ocorrem isoladas. Desta forma os autores conseguem apresentar uma nuvem de *tags* com uma distribuição visual mais coerente do que as nuvens ordenadas alfabeticamente e também ajuda o usuário a inferir a semântica contida nas *tags* pelo relacionamento com suas *tags* vizinhas. A desvantagem desta abordagem é a utilização da técnica *K-means*, pois esta exige que se determine previamente o número de agrupamentos que se deseja formar. Definir o número de agrupamentos não é trivial e um número mal dimensionado afeta a qualidade dos agrupamentos formados.

O projeto de um sistema de recuperação da informação baseado em coocorrência de *tags* e no subsequente agrupamento é apresentado no trabalho de Knautz et al. (2010). Este sistema permite que os usuários possam acessar dados digitais por meio da visualização e navegação pelos agrupamento de *tags*. Além disso, para os autores, o agrupamento de *tags* permite uma nova forma de expansão da consulta de forma mais amigável. Essa expansão acontece a medida que o usuário navega pelos vértices (*tags*) ou pelas arestas (relacionamento entre *tags*) do agrupamento.

Passaremos agora a analisar trabalhos baseados na detecção de comunidades. Um dos algoritmos divisivos mais populares foi proposto por Girvan e Newman (2002) e utiliza o conceito de *Betweenness* para remover arestas que conectam comunidades. *Betweenness* é uma medida usada para identificar arestas que conectam comunidades, atribuindo-lhes valores altos e penalizando arestas que conectam vértices pertencentes a mesma comunidade. Para o seu cálculo é necessário obter o caminho mínimo entre dois vértices, dado por (3.1), onde $\sigma(i, u, j)$ é o número de caminhos mínimos entre os vértices i e j que passam pelo vértice ou aresta u; $\sigma(i, j)$ é o número total de caminhos mínimos entre i e j e o somatório se aplica a todos os pares i e j de vértices distintos.

$$B_u = \sum_{ij} \frac{\sigma(i, u, j)}{\sigma(i, j)} \tag{3.1}$$

Considerando-se uma rede com duas comunidades ligadas por um pequeno número de arestas, tem-se que todos os caminhos da rede com origem em um vértice de uma comunidade e destino em um vértice da outra comunidade devem passar por alguma destas arestas que conectam as comunidades. Desta forma, estas arestas terão alto valor de *Betweenness* e as arestas dentro de uma mesma comunidade terão um valor menor. O algoritmo de Girvan e Newman Girvan e Newman (2002) remove as arestas com alto valor de *Betweenness* iterativamente e então recalcula o *Betwenness* das arestas remanescentes. A principal desvantagem do algoritmo é seu custo computacional. Em um grafo com M arestas e N vértices, a complexidade total para o cálculo do *Betwenness* é $O(M^2N)$.

Em trabalho posterior Newman (2004) propôs um algoritmo de detecção de comunidades, conhecido como medida de modularidade, que apresenta um melhor custo computacional, com complexidade $O((N + M)N)$ e proporciona resultados qualitativamente similares aos da medida *Betweenness*.

Considere e_{ij} como a metade da fração das arestas da rede que conectam vértices do grupo i com os vértices do grupo j. Portanto a fração total de arestas é igual a $e_{ij} + e_{ij}$. A única exceção ocorre com elementos ortogonais e_{ii}, que são iguais à fração de arestas dentro do grupo i. Desta forma $\sum_i e_{ii}$ é a fração total das arestas dentro de um grupo. Todas as outras arestas conectam vértices entre grupos. O valor máximo desta somatória é igual a 1, ou seja, valores próximos a 1 indicam que a divisão da rede em comunidades é boa. Entretanto, o somatório

por si só não é um indicador de qualidade para medir a estrutura da comunidade, visto que por exemplo, colocando-se todos os vértices em uma única comunidade retornaria o valor máximo 1, porém sem utilidade para gerar estruturas de comunidade.

Uma forma mais útil para se calcular a estrutura de comunidade é dada por $\sum_i e_{ii}$ subtraído do valor que este somatório teria se existissem arestas colocadas aleatoriamente. Define-se a_i como a fração de todas as arestas que se conectam a vértices no grupo i. Pode-se calcular a_i como $a_i = \sum_j e_{ij}$. Se as arestas são conectadas aleatoriamente em conjunto, a fração de arestas resultantes que conectam vértices dentro do grupo i é dado por a_i^2. Define-se então a medida de modularidade como:

$$Q = \sum_i (e_{ii} - a_i^2). \tag{3.2}$$

Se uma dada divisão não gera mais arestas dentro da comunidade do que seria esperado por uma formação aleatória, a modularidade $Q = 0$. Valores maiores que 0 indicam desvio da aleatoriedade, e na prática valores maiores que $0,3$ indicam estruturas de comunidade significantes.

Da forma como foi originalmente definida, Q envolve processos de buscas e divisões iterativas de alto custo computacional, uma vez que deveria-se calcular Q para todas as possíveis formações de comunidades na rede, tornando o cálculo inviável para sistemas com mais de 20 ou 30 vértices. Desta forma, foi necessário mesclar à função alguma heurística que permitisse a redução do custo computacional, sendo adotado um algoritmo de otimização guloso (*greedy*).

Partindo-se de um estado no qual cada vértice da rede representa uma comunidade, comunidades são conectadas duas a duas, repetidamente, até que seja selecionada a conexão que resulte no maior valor de Q. A mudança em Q ao unir duas comunidades é calculada por (3.3), onde e_{ij} é a fração das arestas que conectam a comunidade i à comunidade j, a_i é fração total de arestas que conectam a comunidade i às demais comunidades da rede e pode ser calculada por $a_i = \sum_k e_{ik}$, assim como a_j é a fração total de arestas que conectam a comunidade j às demais comunidades da rede e pode ser calculada da mesma forma que a i.

$$\Delta Q = 2(e_{ij} - a_i a_j), \tag{3.3}$$

O algoritmo proposto por Blondel et al. (2008) é um método heurístico baseado na otimização de **modularidade**. O algoritmo localiza partições de alta modularidade em redes complexas num curto período de tempo, gerando uma estrutura de comunidade hierárquica completa para a rede.

O algoritmo é dividido em duas fases que se repetem iterativamente. Considere uma rede ponderada de N nós. Primeiro, cada nó é atribuído a uma comunidade distinta, ou seja, inicialmente o número de comunidades é igual ao número de nós. Para cada nó i e seus j nós vizinhos, avalia-se o ganho de modularidade que se poderia obter ao remover i de sua comunidade colocando-o na comunidade j. O nó i será colocado na comunidade que gerar o máximo de ganho na modularidade, que também precisa ser positiva. Se não houver ganho positivo, i permanece na sua comunidade original. Este processo é aplicado repetidamente e sequencialmente para todos os nós até que nenhum ganho adicional na modularidade seja possível. Assim completa-se a primeira fase.

Parte da eficiência do algoritmo resulta do fato de que o ganho em modularidade ΔQ obtido ao mover um nó isolado i para a comunidade C pode ser facilmente calculado pela Equação (3.4), onde *in* é o somatório dos pesos dos *links* dentro de C, tot é o somatório dos pesos dos *links* incidentes aos nós em C, k_i é o somatório dos pesos dos *links* incidentes ao nó i, $K_{i,in}$ é o

somatório dos pesos dos *links* de i para os nós em C e m é o somatório dos pesos de todos os *links* na rede.

$$\Delta Q = \left[\frac{\sum in + 2K_{i,in}}{2m} - \left(\frac{\sum tot + k_i}{2m} \right)^2 \right] - \left[\frac{\sum in}{2m} - \left(\frac{\sum tot}{2m} \right)^2 - \left(\frac{K_i}{2m} \right)^2 \right]. \tag{3.4}$$

Uma expressão similar é usada para calcular a mudança de modularidade quando i é removido de sua comunidade. Na prática, calcula-se a mudança de modularidade removendo-se i de sua comunidade e então movendo-o para dentro da comunidade vizinha.

A segunda fase do algoritmo consiste em construir uma nova rede cujos nós são agora as comunidades encontradas durante a primeira fase. Fazendo isso, os pesos dos *links* entre os novos nós são dados pelo somatório dos pesos dos links entre os nós nas duas correspondentes comunidades.

Uma vez terminada a segunda fase, é possível reaplicar a primeira fase do algoritmo para esta nova rede de forma iterativa. Denominando-se *passo* a combinação dessas duas fases, o número de meta-comunidades diminui a cada passo executado e consequentemente a maior parte do tempo de processamento é gasto na primeira fase. Os passos são iterados até que não haja mais mudanças e a máxima modularidade é alcançada.

O algoritmo incorpora a noção de hierarquia, onde comunidades de comunidades são construídas durante o processo e a profundidade ou altura da hierarquia é dada pelo número de passos que se executa.

A representação deste algoritmo pode ser vista na Figura 3.1, onde cada passo é composto de duas fases: uma onde a modularidade é otimizada, permitindo apenas mudanças locais de comunidades; outra onde as comunidades encontradas são agregadas para construir uma nova rede de comunidades. As passagens são repetidas iterativamente até que não seja possível aumentar a modularidade.

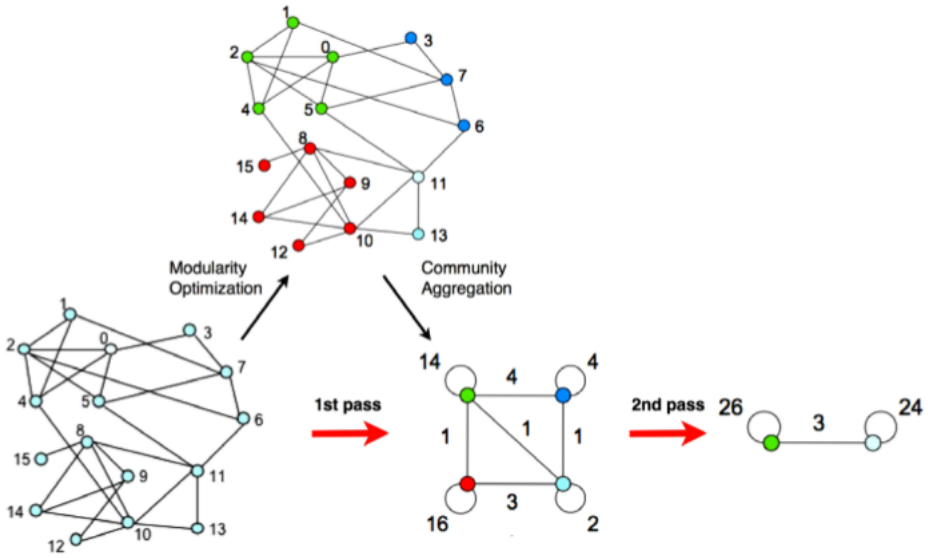

Figura 3.1: Ilustração do algoritmo de Blondel

fonte: Blondel et al. (2008)

A estrutura modular de um grafo pode ser considerada uma descrição comprimida da matriz de adjacências do grafo. Baseado nessa ideia Rosvall e Bergstrom (2008) desenvolveram o método *Infomod*, no qual a partição do grafo em comunidades é encarada como uma informação a ser comprimida numa comunicação entre um emissor e um receptor. O receptor deve tentar inferir a topologia original do grafo a partir da informação comprimida.

Em outro trabalho, Bohlin et al. (2014) seguem a mesma ideia de Rosvall e Bergstrom (2008), descrevendo um grafo usando menos informações do que as codificadas na matriz de adjacência completa, com o objetivo de otimizar a compressão das informações necessárias para descrever o processo de difusão da informação em todo o grafo.

Um passeio aleatório é utilizado para a difusão das informações. Por meio do algoritmo conhecido como ***Infomap***, o grafo é codificado numa estrutura de dois níveis, na qual atribuem-se nomes únicos às estruturas importantes do grafo e aos vértices dentro da mesma estrutura. Porém reaproveita-se os nomes dos vértices entre diferentes estruturas, de forma a obter uma descrição mais compacta (*codebooks*) do que simplesmente codificar todos os vértices com diferentes nomenclaturas. É semelhante ao procedimento adotado em mapas geográficos, onde as estruturas correspondem a cidades e os nomes de ruas se repetem em cidades diferentes, desde que exista uma única rua com esse nome numa mesma cidade. Para a caminhada aleatória, as estruturas mencionadas são comunidades, onde os caminhantes passam muito tempo dentro delas, desempenhando um papel crucial no processo de difusão da informação.

O agrupamento de grafos enfrenta então o seguinte problema de codificação: encontrar a partição que produz o comprimento mínimo da descrição de uma caminhada aleatória infinita (*map equation*). Exceto por seguir uma sequência randômica e não sequencial, o núcleo do algoritmo de detecção de comunidades proposto por Bohlin et al. (2014) segue o mesmo algoritmo proposto por Blondel et al. (2008): nós adjacentes são agrupados em módulos, os quais são agrupados em supermódulos e assim sucessivamente.

Inicialmente cada nó constitui um módulo. Em seguida, numa sequência aleatória, cada nó é movido para o módulo vizinho que apresente o maior decréscimo do *map equation*, caso contrário ele permanece no seu módulo atual. Este procedimento é repetido, seguindo-se uma nova sequência randômica a cada vez que é executado, até que mais nenhum movimento resulte em decréscimo do *map equation*. A rede é então reconstruída, com os módulos da fase anterior formando os vértices para esta nova fase e assim repetem-se os passos iterativamente. Esta reconstrução hierárquica da rede é repetida até que *map equation* não possa ser reduzida.

Detectar comunidades pelo **mapeamento de fluxo** é conceitualmente uma abordagem bastante diferente de inferir atribuição de módulos nos modelos de rede subjacentes. Enquanto a primeira foca na interdependência entre links e na dinâmica na rede uma vez que é formada, a segunda foca na interação dos pares e no próprio processo de formação. Uma vez que *Map Equation* e Modularidade consideram essas abordagens distintas, é interessante ver como isso difere na prática. *Map equation* captura pequenos módulos com longos tempos de persistência e a modularidade captura pequenos módulos com mais do que o número esperado de links-fins, entrantes ou saintes.

A Figura 3.2, baseada em Rosvall et al. (2009), mostra duas redes diferentes, cada uma particionada de duas maneiras diferentes. Ambas redes são geradas a partir do mesmo modelo de rede subjacente no sentido da modularidade: 20 links direcionados conectam 16 vértices em quatro módulos, com peso total de entrada (*in*) e saída (*out*) equivalentes em todos os módulos.

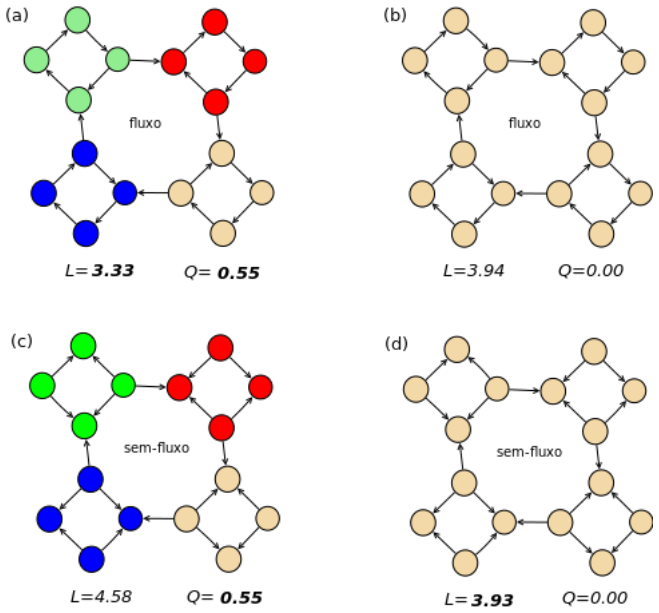

Figura 3.2: Comparativo entre *Map equation* e Modularidade

As duas amostras de rede, *fluxo* e *sem-fluxo*, são idênticas, exceto pela direção de dois links em cada grupo de quatro nós. A cor dos nós ilustra partições alternativas. As soluções ótimas para *map equation* (mínimo L) e para modularidade (máximo Q) são destacadas em negrito. Os links direcionados nas redes em (a) e (b) conduzem a um fluxo por toda a rede com relativamente longo tempo de persistência nos módulos em (a). Entretanto, a partição de quatro módulos em (a) minimiza *map equation*.

