Nelson Jacob Dressler

Sistemas de Recomendação Baseada em Conteúdo para Sistemas de Bibliotecas Universitárias

São Paulo – Brasil

2017

Nelson Jacob Dressler

Sistemas de Recomendação Baseada em Conteúdo para Sistemas de Bibliotecas Universitárias

Monografia apresentada na disciplina Trabalho de Conclusão de Curso II, como parte dos requisitos para obtenção do título de Bacharel em Ciência da Computação.

Centro Universitário Senac

Bacharelado em Ciência da Computação

Orientador: Prof. Ms. Leonardo Takuno

São Paulo – Brasil

2017

Nelson Jacob Dressler

Sistemas de Recomendação Baseada em Conteúdo para Sistemas de Bibliotecas Universitárias

Monografia apresentada na disciplina Trabalho de Conclusão de Curso II, como parte dos requisitos para obtenção do título de Bacharel em Ciência da Computação.

Prof. Ms. Leonardo Takuno
Orientador

Profa. Ms. Danielle Mingatos
Professora da Disciplina

Prof. DSc. Alexandre Lopes Machado
Professor Convidado

São Paulo – Brasil
2017

Dedico este trabalho a minha noiva e companheira pra vida toda Ana Gabriela e aos meus pais José Manoel e Selma por terem me apoiado e me ajudado sempre e terem acreditado em mim em mais esse momento importante da vida.

"Um homem não deveria nunca parar de aprender, nem no seu último dia.
(Moises Maimônides)

"O único lugar onde o sucesso vem antes do trabalho é no dicionário.
(Albert Einstein)

Agradecimentos

Em primeiro lugar, agradeço ao Todo Poderoso, Criador do Universo, pela saúde, sabedoria e sucesso que tem me proporcionado até hoje.

Em segundo lugar, gostaria de agradecer à minha noiva e companheira para a vida toda, Ana Gabriela Ungierowicz, por me acompanhar nos últimos momentos e mais difíceis do curso, me auxiliando de diversas maneiras, principalmente para a realização deste trabalho de conclusão.

Em terceiro, gostaria de agradecer aos meus pais, José Manoel Dressler e Selma Kalik Dressler, que, desde sempre, me acompanham nos bons e turbulentos momentos da vida e me apoiam em todas as minhas decisões, além da excelente educação que me foi dada desde pequeno.

Ao Centro Universitário Senac, por ter me dado a oportunidade de entrar no curso de Bacharelado em Ciência da Computação e por ter oferecido sempre uma ótima qualidade de ensino, com bons equipamentos e professores renomados, tanto no meio acadêmico quanto no mercado de trabalho.

À professora, mestre e coordenadora do curso, Danielle Mingatos, por ter me ajudado sempre que lhe foi solicitada. Além disso, ao ex-coordenador do curso, o professor e mestre Eduardo Heredia.

Ao meu orientador, o professor e mestre Leonardo Takuno, por ter me aconselhado e me mostrado um caminho objetivo e que, sem dúvida alguma, permitiu a conclusão deste trabalho com êxito.

Ao meu colega Gabriel Fontenelle por ter me acompanhado e apoiado nas questões técnicas e práticas deste trabalho.

Aos meus colegas, amigos e conhecidos que colaboraram com a realização dos testes do sistema desenvolvido para este trabalho.

Por fim, aos meus professores e colegas que fizeram parte do período em que estive estudando na instituição, compreendido de Fevereiro de 2014 à Dezembro de 2017.

Resumo

A Ciência de Dados é uma área que está em evidência na atualidade, fazendo parte das novas tendências da Era da Informação, graças à evolução dos Sistemas de Informação e ao surgimento do *Big data*. Conjuntamente, as áreas derivadas desta como Análise de Dados, Mineração de Dados, Mineração de Textos e Aprendizagem Computacional estão sendo muito estudadas e aplicadas, tanto no mercado quanto no ambiente acadêmico. Uma das aplicações dessas áreas são os Sistemas de Recomendação, que tem como base os conceitos de Filtragem de Informações e Recuperação de Informações. Estes sistemas aplicam tanto a técnica de Aprendizagem Computacional, quanto a de Mineração de Dados. Um dos cenários de aplicação dos Processos de Recomendação é a Biblioteca Universitária que necessita de um serviço diferenciado e personalizado ao usuário para a busca de documentos, conforme seus interesses e necessidades. A proposta deste trabalho foi o desenvolvimento de um Sistema de Recomendação para Bibliotecas Universitárias, com a aplicação das técnicas de filtragem baseada em conteúdo e recuperação de informações, visando a personalização do serviço de empréstimos e o alcance de melhores níveis de satisfação dos usuários, sem a intervenção humana.

Palavras-chaves: Ciência de Dados. Análise de Dados. Mineração de Dados. Mineração de Textos. Aprendizagem Computacional. Sistemas de Recomendação. Filtragem Baseada em Conteúdo. Recuperação de Informações.

Abstract

For the last years, Data Science has been one of the new tendency of Information Era, thanks to the evolution of Information Systems and the beginning of Big Data. The derivative areas such as Data Analysis, Data Mining, Text Mining and Machine Learning are being studied and applied, in the market as much as in the academic field. One of the applications of this area is Recommendation Systems, which is based on the concepts of Information Filtering and Information Retrieval. These systems apply Machine Learning and Data Mining techniques. One scenario that recommendation processes can be used are University Libraries, in which, requires a differential and personalized service to users that are looking for documents, according to their interests and needs. In this context, the work proposes a Recommendation Systems to University Libraries , applying the techniques of Content Based Filtering and Information Retrieval. This system aims for providing personal loan services and reach better levels of user satisfaction without human intervention.

Keywords: Data Science. Data Analysis. Data Mining. Text Mining. Machine Learning. Recommendation Systems. Content Based Filtering. Information Retrieval.

Lista de ilustrações

Lista de abreviaturas e siglas

AM	Aprendizado de Máquina
API	Application Programming Interface (Interface de Programação de Aplicativos)
BD	Banco de Dados
BOW	Bag of Word (Depósito de Palavras)
CBF	Content-Based Filtering (Filtragem Baseada em Conteúdo)
CF	Collaborative Filtering (Filtragem Colaborativa)
CRISP-DM	Cross Industry Standard Process for Data Mining (Processo Padrão Genérico para Mineração de Dados)
CRUD	Create, Read, Update, Delete (Criar, Ler, Atualizar, Excluir)
DCBD	Descoberta de Conhecimento em Base de Dados
DER	Diagrama Entidade Relacionamento
DM	Data Mart
DSI	Disseminação Seletiva da Informação
DW	Data Warehouse
ENIAC	Electronic Numerical Integrator and Computer (Computador Integrador Numérico Eletrônico)
ETEC	Escola Técnica Estadual de São Paulo
ETL	Extract, Transform and Load (Extração, Transformação e Carregamento)
HF	Hybrid Filtering (Filtragem Híbrida)
ID	Identity (Identidade)
KDD	Knowledge Discovery from Data (Descoberta de Conhecimento a partir de Dados)
MER	Modelo Entidade Relacionamento

MD	Mineração de Dados
MVC	Model – View – Controller (Modelo – Visualização – Controlador)
MT	Mineração de Textos
MTV	Model – Template – View (Modelo – Página – Visualização)
OLTP	Online Transaction Processing (Processamento de Transações em Tempo Real)
PAM	Probabilistic Advanced Modeling (Modelagem Probabilística Avançada)
RI	Recuperação de Informação
SGBD	Sistema de Gerenciamento de Banco de Dados
SI	Sistema de Informação
SR	Sistema de Recomendação
TF-IDF	Term Frequency – Inverse Document Frequency (Frequência de Termos – Frequência Inversa de Documentos)
T–SQL	Transact – Structured Query Language (Transações da Linguagem de Consulta Estruturada)
URL	Uniform Resource Locator (Localizador Uniforme de Recursos)
VSM	Vector Space Model (Modelo de Espaço Vetorial)
XML	eXtensible Markup Language (Linguagem de Marcação Extendida)

Sumário

1 Introdução

O conceito *Data Science* ou Ciência de Dados existe desde os anos 1960 (AMARAL, 2016). Desde os primórdios do surgimento da computação e da tecnologia havia a preocupação em armazenar e trabalhar sobre os dados gerados por uma empresa ou organização, a fim de melhorar os processos internos, tornar o negócio mais rentável e melhorar o relacionamento com seus clientes. Essa preocupação envolve todo o processo de informatização de um ambiente corporativo e de controle dos dados que são gerados e armazenados em seus sistemas. Estes aspectos também se aplicam a instituições de ensino.

Entretanto, a ciência de dados é uma área que está em evidência, devido ao crescente volume de dados que são gerados diariamente por diferentes dispositivos, conhecido também como *Big Data*, e a preocupação em transformar os mesmos em informações úteis e conhecimento para as empresas (AMARAL, 2016).

Uma das aplicações da ciência de dados está em sistemas de recomendação que utilizam técnicas para analisar dados de usuários e itens, assim como seus relacionamentos ou interações realizadas, presentes numa base de dados, visando recomendar novos itens que ainda não chegaram ao conhecimento do usuário, mas que possam fazer parte de seu interesse. Estes sistemas estão mais presentes na área de *e-commerce* e de negócios, como *Amazon*, *Ebay*, *Netflix* e *Last.fm* (TAKAHASHI, 2015; REATEGUI; CAZARELLA, 2005).

Nas bibliotecas universitárias, a presença desses sistemas não é tão comum (KREBS, 2013). Os sistemas possuem dados estáticos nos quais oferecem aos usuários apenas a opção de busca de informações, e não existe um envolvimento do sistema com o usuário, apenas na presença de um ser humano que se relacione e conheça os usuários, seus interesses e seu perfil. Somente nessas condições é possível recomendar publicações aos usuários.