Modularidade, por considerar padrões com altos pesos em links dentro dos módulos, prefere a solução de quatro módulos em (a) do que a rede não particionada em (b). Os links direcionados na rede em (c) e (d) representam falta de movimento entre os nós, porém interação entre os pares, e a estrutura *sem-fluxo* não gera fluxo. Sem fluxo e sem regiões com longos tempos de persistência, não há como usar múltiplos *codebooks* e a rede não particionada em (d) otimiza *map equation*. Mas para modularidade que conta somente os pesos dos links e o grau de links entrantes e saintes nos módulos, não há diferença entre as soluções em (a)/(c) e (b)/(d) respectivamente, e novamente a solução de quatro módulos maximiza a modularidade.

Considerando que não é objetivo deste trabalho a implementação de algoritmos de agrupamento e que simplesmente faremos uso de ferramentas disponíveis, consideramos útil analisar alguns trabalhos relacionados à avaliação de desempenho de grafos, sendo a mais popular proposta por Girvan e Newman (GN) Girvan e Newman (2002), onde a rede ou o grafo é composto por 128 nós, todos com grau 16 e divididos em 4 grupos de 32.

A avaliação comparativa GN é normalmente usada para testar algoritmos de detecção de comunidades e são comparados baseados em sua performance. Entretanto, segundo Lancichinetti e Fortunato (2009b), avaliação GN tem duas desvantagens: 1) todos os nós tem o mesmo grau; 2) todas as comunidades tem o mesmo tamanho; aspectos que não condizem com redes complexas reais que são caracterizadas por distribuição heterogênea de graus e comunidades com tamanhos distintos.

Em Lancichinetti et al. (2008); Lancichinetti e Fortunato (2009a), os autores introduzem uma nova classe de avaliação de desempenho de grafos, *Lancichinetti–Fortunato–Radicchi (LFR) benchmark*, que generaliza a avaliação GN permitindo tamanho de comunidade e distribuição de grau que seguem uma lei de potência (*Power Law* (Easley e Kleinberg, 2010, cap. 18, p. 545)). Lancichinetti e Fortunato (2009b) executam então testes de desempenho com algoritmos de

detecção de comunidades e concluem que os melhores resultados são atribuídos ao método Infomap Rosvall et al. (2009).

Para o presente trabalho, essa última pesquisa tem peso fundamental, pois a estrutura formada pelo conjunto de RE, suas etiquetas e seus relacionamentos podem ser exploradas como uma grande rede, e usando *Infomap* Rosvall et al. (2009) pretende-se extrair os agrupamentos necessários para embasar a proposta de busca por RE.

3.4 Ranqueamento dos resultados no processo de busca

Para entendimento de alguns dos tradicionais algoritmos de ranqueamento (*ranking*), passamos primeiro pelo ***PageRank*** Brin e Page (1998). Trata-se do algoritmo adotado pelo motor de busca Google que calcula a importância de uma página baseado na quantidade e na qualidade dos *links* que apontam para ela. *PageRank* baseia-se em grafos direcionados ou dígrafos onde a web é considerada uma rede de citações, sendo cada página um nó e cada aresta corresponde a uma referência (*link*) de uma página para outra. O cálculo de *PageRank* de uma página é dado pela Equação (2.1) citado em (2.2.1).

FolkRank Hotho et al. (2006a,b) por sua vez, é um algoritmo de ranqueamento que é uma adaptação do algoritmo *PageRank* por considerar a estrutura de folksonomia para tratar o ranqueamento de itens. O algoritmo *FolkRank* baseia-se em grafos não direcionados com fechamento triádico formado pelo conjunto dos principais componentes da folksonomia: os usuário, as etiquetas e os recursos etiquetados.

FolkRank implementa o esquema de classificação por distribuição de peso sobre a folksonomia. A ideia original no *PageRank* é a de que uma página é importante se muitas páginas também importantes apontam para ela Brin e Page (1998). Analogamente, *FolkRank* emprega o mesmo princípio para o esquema de ranqueamento baseado em folksonomia: um recurso torna-se importante se lhe for atribuída uma *tag* importante por um usuário importante. O mesmo vale simetricamente para as *tags* e usuários. Assim, tem-se um grafo de vértices que se reforçam mutuamente, espalhando seus pesos Hotho et al. (2006a,b). Portanto, diz-se que este algoritmo não representa uma medida de popularidade, mas sim uma medida de reputação Rochadel (2016).

O presente trabalho não explora relevância dada pela popularidade ou pela reputação dos objetos ou usuários, mas explora a quantidade e peso dos termos correlacionados ao termo de busca.

3.5 Considerações do capítulo

Os trabalhos de de Souza et al. (2008); Patrocinio e Ishitani (2009); Costa et al. (2013) apontam algumas das dificuldades existentes no processo de busca por RE em repositórios digitais, como buscas restritas a análise sintática e com resultados poucos relevantes. Analisando os trabalhos expostos neste capítulo, destacamos as abordagens consideradas relevantes para este trabalho:

- considerar o uso das *tags* agrega valor ao processo de busca Morrison (2008); Patrocinio e Ishitani (2009); Coelho et al. (2012); Saoud e Kechid (2016); Li et al. (2016). Este é um ponto fundamental para o desenvolvimento deste trabalho. É por meio desta unidade de informação que todo o processo de busca idealizado se viabiliza.
- o universo de recursos e suas *tags* pode ser mapeado como uma grande rede Rosvall et al. (2009); Bohlin et al. (2014), e grupos fortemente intra-conectados podem ser mapeados. Os

agrupamentos que se formam a partir das *tags* dão o potencial semântico Hassan-Montero e Herrero-Solana (2006); Knautz et al. (2010); Saoud e Kechid (2016) necessário para combater as restrições impostas pela busca sintática.

- a expansão de termos correlacionados dada pelo agrupamento de *tags* no processo de busca enriquece os resultados obtidos Morrison (2008); Sun et al. (2009); Patrocinio e Ishitani (2009); Coelho et al. (2012); Saoud e Kechid (2016), sendo possível minimizar o problema de retornar poucos resultados relevantes na busca.

- pontuar e ranquear RE considerando agrupamentos de *tags* pode levar a uma maior quantidade de resultados significativos e relevantes Morrison (2008); Sun et al. (2009); Knautz et al. (2010), pois serão levados em consideração também os termos correlacionados ao termo de busca.

Apesar da relevância das abordagens apresentadas nos trabalhos analisados, consideramos que alguns pontos merecem ser melhor avaliados para o desenvolvimento do presente trabalho:

- utilizar as *tags* no processo de busca como em Hassan-Montero e Herrero-Solana (2006); Morrison (2008); Patrocinio e Ishitani (2009) sem considerar a formação de agrupamentos limitam os benefícios que podem ser obtidos pela sua utilização. Por exemplo, sem o suporte dos agrupamentos de *tags* similares fica inviável expandir uma busca com termos correlacionados.

- a abordagem de Hassan-Montero e Herrero-Solana (2006) apesar de utilizar o agrupamento de *tags*, utiliza *K-means*, que além de ser um algoritmo de alto custo computacional, exige que se forneça o número de agrupamentos a ser formado.

- o trabalho que mais se assemelha ao presente estudo é o de Knautz et al. (2010), porém esta abordagem exige a interação do usuário, que por meio dos cliques nas *tags* ou nas arestas precisa compor a sua busca. Nossa abordagem difere desta, pois utiliza o agrupamento de *tags* de forma automática como suporte para o processo de busca, sem exigir a interação do usuário. A ideia é auxiliar o usuário de forma transparente, fornecendo-lhe os benefícios dos termos correlacionados obtidos pela utilização do agrupamento de *tags*.

Capítulo 4

Busca e ranqueamento de recursos educacionais com suporte de agrupamento de *tags*

Este capítulo apresenta o modelo para a busca e ranqueamento de recursos educacionais com suporte na estrutura formada pelo agrupamento de *tags*, cujo objetivo é melhorar o resultado da busca, aumentando a probabilidade de retornar RE mais relevantes aos termos pesquisados.

Apesar de não serem especificamente relacionados ao contexto de repositórios de RE, existem trabalhos que realizam o agrupamento de *tags* por uma medida de similaridade para serem utilizadas no processo de busca, sendo que alguns como os de Hassan-Montero e Herrero-Solana (2006); Knautz et al. (2010) focam na apresentação dos agrupamentos de *tags* como uma alternativa à nuvem de *tags*, nos quais o usuário pode visualizar as *tags* relacionadas e realizar suas buscas clicando nas próprias *tags* ou nas arestas que representam seus relacionamentos. Outros como Begelman et al. (2006) utilizam os agrupamentos de *tags* para sugerir ao usuário termos similares no momento da busca.

O presente trabalho difere dos citados porque (i) expande automaticamente os resultados da busca por considerar as *tags* correlacionadas para buscar os recursos associados a estes termos, fazendo uma espécie de ampliação semântica da busca; e (ii) combina estes resultados aos do motor de busca tradicional, reclassificando todos os resultados dando maior relevância aos resultados que aparecem em ambas as buscas, bem como aos resultados obtidos por meio dos termos correlacionados. A Seção 4.1 traz uma visão geral da arquitetura proposta e as suas subseções detalham todos os processos desta arquitetura.

4.1 Visão geral do processo de busca e ranqueamento

A Figura 4.1 mostra uma visão geral das tarefas necessárias, bem como o fluxo entre tarefas, para a realização de uma busca e ranqueamento de RE com suporte do agrupamento de *tags*. Neste modelo há dois grandes grupos de processos, identificados por **P1-Agrupamento** e **P2-Busca e ranqueamento**.

As tarefas do grupo **P1-Agrupamento** têm o objetivo principal de formar os agrupamentos de *tags*:

- **T1** - Recuperação da lista de RE e sua *tags*: nesta tarefa é recuperada a lista de todos os recursos educacionais e suas correspondentes *tags* a partir do banco de dados do repositório digital que armazena os RE.

- **T2** - Mapeamento da coocorrência entre *tags*: a partir da lista de RE e suas *tags*, esta tarefa faz o mapeamento de todos os pares de *tag* utilizadas para anotar um mesmo RE. Além disso, calcula o coeficiente de similaridade para todo par de *tags* coocorrentes.
- **T3** - Geração do grafo: utilizando as informações sobre as *tags* (nós), as coocorrências (arestas) e seus coeficientes de similaridade (peso nas arestas), esta tarefa transforma esse conjunto de informações em um grafo não direcionado.
- **T4** - Cálculo e formação dos agrupamentos: com base no grafo formado na tarefa [T3], aqui realiza-se o agrupamento das *tags* conforme o coeficiente de similaridade das mesmas.

Estas tarefas devem ser executadas periodicamente, gerando e atualizando as informações necessárias para possibilitar as buscas que são realizadas pelas tarefas do grupo P2-Busca e ranqueamento.

As tarefas do grupo **P2-Busca e ranqueamento** por sua vez, tem o objetivo de realizar a busca de RE a partir de um termo de pesquisa:

- **T5** - representa a entrada de um termo de busca.
- **T6** - Busca via motor de busca: nesta tarefa o motor de busca recupera uma lista de RE considerados correspondentes ao termo pesquisado classificados conforme algoritmo de ranqueamento utilizado pelo motor de busca.
- **T6'** - Busca via agrupamento de *tags*: esta tarefa também recupera uma lista de RE, mas para isso considera além do termo original da busca, todas as *tags* pertencentes ao mesmo agrupamento do termo original, retornando todos os RE relacionados a esse conjunto de termos.
- **T7** - Mescla e recalcula ranqueamento: esta tarefa faz a união das duas listas de resultado gerados em [T6] e [T6'], recalculando e reclassificando os RE antes de retornar o resultado final para o usuário.

4.1.1 Recuperação da lista de recursos educacionais e suas *tags*

Primeiro abordaremos as tarefas do grupo **P1-Agrupamento**. Na tarefa **[T1] Recuperação da lista de RE e suas *tags***, é realizada a coleta de todos os RE com as respectivas *tags* que lhe foram atribuídas. Supondo que os recursos disponíveis em um repositório sejam representados pelas Tabelas 4.1, 4.2, 4.3, sendo elas o universo de RE, de *tags* e da associação de *tags* aos RE respectivamente, o conjunto resultante neste passo seria dado por: $C_{rt} = \{r1 : \{t10, t12\}, r2 : \{t10, t11, t13\}, r3 : \{t10, t11\}\}$. Por questões de otimização os dados são tratados na arquitetura pelos seus códigos e não por seus nomes extensos.

Tabela 4.1: Lista de Recursos Educacionais

código RE	**Nome RE**
r1	Motores moleculares
r2	Geometria molecular
r3	The Drosophila Molecular

Tabela 4.2: Lista de *tags*

código tag	**Nome tag**
t10	molécula
t11	biologia
t12	movimento
t13	geometria molecular

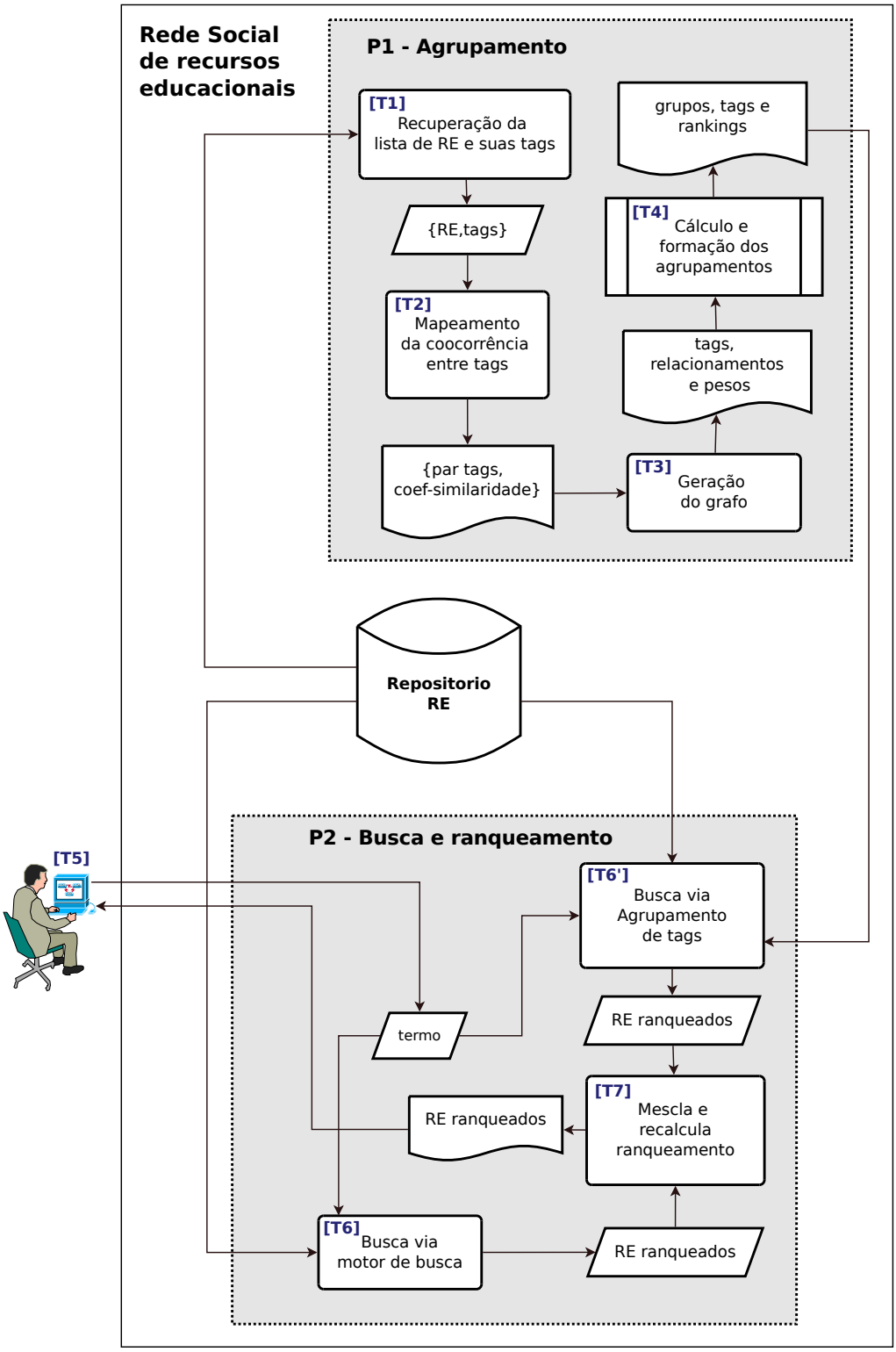

Figura 4.1: Representação do processo de busca e ranqueamento de recursos educacionais

Tabela 4.3: Lista de RE e suas respectivas *tags*

Nome RE	***Nome tag***
Motores moleculares	molécula, movimento
Geometria molecular	molécula, biologia, geometria molecular
The Drosophila Molecular	molécula, biologia

4.1.2 Mapeamento das *tags* coocorrentes

A tarefa **[T2] Mapeamento da coocorrência entre *tags*** do grupo **P1-Agrupamento**, utiliza como entrada o conjunto C_{rt} da tarefa **[T1]** para realizar o mapeamento de todos os pares de *tags* coocorrentes. Entende-se que duas *tags* são coocorrentes se ambas são utilizadas para anotar um mesmo RE. Continuando com o conjunto C_{rt} do exemplo anterior, temos que a *tag* $t10$ coocorre com as *tags* $t11$, $t12$ e $t13$. Além disso, calcula-se a quantidade de vezes que cada par de *tags* coocorre. Todas essas informações são mapeadas para uma lista com o seguinte formato: $M_{tc} = \{t10 : \{t11 : 2, t12 : 1, t13 : 1\}, t11 : \{t10 : 2\}, t12 : \{t10 : 1\}, t13 : \{t10 : 1, t11 : 1\}\}$, até que todos os pares de *tags* coocorrentes e as respectivas quantidade de vezes que elas coocorrem no repositório sejam mapeados. Analisando o conjunto M_{tc}, a *tag* $t10$ coocorre com a *tag* $t11$ duas vezes e com $t12$ e $t13$ uma única vez e assim sucessivamente com todos os pares coocorrentes.

Calcula-se em seguida o coeficiente de similaridade dos pares de *tags* coocorrentes pela similaridade cosseno, conforme Equação (2.4) da Subseção 2.2.1. Para a lista M_{tc} do exemplo, a lista resultante com os coeficientes de similaridade ficaria assim representada: $M_{cs} = \{t10 : \{t11 : 0.82, t12 : 0.58, t13 : 0.58\}, t11 : \{t10 : 0.82\}, t12 : \{t10 : 0.58\}, t13 : \{t10 : 0.58, t11 : 0.71\}\}$.

4.1.3 Geração do grafo não direcionado

Passa-se então para a tarefa **[T3] Geração do grafo**, pois neste ponto já temos as informações necessárias à composição de um grafo não direcionado $G(V, A, P)$, constituído de vértices V, um conjunto de arestas A com seus respectivos pesos P. Cada vértice v_i pertencente a V representa uma *tag* do conjunto M_{cs}, e haverá uma aresta entre v_i e v_j se tag_i for coocorrente com tag_j ou vice-versa, e seu peso $\mathrm{P}_{i,j}$ será o coeficiente de similaridade desse par de *tags*. A Figura 4.2 representa o grafo formado pelo conjunto M_{cs}.

A representação do conjunto M_{cs} em estrutura de grafo é necessária pois é o elemento de entrada para o algoritmo de agrupamento.

4.1.4 Agrupamento de *tags* similares

Passamos finalmente à tarefa **[T4] Cálculo e formação dos agrupamentos**. Como não é objetivo deste trabalho o desenvolvimento de um algoritmo específico para realizar o agrupamento, por isso não entraremos no mérito de como calcular o agrupamento. Nesta etapa deve ser utilizado um algoritmo ou ferramenta disponível que calcule e forme os agrupamentos de *tags* baseado na medida de similaridade das *tags* cocorrentes, informações disponibilizadas pelo grafo formado na tarefa **[T3]**. Cada agrupamento será composto por um conjunto de *tags* ranqueadas conforme sua relevância para o agrupamento. Com a formação dos grupos as tarefas do grupo **P1-Agrupamento** são concluídas e as informações necessárias para dar suporte às tarefas do grupo **P2-Busca e ranqueamento** estão disponíveis.

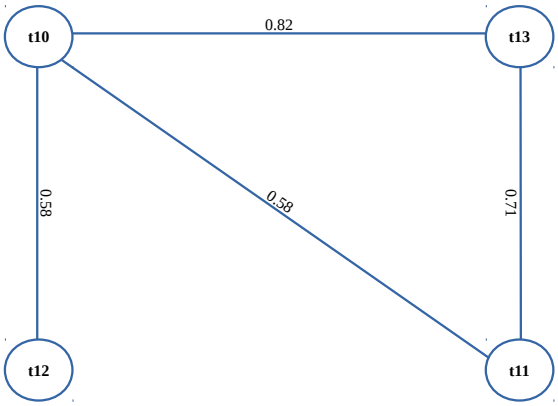

Figura 4.2: *Tags* coocorrentes, seus relacionamentos e coeficientes de similaridade

4.1.5 Busca de recursos educacionais via motor de busca

Passaremos agora a descrever as tarefas do grupo **P2-Busca e ranqueamento** da arquitetura exposta na Figura 4.1. As tarefas deste grupo são executadas quando se deseja realizar a busca por recursos educacionais a partir de um ou mais termos de pesquisa. Todo o processo inicia-se na tarefa identificada como **[T5]**, onde é feita a entrada do termo de busca, que é então repassado para duas tarefas **[T6]** e **[T6']**. A tarefa **[T6] Busca via motor de busca** inicia a tarefa de busca de RE por meio de um motor de busca, que ao final retornará um conjunto de RE ranqueados (C_{mb}) conforme a relevância calculada em relação ao termo pesquisado. Também não é objetivo deste trabalho o desenvolvimento de um motor de busca tradicional, por isso deve-se utilizar ferramenta disponível que realize esta funcionalidade.

4.1.6 Busca de recursos educacionais via agrupamento de *tags*

A tarefa **[T6'] Busca via agrupamento de *tags*** realiza a busca com o suporte do agrupamento de *tags*. Ao receber o termo de pesquisa, primeiro identifica-se a qual agrupamento o termo pertence. Encontrado o agrupamento correspondente, recuperam-se as *tags* correlacionadas ao termo pesquisado, fazendo-se uma espécie de expansão da busca, tanto no sentido quantitativo como semântico. Quantitativo por aumentar o número de termos de pesquisa, e semântico por considerar termos considerados similares. Nesta abordagem serão buscados os RE que são correspondentes tanto ao termo de pesquisa original, como também os RE que são correspondentes às *tags* correlacionadas. Desta forma é possível recuperar RE diferentes daqueles recuperados pelo motor de busca.

Para o modelo aqui proposto, empregam-se equações para cálculo de relevância e ranqueamento, que foram criadas e testadas de forma empírica, como as Equações (4.1), (4.2) e (4.3). Trata-se de um conjunto de equações simples, de fácil implementação, e que apresentam bons resultados, descritos e avaliados no decorrer deste trabalho.

Para calcular a relevância e ranquear os RE encontrados, primeiro considera-se que a *tag* correspondente ao termo original de pesquisa é o elemento do agrupamento que tem a maior relevância para esta busca. Para diferenciá-la denominamo-na de *tag* principal. A relevância das outras *tags* correlacionadas é recalculada relativamente à *tag* principal. Seus pesos serão proporcionais à distância (diferença) que se situam no agrupamento em relação à *tag* principal. Quanto maior a distância, menos relevante.

Tabela 4.4: *Tags* coocorrentes com o termo “Sagitário”

Tag	**Peso original do grupo**	**Peso recalculado** ($rank$)
Estrela	0.00021569	0.5
Peony	3.11466e-05	0.933505437256604
Shine	3.11466e-05	0.933505437256604
Estrela anã	2.42713e-05	0.9893778666944993
Estrela luminosa	2.42713e-05	0.9893778666944993
Sagitário	2.31473e-05	1.0
W5	1.55874e-05	0.9114198887466596
Movimento das estrelas diurnal	1.46697e-05	0.8978267590590719
Galáxia Roda de Carroça	1.18271e-05	0.8495753919004235
Recycle	1.11963e-05	0.8372969365374049
Disco protoplanetário	6.47752e-06	0.7147040914896332
Diagrama de Hertzsprung Russell	6.13133e-06	0.7023996753547234

Os valores originais dos pesos das *tags* em um agrupamento podem ter ordens de grandeza bastante variadas (exemplo na Tabela 4.4). Simplesmente calcular a diferença entre os pesos originais não permitiria obter novos pesos numa escala linear (por exemplo entre 0 e 1). Para contornar esse problema, lançou-se mão da escala logarítmica, e desta forma o resultado do novo ranqueamento pode ser dado de forma linear.

Primeiro obtem-se a medida da distância entre a *tag* principal e as demais *tags* do agrupamento, calculada usando-se a escala logarítmica dada por $dist_{(t_p,t_1)} = |(\log_2 rank_{t_p}) - (\log_2 rank_{t_1})|$, onde $rank_{t_p}$ e $rank_{t_1}$ são os valores dos pesos originais atribuídos para as *tags* t_p e t_1 respectivamente. Quanto menor o valor resultante $dist$, mais próximas e mais similares são consideradas as *tags*. Portanto a relevância (peso) da *tag* t_1 deve ser inversamente proporcional ao valor da distância desta em relação à *tag* principal t_p, e é dada então pela Equação (4.1). O fator c apenas controla a amplitude da escala, sendo 1 a máxima relevância e c a mínima relevância, que deve ser próxima a zero.

$$rank_{t_1} = 1 - \left(\frac{dist_{(t_p,t_1)}}{1 + c} \right) \tag{4.1}$$

A Tabela 4.4 representa um exemplo desta tarefa. Considere o termo de pesquisa “Sagitário”. A primeira coluna da tabela mostra todas as *tags* do agrupamento ao qual pertence o termo “Sagitário”. A segunda coluna mostra o peso original calculado para as *tags* dentro do agrupamento no processo de formação dos agrupamentos. Já a terceira coluna mostra o peso recalculado das *tags*, usando a Equação (4.1), levando-se em consideração a *tag* principal “Sagitário”.

Uma vez que os pesos das *tags* foram recalculados, podemos finalmente buscar os RE correspondentes. Baseado na ideia do TF, para cada RE encontrado, calcula-se a pontuação do RE pelo somatório dos pesos das *tags* coocorrentes que ela possui: $Rank_{RE} = \sum_{i=1}^{n} rank_i$, onde n é a quantidade de *tags* do *cluster* que o RE possui e $rank_i$ é o peso recalculado da *tag* i. Ao final desta tarefa, são retornados todos os RE encontrados que foram classificados conforme a pontuação total de suas *tags*, e seu conjunto será dado por C_{at}. A Tabela 4.5 traz uma amostra de RE encontrados para a busca pelo termo “Sagitário” via agrupamento de *tags*. Considerando os pesos das *tags* do agrupamento listadas na primeira linha da tabela, as pontuações de cada RE são dadas na primeira coluna da tabela. Por exemplo, para o RE “Estrelas e Diagrama HR”, que

possui as *tags* “Diagrama de Hertzsprung Russell”, “Estrela”, “Estrela anã” e “Estrela luminosa”, o cálculo da sua pontuação é dada por $Rank_{RE} = 0,70 + 0,50 + 0,99 + 0,99 = 3,18$.

Tabela 4.5: Busca pelo termo “Sagitário” via agrupamento de *tags*

Termo pesquisado=“Sagitário” *Tags* coocorrentes (peso): Estrela anã (0,99); Estrela luminosa (0,99); Peony (0,93);Shine (0,93); W5 (0,91); Movimento das estrelas diurnal (0,90);Galáxia Roda de Carroça (0,85); Recycle (0,84); Disco protoplanetário (0,71); Diagrama de Hertzsprung Russell (0,70); Estrela (0,50)		
Pontuação ($Rank_{RE}$)	**Nome RE**	***Tags***
3,1812	Estrelas e Diagrama HR	Educação Superior; Ciências Exatas e da Terra; Astronomia; Astronomia Fundamental; Diagrama de Hertzsprung Russell; Estrela; Estrela anã; Estrela luminosa;
2,3670	Peony star	Educação Superior; Ciências Exatas e da Terra; Astronomia; Astrofísica Estelar; Astronomy; Brilho; Estrela; NASA; Peony; Shine; Star; Universe; Universo;
1,5000	Milky way 2	Educação Básica; Ensino Fundamental Final; Ciências Naturais; Terra e universo; Ensino Médio; Física; Universo, terra e vida; Educação Superior; Ciências Exatas e da Terra; Astronomia; Galáxias; Astronomy; Constelação; Constellation; Espaço; Estrela; Sagitário; Space; Star; Universe; Universo;

4.1.7 Mesclando e recalculando a classificação dos resultados

Partimos então para a última tarefa do grupo **P2 Busca e ranqueamento**. A tarefa **[T7] Mescla e recalcula ranqueamento** recebe como entrada os dois conjuntos de RE, C_{mb} resultado da tarefa **[T6]** e C_{at} resultado da tarefa **[T6’]**. Primeiramente os dois conjuntos passam por uma normalização dos pesos de seus RE, pois os pesos tem dispersões distintas e por isso é necessário ajustar a escala de valores para que seja possível unir os dois conjuntos. Para isso, utiliza-se uma equação baseada na normalização linear. No caso do resultado gerado pelo motor de busca, a normalização da pontuação é dada por (4.2), onde min_score e max_score são respectivamente a pontuação mínima e máxima encontradas no conjunto C_{mb}, d é um coeficiente de ajuste para evitar valores iguais a 0 e *boost* trata-se do fator de impulsão responsável por controlar a relevância dos recursos no processo de ranqueamento (mencionado na Subseção 2.2.2).

$$rank_{mb} = boost \times \left(\frac{score \times (1 + d) - min\_score}{max\_score - min\_score} \right) \tag{4.2}$$

A normalização do segundo conjunto C_{at} gerado com suporte do agrupamento de *tags* é dada por (4.3), onde min_score e max_score são respectivamente a pontuação mínima e máxima encontradas no conjunto C_{at}, d é o coeficiente de ajuste para evitar valores iguais a 0 e max_boost é o valor máximo de *boost* encontrado no conjunto C_{mb}. O maior diferencial desta proposta talvez seja neste ponto, pois aplica-se o valor máximo de boost (max_boost) nesse segundo conjunto, justamente para impulsionar a pontuação dos RE, assim como é feito no motor de busca. Partindo-se da premissa de que os RE do conjunto C_{at} foram retornados somente porque são considerados similares pela correlação entre as *tags* utilizadas, consideramos razoável aplicar max_boost aos RE deste conjunto para que não haja prejuízo na relevância destes em relação aos do conjunto C_{mb}.

$$rank_{at} = max\_boost \times \left(\frac{score \times (1 + d) - min\_score}{max\_score - min\_score} \right) \tag{4.3}$$

Após a normalização das pontuações nos dois conjuntos, o resultado final é dado por C_{mix} que é a união do conjunto C_{mb} e C_{at}. Por último, mas não menos importante, outra medida adotada é embasada na constatação de Morrison (2008): recursos retornados tanto pelo motor de busca como pelo agrupamento de *tags* devem ser considerados mais relevantes em relação aos

demais. Por isso a pontuação final do RE em C_{mix} para RE que ocorre tanto em C_{mb} como em C_{at} é dada pela soma das pontuações que o RE teve em C_{mb} e em C_{at}, ou seja $rank_{mb} + rank_{at}$. Finalmente a tarefa **[T7]** pode retornar o resultado final dado pelo conjunto C_{mix} com os RE reclassificados pela nova pontuação.