Entretanto, num cenário onde os usuários não frequentem presencialmente, ou quando existe um número maior de usuários presentes na biblioteca ou um número menor de bibliotecários disponíveis para atendimento, se torna difícil o conhecimento de cada usuário e, consequentemente, o serviço acaba não sendo satisfatório, além de não ser possível atingir aqueles que não compareçam ao local (KREBS, 2013).

Seria interessante que houvesse uma forma de recomendar livros e artigos para cada usuário automaticamente, via sistema, de acordo com seu perfil, interesses ou histórico de locações, a fim de criar mais um recurso para sistemas bibliotecários e disponibilizar mais um serviço para os estudantes, proporcionando uma satisfação maior aos usuários.

O tema deste trabalho é a ciência de dados aplicada à sistemas de bibliotecas universitárias, utilizando técnicas de mineração de dados e aprendizagem computacional e,

mais especificamente, algoritmos de recomendação para gerenciar estes sistemas, analisando os dados dos usuários e dos itens existentes para sugerir novos itens a usuários que estejam cadastrados e ativos para empréstimos, conforme seu perfil, interesses em comum com outros estudantes e interações anteriores.

O objetivo geral deste trabalho é criar um sistema de recomendação de itens aos usuários de bibliotecas universitárias e a personalização do serviço de empréstimos.

Como objetivos específicos estão:

- A implementação de um algoritmo e técnica de recomendação baseada na análise de conteúdo dos itens;
- O relacionamento de perfis de usuários, aplicando cálculos estatísticos e matemáticos sobre uma base de dados de livros;
- A modelagem e simulação de um cenário onde estudantes retiram livros e artigos periodicamente; e
- A análise dos resultados conforme o grau de satisfação e avaliação dos usuários finais.

Este trabalho está estruturado da seguinte forma: no capítulo 1 é apresentada uma introdução, na qual são descritos a motivação e a contextualização do tema, a problemática, as hipóteses e os objetivos gerais e específicos para a execução deste trabalho. No capítulo 2, é apresentado um referencial teórico para a solução do problema. No capítulo 3, é descrito como o trabalho foi desenvolvido na parte prática e apresentado todos os detalhes de projeto e implementação do sistema proposto. No capítulo 4, são descritos os testes realizados com o usuário e os resultado obtidos, além de uma análise crítica sobre o desempenho deste trabalho. Por fim, são apresentadas as conclusões e trabalhos futuros no capítulo 5.

2 Revisão de Literatura

Este capítulo tem como objetivo principal trazer um referencial teórico aos conceitos estudados e necessários para o tema proposto neste trabalho. Inicialmente, é definido, contextualizado e descrito sobre a ciência de dados, bem como a sua área de atuação e as possíveis aplicações. Posteriormente, é explicado sobre alguns conceitos relativos à ciência de dados que introduziram a necessidade para a sua existência e motivaram o seu crescimento nos dias atuais. Após o esclarecimento desses tópicos, exploramos duas áreas que estão intrinsecamente relacionadas, apesar de diferenças tênues as separando: a mineração de dados e a aprendizagem computacional. Estas são abordagens que estão presentes principalmente na fase de análise de dados e utilizam métodos matemáticos e estatísticos conjuntamente com algoritmos computacionais e modelos de banco de dados para a execução de sua análise. Por fim, nos aprofundamos numa aplicação comum de aprendizagem computacional: os sistemas de recomendação, além de ser apontada sua real necessidade no cenário de bibliotecas universitárias.

2.1 Ciência de Dados

Historicamente, desde o surgimento da computação e dos primeiros sistemas de informação, houve a necessidade de se trabalhar dos dados a fim de gerar informações e conhecimento para as empresas e/ou organizações. Essa necessidade é algo constante, porém a forma com que os dados foram sendo gerados, assim como o aumento do volume e da quantidade dos mesmos e a evolução da tecnologia como um todo, exigiu um grau maior de tratamento, transformação e adequação desses para se obter resultados compatíveis com o tempo e precisão esperados pelos usuários finais.

O termo **Ciência de Dados** (ou ***Data Science***, em inglês) surgiu ainda nos anos 1960, entretanto, é caracterizada como uma ciência nova devido às mudanças que ocorreram no cenário mundial citadas acima (AMARAL, 2016).

Um sistema de informação (SI) é um sistema que permite a coleta, armazenamento, processamento, recuperação e disseminação de informações para usuários finais. Um programa, basicamente, é formado por três etapas principais: entrada, processamento e saída (PRESSMAN, 2011). Ao longo destas, há a presença do dado e/ou informação na qual todo o processo é visado e seu funcionamento se deve graças e para a obtenção dos mesmos.

Segundo (AMARAL, 2016), ciência de dados é definida como os processos, modelos e tecnologias que estudam os dados ao longo de todo o seu ciclo de vida. Para o autor

(BAUDISCH, 2016), significa formular questões a fim de encontrar padrões profundos e ocultos num oceano de dados em diversos formatos. Em outras palavras, produz dados a fim de trazer respostas para perguntas preditivas, agregando valor aos dados brutos e motivando outros usuários a usarem os resultados gerados.

A ciência de dados é baseada em técnicas e teorias originadas das engenharias e ciências básicas, formando uma nova área altamente interdisciplinar (PORTO; ZIVIANI, 2014). Esta multidisciplinaridade está intrinsecamente ligada às áreas de Ciência da Computação, Matemática e Estatística, Especialização Científica e Design Gráfico (BAUDISCH, 2016). Podemos especificar e descrever cada subárea da seguinte forma:

- **Ciência da Computação:** envolve o armazenamento, a obtenção e o tratamento de dados por meio de sistemas de informação e programas desenvolvidos para esse fim;
- **Matemática e Estatística:** envolve a aplicação de técnicas de filtragem, generalização e especialização, classificação e agrupamento dos dados obtidos e gerados;
- **Especialização Científica:** envolve métodos científicos e a formulação de perguntas pertinentes à presença de padrões nos dados obtidos e gerados.
- **Design Gráfico:** envolve formas de visualização e refinamento das informações;

Na figura 1, é apresentado o diagrama de *Venn* onde são relacionadas as áreas comentadas acima, formando a área da ciência de dados.

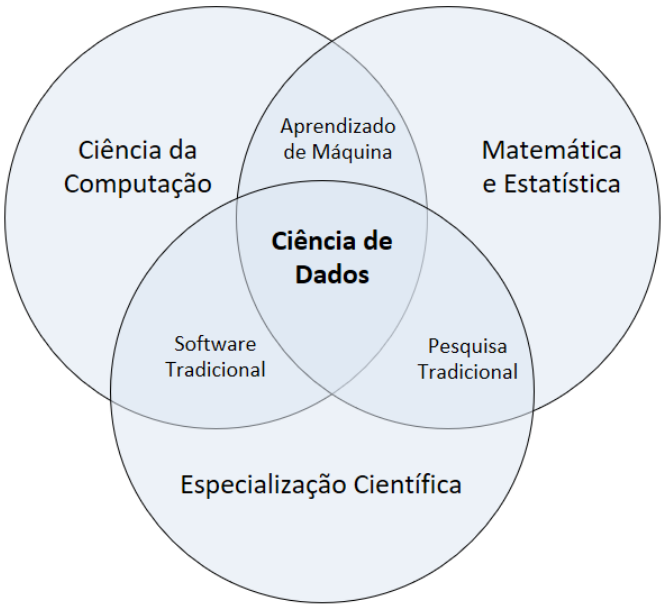


Figura 1 – Diagrama de Venn da ciência de dados (adaptado de (BAUDISCH, 2016))

Com este relacionamento, surgem algumas áreas secundárias e/ou perfis de indivíduos que não exigem ou não dispõem de todas as áreas principais. São estas:

- **Aprendizado de Máquina:** envolve a manipulação e análise de dados com cálculos matemáticos e estatísticos, as quais não exigem especialização científica para desempenhar tal função;
- **Software Tradicional:** envolve a extração e estrturação de dados por intermédio de funções já desenvolvidas, as quais não exigem o conhecimento prévio de sua implementação e, por consequência, não exigem o conhecimento de conceitos matemáticos e estatísticos para o seu uso. Porém, o manipulador dos dados deve utilizar um método científico a fim de encontrar padrões nos dados;
- **Pesquisa Tradicional:** envolve o estudo e aprofundamento na aplicação de métodos científicos para manipular e analisar dados, por intermédio de cálculos matemáticos e estatísticos. Neste caso, entretanto, os cientistas não dispõem de conhecimentos relacionados a tecnologia.

É importante observar que a área de design gráfico está ligada com cada uma das demais, pois todas têm seu fim na visualização de dados.

Segundo (AMARAL, 2016), a ciência de dados é formada pelas seguintes etapas:

(i) **Produção de Dados:** onde dados são gerados por diversos tipos de dispositivos, diversos componentes físicos de hardware e por diversas formas e mecanismos de coleta. Exemplo: digitação de um texto em um computador, acionamento de um sensor em um veículo ao ser freado, obtenção de uma imagem por uma câmera digital ao se tirar uma foto, etc;

(ii) **Armazenamento:** onde os dados obtidos são persistidos em alguma mídia específica, nos mais diversos formatos, como *XML*, texto plano, registros em um banco de dados relacional, entre outros;

(iii) **Armazenamento Analítico:** onde ocorre o processo de **ETL** (extração, transformação e carregamento ou *extract, transform and load*, em inglês) de dados e a geração de depósitos de dados corporativos voltados ao apoio à decisão chamados de *data warehouse* (DW) e *data mart* (DM);

(iv) **Análise de Dados:** onde ocorre a execução de operações que visam extrair informação e conhecimento dos dados existentes. Envolve conceitos de Matemática e Estatística, técnicas de mineração de dados (ou *data mining*, em inglês), aprendizagem computacional (ou *machine learning*, em inglês), aprendizagem profunda (ou *deep learning*, em inglês), redes neurais e inteligência artificial;

(v) **Visualização de Dados:** onde ocorre a apresentação propriamente dita, além de pequenas adequações sobre os dados a fim de serem apresentados de forma mais clara e intuitiva ao consumidor.