A Tabela 4.6 mostra um comparativo das pontuações retornadas das tarefas [T6] e [T6'], bem como a nova pontuação recalculada após a mesclagem dos dois resultados. Considere para este exemplo $boost = 10$, $max\_boost = 10$, $d = 0,05$, $min\_score = 0,8373$ e $max\_score = 3,1812$ para o conjunto C_{at}; $min\_score = 147,6802$ e $max\_score = 9,1596$ para o conjunto C_{mb}. Portanto, para o RE "16289" que foi retornado tanto pelo motor de busca como pelo agrupamento de *tags*, sua pontuação foi calculada como $rank_{at} + rank_{mb} = (10 * (1,5 * (1 + 0,05) - 0,8373)/(3,1812 - 0,8373)) + (10 * (147,6802 * (1 + 0,05) - 9,1596)/(147,6802 - 9,1596)) = 13,68$. Já para o RE "2095" que foi retornado somente pelo agrupamento de *tags*, sua pontuação é dada por $rank_{at} = 10 * (3,1812 * (1 + 0,05) - 0,8373)/(3,1812 - 0,8373) = 10,67$.

Tabela 4.6: Exemplo das pontuações resultantes do motor de busca, via agrupamento de *tags* e o resultado final mesclado

	Agrupamento de *tags*		**Elasticsearch**		**Mescla**	
Classificação	**RE id**	**Pontuação**	**RE id**	**Pontuação**	**RE id**	**Pontuação**
1	2095	3,1812	16289	147,6802	16289	13,6804
2	10667	2,3670	2538	11,1470	2095	10,6786
3	16289	1,5000	1917	10,6484	10667	7,0314
4	10837	1,4114	3091	10,3336	10837	2,7506
5	557	1,4114	17001	9,9435	557	2,7506
6	2642	1,3978	17393	9,9435	2642	2,6897
7	11324	1,3496	3987	9,5448	11324	2,4735
8	7105	1,3373	6078	9,5448	7105	2,4185
9	5482	1,2147	15537	9,5448	5482	1,8693
10	11438	0,8373	9508	9,1596	11438	0,1786

Capítulo 5

Avaliação do modelo proposto

O Portalmec, sistema em desenvolvido no C3SL (Centro de Computação Científica e Software Livre) para o MEC (Ministério da Educação). Trata-se de um portal de objetos educacionais em nuvem e fornece mecanismos de rede social, envio, busca e ranqueamento de objetos, bem como mecanismos para seguir coleções e autores.

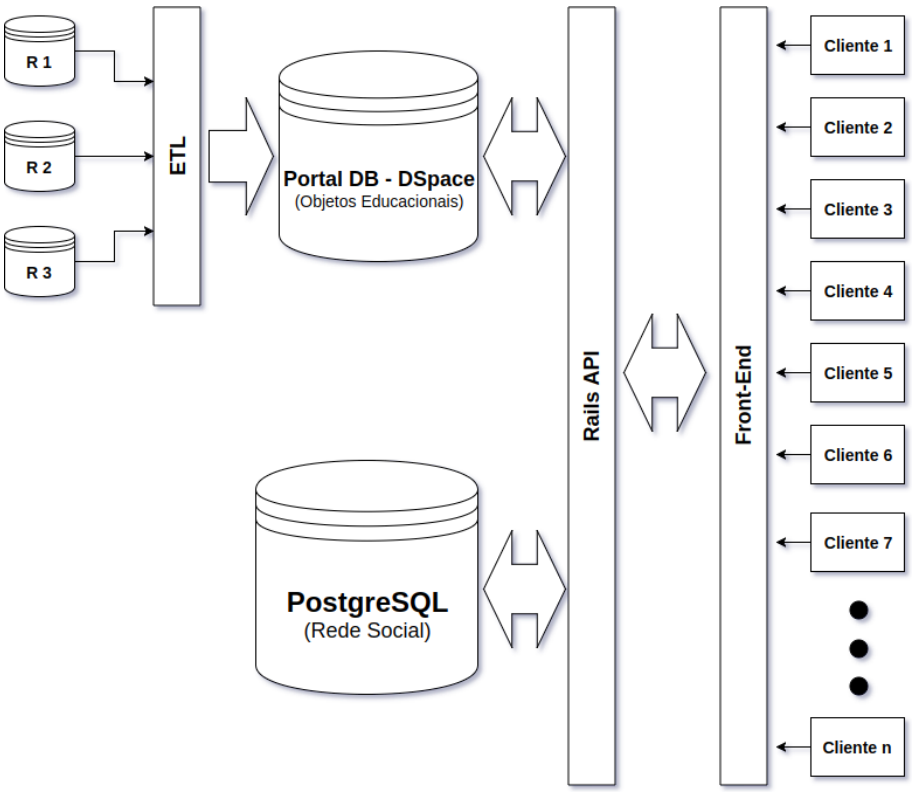

Figura 5.1: Componentes do Portalmec

Para o desenvolvimento do Portalmec, são adotados somente softwares livres e com códigos abertos, de modo a garantir a disseminação do conhecimento produzido e a possibilidade de cooperação na construção da própria plataforma. A figura 5.1 representa a arquitetura construída para implementação do Portalmec. A camada ETL (*Extract, Transform, Load*) permite que RE possam ser trazidos de outros repositórios. Os repositórios de RE são armazenados localmente em um banco de dados DSpace. A *API Rails* gerencia e permite acesso aos conteúdos do DSpace e do banco de dados PostgreSQL que armazena informações referentes à rede social. As buscas no Portalmec são realizadas pelo motor de busca Elasticsearch. A camada *Front-End*

recebe as conexões dos usuários e se comunica com a API Rails, repassando as informações necessárias aos usuários.

A Figura 5.2 traz a tela inicial do Portalmec; a Figura 5.3 mostra os resultados de uma busca realizada no Portalmec; e a Figura 5.4 mostra os detalhes de um RE no Portalmec.

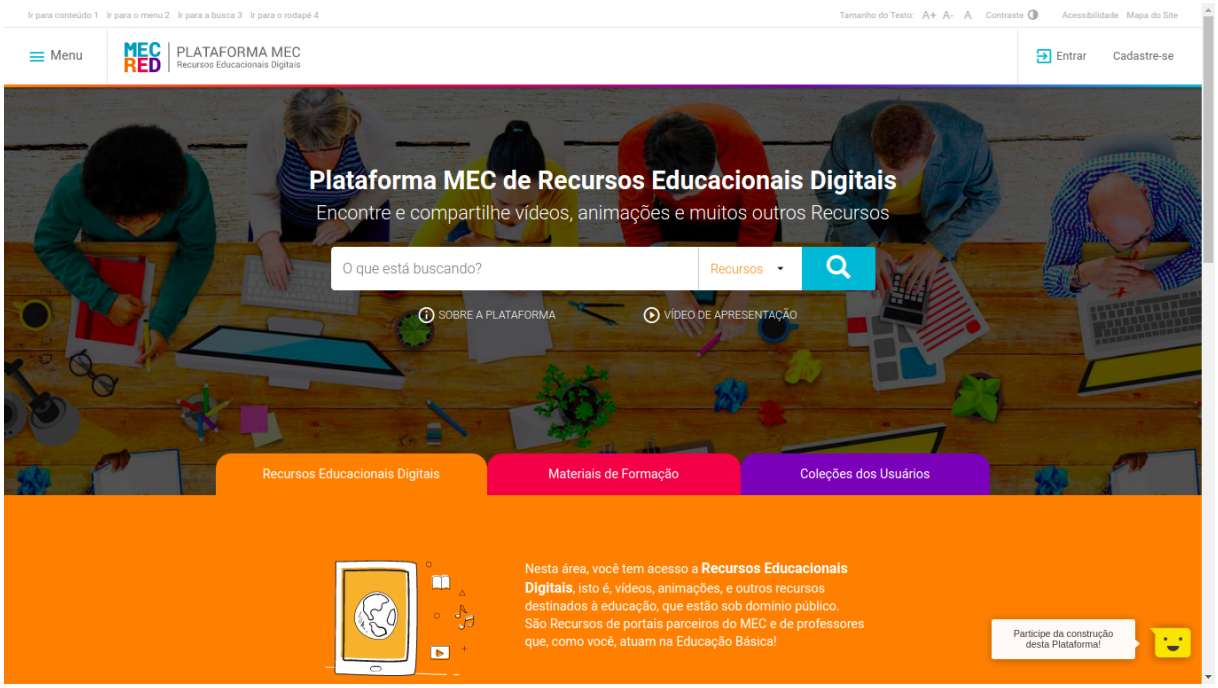

Figura 5.2: Página inicial do Portalmec

Com o objetivo de avaliar o modelo de busca e ranqueamento de RE proposto no Capítulo 4, foi utilizada a mesma infraestrutura e base de dados utilizados pelo sistema Portalmec.

Considerando que o Portalmec até a data de execução dos experimentos não havia sido lançado, há de se considerar o problema conhecido como *cold start* Bobadilla et al. (2012); Kim et al. (2010). Este problema resumidamente ocorre quando não há dados prévios sobre usuários, recursos ou comunidades, afetando principalmente os sistemas de recomendação. Em geral esse problema ocorre com novos usuários que se cadastram em um sistema, ou com recursos/objetos recém criados ou com comunidades novas.

Em particular para o presente trabalho, o problema se concentra especificamente sobre os RE. Vale ressaltar que o Portalmec recebeu uma carga com RE coletados de outros portais MEC já existentes. Para contornar o problema de *cold start* em relação à ausência de *tags* atribuídas pelos usuários aos RE, todas as informações disponíveis nos metadados palavra-chave e assunto de cada RE foram transformadas em *tags*, tal que o repositório do Portalmec passou a contar com 19.159 RE e 23.808 *tags*, suficientes para a realização dos experimentos para a avaliação do modelo proposto neste trabalho.

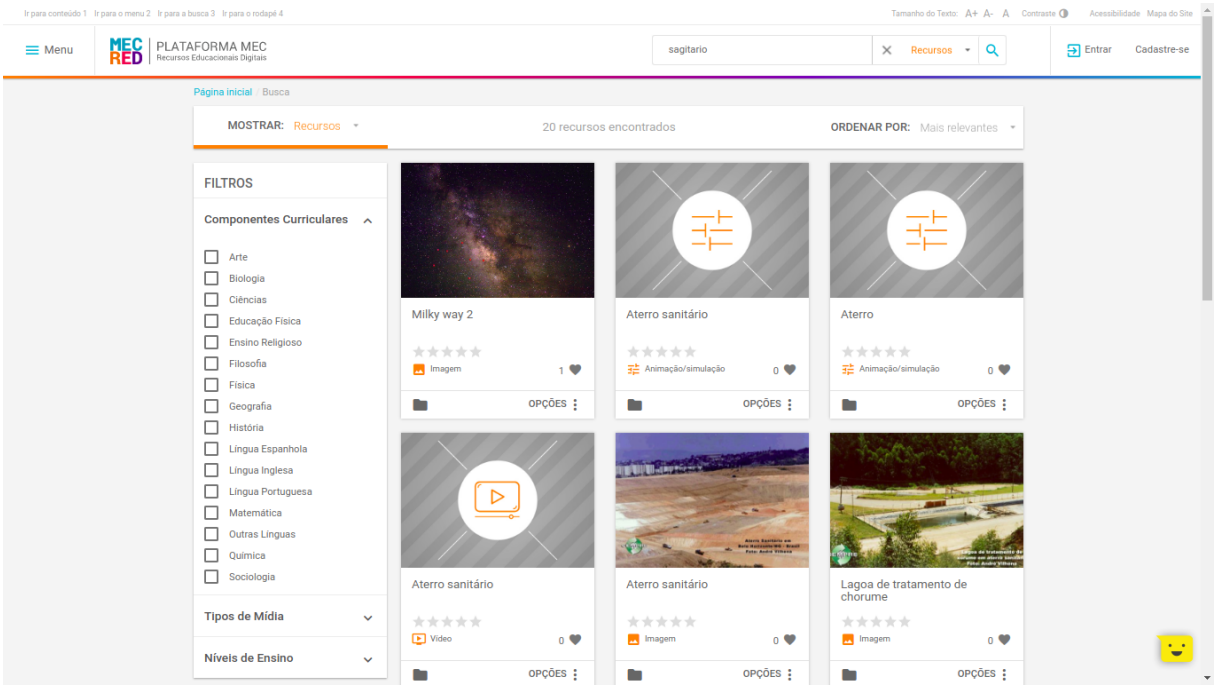

Figura 5.3: Página com resultados de uma busca no Portalmec

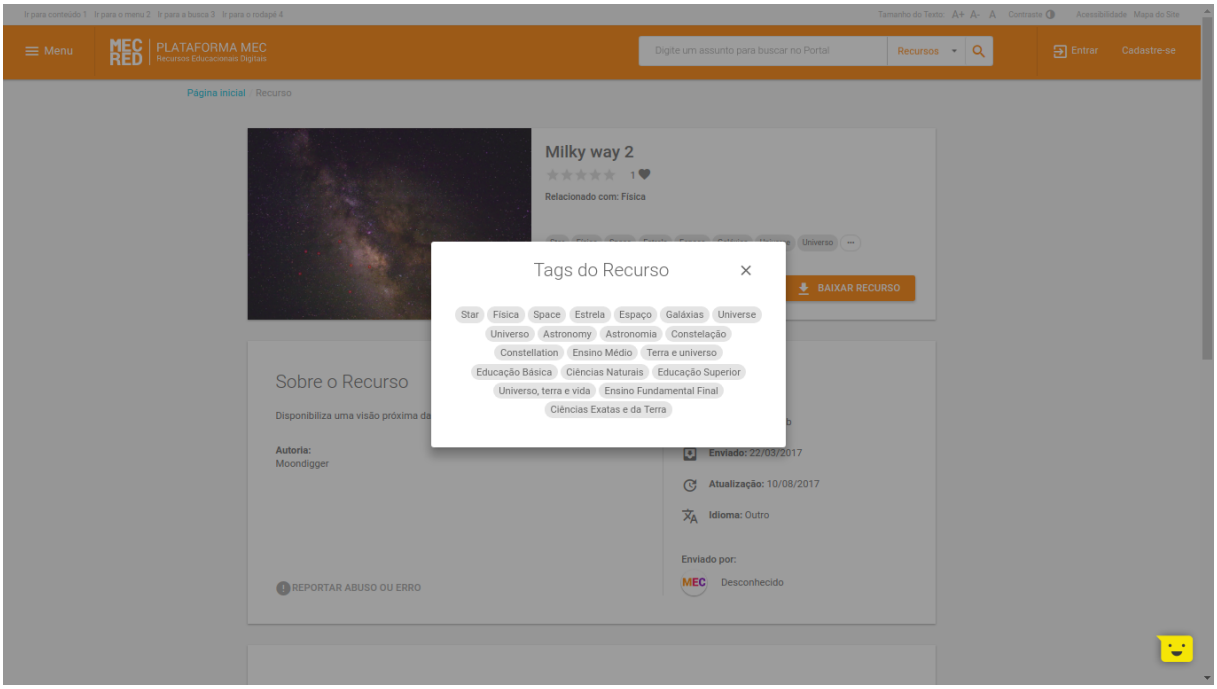

Figura 5.4: Página com detalhes de uma RE no Portalmec

A Seção 5.2 detalha os experimentos de busca e ranqueamento de RE realizados pela execução do sistema implementado conforme modelo proposto para Busca e Ranqueamento de RE com suporte no Agrupamento de *Tags* descrito no Capítulo 4.

5.1 Implementação

Para viabilizar a implementação do modelo proposto no contexto do Portalmec, foi desenvolvido um sistema que possibilita a execução de todas as tarefas do grupo **P1-Agrupamento** e do grupo **P2-Busca e ranqueamento**. Os programas foram desenvolvidos na linguagem de programação *Ruby*, utilizando o *framework Rails* com a gema (biblioteca em Ruby) *Searchkick* para realizar a comunicação com o *Elasticsearch*, e também foi utilizado o algoritmo Infomap Bohlin et al. (2014) para realizar o agrupamento das *tags*. Os arquivos de dados e resultados bem como os códigos-fonte deste trabalho podem ser acessados no repositório Gitlab do C3SL[1]. Os detalhes da implementação são descritos na sequência.

A tarefa **[T1] Recuperação da lista de RE e suas tags** busca todos os RE e suas respectivas *tags* do repositório de RE do Portalmec, armazenadas no Sistema de Gerenciamento de Banco de Dados *PostgreSQL*. Tendo a lista dos RE com suas *tags* é possível executar as tarefas **[T2] Mapeamento da coocorrência das *tags*** e **[T3] Geração do grafo**. Para possibilitar a geração dos agrupamentos de *tags*, tarefa **[T4] Cálculo e formação dos agrupamentos**, adotamos o *framework* Mapequation Bohlin et al. (2014). Trata-se de um conjunto de ferramentas para formação e visualização de agrupamentos. A partir do grafo obtido na saída da tarefa **[T3]**, representado num formato específico denominado *PAJEK* (.net), utilizamos a ferramenta *Infomap* para calcular e gerar os agrupamentos de *tags*. Além de gerar os agrupamentos, *Infomap* gera um ranqueamento das *tags* dentro de cada agrupamento formado. O resultado deste processo é gravado num arquivo de saída (.ftree), onde são armazenadas informações sobre todos os agrupamentos formados, com suas respectivas *tags* e pontuações, sendo assim a base de informação para apoiar a tarefa **[T6'] Busca via agrupamento de tags**.

```
*Vertices 23807
15 "Iniciação científica"
16 "Laboratório"
17 "Pesquisa científica"
19 "Medicina"
...
1442 "Taxonomia"
1443 "Teorema de Pitágoras"
1448 "Dinâmica"
1449 "Princípio de Bernoulli"
1450 "Tensão superficial"
1451 "Compostos químicos"
1453 "Oxigênio"
1454 "Figuras geométricas planas"
...
*Edges
1 25 0.8548921527895664
1 3 0.4119236190465282
1 1233 0.008972309321891575
1 215 0.3237898708510765
...
13872 13874 1.0
146 147 0.2581988897471611
146 148 0.23570226039551587
13645 13646 0.7071067811865475
20732 20734 1.0
...
```

Figura 5.5: Trecho do arquivo *tags.net*

As Figuras 5.5 e 5.6 representam uma amostra dos arquivos (.net) e (.ftree), respectivamente.

[1]https://gitlab.c3sl.ufpr.br/portalmec/tag-clustering

```
# '--ftree /home/portalmec/portalmec/tmp/tags.net /home/portalmec/portalmec/tmp'
-> 23807 nodes and 173038 links partitioned in 3s from codelength 13.755221776
in one level to codelength 6.251978645 in 5 levels.
# path flow name node:
1:1:1 0.00306486 "Educação Básica" 1
1:1:2 0.00258015 "Ensino Médio" 25
1:1:3 0.00165008 "Ensino Fundamental Final" 2
1:1:4 0.00121506 "Química" 204
1:1:5 0.000714053 "Transformações: caracterização, aspectos energéticos,
aspectos dinâmicos" 205
...
1:3:1:2 0.000122887 "separação do lixo" 10524
1:3:1:3 0.000110899 "resíduo radioativo" 16370
1:3:1:4 0.000110899 "resíduos biológicos" 16371
1:3:1:5 0.000110899 "resíduos químicos" 16372
1:3:1:6 0.000110899 "aterro industrial" 16365
1:3:1:7 0.000110899 "destinação do lixo" 16366
1:3:1:8 0.000110899 "incineração do lixo" 16367
...
*Links undirected
#*Links path exitFlow numEdges numChildren
*Links root 0.0 1072 603
1 6 0.00155962
1 2 0.00145746
1 3 0.00138853
1 5 0.00115087
1 4 0.000980317
1 8 0.000825044
3 5 0.000739728
1 19 0.000724155
3 14 0.000715477
3 8 0.00066951
3 11 0.000629576
9 10 0.000616566
3 9 0.000610781
...
```

Figura 5.6: Trecho do arquivo *tags.ftree*

Do total de 23.807 *tags* (nós) e 173.038 coocorrências[2] (*links*), foram gerados a partir da execução do *Infomap* 8.568 agrupamentos. Para permitir uma melhor visualização dos agrupamentos formados com a execução da etapa **[T4]-Cálculo e formação dos agrupamentos** utilizamos as ferramentas disponíveis no *framework Map Equation* para gerar amostras representativas dos agrupamentos. As Figuras 5.7, 5.8 e 5.9 apresentam amostras dos agrupamentos com foco geral, em educação básica e matemática respectivamente.

Desta forma, todos os processos do grupo **P1-Agrupamento** foram viabilizados formando os agrupamentos de *tags*. Podemos então focar nos processos do grupo **P2-Busca e ranqueamento**. A tarefa **[T5]** é viabilizada fornecendo-se o termo de pesquisa como argumento para o sistema desenvolvido. O motor de busca utilizado no Portalmec é o Elasticsearch, por isso para executar a tarefa **[T6] Busca via motor de busca** utilizamos a biblioteca *Searchkick* para fazer a comunicação do sistema com o Elasticsearch. Desta forma é possível obter a lista de RE recuperados pelo motor de busca. Neste ponto ressaltamos a utilização do parâmetro *explain* na execução do Elasticsearch (conforme explicado no Capítulo 2 Seção 2.2.2). Com isso é possível capturar o valor de *boost* aplicado pelo Elasticsearch no processo de ranqueamento dos RE retornados. O valor máximo de *boost* é utilizado posteriormente para que seja possível também impulsionar os RE encontrados via agrupamento de *tags*.

[2]Número de vezes que um par de *tags* é utilizado para descrever um mesmo RE.

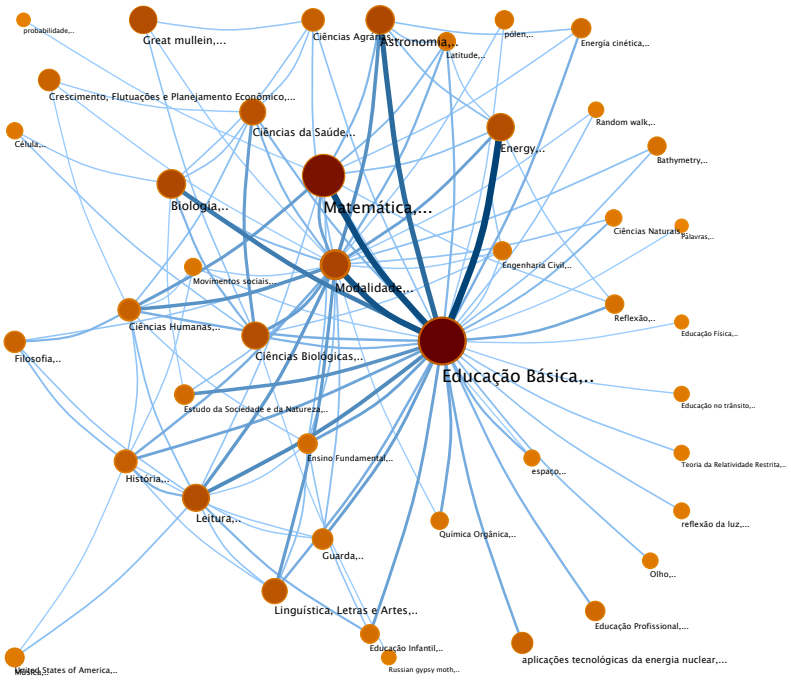

Figura 5.7: Agrupamentos das *tags* dos recursos educacionais do Portalmec

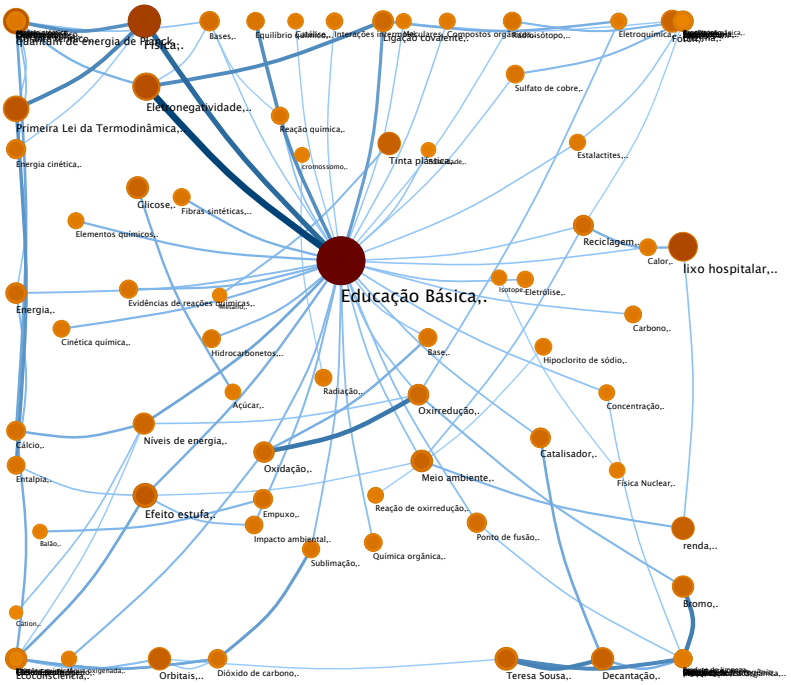

Figura 5.8: Agrupamentos das *tags* do Portalmec, com foco no grupo Educação Básica

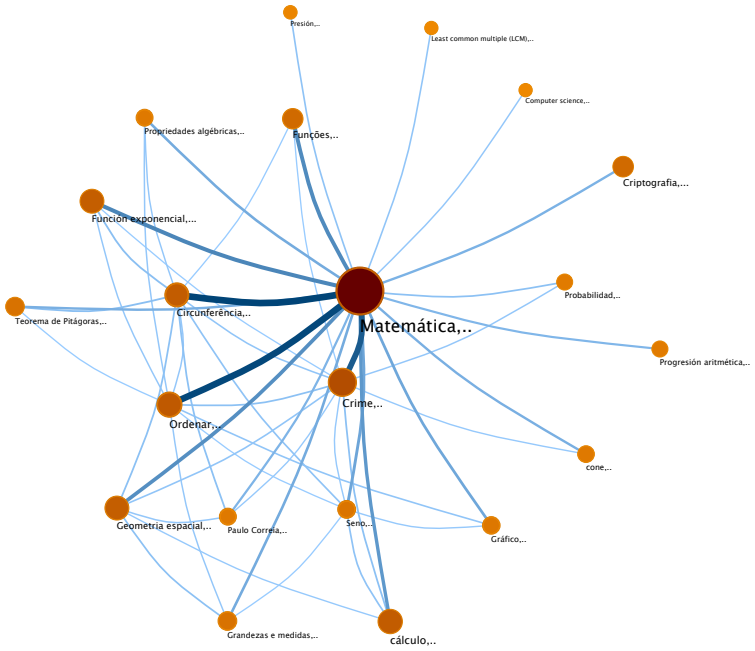

Figura 5.9: Agrupamentos das *tags* do Portalmec focando no grupo Matemática

As tarefas **[T6'] Busca via agrupamento de *tags*** e **[T7] Mescla resultados** foram implementadas no sistema viabilizando a recuperação de RE via agrupamento de *tags* disponibilizado na tarefa **[T4]**, possibilitando ao final produzir uma lista ranqueada de RE considerados correspondentes ao termo pesquisado.

5.2 Experimentos

Os experimentos foram realizados em uma máquina com um processador AMD FX(tm)-6300 Six-Core Processor 3.5GHz, memória RAM DIMM DDR3 8GB 667MHz, disco rígido SATA 320GB 7200RPM, rodando o Sistema Operacional Ubuntu GNOME 16.04.

Os dados necessários para realização dos testes foram coletados do repositório do Portalmec, totalizando uma massa de dados com 19.159 RE e 23.808 *tags*.

Foi realizado um conjunto de buscas, e para cada termo de busca utilizado no experimento tem-se 4 tabelas que apresentam: 1) lista de RE retornados pelo agrupamento de *tags*; 2) lista de RE retornados pelo motor de busca Elasticsearch; 3) resultado final com RE mesclados e reclassificados originados de 1) e 2); e 4) resumo comparativo da ordem de classificação final dos RE e em cada uma das buscas (via motor de busca e agrupamento de *tags*).

Os experimentos foram realizados com um conjunto de termos ou *tags* escolhidos aleatoriamente, a partir das quais foram realizadas as buscas por RE, buscando resultados por meio do motor de busca Elasticsearch e também por meio do algoritmo de busca baseado no agrupamento de *tags*. Como já afirmando por Hotho et al. (2006a), não existe ranqueamento ou classificação padrão-ouro neste tipo de busca, por isso os resultados foram avaliados empiricamente.

Na Tabela 5.1 pode-se ver uma pequena amostragem das informações obtidas do agrupamento de *tags*. Para cada *tag* dentro do *cluster* é atribuído um peso, que é utilizado para o cálculo do ranqueamento dos RE relacionados a estas *tags*. Pelos poucos exemplos, pode-se constatar a variedade de termos que podem ser ampliados numa busca quando se expande os

termos correlacionados fornecidos pelos agrupamentos de *tags*, mostrando assim que a ampliação semântica dada pelo modelo proposto é viável.

Tabela 5.1: Exemplos de resultados de *tags* correlacionadas nos agrupamentos

Termo	***Tags* coocorrentes (peso)**
Sagitário	Estrela anã (0,99); Estrela luminosa (0,99); Peony (0,93); Shine (0,93); W5 (0,91); Movimento das estrelas diurnal (0,90); Galáxia Roda de Carroça (0,85); Recycle (0,84); Disco protoplanetário (0,71); Diagrama de Hertzsprung Russell (0,70); Estrela (0,50)
DNA	RNA(0,91); Guanina (0,82); Nucleotídeo (0,77); Nucleosídeo(0,67); Tradução protéica (0,62); enzymes restriction endonucleases(0,62); Impuesto Sobre El Valor Añadido (IVA) (0,61); Proteína homóloga(0,61); Ácido nucleico (0,59); Dupla-hélice (0,54); Armadeira (0,54); Capsídeo de proteínas (0,50); Guanine(0,82); Thymine(0,82); Timine (0,82);
Força gravitacional	Modelo de Ptolomeu (1,00); Movimento retrogrado (1,00); Posição dos planetas (0,94); Epiciclo (0,94); Deferente (0,87); Luminosidad (0,66); Einstein (0,64); Circunferência maior (0,50)
Descobrimento do Brasil	Pau-Brasil(0,76); Tupinambá (0,76); História indígena (0,5)
Corrosão	Eletroquímica (0,917); Óxido-redução (0,95); pilha de concentração (0,5)

Dentre os experimentos realizados, citam-se os experimentos que representam resultados com características distintas entre si, para que seja possível analisar os resultados obtidos pela aplicação do modelo proposto.

Os experimentos com o termo "Sagitário" mostram de maneira mais acentuada os ganhos trazidos pela abordagem proposta. A Tabela 5.2 mostra os resultados obtidos pela busca realizada por meio do agrupamento de *tags*. Lista todas as *tags* pertencentes ao mesmo agrupamento do termo de busca original e que foram utilizadas para buscar os RE, bem como os pesos dos termos do agrupamento. São também apresentados os dez RE melhor ranqueados, suas pontuações e o conjunto de *tags* que cada RE possui. A Tabela 5.3 mostra a relação dos RE retornados pelo motor de busca ranqueados com suas devidas pontuações. A Tabela 5.4 mostra um resumo dos resultados via agrupamento de *tags*, via motor de busca e também o resultado final após a mesclagem dos RE que são reclassificados baseado em seus novos pesos recalculados. A Tabela 5.5 apresenta o resultado final de outra forma, trazendo desta vez o nome do RE e resume a posição em que o mesmo apareceu na busca via agrupamento de *tags* (At), via motor de busca (E) e sua classificação final após mesclagem (M). Esta interpretação é válida para os demais experimentos, onde somente os termos pesquisados são distintos.

Neste experimento com o termo "Sagitário" pode-se verificar que a ampliação da busca realizada via agrupamento de *tags* aumenta de forma significativa o resultado final da busca com resultados relacionados ao termo pesquisado. Na busca via Elasticsearch a maioria dos resultados encontrados referem-se ao termo "Sanitário", trazendo RE não relacionados com o termo de busca. Dentre os dez resultados obtidos pelo motor de busca, somente um, "Milk way 2", pode ser considerado relacionado ao termo de busca.