A figura 2 mostra uma visão panorâmica da ciência de dados, relacionando todas as etapas, com exemplos de tecnologias, plataformas e técnicas existentes.

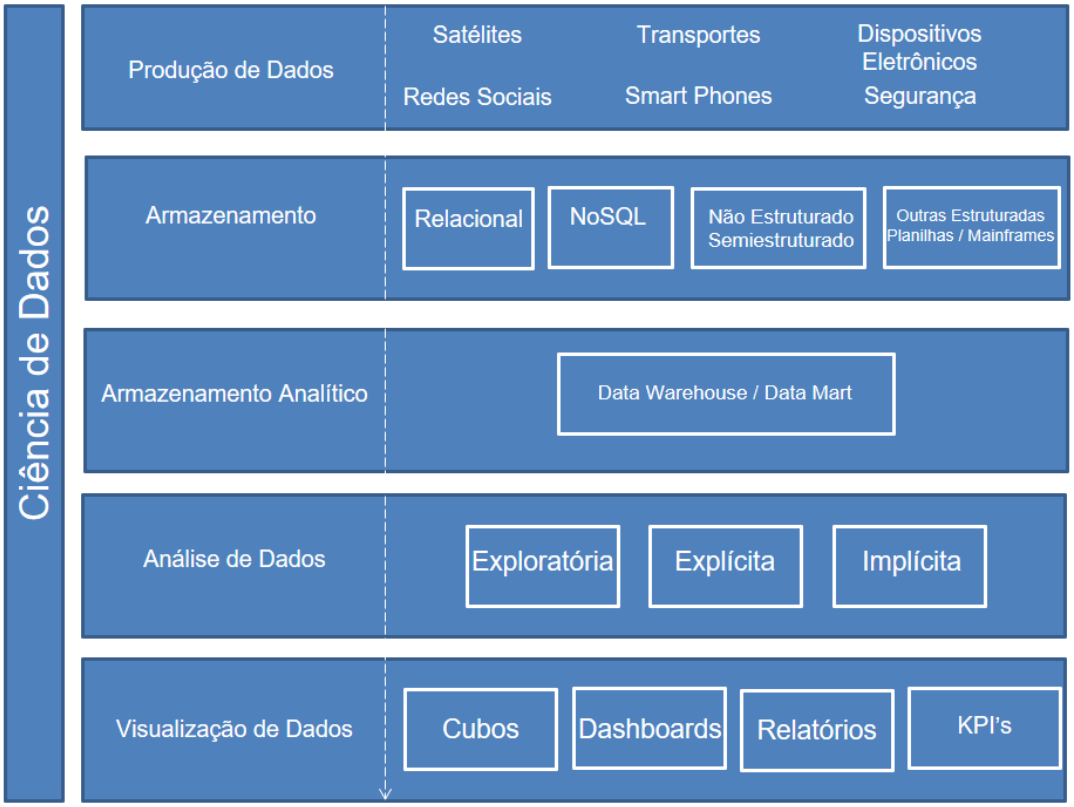


Figura 2 – Visão panorâmica da ciência de dados (adaptado de (AMARAL, 2016))

Segundo (PORTO; ZIVIANI, 2014), existem três linhas de pesquisa que estão relacionadas e são dependentes para o futuro da ciência de dados, considerando a larga escala dos dados a serem analisados bem como seu dinamismo:

- **Gestão de Dados:** envolvem desafios da representação adequada para os dados coletados, da definição de um grau de confiança num conjunto de dados específico e sua precisão, seu devido particionamento entre nós de um cluster de computadores, sobretudo sobre aplicações que acessam grandes volumes de dados e de natureza distinta das aplicações convencionais e, finalmente, do processamento de grandes volumes de dados por meio de programas *ad-hoc* para modelos de processo baseados em *dataflows*;
- **Análise de Dados:** envolve um processo de análise aplicada a dados volumosos que vai desde a seleção dos dados até a produção do conhecimento, que é o principal

produto da análise. Este processo é apoiado por técnicas e métodos baseados em mineração de dados (pré-processamento, classificação, agrupamento, associação e visualização) e em modelagem computacional (algoritmos relacionados: *k-means*, PAM, árvores de decisão, redes neurais e VSM). Um dos grandes desafios dessa linha de pesquisa é a escolha de uma plataforma mais adequada e de métodos que otimizem o funcionamento e o resultado da análise de dados. Para isso, cabe ao pesquisador formular hipóteses, com base nos dados disponíveis, e validá-las por meio das ferramentas computacionais, usando como suporte a gestão dos dados;

- **Análise de Redes Complexas:** também chamado de ciência de redes, envolve o desenvolvimento de novas ferramentas, métodos e tecnologias que visam extrair conhecimento de enormes volumes de dados de alguma maneira interconectados, explorando os modelos e estruturas das redes de comunicação, bem como os processos dinâmicos que ocorrem por meio destas.

Como aplicação da *data science*, podemos citar de modo geral três eixos: ciência, indústria e governo (PORTO; ZIVIANI, 2014). Em cada um destes, é possível visualizar uma necessidade específica onde pode ser aplicada a ciência de dados: uma forma de investigação sobre uma imensidão de dados coletados sobre observações e pesquisas científicas (ciência), a análise preditiva num ambiente empresarial de computação em nuvem (indústria) e uma forma de gerar planejamento visando a melhoria de serviços ao cidadão (governo).

Mais especificamente, existem outras áreas que está presente a necessidade da ciência de dados. São elas: saúde, petróleo, energia, financeira, esporte, astronomia, bioinformática, Internet, mobilidade urbana, defesa cibernética, comunicação móvel, biodiversidade, entre outras (PORTO; ZIVIANI, 2014).

Na figura 3 é possível visualizar uma relação direta das linhas de pesquisa com as aplicações da ciência de dados nos eixos ciência-indústria-governo.

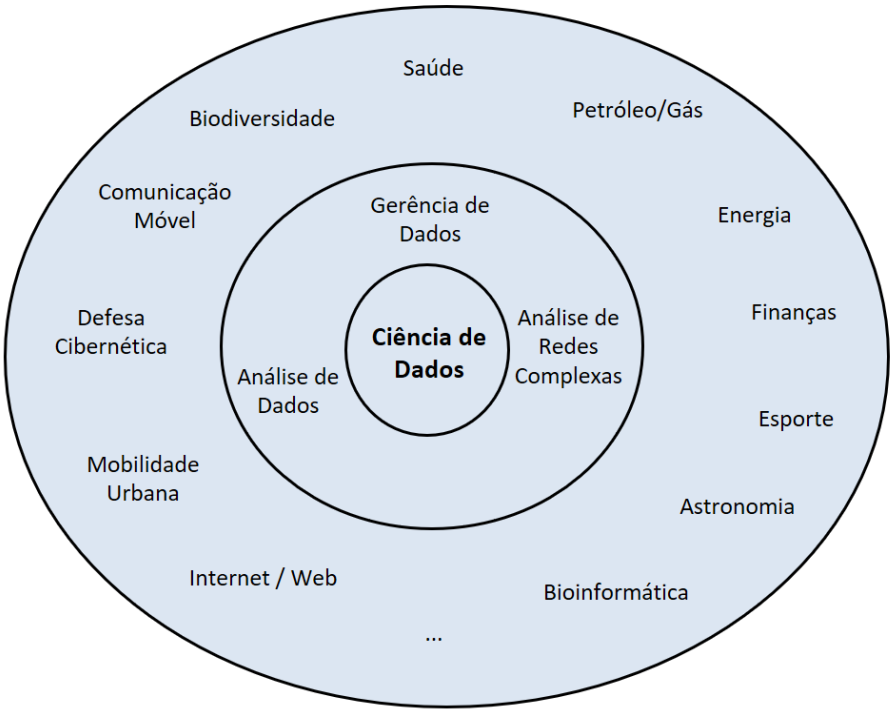


Figura 3 – Desafios da pesquisa básica em ciência de dados (adaptado de (PORTO; ZIVIANI, 2014))

2.1.1 Classificação dos Dados

No processo de ciência de dados como um todo, os dados podem assumir três denominações diferentes, conforme o seu valor e o seu significado: dado, informação e conhecimento. Estes três podem se basear em diferentes abordagens de acordo com a literatura.

Segundo (AMARAL, 2016), os três são a matéria prima dos sistemas de informação e ocorrem em momentos distintos do processo da ciência de dados: dados ocorrem no momento que são coletados, podendo ou não ser armazenados; informação ocorre na etapa de análise, trazendo algum significado consigo; e finalmente, conhecimento ocorre na interpretação da informação, na qual a mesma é compreendida e aplicada para algum fim. Esta última etapa ocorre também na análise de dados, porém num momento posterior, quando é feita a avaliação e é validado o que foi analisado em primeira instância.