Tabela 5.2: Resultados da busca pelo termo “Sagitário” via Agrupamento de *Tags*

Busca via Agrupamento de tags termo = Sagitário			
Tags coocorrentes (peso): Estrela anã (0,99); Estrela luminosa (0,99); Peony (0,93); Shine (0,93); W5 (0,91); Movimento das estrelas diurnal (0,90); Galáxia Roda de Carroça (0,85); Recycle (0,84); Disco protoplanetário (0,71); Diagrama de Hertzsprung Russell (0,70); Estrela (0,50)			
RE id	**Peso**	**Nome RE**	**Tags**
2095	3,1812	Estrelas e Diagrama HR	Educação Superior; Ciências Exatas e da Terra; Astronomia; Astronomia Fundamental; Diagrama de Hertzsprung Russell; Estrela; Estrela anã; Estrela luminosa;
10667	2,3670	Peony star	Educação Superior; Ciências Exatas e da Terra; Astronomia; Astrofísica Estelar; Astronomy; Brilho; Estrela; NASA; Peony; Shine; Star; Universe; Universo;
16289	1,5000	Milky way 2	Educação Básica; Ensino Fundamental Final; Ciências Naturais; Terra e universo; Ensino Médio; Física; Universo, terra e vida; Educação Superior; Ciências Exatas e da Terra; Astronomia; Galáxias; Astronomy; Constelação; Constellation; Espaço; Estrela; Sagitário; Space; Star; Universe; Universo;
10837	1,4114	W5 (Allen)	Educação Superior; Ciências Exatas e da Terra; Astronomia; Astrofísica Estelar; Ciência; Estrela; Universo; W5;
557	1,4114	W-5 Star-Forming Region	Educação Superior; Ciências Exatas e da Terra; Astronomia; Astrofísica Estelar; Estrela; NASA; Star; W5;
2642	1,3978	O Movimento diurno das estrelas criado pela rotação da Terra	Educação Superior; Ciências Exatas e da Terra; Astronomia; Astronomia Fundamental; Estrela; Movimento; Movimento das estrelas diurnal; Planeta; Terra;
11324	1,3496	Cartwheel galaxy	Educação Superior; Ciências Exatas e da Terra; Astronomia; Galáxias; Estrela; Galáxia; Galáxia Roda de Carroça; Universo;
7105	1,3373	Robot Astronomy Talk Show 6	Educação Superior; Ciências Exatas e da Terra; Astronomia; Astronomia Espacial; Astronomy; Estrela; Lixo espacial; Reciclagem; Recycle; Space trash; Star; Supernova; Universe; Universo;
5482	1,2147	Inner Gap in Circumstellar Disk	Educação Superior; Ciências Exatas e da Terra; Astronomia; Astronomia Espacial; Disco protoplanetário; Estrela;
11438	0,8373	Space Trash	Educação Básica; Ensino Fundamental Final; Ciências Naturais; Terra e universo; Vida e ambiente; Ensino Fundamental Inicial; Ambiente; Ensino Fundamental; Séries Finais; Astronauts; NASA Connect; Open Video; Recycle; Reduce; Space Trash; Trash Cans;

Na busca via agrupamento de *tags*, pode-se considerar que entre os dez resultados obtidos, nove são relacionados ao termo pesquisado. O fato de dar relevância aos termos expandidos, ou seja, os que são correlacionados com as *tags* coocorrentes, faz com que os RE retornados pelo motor de busca que não são relevantes ao termo “Sagitário” efetivamente percam a relevância na classificação final quando são mesclados a resultados mais relevantes trazidos pelo agrupamento de *tags*. Isso é viável porque o próprio motor de busca, ao trazer RE pouco relevantes, atribui um valor de *boost* próximo a 1 ao campo pesquisado, o que significa pouca relevância para o Elasticsearch, sendo que para campos considerados relevantes, o valor de *boost* normalmente é igual a 10. No resultado final pode-se considerar que dos dez resultados, nove podem ser considerados relevantes à busca. Desta forma, os resultados mostram que a abordagem proposta neste trabalho melhora o resultado final retornando RE relevantes ao termo pesquisado. Somente com a utilização do motor de busca, temos 10% de resultados relevantes, e com a abordagem proposta neste trabalho o percentual de RE relevantes sobe para 90%.

Tabela 5.3: Resultados da busca pelo termo “Sagitário” via Elasticsearch

Busca via Elasticsearch termo = Sagitário		
RE id	**Peso**	**Nome RE**
16289	147,6802	Milky way 2
2538	11,1470	Aterro sanitário
1917	10,6484	Almanaque Sonoro de Química - Química na Agricultura - Parte 3
3091	10,3336	Conversa Periódica - Perigos do Lixo
17001	9,9435	Aterro sanitário
17393	9,9435	Aterro sanitário
3987	9,5448	Minas sem lixões - Parte 2
6078	9,5448	Lagoa de tratamento de chorume
15537	9,5448	Minas sem lixões - Parte 3
9508	9,1596	Tratamento de chorume

Tabela 5.4: Comparativo das pontuações e classificações da busca por “Sagitário”

Termo = Sagitário						
	Agrup. Tags (At)		**Elasticsearch (E)**		**Mescla (M)**	
Classificação	**RE id**	**Pontuação**	**RE id**	**Pontuação**	**RE id**	**Pontuação**
1	2095	3,1812	16289	147,6802	16289	13,6804
2	10667	2,3670	2538	11,1470	2095	10,6786
3	16289	1,5000	1917	10,6484	10667	7,0314
4	10837	1,4114	3091	10,3336	10837	2,7506
5	557	1,4114	17001	9,9435	557	2,7506
6	2642	1,3978	17393	9,9435	2642	2,6897
7	11324	1,3496	3987	9,5448	11324	2,4735
8	7105	1,3373	6078	9,5448	7105	2,4185
9	5482	1,2147	15537	9,5448	5482	1,8693
10	11438	0,8373	9508	9,1596	11438	0,1786

Tabela 5.5: Resumo comparativo da busca pelo termo “Sagitário”

		Classificação		
RE id	**Nome RE**	**(At)**	**(E)**	**(M)**
16289	Milky way 2	3	1	1
2095	Estrelas e Diagrama HR	1	-	2
10667	Peony star	2	-	3
10837	W5 (Allen)	4	-	4
557	W-5 Star-Forming Region	5	-	5
2642	O Movimento diurno das estrelas criado pela rotação da Terra	6	-	6
11324	Cartwheel galaxy	7	-	7
7105	Robot Astronomy Talk Show 6: Sculpting With Stars	8	-	8
5482	Inner Gap in Circumstellar Disk	9	-	9
11438	Space Trash	10	-	10

As Tabelas 5.6, 5.7, 5.8 e 5.9 mostram os resultados do experimento de busca realizado com o termo "Força gravitacional". Neste experimento, dois dos dez RE na classificação final são trazidos do agrupamento de *tags*. Três (RE id: 8958, 8214 e 14722) foram retornados por ambos (motor de busca e agrupamento de *tags*), fazendo com que itens que tinham sido melhores classificados nas duas abordagens separadas perdessem a relevância perante esses três no ranqueamento final.

Tabela 5.6: Resultados da busca pelo termo "Força gravitacional" via Elasticsearch

Busca via Elasticsearch termo = Força gravitacional		
RE id	**Peso**	**Nome RE**
17961	222,6133	Mecânica - Gravidade
15426	222,0840	Orbits around and through a Sphere
13975	213,6619	Mecânica - Lei da conservação da energia mecânica
3282	205,7784	Queda de moedas
10198	195,9963	Campo elétrico e gravitacional
8214	194,7455	Gravitação
14722	193,6935	Comparando tempos de queda
5047	191,8774	Balanço impossível
4772	188,2813	Módulo de pouso lunar
8958	187,1566	A física e o cotidiano: Gavitação - Parte I

Vale aqui ressaltar que sem a análise de especialistas no assunto do termo pesquisado é difícil ponderar se o resultado final é melhor do que os resultados obtidos somente pelo motor de busca, pois neste exemplo são diferenças muito sutis entre os dois resultados. É possível afirmar somente que o que foi proposto no modelo está sendo aplicado na implementação, pois resultados que aparecem em ambas as buscas recebem maior destaque na composição do resultado final.

Tabela 5.7: Resultados da busca pelo termo “Força gravitacional” via Agrupamento de *Tags*

Busca via Agrupamento de tags termo = Força gravitacional			
Tags coocorrentes (peso): Modelo de Ptolomeu (1,00); Movimento retrogrado (1,00); Posição dos planetas (0,94); Epiciclo (0,94); Deferente (0,87); Luminosidad (0,66); Einstein (0,64); Circunferência maior (0,50)			
RE id	**Pontuação**	**Nome RE**	**Tags**
14641	5,2314	Modelo de Ptolemeu para um planeta inferior	Educação Superior; Ciências Exatas e da Terra; Astronomia; Sistema Planetário; Circunferência maior; Deferente; Epiciclo; Modelo de Ptolomeu; Movimento retrogrado; Posição dos planetas;
15850	4,7314	Modelo de Ptolomeu para um planeta superior	Educação Superior; Ciências Exatas e da Terra; Astronomia; Sistema Planetário; Deferente; Epiciclo; Modelo de Ptolomeu; Movimento retrogrado; Planeta; Posição dos planetas;
16374	2,5764	A física e o cotidiano: Gavitação - Parte III	Educação Básica; Ensino Médio; Física; Universo, terra e vida; Cometa; Copérnico; Einstein; Elipse; Epiciclo; Força gravitacional; Gravidade; Gravitação; Kepler; Leis de Kepler; Newton; Órbita;
5864	2,5764	A física e o cotidiano: Gravitação - Parte II	Educação Básica; Ensino Médio; Física; Universo, terra e vida; Cometa; Einstein; Elipse; Epiciclo; Força gravitacional; Gravidade; Gravitação; Leis de Kepler; Newton; Órbita;
18197	2,5764	A física e o cotidiano: Gravitação - Parte IV	Educação Básica; Ensino Médio; Física; Universo, terra e vida; Cometa; Copérnico; Einstein; Elipse; Epiciclo; Força gravitacional; Gravidade; Gravitação; Kepler; Leis de Kepler; Órbita;
8958	2,5764	A física e o cotidiano: Gavitação - Parte I	Educação Básica; Ensino Médio; Física; Universo, terra e vida; Cometa; Copérnico; Einstein; Elipse; Epiciclo; Força gravitacional; Gravidade; Gravitação; Kepler; Leis de Kepler; Newton; Órbita;
15470	1,2961	Representación de una historia y su cono de luz [Conceptos de relatividad]	Educação Básica; Ensino Médio; Física; Movimento, variações e conservações; Diagrama; Einstein; Espacio; Luminosidad; Movimiento;
8214	1,0000	Gravitação	Educação Básica; Ensino Médio; Física; Universo, terra e vida; Força gravitacional; Gravidade;
14722	1,0000	Comparando tempos de queda	Educação Básica; Ensino Fundamental Final; Ciências Naturais; Vida e ambiente; Aceleração da gravidade; Força gravitacional; Massa; Resistência do ar;
16139	1,0000	Queda livre	Educação Básica; Ensino Médio; Física; Movimento, variações e conservações; Educação Superior; Ciências Exatas e da Terra; Relatividade e Gravitação; Força gravitacional; Movimento uniforme variável; Queda livre; Resistência do ar;

Tabela 5.8: Comparativo das pontuações e classificações da busca por “Força gravitacional”

Termo = Força gravitacional						
	Agrup. Tags (At)		**Elasticsearch (E)**		**Mescla (M)**	
Classificação	**RE id**	**Pontuação**	**RE id**	**Pontuação**	**RE id**	**Pontuação**
1	14641	5,2314	17961	222,6133	17961	13,1392
2	15850	4,7314	15426	222,0840	15426	12,9825
3	16374	2,5764	13975	213,6619	14641	10,6182
4	5864	2,5764	3282	205,7784	13975	10,4884
5	18197	2,5764	10198	195,9963	15850	9,3774
6	8958	2,5764	8214	194,7455	3282	8,1538
7	15470	1,2961	14722	193,6935	8958	6,6692
8	8214	1,0000	5047	191,8774	10198	5,2570
9	14722	1,0000	4772	188,2813	8214	5,0047
10	16139	1,0000	8958	187,1566	14722	4,6932

Tabela 5.9: Resumo comparativo da busca pelo termo “Força gravitacional”

		Classificação		
RE id	**Nome RE**	**(At)**	**(E)**	**(M)**
17961	Mecânica – Gravidade	-	1	1
15426	Orbits around and through a Sphere	-	2	2
14641	Modelo de Ptolemeu para um planeta inferior	1	-	3
13975	Mecânica - Lei da conservação da energia mecânica	-	3	4
15850	Modelo de Ptolomeu para um planeta superior	2	-	5
3282	Queda de moedas	-	4	6
8958	A física e o cotidiano: Gavitação - Parte I	6	10	7
10198	Campo elétrico e gravitacional	-	5	8
8214	Gravitação	8	6	9
14722	Comparando tempos de queda	9	7	10

As Tabelas 5.10, 5.11, 5.12 e 5.13 mostram os resultados do experimento de busca realizado com o termo “DNA”. Neste experimento o próprio motor de busca retorna muitos resultados relevantes. Ainda assim três dos dez RE do resultado final são RE que não foram encontrados pelo motor de busca e foram retornados via agrupamento de *tags*. Neste caso pode-se considerar que houve uma diversificação de resultados devido à expansão da consulta com termos correlacionados. Mesmo o motor de busca retornando muitos RE relevantes, a mesclagem conseguiu ressaltar os RE que continham muitas *tags* correlacionadas, pois na mesclagem os termos correlacionados também recebem a impulsão, como é feito pelo Elasticsearch para o cálculo do ranqueamento final.

Tabela 5.10: Resultados da busca pelo termo “DNA” via Agrupamento de *Tags*

Busca via Agrupamento de tags termo = DNA			
Tags coocorrentes (peso): RNA (0,91); Guanine (0,82); Thymine (0,82); Timine (0,82); Guanina (0,82); Nucleotídeo (0,77); Nucleosídeo (0,67); Tradução protéica (0,62); enzymes restriction endonucleases (0,62); Impuesto Sobre El Valor Añadido (IVA) (0,61); Proteína homóloga (0,61); Ácido nucleico (0,59); Dupla-hélice (0,54); Armadeira (0,54); Capsídeo de proteínas (0,50)			
RE id	**Peso**	**Nome RE**	**Tags**
6681	4,2880	DNA simples 2	Educação Básica; Ensino Fundamental Final; Ciências Naturais; Ser humano e saúde; Ensino Médio; Biologia; Identidade dos seres vivos; Transmissão da vida, ética e manipulação genética; Adenina; Adenine; Citosina; Cytosine; DNA; Genetics; Guanina; Guanine; Thymine; Timine;
4137	3,3455	Composição dos ácidos nucléicos	Educação Superior; Ciências Biológicas; Biologia Geral; Base nitrogenada; Carboidrato; DNA; Nucleosídeo; Nucleotídeo; RNA;
16004	2,7298	Guanina	Educação Básica; Ensino Médio; Biologia; Moléculas, células e tecidos; Química; Modelos de constituição: substâncias, transformações químicas; Educação Superior; Ciências Exatas e da Terra; Estrutura, Conformação e Estereoquímica; DNA; Estrutura química; Guanina; RNA;
17631	2,6763	Resumo sobre origem da vida	Educação Básica; Ensino Médio; Biologia; Origem e evolução da vida; Biodiversidade; DNA; Macromolécula; Nucleotídeo; Organismo; RNA; Ácido nucléico;
13756	2,5321	Tabela do código genético	Educação Superior; Ciências Biológicas; Genética; Genética Humana e Médica; DNA; RNA; Tradução protéica;
4969	2,5176	A sequência de bases determina a constituição das proteínas	Educação Básica; Ensino Médio; Biologia; Origem e evolução da vida; Transmissão da vida, ética e manipulação genética; DNA; Proteína homóloga; RNA;
7318	2,4079	Cytomegalovirus Scheme (CMV)	Educação Básica; Ensino Médio; Biologia; Diversidade da vida e hereditariedade; Capsídeo de proteínas; DNA; RNA; Vírus;
3114	1,9079	DNA words are three letters long	Educação Superior; Ciências Biológicas; Genética; Genética Molecular e de Microorganismos; DNA; RNA; genetics;
12242	1,9079	Adenina	Educação Básica; Ensino Médio; Biologia; Moléculas, células e tecidos; Química; Modelos de constituição: substâncias, transformações químicas; Educação Superior; Ciências Exatas e da Terra; Estrutura, Conformação e Estereoquímica; Adenina; DNA; Estrutura química; RNA;
18329	1,9079	Citosina	Educação Básica; Ensino Médio; Biologia; Moléculas, células e tecidos; Química; Modelos de constituição: substâncias, transformações químicas; Educação Superior; Ciências Exatas e da Terra; Estrutura, Conformação e Estereoquímica; Citosina; DNA; Estrutura química; RNA;

Tabela 5.11: Resultados da busca pelo termo “DNA” via Elasticsearch

Busca via Elasticsearch termo = DNA		
RE id	**Peso**	**Nome RE**
15739	122,9691	Cell Transcription and Translation
16456	117,7556	Extensão do DNA
15037	116,2638	DNA
11025	116,1966	DNA simple 2
13044	108,7753	DNA orbit animated small
15699	108,7753	Do DNA à proteína
1562	108,6257	Stretching DNA
7763	106,9415	DNA Replication
11959	106,9415	DNA Transcrição
10279	105,9890	DNA e proteínas - Parte II

Da mesma forma que no experimento anterior, somente um especialista no assunto do termo pesquisado poderia fazer uma análise mais apurada se o resultado final é ou não mais relevante do que o resultado apenas via motor de busca.

Tabela 5.12: Comparativo das pontuações e classificações da busca por “DNA”

Termo = DNA						
	Agrup. Tags (At)		Elasticsearch (E)		Mescla (M)	
Classificação	**RE id**	**Pontuação**	**RE id**	**Pontuação**	**RE id**	**Pontuação**
1	6681	4,2880	15739	122,9691	15739	13,6210
2	4137	3,3455	16456	117,7556	6681	10,9008
3	16004	2,7298	15037	116,2638	16456	10,3971
4	17631	2,6763	11025	116,1966	15037	9,4746
5	13756	2,5321	13044	108,7753	11025	9,4331
6	4969	2,5176	15699	108,7753	4137	6,7430
7	7318	2,4079	1562	108,6257	13044	4,8440
8	3114	1,9079	7763	106,9415	15699	4,8440
9	12242	1,9079	11959	106,9415	1562	4,7514
10	18329	1,9079	10279	105,9890	16004	4,0266

Tabela 5.13: Resumo comparativo da busca pelo termo “DNA”

		Classificação		
RE id	**Nome RE**	**(At)**	**(E)**	**(M)**
15739	Cell Transcription and Translation	-	1	1
6681	DNA simples 2	1	-	2
16456	Extensão do DNA	-	2	3
15037	DNA	-	3	4
11025	DNA simple 2	-	4	5
4137	Composição dos ácidos nucléicos	2	-	6
13044	DNA orbit animated small	-	5	7
15699	Do DNA à proteína	-	6	8
1562	Stretching DNA	-	7	9
16004	Guanina	3	-	10

As Tabelas 5.14, 5.15, 5.16 e 5.17 mostram os resultados do experimento de busca realizado com o termo “corrosão”. Pode se ver que os resultados que ocorrem em ambas as buscas (via motor de busca e agrupamento de *tags*) ganham relevância no ranqueamento final, deslocando por exemplo o décimo item do motor de busca para a primeira colocação na classificação final.

Tabela 5.14: Resultados da busca pelo termo “corrosão” via Agrupamento de *Tags*

Busca via Agrupamento de tags termo = corrosão			
Tags coocorrentes (peso): Eletroquímica (0,91); óxido-redução (0,95); pilha de concentração (0,5)			
RE id	**Peso**	**Nome RE**	**Tags**
349	2,8716	Gota salina: parte 2 : vídeo	Educação Básica; Ensino Médio; Química; Transformações: caracterização, aspectos energéticos, aspectos dinâmicos; Educação Profissional; Controle e Processos Industriais; Técnico em Análises Químicas; Técnico em Química; corrosão; eletroquímica; óxido-redução;
12427	2,8716	Gota salina: parte 1 : experimento prático	Educação Básica; Ensino Médio; Química; Transformações: caracterização, aspectos energéticos, aspectos dinâmicos; Educação Profissional; Controle e Processos Industriais; Técnico em Análises Químicas; Técnico em Química; Recursos Naturais; corrosão; eletroquímica; óxido-redução;
17461	1,4172	Pilha de moedas	Educação Básica; Ensino Médio; Química; Transformações: caracterização, aspectos energéticos, aspectos dinâmicos; eletroquímica; pilha de concentração;
3237	1,0000	Corrosão	Educação Básica; Ensino Médio; Química; Propriedades das substâncias e dos materiais; corrosão;

Tabela 5.15: Resultados da busca pelo termo “corrosão” via Elasticsearch

Busca via Elasticsearch termo = corrosão		
RE id	**Peso**	**Nome RE**
1944	157,2310	A química nossa de cada dia : A Química da Corrosão
17317	144,0521	Corrosão do ferro
18078	137,0151	Não fique nervoso: parte1: experimento prático
57	136,9047	Eletrólise e banho de metais
12226	136,9047	A corrosão de aço carbono, alumínio, cobre e magnésio
3237	136,5472	A química entre nós: Corrosão
17806	135,0743	Conversa Periódica - Pilhas e Baterias - Corrosão
3160	130,3338	Quiz: eletroquímica
790	128,6072	Funções químicas e suas reatividades - Diferentes reatividades de metais
349	128,3022	Gota salina: parte 2: vídeo

Tabela 5.16: Comparativo das pontuações e classificações da busca por “corrosão”

Termo = corrosão						
	Agrup. Tags (At)		Elasticsearch (E)		Mescla (M)	
Classificação	RE id	Pontuação	RE id	Pontuação	RE id	Pontuação
1	349	2,8716	1944	157,2310	349	12,9847
2	12427	2,8716	17317	144,0521	1944	12,7175
3	17461	1,4172	18078	137,0151	12427	10,7671
4	3237	1,0000	57	136,9047	17317	7,9341
5			12226	136,9047	3237	5,4773
6			3237	136,5472	18078	5,3800
7			17806	135,0743	57	5,3399
8			3160	130,3338	12226	5,3399
9			790	128,6072	17806	4,6756
10			349	128,3022	3160	2,9549

Tabela 5.17: Resumo comparativo da busca pelo termo “corrosão”

		Classificação		
RE id	Nome RE	(At)	(E)	(M)
349	Gota salina: parte 2 : vídeo	1	10	1
1944	A química nossa de cada dia : A Química da Corrosão	-	1	2
12427	Gota salina: parte 1 : experimento prático	2	-	3
17317	Corrosão do ferro	-	2	4
3237	A química entre nós : Corrosão	4	6	5
18078	Não fique nervoso: parte 1: experimento prático	-	3	6
57	Eletrólise e banho de metais	-	4	7
12226	A corrosão de aço carbono, alumínio, cobre e magnésio	-	5	8
17806	Conversa Periódica - Pilhas e Baterias - Corrosão	-	7	9
3160	Quiz: eletroquímica	-	8	10

As Tabelas 5.18, 5.19, 5.20 e 5.21 mostram os resultados do experimento de busca realizado com o termo “Aquecimento global”.

Tabela 5.18: Resultados da busca pelo termo “Aquecimento global” via Elasticsearch

Busca via Elasticsearch termo = Aquecimento global		
RE id	**Peso**	**Nome RE**
7689	241,2583	Aquecimento Global
8834	240,1201	Aquecimento global - 2
8403	232,6323	Aprender para não aquecer: parte 7 [Conhecimento global]
11471	230,0721	Aquecimento global
2453	225,1099	Pesquisa no Ártico sobre o aquecimento global
468	223,2638	Aquecimento global - 1
16066	222,9591	Aquecimento global - 3
6230	219,3686	Aquecendo os neurônios - Episódio 2 – Aquecimento global
11870	218,3520	Aquecimento global 1
9583	216,0336	Aprender para não aquecer 3 [Conhecimento global]

Neste experimento, o próprio motor de busca já retorna muitos resultados relevantes, mesmo assim três dos dez RE do resultado final, são RE que não foram encontrados pelo motor de busca e foram retornados via agrupamento de *tags*. Neste caso também houve uma diversificação de resultados devido a expansão da consulta com termos correlacionados, e mesmo o motor de busca retornando muitos RE relevantes, a mesclagem conseguiu ressaltar os RE que continham muitas *tags* correlacionadas.

Lembrando que na mesclagem, para o cálculo do ranqueamento final, os termos correlacionados também recebem a impulsão, como é feito pelo Elasticsearch.

Tabela 5.19: Resultados da busca pelo termo “Aquecimento global” via Agrupamento de *Tags*

Busca via Agrupamento de tags termo = Aquecimento global			
Tags coocorrentes (peso): Aquecimento Global (1,00); Gas poluente (0,93); Emissão de gás (0,88); Escamas (0,87); Ação Humana (0,86); Mudança climática (0,84); Fase Clara (0,83); Fase Escura (0,83); Gases tóxicos (0,82); Fogão Solar (0,80); Derretimento de geleira (0,78); Efeito Estufa (0,50)			
RE id	**Peso**	**Nome RE**	**Tags**
15143	3,0475	Mudanças climáticas	Educação Básica; Ensino Fundamental Final; Geografia; Natureza e as questões socioambientais; Camada de ozônio; Derretimento de geleira; Efeito Estufa; Gas poluente; Mudança climática; Urbanização;
17227	2,3626	A física e o cotidiano - Fique sabendo ! - Aquecimento Global	Educação Básica; Ensino Médio; Física; Universo, terra e vida; Aquecimento Global; Ação Humana; Efeito Estufa;
5321	2,3183	A física e o cotidiano: Efeito Estufa	Educação Básica; Ensino Médio; Física; Universo, terra e vida; Aquecimento Global; Efeito Estufa; Gases tóxicos;
15373	2,1505	Reações fotoquímicas - Química da fotossíntese	Educação Básica; Ensino Médio; Química; Modelos de constituição: substâncias, transformações químicas; Cadeia Alimentar; Ciclo da Cavin; Consumidores; Decompositores; Efeito Estufa; Fase Clara; Fase Escura; Fotossíntese; Produtores; Reações fotoquímicas;
16302	1,7124	Aprender para não aquecer 4 [Conhecimento global]	Educação Básica; Ensino Fundamental Final; Meio Ambiente; Sociedade e meio ambiente; Aquecimento global; Aumento da temperatura; Emissão de gás; Mudança climática;
17102	1,2970	A física e o cotidiano - Laboratório virtual :O Fogão Solar	Educação Básica; Ensino Médio; Física; Calor, ambiente e usos de energia; Efeito Estufa; Fogão Solar; Termodinâmica;
5699	1,2970	A física e o cotidiano - Experimentos Educacionais : O Fogão Solar	Educação Básica; Ensino Médio; Física; Equipamentos elétricos e telecomunicações; Efeito Estufa; Fogão Solar; Termodinâmica;
1135	1,0000	Usando o rádio para divulgar e ensinar ciência	Educação Básica; Ensino Médio; Química; Transformações: caracterização, aspectos energéticos, aspectos dinâmicos; Educação Superior; Ciências Humanas; Educação; Ensino-Aprendizagem; Métodos e Técnicas de Ensino; Aquecimento Global; Efeito estufa; Rádio;
9823	0,9268	Chuva ácida	Educação Básica; Ensino Fundamental Final; Ciências Naturais; Vida e ambiente; Atmosfera; Chuva ácida; Gas poluente;
9155	0,8764	Nanopartículas combatendo a poluição	Educação Superior; Multidisciplinar; Interdiciplinar; Engenharia; Tecnologia; Gestão; Emissão de gás; Nanopartícula; Nanotecnologia;

Tabela 5.20: Comparativo das pontuações e classificações da busca por “Aquecimento global”

Termo = Aquecimento global						
	Agrup. Tags (At)		**Elasticsearch (E)**		**Mescla (M)**	
Classificação	**RE id**	**Pontuação**	**RE id**	**Pontuação**	**RE id**	**Pontuação**
1	15143	3,0475	7689	241,2583	7689	14,7822
2	17227	2,3626	8834	240,1201	8834	14,3084
3	5321	2,3183	8403	232,6323	8403	11,1915
4	15373	2,1505	11471	230,0721	15143	10,7018
5	16302	1,7124	2453	225,1099	11471	10,1258
6	17102	1,2970	468	223,2638	2453	8,0603
7	5699	1,2970	16066	222,9591	17227	7,3899
8	1135	1,0000	6230	219,3686	468	7,2918
9	9823	0,9268	11870	218,3520	5321	7,1756
10	9155	0,8764	9583	216,0336	16066	7,1650

Tabela 5.21: Resumo comparativo da busca pelo termo “Aquecimento global”

		Classificação		
RE id	**Nome RE**	**(At)**	**(E)**	**(M)**
7689	Aquecimento Global	-	1	1
8834	Aquecimento global - 2	-	2	2
8403	Aprender para não aquecer: parte 7 [Conhecimento global]	-	3	3
15143	Mudanças climáticas	1	-	4
11471	Aquecimento global	-	4	5
2453	Pesquisa no Ártico sobre o aquecimento global	-	5	6
17227	A física e o cotidiano - Fique sabendo ! - Aquecimento Global	2	-	7
468	Aquecimento global - 1	-	6	8
5321	A física e o cotidiano: Efeito Estufa	3	-	9
16066	Aquecimento global - 3	-	7	10

5.3 Considerações do capítulo

Com a implementação do modelo e com os resultados dos experimentos foi possível avaliar a viabilidade do modelo proposto. Considerar as *tags* correlacionadas para realizar a expansão da consulta original tornou possível, em alguns casos, mais do que dobrar o percentual de resultados relacionados ao termo de busca original, como foi o caso do experimento com o termo “Sagitário”. Para os demais casos, onde o próprio motor de busca retorna uma boa quantidade de resultados relevantes, a avaliação é empírica e subjetiva, pois somente um especialista ligado ao tema da busca, ou um usuário com propósitos de busca específicos poderiam avaliar com maior exatidão a relevância dos RE ranqueados nos resultados.

A proposta de Knautz et al. (2010) apresenta um modelo baseado na apresentação de uma nuvem de *tags*, em que o usuário precisa clicar nas *tags* ou nas arestas para acessar os recursos relacionados à *tag* específica. Comparada à proposta de Knautz et al. (2010), pode-se considerar que para um usuário obter os mesmos resultados trazidos pelo modelo aqui proposto, o mesmo precisaria realizar várias consultas distintas. Desta forma, um professor precisaria dispender muito mais tempo e esforço para obter os resultados que nesta abordagem são trazidos automaticamente com apenas uma consulta. Tomando o exemplo do experimento com o termo “DNA”, em que o agrupamento reúne 16 termos correlacionados, o usuário precisaria realizar 16 consultas no modelo proposto por Knautz et al. (2010) para obter resultados similares aos obtidos pelo modelo proposto neste trabalho.

O processo de ranqueamento, pela simples soma das pontuações das *tags* do agrupamento que o RE possui, também mostrou-se viável e apresentou bons resultados. Resultados pouco relevantes retornados pelo motor de busca foram tratados com pouco destaque na classificação final em relação a RE considerados mais relevantes em relação aos termos correlacionados retornados pela busca baseada no agrupamento de *tags*.

Capítulo 6

Conclusões

O presente trabalho apresentou uma nova abordagem do processo de busca por RE em repositórios digitais no intuito de facilitar aos professores e outros usuários interessados à encontrar RE relevantes dentro deste ambiente. Para alcançar este objetivo, foi proposto e implementado um processo de busca de RE em repositórios digitais que mescla RE encontrados por meio de agrupamento de *tags* com os RE encontrados via motor de busca, recalculando as pontuações de cada RE bem como reordenando-os conforme essa nova pontuação, gerando-se desta forma um novo ranqueamento de RE.

Para viabilizar a formação do agrupamento de *tags*, foi necessário mapear a similaridade entre as *tags* e para isso lançou-se mão da coocorrência entre *tags* e a partir desse mapeamento, implementou-se o cálculo do coeficiente de similaridade entre as *tags*, viabilizando a construção de um grafo não direcionado, formado então pelas *tags* que são os vértices deste grafo; a relação de coocorrência são as arestas do grafo, ligando uma *tag* a outra se houver coocorrência entre elas; além disso as arestas são ponderadas pelo coeficiente de similaridade calculado para o par de *tags* coocorrentes.