Segundo (ZAIDAN, 2008), a classificação é feita levando em consideração o tamanho e o valor agregado. O dado é a matéria prima e menor unidade possível, não tem a capacidade de informar nada e de levar a conclusão alguma. Já, a informação é formada por um conjunto de dados organizados de tal forma que adquirem um valor adicional e que possuem algum significado, pois são dados trabalhados, tratados e inseridos num contexto específico. A partir da informação, é possível tomar decisões para atingir as metas de uma organização. Por fim, o conhecimento é um conjunto de informações interpretadas e

aplicadas a um cenário ou simulação, por meio de pessoas e recursos computacionais, que resultam num valor maior e num aprendizado para uma organização.

Uma forma de compreender o que significa cada termo é através do exemplo de um *Boeing* 787, que produz meio *Terabytes* de dados durante um voo (FINNEGAN, 2013). Nessa aeronave, existem alguns sensores instalados nos seus flaps, que são extensões das asas usados para aumentar a sua sustentação. Estes sensores são usados também em procedimentos de pousos. Os sinais de vibração emitidos pelos sensores são dados. O fato de os mesmos serem emitidos no momento de pouso da aeronave é uma informação. E, por fim, o fato de um flap vibrar durante o pouso é um conhecimento (AMARAL, 2016).

2.1.2 Ciclo de Vida do Dado

O ciclo de vida do dado é composto pelas mesmas etapas que ocorrem na ciência de dados, pois, conforme mencionado anteriormente, estão diretamente relacionadas (AMARAL, 2016). São elas: produção, armazenamento, transformação, armazenamento analítico, análise e descarte. As etapas intermediárias de armazenamento e transformação não necessariamente ocorrem com todas as fontes de dados, pois existem aquelas que não necessitam de transformação a partir de sua produção ou aquelas que não são armazenadas, sendo descartadas logo após sua produção. Em geral, a definição do processo dos dados depende de sua natureza e finalidade, bem como as normas corporativas e a legislação vigente.

A figura 4 ilustra a composição básica do ciclo de vida do dado.

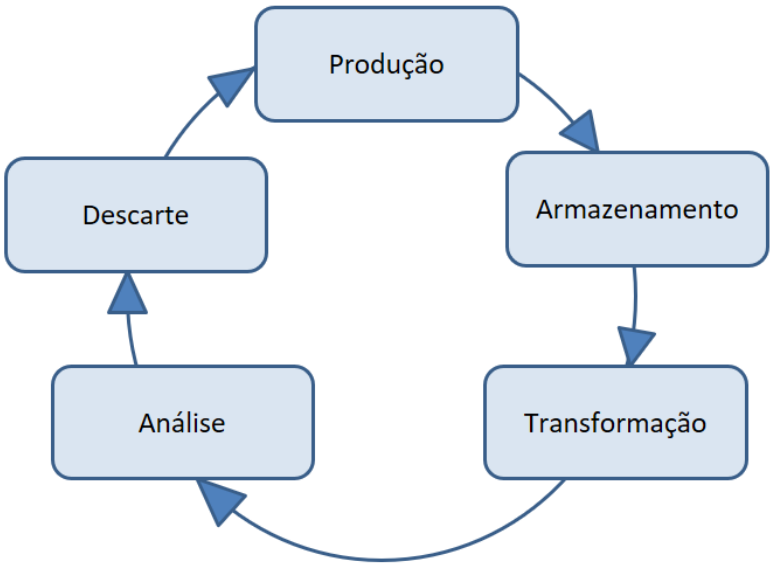


Figura 4 – Ciclo de vida do dado (adaptado de (AMARAL, 2016))

Segundo (AMARAL, 2016), esta é uma visão mais resumida do que a visão

panorâmica apresentada na figura 2, pois cada uma das etapas anteriormente comentadas estão sendo representadas nesta figura, com a exceção da etapa de visualização, visto que a mesma não está relacionada diretamente ao ciclo de vida do dado, apenas ao resultado de sua análise antecessora. Por fim, o descarte representa o momento em que o dado deixa de ser útil e, por consequência, é descartado.

2.1.3 *Big Data*

Segundo (AMARAL, 2016), ***Big Data*** é definido como um fenômeno no qual dados são produzidos em vários formatos e armazenados por uma grande quantidade de dispositivos e equipamentos. Esse evento está causando uma revolução na Era da Informação e fomentando uma nova fase da Revolução Industrial e, certamente, é um dos grandes motivadores à pesquisa e investimento na área de ciência de dados.

Conforme a definição formal, o *Big Data* é composto por cinco Vs, três básicos e comuns: volume, velocidade e variedade; e mais dois adicionais: veracidade e valor (AMARAL, 2016). Estes cinco são critérios de avaliação para um conjunto de dados.

Segundo (AMARAL, 2016), existem três principais causas para o aparecimento do fenômeno *Big Data*: o barateamento, a miniaturização e o aumento da capacidade de processamento, pois por meio destes é possível existir uma imensidão de tecnologias, equipamentos, dispositivos e processos que produzam e armazenem dados.

2.2 Análise de Dados

Conforme citado anteriormente, a **Análise de Dados** é uma das fases da ciência de dados que tem como objetivo principal a transformação de dados brutos em informação útil e conhecimento para as organizações e/ou empresas. Esta área está envolvida e dependente de conceitos de várias outras e, atualmente, está ligada, principalmente, às áreas da computação e processamento de dados, devido a presença do *Big Data*, descrito acima.

Segundo (AMARAL, 2016), existem três tipos de análise, um introdutório e dois que estão relacionados com duas abordagens distintas. São eles: exploratório, explícito e implícito. O primeiro ocorre quando o intuito é apenas conhecer os dados, saber como estão distribuídos e relacionados, bem como suas características estatísticas, através de técnicas quantitativas ou visuais. O segundo ocorre quando a informação e o conhecimento estão presentes de forma explícita nos dados, apenas sendo necessária uma operação de baixa complexidade para destacá-los. O terceiro e último ocorre quando a informação e o conhecimento não estão disponíveis claramente no conjunto de dados, fazendo-se necessária a aplicação de um conjunto de funções baseadas em aprendizagem computacional ou em

estatística. Esses dois últimos ocorrem após a triagem da primeira análise e, dessa forma, é possível escolher qual é a melhor abordagem para a resolução do problema.

Na análise implícita, existem duas áreas que trabalham com modelagem de dados e estão em alta no mercado e no ambiente acadêmico: a mineração de dados e a aprendizagem computacional. Estas apresentam características e técnicas bem parecidas e que convergem num propósito apenas: a geração de informações úteis e de conhecimento. A seguir, analisamos separadamente cada uma, apontando seus pontos principais, algumas de suas técnicas, algoritmos e fluxo de tratamento de dados, bem como suas particularidades e aplicações práticas. Finalmente, comparamos as duas, citando diferenças e semelhanças.

2.2.1 Mineração de Dados

Segundo (HAN; KAMBER, 2001), a **Mineração de Dados** (**MD**, ou **Data Mining**, em inglês) é o processo no qual é possível descobrir padrões relevantes em uma vasta quantidade de dados, que podem estar armazenados em diversas fontes de dados, seja em banco de dados, *data warehouses*, entre outras. Já (HAND; MANNILA; SMYTH, 2001), define como a análise de grandes conjuntos de dados em observação a fim de encontrar relacionamentos pré-existentes e formatar tais dados em um nível que possa ser compreendido por quem os detém.

A mineração de dados está presente em dois padrões: no Processo Padrão Genérico para Mineração de Dados (CRISP-DM, ou *Cross Industry Standard Process for Data Mining*, em inglês) e no Descoberta de Conhecimento a partir de Dados (KDD, ou *Knowledge Discovery from Data*, em inglês) (AMARAL, 2016). Ambos podem ser definidos como um processo iterativo de etapas, nas quais os dados são trabalhados, resultando em conhecimento (HAN; KAMBER, 2001). Como exemplos de aplicação que utilizam essas abordagens respectivamente, é possível citar os sistemas Clementine-SPSS e SAS-Enterprise Miner. Estes sistemas passam basicamente pelas mesmas etapas: coleta de dados, depuração e análise, podendo resultar em modelos descritivos ou até preditivos (BRAGA, 2005).

O termo KDD foi criado em 1995 com o intuito de criar um ambiente de processos, técnicas e abordagens nos quais deve ocorrer a mineração de dados, aplicando o método científico moderno aos problemas do mundo dos negócios (BRAGA, 2005). Conceitualmente, há divergência entre os autores da literatura nas definições de mineração de dados e KDD, pois há quem diga que são sinônimos, porém existem descrições de que a mineração de dados seja apenas um dos passos dentro do processo do KDD. Apesar de ser apenas um dos passos, os primeiros autores consideram como a parte mais importante do processo como um todo e por isso a alusão (CARVALHO, 2014).

Segundo (HAN; KAMBER, 2001), esse processo de descoberta é formado e descrito

por três grandes etapas:

- **Pré-Processamento de Dados:** é a etapa de preparação dos dados e tem como objetivo principal permitir uma mineração mais eficiente dos mesmos. É formada pelos seguintes passos: a extração e integração que trata os dados e visa torná-los uma fonte única e uma única estrutura como, por exemplo, um *data warehouse*; a limpeza que remove inconsistências e pequenos ruídos nos dados, os quais, possam acabar comprometendo a descoberta de padrões, levando a certos desvios e falta de precisão; a seleção que analisa os atributos ou características dos dados e visa selecionar os que possuem relevância e são mais completos para o processo de mineração; e, finalmente, a transformação que aplica operações de redução e agregação nos dados e visa a filtragem do que realmente pode determinar os padrões corretos na próxima etapa;
- **Mineração de Dados:** é a etapa composta por técnicas que executam a mineração propriamente dita e resultam na descoberta de padrões. Os padrões podem ser classificados por dois grupos principais de técnicas: preditiva e descritiva. A primeira visa induzir modelos ou teorias a partir de um conjunto de dados e é formada pelas seguintes técnicas: classificação, regressão e agrupamento (ou *clustering*, em inglês). A segunda visa a exploração de dados, analisando suas características e é formada pelas seguintes técnicas: caracterização, discriminação e regras de associação. (As técnicas citadas anteriormente utilizam algoritmos de modelagem computacional que são a base da Aprendizagem de Máquina);
- **Pós-Processamento de Dados:** após a mineração de dados e o reconhecimento de padrões nos dados, se faz necessário pequenas transformações sobre os mesmos a fim de ter como resultado, de fato, um conhecimento. É formada por dois passos: a preparação de padrões que se preocupa em determinar quais dos padrões gerados são relevantes para o contexto e a visualização do conhecimento que são representações gráficas para o usuário final para que o mesmo seja capaz de interpretar de forma simples os dados gerados.

A mineração de dados em conjunto com a análise de dados só é possível graças à presença do *data warehouse* (DW), pois este possui uma estrutura multidimensional (BRAGA, 2005), permitindo redundâncias e retirando as normalizações presentes no modelo relacional, também conhecido como Processamento de Transações em Tempo Real (OLTP, ou *Online Transaction Processing*, em inglês) (AMARAL, 2016). Essa nova estrutura criada viabiliza uma manipulação dos dados com menor complexidade e, consequentemente, com maior velocidade e menor exigência do poder de hardware.

A figura 5 descreve todo o processo iterativo do KDD, relacionando cada passo de execução com as fontes de dados de origem e destino.

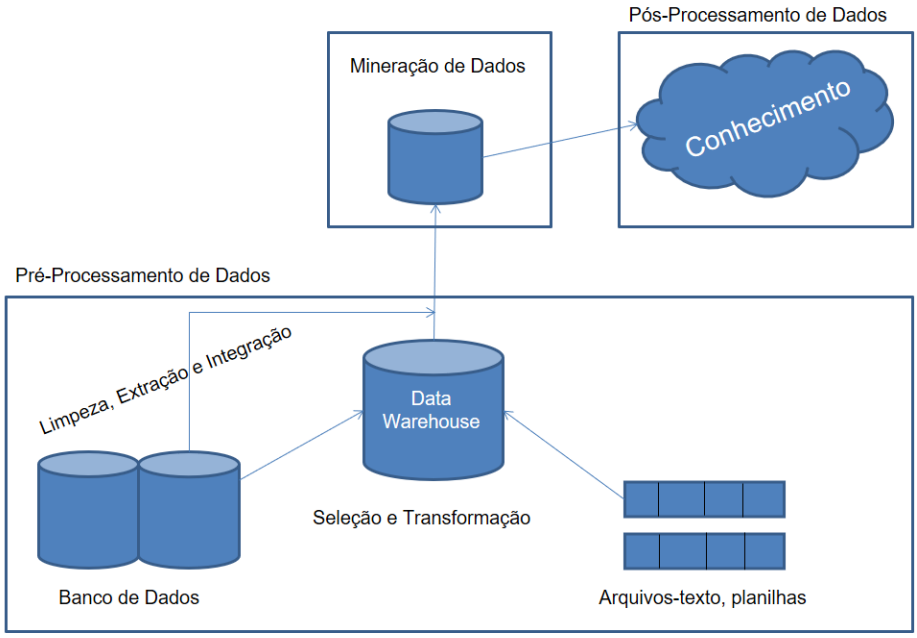


Figura 5 – Sequência de passos do processo do KDD (adaptado de (CARVALHO, 2014))

Para (BRAGA, 2005), um projeto de mineração de dados pode possuir uma sequência genérica de etapas, como segue:

(i) **Definição do Problema:** envolve a descoberta da real necessidade do usuário final, levando em consideração o ambiente de software e hardware da empresa;

(ii) **Aquisição de Avaliação dos Dados:** envolve a obtenção e a formatação dos dados, a validação destas atividades, a criação de amostras de trabalho e o particionamento dos dados;

(iii) **Extração de Características e Realce:** envolve o destaque dos atributos que podem auxiliar na resolução do problema, assim como a redução dos demais que não possuem relevância;

(iv) **Plano de Prototipagem, Prototipagem e Desenvolvimento:** envolve o desenvolvimento de hipóteses e plano de testes, a criação de protótipos e a apresentação de modelos descritivos e preditivos;

(v) **Avaliação do Modelo:** envolve a validação do protótipo e avaliação dos resultados obtidos;

(vi) **Implementação:** envolve a entrega do produto final;

(vii) **Avaliação do Retorno do Investimento (Pós-Projeto):** envolve o papel da gerência de avaliar se as mudanças consequentes do projeto tiveram um ganho material, em forma de maior receita ou maior lucro considerável para a empresa.

Em outras palavras, a mineração de dados está relacionada com duas das principais etapas do ciclo de vida do dado e, consequentemente, da ciência de dados: o armazenamento analítico e a análise de dados, pois está envolvida com a preparação dos dados, passando pela etapa de reconhecimento de padrões até a geração e a utilização do conhecimento para os processos de negócios internos e externos da organização e/ou empresa.

A construção de um modelo no processo do KDD deve refletir ao paradigma no qual vivem as empresas num determinado período. Conforme o antigo paradigma, o negócio era dividido em áreas funcionais: marketing, finanças, engenharia, produção, etc. Hoje em dia, é dado mais ênfase ao lado do cliente, criando um ambiente centralizado no mesmo. Esta mudança se deu por três principais contribuições: o surgimento da lealdade/assiduidade do cliente, a classificação de segmentos de clientes para nichos específicos e a oferta da customização radical do produto/serviço ao cliente (BRAGA, 2005).

Como aplicações da MD no mercado brasileiro, é possível citar algumas áreas (GOLDSCHMIDT; BEZERRA, 2016): telecomunicações (classificação de clientes de acordo com seu potencial de compra de serviços), comércio (associação de produtos que são vendidos de forma conjunta e frequente numa loja, com base na captação de dados num determinado período e num determinado ponto de vendas, a fim de oferecer promoções e estimular a venda combinada de certos itens), finanças (geração de um modelo de classificação para clientes que são pontuais com relação a pagamento, pagam com atraso e que não quitam as suas dívidas; construção e avaliação de modelos de predição de séries temporais a partir do histórico de cotações de ações na bolsa de valores; ou ainda o desenvolvimento de mecanismos de detecção de fraudes em compras de cartão de crédito a partir do comportamento prévio de cada cliente), medicina (extração de conhecimento a partir de imagens e dados sobre patologias para detecção e prevenção de doenças), ação social (análise de dados de moradores de rua e pessoas necessitadas para a recomendação de programas de reintegração social mais adequados a elas, avaliando se as mesmas foram ou não reintegradas à sociedade), educação (análise de dados coletados da utilização de laptops introduzidos em programas sociais nas escolas a fim de avaliar o efeito no aprendizado dos alunos), energia (realização de previsão de demanda de consumo de energia elétrica por regiões com base nos períodos anteriores), indústria (realização de previsão de demanda para o consumo de produtos industrializados), seguros (desenvolvimento de um sistema que sugere um valor de apólice compatível com o perfil do usuário; ou ainda a construção de um modelo preditivo para a detecção de fraudes, com base nas ocorrências), arrecadação de impostos (construção de modelos de conhecimento de identificação de futuras fraudes no pagamento de impostos, com base no histórico de comportamento do indivíduo), comportamentos de redes sociais online (construção de modelos de análise que visam compreender o comportamento dos usuários nas redes sociais, tais como opinião e sentimento, com base no histórico de interações ou comentários), entre outras.

Dentro da área de MD, existe uma área específica denominada mineração de textos (MT, ou *text mining*, em inglês). Esta, como o próprio nome já diz, trabalha com identificação de padrões em textos, indexando palavras chaves que se aplicam a motores de busca e a sistemas de recomendação (TAKAHASHI, 2015; TAN, 2017). Este tema será abordado com mais detalhes nos próximos tópicos referentes a técnicas de sistemas de recomendação.

2.2.2 Aprendizagem Computacional

A **Aprendizagem Computacional** ou **Aprendizado de Máquina** (**AM**, ou ***Machine Learning***, em inglês) é uma área derivada da inteligência artificial e baseada na ciência da computação e estatística que tem como objetivo principal aprender com base em experiências passadas, assim como um ser humano possui esta capacidade. Esta necessidade não é algo novo, pois desde o surgimento dos primeiros computadores na década de 40, como o ENIAC, havia esta preocupação (TAKAHASHI, 2015; CARVALHO, 2014). Além das áreas citadas acima, está relacionada também à ciência de dados em geral e, especificamente, à análise de dados implícita (AMARAL, 2016), conforme descrito anteriormente.

Segundo a *SAS* (Sistema de Análise Estatística) (SAS, 2016), a aprendizagem computacional é um método de análise de dados que automatiza o desenvolvimento de modelos analíticos, no qual, usa algoritmos que aprendem interativamente a partir de uma instância de dados. Dessa forma, é possível encontrar padrões ocultos, os quais não seria possível por meio de análises explícitas (AMARAL, 2016). Para (MITCHELL, 1997), a AM pode ser definida como o estudo de métodos computacionais com o intuito de gerar novos conhecimentos, novas habilidades e novos meios de organizar o conhecimento já existente.