A partir do grafo formado, foi possível a construção do agrupamento de *tags*, formando grupos ou comunidades de *tags* fortemente correlacionadas. Esses agrupamentos são a base para apoiar a busca por RE, viabilizando a ampliação semântica dos termos de busca, aumentando assim a chance de se encontrar RE relevantes à busca do usuário.

Após a ampliação dos resultados da busca por meio do suporte do agrupamento de *tags*, que foi somado aos resultados encontrados por um motor de busca, nosso trabalho foi além. Com o novo conjunto de resultados, trabalhou-se então sobre o ranqueamento da lista final, pois para o usuário é crucial uma boa classificação dos resultados, de forma que os RE relevantes apareçam no topo da lista retornada. Nosso algoritmo de ranqueamento passa então por um processo que normaliza os resultados e destaca a relevância de resultados que aparecem nas duas abordagens (motor de busca e agrupamento de *tags*). Além disso, impulsiona os RE que foram encontrados por meio das *tags* correlacionadas para equiparar a relevância com o termo original de busca. Resultados pouco relevantes trazidos pelo motor de busca são ranqueados com menor relevância nesta nova abordagem.

Tendo aplicado a proposta de busca e ranqueamento de recursos educacionais com suporte de agrupamento de *tags* utilizando a infraestrutura do Portalmec para sua implementação, foi possível avaliar e realizar experimentos que mostram a viabilidade do modelo proposto. Primeiro, destacamos a utilidade de considerar os termos correlacionados para realizar a expansão da consulta original. Essa técnica viabilizada pelos agrupamentos de *tags* coocorrentes auxilia o usuário de forma transparente.

Os experimentos foram realizados considerando a recuperação de uma lista de no máximo dez resultados, pois conforme estudo de Silverstein et al. (1999), os usuários costumam considerar somente a primeira página ou os primeiros dez resultados. No experimento realizado com o termo de busca "Sagitário", por meio do motor de busca era possível recuperar somente um RE considerado relevante ao termo pesquisado. Os demais RE eram pouco relevantes, pois tinham relação com o termo "sanitário". Após aplicar a nova abordagem proposta, nove entre os dez resultados retornados podem ser considerados relevantes ao termo pesquisado.

Já no experimento com o termo "DNA", muitos resultados relevantes foram retornados pelo motor de busca, mas pode-se ressaltar que três dos dez, são RE que não foram encontrados pelo motor de busca e foram retornados via agrupamento de *tags*. No experimento com o termo "Força gravitacional", dois dos dez são RE novos trazidos pelo agrupamento de *tags*, três foram retornados por ambos, fazendo com que itens que tinham sido melhor classificados nas duas abordagens separadas perdessem a relevância perante esses três no ranqueamento final.

Considerando a implementação do modelo proposto e pelos resultados obtidos nos experimentos com dados do repositório de recursos educacionais do próprio Portalmec, podemos verificar a viabilidade do modelo, principalmente na ampliação de resultados relevantes nos casos em que o motor de busca não é capaz de encontrar recursos pelo simples fato de não considerar termos similares ou correlacionados. Isso significa que neste tipo de situação, comparado ao processo de busca utilizado pelo motor de busca Elasticsearch, que é baseado em MEV e TF-IDF, os resultados obtidos pelo modelo proposto são considerados empiricamente mais significativos. Para os demais casos, onde o próprio motor de busca consegue retornar uma boa quantidade de resultados relevantes, somente um especialista ligado ao tema da busca, ou o próprio usuário que busca por informação, poderiam avaliar a relevância dos RE ranqueados entre as distintas abordagens.

Comparada à proposta de Knautz et al. (2010) pode-se considerar que para um usuário obter os mesmos resultados trazidos pela nossa abordagem, o mesmo precisaria realizar várias consultas distintas, dispendendo muito mais tempo e esforço do usuário para obter os resultados que em nossa abordagem são trazidos automaticamente com apenas uma consulta.

O processo de ranqueamento também mostrou sua aplicabilidade, pois resultados pouco relevantes que foram retornados pelo motor de busca são devidamente tratados com pouco destaque na classificação final, já que RE mais relevantes foram retornados pela busca baseada no agrupamento de *tags*. Foi proposto e aplicado um conjunto de equações simples e que se mostrou adequado pelos resultados obtidos. Verifica-se desta forma que não necessariamente precisa-se recorrer a modelos e cálculos complexos para obter melhorias no ranqueamento.

Desta forma, considera-se que o modelo proposto neste trabalho pode ser aplicado a qualquer repositório digital que permita a atribuição de *tags* a seus objetos.

Assim que o Portalmec esteja em pleno funcionamento, com usuários ativos realizando a etiquetagem colaborativa, permitirá a realização de novos experimentos com um maior volume de dados reais, possibilitando uma reavaliação dos resultados. Também seria de grande valia os próprios usuários realizarem o julgamento dos resultados para avaliação do modelo proposto neste trabalho.

Como trabalhos futuros pode-se destacar ainda a possibilidade de se aplicar técnicas do Processamento de Linguagem Natural como radicalização, lematização, remoção de *stopwords* e desambiguação para melhorar a qualidade dos agrupamentos de *tags* e, consequentemente, melhorar ainda mais o resultado das buscas.

Outro ponto que merece pesquisa adicional é sobre as técnicas que podem ser aplicadas para realizar o ranqueamento de RE, pois dependendo das equações e métodos aplicados pode-se classificar os resultados finais de muitas maneiras distintas.

Para finalizar, pode-se citar a exploração dos agrupamentos de *tags* para possibilitar a recomendação de RE deve ser a os usuários que navegam pelo portal de repositórios digitais, ou ainda, para recomendação de *tags* no momento em que o usuário esteja realizando novas anotações nos RE.

Referências Bibliográficas

Aggarwal, C. C. e Reddy, C. K. (2013). *Data clustering: algorithms and applications*. CRC press.

Aguiar, J. J., Santos, S. I., Fechine, J. M. e Costa, E. B. (2014). Um mapeamento sistemático sobre iniciativas brasileiras em sistemas de recomendação educacionais. *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, 1:1123–1132.

Apache (2006). Apache solr. `https://lucene.apache.org/solr/`. Acessado em 15/01/2017.

Baca, M. (2008). *Introduction to metadata*. Getty Publications.

Baeza-Yates, R. e Ribeiro-Neto, B. (1999). *Modern information retrieval*, volume 463. ACM Press New York.

Begelman, G., Keller, P. e Smadja, F. (2006). Automated tag clustering: Improving search and exploration in the tag space. Em *Collaborative Web Tagging Workshop at World Wide Web Conference 2006*, páginas 15–33, Edinburgh - Scotland.

Białecki, A., Muir, R., Ingersoll, G. e Imagination, L. (2012). Apache lucene 4. Em *SIGIR 2012 Workshop on Open Source Information Retrieval*, páginas 17–24, Portland - Oregon - USA.

Blondel, V. D., Guillaume, J.-L., Lambiotte, R. e Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008.

Bobadilla, J., Ortega, F., Hernando, A. e Bernal, J. (2012). A collaborative filtering approach to mitigate the new user cold start problem. *Knowledge-Based Systems*, 26:225–238.

Bohlin, L., Edler, D., Lancichinetti, A. e Rosvall, M. (2014). Community detection and visualization of networks with the map equation framework. Em *Measuring Scholarly Impact*, páginas 3–34. Springer.

Brin, S. e Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1):107–117.

Catarino, M. E. e Baptista, A. A. (2007). Folksonomia: um novo conceito para a organização dos recursos digitais na web. *DataGramaZero-Revista de Ciência da Informação*, 8(3):1–20.

Coelho, G. O. (2009). Recuperação de objetos de aprendizagem usando a web 2.0. Dissertação de Mestrado, Pontifícia Universidade Católica de Minas Gerais, Belo Horizonte - MG - Brazil.

Coelho, G. O., Ishitani, L. e Nelson, M. A. V. (2012). Vitae: recuperação de objetos de aprendizagem baseada na web 2.0. *ETD-Educação Temática Digital*, 14(2):238–257.

Costa, E., Aguiar, J. e Magalhães, J. (2013). Sistemas de recomendação de recursos educacionais: conceitos, técnicas e aplicações. Em *Jornada de Atualização em Informática na Educação*, volume 1, páginas 57–78, Campinas - SP - Brazil.

de Souza, A. B., da Silva, J. P., de Oliveira, W. C. C., Kuma, T. H. e Silveira, I. F. (2008). Recuperação semântica de objetos de aprendizagem: Uma abordagem baseada em tesauros de propósito genérico. Em *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 1, páginas 603–612, Fortaleza - CE - Brazil.

dos Santos, H., Cechinel, C., Araújo, R. e Brauner, D. (2015). Recomendação de objetos de aprendizagem utilizando filtragem colaborativa: Uma comparação entre abordagens de pré-processamento por meio de clusterização. Em *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 26, páginas 1127–1136, Maceió - AL - Brazil.

Easley, D. e Kleinberg, J. (2010). *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge University Press.

Elastic (2015). Elasticsearch: Restful, distributed search & analytics. `https://www.elastic.co/products/elasticsearch`. Acessado em 15/01/2017.

Gemmell, J., Shepitsen, A., Mobasher, B. e Burke, R. (2008). Personalization in folksonomies based on tag clustering. *Intelligent techniques for web personalization & recommender systems*, 12:37–48.

Girvan, M. e Newman, M. E. (2002). Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12):7821–7826.

Goffman, W. (1964). A searching procedure for information retrieval. *Information Storage and Retrieval*, 2(2):73–78.

Golder, S. A. e Huberman, B. A. (2006). Usage patterns of collaborative tagging systems. *Journal of information science*, 32(2):198–208.

Gormley, C. e Tong, Z. (2015). *Elasticsearch: The Definitive Guide*. O'Reilly Media, Inc.

Hassan-Montero, Y. e Herrero-Solana, V. (2006). Improving tag-clouds as visual information retrieval interfaces. Em *International conference on multidisciplinary information sciences and technologies*, páginas 25–28, Mérida - Spain.

Hotho, A., Jäschke, R., Schmitz, C. e Stumme, G. (2006a). Folkrank: A ranking algorithm for folksonomies. Em *LWA 2006: Lernen - Wissensentdeckung - Adaptivität*, volume 1, páginas 111–114, Hildesheim - German.

Hotho, A., Jäschke, R., Schmitz, C. e Stumme, G. (2006b). Information retrieval in folksonomies: Search and ranking. Em *European Semantic Web conference*, páginas 411–426, Budva - Montenegro. Springer.

Isotani, S., Mizoguchi, R., Bittencourt, I. I. e Costa, E. (2009). Estado da arte em web semântica e web 2.0: potencialidades e tendências da nova geração de ambientes de ensino na internet. *Revista brasileira de informática na educação*, 17(1):30–42.

Jain, A. K. e Dubes, R. C. (1988). *Algorithms for clustering data*. Prentice-Hall, Inc.

Jain, A. K., Murty, M. N. e Flynn, P. J. (1999). Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3):264–323.

Kaufman, L. e Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis*, volume 344. John Wiley & Sons.

Kim, H.-N., Ji, A.-T., Ha, I. e Jo, G.-S. (2010). Collaborative filtering based on collaborative tagging for enhancing the quality of recommendation. *Electronic Commerce Research and Applications*, 9(1):73–83.

Knautz, K., Soubusta, S. e Stock, W. G. (2010). Tag clusters as information retrieval interfaces. Em *System Sciences (HICSS), 2010 43rd Hawaii International Conference on*, páginas 1–10, Honolulu - HI - USA. IEEE.

Lagoze, C., Lynch, C., Waters, D., Van de Sompel, H. e Hey, T. (2006). Augmenting interoperability across scholarly repositories. Em *Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL'06)*, página 85, Chapel Hill - NC - USA. IEEE.

Lancichinetti, A. e Fortunato, S. (2009a). Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Physical Review E*, 80(1):016118.

Lancichinetti, A. e Fortunato, S. (2009b). Community detection algorithms: a comparative analysis. *Physical review E*, 80(5):056117.

Lancichinetti, A., Fortunato, S. e Radicchi, F. (2008). Benchmark graphs for testing community detection algorithms. *Physical review E*, 78(4):046110.

Lemur, P. (2000). Indri - language modeling meets inference networks. `https://www.lemurproject.org/indri/`. Acessado em 15/01/2017.

Li, X., Uricchio, T., Ballan, L., Bertini, M., Snoek, C. G. e Bimbo, A. D. (2016). Socializing the semantic gap: A comparative survey on image tag assignment, refinement, and retrieval. *ACM Computing Surveys (CSUR)*, 49(1):14–53.

Liu, R. e Niu, Z. (2014). A collaborative filtering recommendation algorithm based on tag clustering. Em *Future Information Technology*, páginas 177–183. Springer, Zhangjiajie - China.

Manning, C. D., Raghavan, P., Schütze, H. et al. (2008). *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge.

Markines, B., Cattuto, C., Menczer, F., Benz, D., Hotho, A. e Stumme, G. (2009). Evaluating similarity measures for emergent semantics of social tagging. Em *Proceedings of the 18th international conference on World wide web*, páginas 641–650, Madrid - Spain. ACM.

McCandless, M., Hatcher, E. e Gospodnetic, O. (2010). *Lucene in Action: Covers Apache Lucene 3.0*. Manning Publications Co.

Mooers, C. N. (1951). Zatocoding applied to mechanical organization of knowledge. *American documentation*, 2(1):20–32.

Morrison, P. J. (2008). Tagging and searching: Search retrieval effectiveness of folksonomies on the world wide web. *Information Processing & Management*, 44(4):1562–1579.

Newman, M. E. (2004). Fast algorithm for detecting community structure in networks. *Physical review E*, 69(6):066133.

Newman, M. E. e Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical review E*, 69(2):026113.

Patrocinio, M. e Ishitani, L. (2009). Associação de recursos semânticos para a anotação de objetos de aprendizagem. Em *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 1, Florianópolis - SC - Brazil.

Peters, I. (2009). *Folksonomies. Indexing and retrieval in Web 2.0*, volume 1. Walter de Gruyter.

Pontes, W. L., França, R. M., Costa, A. P. M. e Behar, P. (2014). Filtragens de recomendação de objetos de aprendizagem: uma revisão sistemática do cbie. Em *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 25, páginas 549–558, Dourados - MS - Brazil.

Rafailidis, D. e Daras, P. (2013). The tfc model: Tensor factorization and tag clustering for item recommendation in social tagging systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 43(3):673–688.

Ribeiro, F. A. A., Fonseca, L. C. C. e de Sousa Freitas, M. (2013). Recomendando objetos de aprendizagem a partir das hashtags postadas no moodle. Em *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 24, páginas 82–91, Campinas - SP - Brazil.

Rochadel, W. (2016). Identificação de critérios para avaliação de ideias: um método utilizando folksonomias. Dissertação de Mestrado, Universidade Federal de Santa Catarina, Florianópolis - SC - Brazil.

Rosvall, M., Axelsson, D. e Bergstrom, C. T. (2009). The map equation. *The European Physical Journal-Special Topics*, 178(1):13–23.

Rosvall, M. e Bergstrom, C. T. (2008). Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences*, 105(4):1118–1123.

Salton, G., Wong, A. e Yang, C.-S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620.

Saoud, Z. e Kechid, S. (2016). Integrating social profile to improve the source selection and the result merging process in distributed information retrieval. *Information Sciences*, 336:115–128.

Shepitsen, A., Gemmell, J., Mobasher, B. e Burke, R. (2008). Personalized recommendation in social tagging systems using hierarchical clustering. Em *Proceedings of the 2008 ACM conference on Recommender systems*, páginas 259–266, Lausanne - Switzerland. ACM.

Silverstein, C., Marais, H., Henzinger, M. e Moricz, M. (1999). Analysis of a very large web search engine query log. Em *ACM SIGIR Forum*, volume 33, páginas 6–12. ACM.

Sinclair, J. e Cardew-Hall, M. (2008). The folksonomy tag cloud: when is it useful? *Journal of Information Science*, 34(1):15–29.

Sun, Y., Han, J., Zhao, P., Yin, Z., Cheng, H. e Wu, T. (2009). Rankclus: integrating clustering with ranking for heterogeneous information network analysis. Em *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*, páginas 565–576, Saint-Petersburg - Russian Federation. ACM.

Xapian (2000). The xapian project. `https://xapian.org/`. Acessado em 15/01/2017.