Formalmente, um sistema de aprendizagem computacional é capaz de aprender com uma experiência E, sobre um grupo de tarefas T e índice de performance P, se a medida P em T aumenta com E (MITCHELL, 1997). Em outras palavras, envolve o uso de algoritmos e técnicas de inteligência artificial e otimização aplicados sobre uma massa de dados que otimizam o desempenho e a precisão de um sistema, levando em consideração uma taxa de erro, com base na experiência obtida por conjuntos de treinamento (TAKAHASHI, 2015).

Uma das principais características dos algoritmos usados na aprendizagem computacional é poder generalizar e criar regras a partir de uma instância de um conjunto de dados de treinamento, de tal forma que gere um conhecimento além dos dados já existentes (CARVALHO, 2014). Porém, com a existência da preocupação eminente de que o modelo gerado na etapa de treinamento possa não ser eficiente para novas entradas, é necessário manter um conjunto de dados separado para a execução de novos testes, a fim de avaliar e garantir a precisão do modelo e de realizar mudanças, caso haja um *feedback* negativo

(DOMINGOS, 2012; CARVALHO, 2014).

Conforme a literatura, a aprendizagem computacional, em geral, pode ser formada pelas seguintes abordagens e níveis de *feedback* possíveis(SAS, 2016; REZENDE, 2003; NORVIG; RUSSEL, 2010):

- **Supervisionada:** ocorre quando um agente dispõe de entradas e saídas e, por meio destas, cria um mapeamento em formato de uma função, de modo que, para cada entrada é resultada em sua saída correspondente. É baseada no aprendizado obtido pelo treinamento de instâncias de dados já classificadas. Dessa forma, é possível induzir um modelo que seja capaz de classificar novas instâncias de forma precisa. Esta abordagem pode ser chamada também de classificador e é utilizada para a realização de predições (CARVALHO, 2014). Para isso, se faz necessário a preparação dos dados descrita no processo de KDD e mineração de dados, citados anteriormente. As técnicas mais utilizadas são a classificação e a regressão, a primeira servindo para valores discretos ou particionados por classes e, a segunda para valores numéricos contínuos. Os principais algoritmos utilizados são: regressão linear, k-Vizinho mais próximo ou *k-nearest neighbor*, redes neurais, árvores de decisão e redes bayesianas. Em termos de uso, é representado como cerca de 70% das abordagens utilizadas (SAS, 2016);

- **Não-Supervisionada:** ocorre quando um agente aprende padrões em dados não categorizados, ou seja, não há a presença de dados de saída, apenas de entrada. Portanto, esta abordagem é mais usada para gerar modelos descritivos (CARVALHO, 2014). A técnica mais comum para esta abordagem é o agrupamento ou *clustering* que tem como objetivo principal detectar potenciais classes a partir de um conjunto de entradas como exemplo. O algoritmo mais utilizado é o k-médias ou *k-means*. Em termos de uso, é representado de 10 a 20% das abordagens utilizadas (SAS, 2016);

- **Semi-Supervisionada:** ocorre quando existem poucos dados rotulados e muitos não rotulados ou quando os dados classificados não são totalmente confiáveis. Neste caso, se faz necessário tanto a abordagem supervisionada quanto a não supervisionada, a fim de garantir a confiabilidade dos dados rotulados. Em termos de custo e complexidade dos algoritmos, existem casos que é mais indicado o uso desta abordagem do que a supervisionada;

- **Por Reforço:** ocorre quando um agente aprende por uma série de recompensas e punições. Ou seja, o algoritmo descobre por tentativa e erro quais ações geram a maior quantidade de recompensa e, assim, quais devem ser priorizadas. Nesta abordagem, existem três principais componentes: o agente, o ambiente e as ações, objetivando na escolha das ações que maximizem a recompensa ou minimizem a

punição, conforme um resultado final esperado, ao longo de um determinado período de tempo.

Como aplicações de AM, temos os seguintes exemplos (AMARAL, 2016; TAKAHASHI, 2015; SAS, 2016): sistemas de recomendação, detecção de fraudes, resultados de pesquisa na *Web*, anúncios em tempo real em páginas da *Web* e dispositivos móveis, análise de sentimento baseada em texto, pontuação de crédito e próximas melhores ofertas, previsão de falhas em equipamento, novos modelos de precificação, detecção de invasão na rede, reconhecimento de padrões e imagem, reconhecimento de fala, filtragem de *spans* no *e–mail*, ferramentas de busca, segmentação de clientes, entre outros.

Uma das aplicações citadas acima são os sistemas de recomendação. Estes, em específico, são muito comuns na atualidade e focaremos nesse tema nos tópicos seguintes, descrevendo com detalhes e explicando com mais profundidade uma das técnicas usadas neste trabalho: a filtragem de informações baseada em conteúdo. O foco deste trabalho é o uso desta técnica para sistemas de bibliotecas universitárias, bem como aplicado aos dados armazenados referentes a usuários, itens e movimentações.

2.2.3 Mineração de Dados e Aprendizagem Computacional

Após a descrição e a explicação dos temas mineração de dados e aprendizagem computacional, foi possível perceber as semelhanças entre estes, pois possuem características e objetivos convergentes, bem como as mesmas aplicações em cenários paralelos. A literatura é divergente com relação à diferenciação dos assuntos. Este tópico trata de algumas opiniões e descrições dos autores referentes às diferenças e similaridades que existem entre os termos em questão.

É possível citar um ponto em comum relacionado à contextualização e história: ambas as áreas ganharam muita força desde o surgimento da Era da Informação, na qual a tecnologia passou a ocupar um espaço maior em nossas vidas (CARVALHO, 2014). Segundo a descrição de (AMARAL, 2016), os termos AM e MD estão diretamente relacionados, possuem o mesmo significado, porém com diferenças sutis: enquanto o primeiro trata de algoritmos que buscam reconhecer padrões em dados, o segundo é a aplicação destes em grandes conjuntos de dados em busca de informação e conhecimento para as empresas. Um dos grandes motivadores do estudo da AM foi o fenômeno do *Big Data* que levantou a necessidade de produzir informação e conhecimento com base num grande volume de dados. Conforme (CARVALHO, 2014), existe uma correlação entre a análise preditiva da MD com a abordagem supervisionada da AM e a análise descritiva da MD com a abordagem não-supervisionada da AM, pois o objetivo da primeira abordagem é fazer predições com base nos dados classificados (entrada e saída), enquanto a segunda possui como objetivo principal a classificação dos dados (apenas entradas), tendo uma característica mais

descritiva. Já (SAS, 2016) explica que os dois temas possuem focos diferentes: enquanto a MD descobre padrões e conhecimento previamente desconhecidos, a AM reproduz padrões e conhecimentos já adquiridos e os aplica sobre outros dados, gerando novas informações úteis para as tomadas de decisões e ações empresariais.

Conforme (SINGH, 2014) descreve, as diferenças entre MD e AM estão em alguns parâmetros. Seguem os principais:

- **Definição:** a primeira é o processo de extração de informação de um conjunto de dados e de transformação para uma estrutura coerente para a posterior tomada de decisões, enquanto que a segunda é formada por algoritmos de análise de dados que visam a construção e estudo de sistemas que possam aprender a partir de exemplos de instâncias de dados;
- **Foco:** a primeira está focada na descoberta de padrões desconhecidos nos dados, enquanto que a segunda está focada no processo de predição, com base em características conhecidas e treinadas a partir dos dados de entrada (similar a descrição de (SAS, 2016));
- **Tamanho da base de dados:** a primeira é aplicada sobre grandes bases de dados num processo automático ou semi-automático, enquanto que a segunda é aplicada em pequenas quantidades de dados a fim de aumentar a precisão dos algoritmos e das técnicas;
- **Classificação:** a primeira é classificada como caráter descritivo ou preditivo, enquanto que a segunda é classificada como abordagem supervisionada, não-supervisionada, semi-supervisionada e por reforço (como citados anteriormente).

2.3 Sistemas de Recomendação

A Era da Informação possui esse nome graças ao crescimento sem proporções da produção de dados por diversos dispositivos e pelo aumento considerável do número de usuários que acessam a *Internet*, tornando os termos ciência de dados, análise de dados, aprendizagem computacional e mineração de dados em ascensão e evidência no ambiente corporativo e acadêmico. Esses fatores criaram também a necessidade e o desafio de apresentar ao usuário apenas aquilo que realmente lhe interessa e que possui, de fato, relevância ao seu contexto. Visando a resolução do problema, ainda que parcialmente, foi criada uma série de motores de busca, como o *Google*, *DevilFinder* e *Altavista*, porém os mesmos não se preocupavam com a customização para cada usuário e a apresentação prévia de suas necessidades, apenas encurtavam o caminho de suas pesquisas. Além disso, quase sempre, um indivíduo não era capaz de realizar escolhas dentre as diversas alternativas

que lhe eram dadas, se fazendo necessário meios que auxiliassem nestas escolhas, sejam pessoas próximas a ele ou comentários e sugestões de desconhecidos que já passaram pela mesma experiência. Por isso, houve a necessidade do desenvolvimento de sistemas de recomendação, que tem como característica principal filtrar esta sobrecarga de dados na Web, de forma automática, a fim de apresentar apenas o que é de preferência de cada usuário (REATEGUI; CAZARELLA, 2005; ISINKAYE; FOLAJIMI; OJOKOH, 2015).

Os **Sistemas de Recomendação** (**SR**) podem ser definidos como uma estratégia que visa decidir automaticamente e aproximar informações úteis aos usuários num cenário onde há um grande volume de dados e, consequentemente, uma grande complexidade de acesso ao que realmente é de seu interesse (YATES; NETO, 1999). Também pode ser definido como uma ferramenta que auxilia aos usuários na busca de informações de real preferência e interesse, frente a um mar de conhecimento, provendo um conteúdo personalizado e exclusivo e, trazendo como consequência uma satisfação bilateral: tanto das empresas prestadoras de serviço quanto dos usuários(ISINKAYE; FOLAJIMI; OJOKOH, 2015; BALABANOVIC; SHOHAM, 1997).

Em termos funcionais, os SR apresentam uma interface na qual ocorre, internamente, o desempenho de duas tecnologias principais: a filtragem e a recuperação de informações, visando a predição dos itens ou partes da informação, as quais o usuário possui um real interesse (LOPES, 2007) e, desta forma, acabam contribuindo para o aumento da capacidade e eficácia do processo de indicação que ocorre, geralmente, na relação social entre os seres humanos (REATEGUI; CAZARELLA, 2005). Estes sistemas usam a modelagem de dados e a aplicação de algoritmos de AM para atribuir uma nota (ou *rating*, em inglês) a cada item de um conjunto de dados que representa o grau de preferência e interesse do usuário sobre o mesmo (RICCI et al., 2011). Além disso, geralmente, estão disponíveis três tipos de dados: informações do item (descrição em forma de texto de cada item), do usuário (receptor das recomendações) e transacionais (históricos de interações dos usuários com os itens) (LOPES, 2007). Em outras palavras, o problema da recomendação é composto pelas entidades usuários e itens e o objetivo é recomendar estes com melhor nota e melhor avaliação àqueles (TAKAHASHI, 2015). Uma hipótese para resolver esta questão é desenvolver um sistema que recebe recomendações de usuários e envia as mesmas aos demais que compartilham dos mesmos interesses, como ocorre na vida real. Um dos grandes desafios para o funcionamento deste é descobrir o relacionamento de interesses entre os usuários (REATEGUI; CAZARELLA, 2005).

O processo de operação e funcionamento dos sistemas de recomendação possui algumas fases. Seguem as principais (ISINKAYE; FOLAJIMI; OJOKOH, 2015):

(i) **Coleta de Informações:** é a fase responsável por coletar informações relevantes sobre o usuário a fim de criar um perfil ou um modelo de predição que inclui atri-

butos, comportamentos ou conteúdos acessados pelo mesmo. Esta etapa pode ser dividida em duas menores: identificação e coleta. Existem duas formas principais para a identificação de usuários: via servidor, na qual, geralmente, é disponibilizado ao usuário um formulário de cadastro com informações pessoais para que o mesmo preencha e, posteriormente, acesse o sistema, entrando com uma forma de identificação personalizada como, por exemplo, *login* e senha, ou via cliente, na qual, é utilizado um mecanismo que identifica o usuário diretamente pela máquina de acesso, salvando as páginas visitadas, também denominado *cookies*. A primeira forma de acesso é, sem dúvida, mais confiável, pois cria um ambiente exclusivo de acesso ao próprio usuário (REATEGUI; CAZARELLA, 2005). Após o processo de identificação, é possível coletar informações sobre o usuário de três maneiras (HERLOCKER; KONSTAN, 2000): explícita, implícita e híbrida. Na primeira, o sistema solicita ao usuário que especifique suas preferências. Na segunda, o sistema infere as preferências e necessidades do usuário com base no monitoramento de suas ações e seu comportamento dentro do sistema. Por fim, na terceira, as duas formas são combinadas visando a minimização de suas fraquezas e o alcance de um desempenho e uma performance melhor;

(ii) **Aprendizado:** esta fase é constituída pela aplicação de um ou mais algoritmos de AM que visam filtrar e explorar as características do usuário coletadas na fase anterior;

(iii) **Predição e Recomendação:** nesta fase ocorre a predição dos tipos de itens que, possivelmente, serão de preferência do usuário, com base na consolidação dos dados coletados na primeira fase com os resultados dos algoritmos da fase anterior;

(iv) ***Feedback:*** por fim, ocorre a fase de avaliação, na qual, o usuário reage a uma recomendação e o sistema recebe um *feedback*, servindo de entrada para um novo ciclo, no qual, as informações estão mais refinadas e mais próximas da realidade sobre este indivíduo. Ou seja, os dados recebidos fazem parte da coleta de informações para uma nova análise. Esta fase pode ocorrer de duas formas: explícita e implícita. Na primeira, o usuário pode indicar se o item foi de real interesse ou não ao usuário. Segundo (RICCI et al., 2011), existem três formas de receber esse *feedback* do usuário: com os valores gostei ou não gostei, por notas, ou ainda por meio de comentários em texto. Este último tem uma complexidade maior, pois requerem técnicas de processamento de linguagem natural, a fim de julgar e decidir se o texto semanticamente é positivo ou negativo. Na forma implícita são observadas as ações tomadas pelo usuário logo após o momento da recomendação do item, dispensando o envolvimento do usuário.

A figura 6 ilustra este processo iterativo de modo geral, apresentando o relacionamento das fases de um SR.

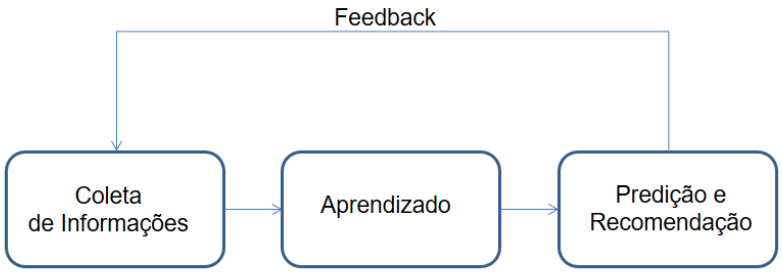


Figura 6 – Fases de um sistema de recomendação (adaptado de (ISINKAYE; FOLAJIMI; OJOKOH, 2015))

As estratégias mais utilizadas na aplicação destes sistemas são (REATEGUI; CAZARELLA, 2005): listas de recomendação (criar uma série de listas de itens organizados por interesses, como os mais vendidos ou os indicados para presentes), avaliações de usuários (comentários e opiniões sobre itens adquiridos), recomendações personalizadas a usuários (através de preferências implícitas ou explícitas), "usuários que se interessaram por X também se interessaram por Y" (através de associações de itens avaliados por usuários) e associação por conteúdo (relação entre o conteúdo dos itens). Em geral, os SR são aplicados em *websites* de comércio eletrônico (TAKAHASHI, 2015; REATEGUI; CAZARELLA, 2005).

Para o desempenho das estratégias mencionadas acima, são necessárias técnicas que identifiquem padrões de comportamento. Estas estão divididas em dois grandes grupos: filtragem de informações e Descoberta de Conhecimento em Base de Dados (DCBD) (REATEGUI; CAZARELLA, 2005). A primeira está relacionada à área de aprendizagem computacional e trabalha com algoritmos específicos organizados num processo de filtragem a fim de entregar previamente às pessoas as informações que realmente necessitam, diferentemente do processo que ocorre na recuperação de informações. A segunda utiliza técnicas de mineração de dados, tais como: regras de associação, classificação e agrupamento, conforme citadas acima, sendo cada uma mais eficiente num determinado tipo de aplicação (REATEGUI; CAZARELLA, 2005). Visando o interesse e o foco deste trabalho, detalhamos a seguir apenas o primeiro grupo.

De acordo com a literatura, a filtragem de informações se dá de algumas maneiras e pelos seguintes tipos de algoritmos (HERLOCKER; KONSTAN, 2000):

- **Filtragem Colaborativa (CF, ou *Collaborative Filtering*, em inglês):** é a técnica, na qual, lida com as informações transacionais (LOPES, 2007) e recomenda um item a um usuário específico, com base na avaliação de outros usuários. Esta,

de fato, desempenha a estratégia de "Quem comprou X também comprou Y" (REATEGUI; CAZARELLA, 2005), comentada acima e, por estar relacionada única e exclusivamente com os dados de interação dos usuários com os itens, por isso, é a mais comum entre os sistemas, pois permite uma implementação fácil e rápida (TAKAHASHI, 2015). As principais vantagens dessa técnica são: a possibilidade de apresentar ao usuário recomendações inesperadas, pois este sequer demonstrava interesse em suas buscas num item específico, o qual foi recomendado através da experiência de outro usuário; e a possível formação de comunidades de usuários pela identificação de interesses em comum. Como desvantagens, podemos citar: a impossibilidade de recomendação de novos itens, os quais ainda não foram interagidos com nenhum usuário anteriormente; cenários em que a quantidade de usuários seja muito pequena em relação ao volume de itens na base de dados, tornando as avaliações muito esparsas; e num outro cenário onde o usuário possui gostos incomuns e, dessa forma, dificilmente, terá boas recomendações (REATEGUI; CAZARELLA, 2005);

- **Filtragem Baseada em Conteúdo (CBF, ou *Content-Based Filtering*, em inglês):** é a técnica, na qual, é analisado o conteúdo de cada item, a fim de criar um relacionamento entre os mesmos, bem como relacioná-los aos interesses de cada usuário, com base em seu perfil, visando verificar se o item possui ou não relevância ao usuário (BALABANOVIC; SHOHAM, 1997). Em outras palavras, são recomendados itens similares àqueles que o usuário adquiriu ou interagiu no passado (TAKAHASHI, 2015). Portanto, esta técnica desempenha a estratégia de associação por conteúdo, citada acima. Como vantagens, é possível apontar os seguintes cenários: novos itens, poucos usuários cadastrados ou poucas avaliações. Como limitações desta, temos: a dificuldade de analisar o conteúdo de itens pouco estruturados ou com sinônimos e palavras que possuem duplo sentido; e a dificuldade de classificar uma informação textual como uma avaliação positiva ou negativa (REATEGUI; CAZARELLA, 2005);

- **Filtragem Híbrida (HF, ou *Hybrid Filtering*, em inglês):** esta técnica procura combinar as duas anteriores, agregando o ponto forte que cada uma oferece e eliminando suas fraquezas (REATEGUI; CAZARELLA, 2005). Esta combinação pode ser feita de algumas formas: executando as duas separadamente e juntando posteriormente os resultados obtidos; incluindo o método de uma na implementação da outra; ou, simplesmente, unificando ambas num mesmo sistema de recomendação (TAKAHASHI, 2015). A figura 7 ilustra a união dos conjuntos de ambos os tipos de filtragem e o ganho que se tem com a presença dos dois juntos.

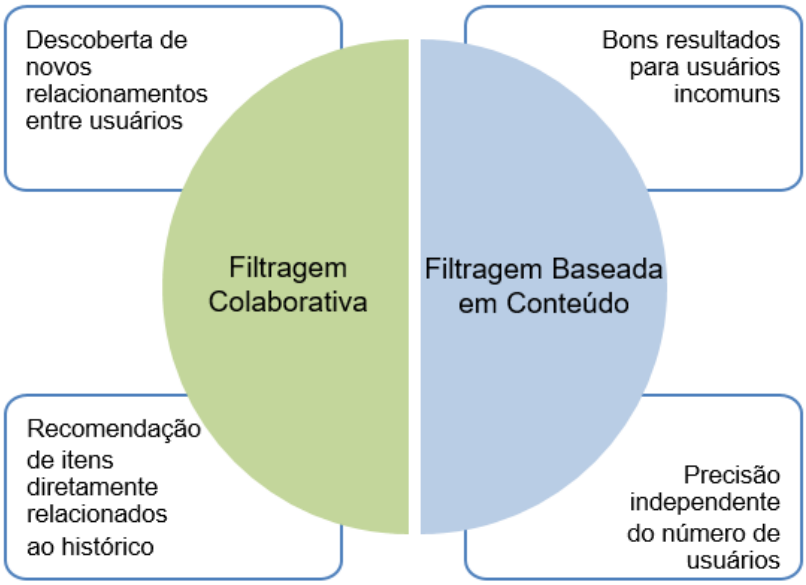


Figura 7 – Características herdadas pela filtragem híbrida (adaptado de (REATEGUI; CAZARELLA, 2005))

Alguns autores citam também mais dois tipos de filtragem: Demográfica e Contextual. A primeira técnica explora os dados de um indivíduo, atribuindo a ele um tipo de estereótipo, a fim de relacionar um item, em específico, com um tipo de indivíduo, com base nas necessidades de um grupo de usuários (MONTANER; LóPEZ; ROSA, 2003). A segunda agrega também o contexto, no qual, levou a uma determinada ocorrência de interação do usuário com um item, em específico, e pode ser representado na seguinte função: *usuário x item x contexto*, resultando numa nota de avaliação ou *rating* por interação (ADOMAVICIUS; TUZHILIN, 2010).

Vistas as técnicas para SR acima, podem ser gerados diferentes tipos de arquiteturas e diferentes aspectos de implementação podem ser aplicados aos Sistemas de Recomendação. Portanto, é possível o desenvolvimento de dois tipos de sistemas: aqueles que utilizam uma etapa intermediária de mineração de dados e aqueles que apenas aplicam algoritmos de AM (filtragem de informações) para obter o resultado final, sem a necessidade de passar pela MD.

A seguir, apresentamos a técnica que é o foco deste trabalho: a filtragem baseada em conteúdo, definimos e descrevemos a mesma, comentando suas fórmulas e uma forma de implementação e modelagem dos dados para a execução da efetiva recomendação. E, logo em seguida, trazemos alguns conceitos do cenário no qual será aplicada a técnica: os sistemas de bibliotecas universitárias.

2.4 Sistemas de Recomendação Baseada em Conteúdo

Os **Sistemas de Recomendação Baseada em Conteúdo**, basicamente, são sistemas que utilizam como técnica de recomendação principal a Filtragem Baseada em Conteúdo. Esta técnica possui este nome por desenvolver a filtragem baseada na análise dos conteúdos dos itens (LOPES, 2007) e, para tanto, possuem um conjunto de algoritmos que analisam o grau de proximidade entre os itens, a fim de gerar predições aos perfis de usuários. Estes, por sua vez, são baseados na coleta de informações de forma implícita, explícita ou híbrida (REATEGUI; CAZARELLA, 2005; HERLOCKER; KONSTAN, 2000), conforme mencionado anteriormente. O processo de recomendação desta técnica consiste em combinar os atributos dos usuários com os dos itens que descrevam suas preferências e abordem seus temas centrais, respectivamente, resultando numa ordem de grandeza que represente o interesse e relevância dos próprios usuários aos objetos combinados (RICCI et al., 2011; ISINKAYE; FOLAJIMI; OJOKOH, 2015; HERLOCKER; KONSTAN, 2000).

Os sistemas que desenvolvem a técnica baseada em conteúdo possuem aplicação e um melhor desempenho quando há o envolvimento de dados não estruturados e que estejam em formato de texto, tais como páginas *Web*, publicações, notícias (ISINKAYE; FOLAJIMI; OJOKOH, 2015) ou até mesmo campos descritivos numa tabela de banco de dados, como é o caso dos modelos clássicos utilizados para a recuperação de informação (**RI**), os quais introduziram o conceito de análise baseada em conteúdo (HERLOCKER; KONSTAN, 2000). As diferenças principais entre os dois modelos é que no primeiro caso a análise ocorre de forma estática e sua necessidade está atrelada a um momento e contexto específico, enquanto que no segundo ocorre de forma dinâmica e está relacionada a um conjunto de interesses constante do usuário (REATEGUI; CAZARELLA, 2005). É possível ainda aplicar o CBF sobre um sistema de RI, como, por exemplo, motores de busca que aplicam este conceito de recomendação e, dessa forma, são capazes de apresentar apenas os resultados de pesquisa que são de real interesse do usuário e, ao mesmo tempo, previnem que apareça informação desnecessária ao mesmo (RICCI et al., 2011).

O processo de funcionamento dos sistemas que aplicam a CBF possui, basicamente, as mesmas fases de um SR, embora haja diferenças sutis. É possível descrever o processo da seguinte forma: os sistemas coletam informações de uma base de dados relativa a itens, usuários e informações transacionais, passam por uma fase de processamento, na qual, são calculados o grau de similaridade entre os itens, gerando a recomendação para os usuários, com base em suas interações passadas (ISINKAYE; FOLAJIMI; OJOKOH, 2015) e, por fim, os usuários avaliam a recomendação com um *feedback* positivo ou negativo e, dessa maneira, é possível que o algoritmo aprenda e refina sua classificação. A figura 8 ilustra este processo.

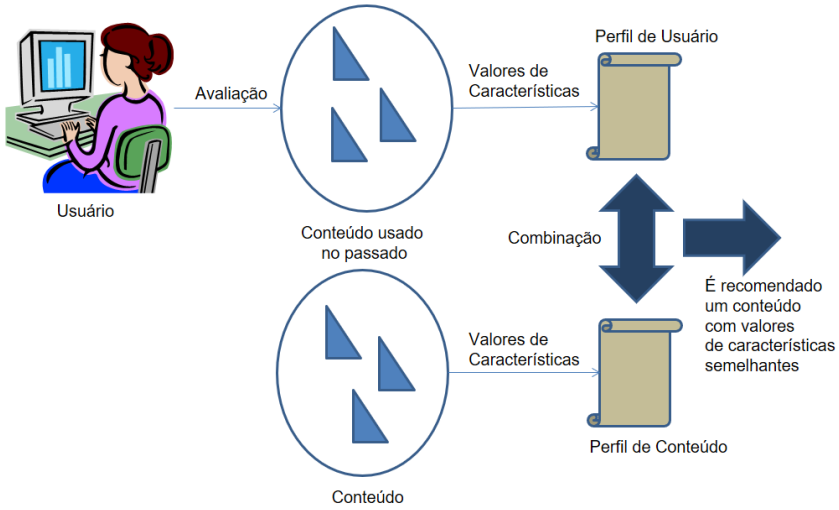


Figura 8 – Processo de um sistema de recomendação baseada em conteúdo (adaptado de (TUAN, 2012))

Na prática, estes sistemas aplicam algoritmos de abordagem tanto supervisionada, quanto não-supervisionada, pois, em primeira instância, os itens não estão classificados de acordo com o seu grau de similaridade, fazendo-se necessária a aplicação de algoritmos descritivos e não supervisionados. Posteriormente, as recomendações são realizadas e é recebido um *feedback* do usuário avaliando a classificação processada pelos primeiros algoritmos, gerando um processo de aprendizado para estes. Por outro lado, existem dados transacionais na base de dados que relacionam itens a usuários e, desta forma, é possível usar algoritmos supervisionados e treinar esses vínculos, criando predições, as quais resultam em forma de recomendação ao usuário. Em resumo, existem duas espécies de classificação na base de dados de um sistema de recomendação baseado em conteúdo: o relacionamento item a item (não supervisionado) e o relacionamento usuário a item (supervisionado).

Os sistemas de recomendação que desenvolvem a técnica de CBF, portanto, possuem três necessidades principais e desafios no que diz respeito ao processo de desenvolvimento: representar os itens; produzir ou gerar os perfis de usuários; e comparar o perfil de usuário com a representação dos itens. Ou seja, é necessário criar formas de representação de modelos, bem como estratégias de relacionamento entre os modelos criados. O autor (RICCI et al., 2011) propõe uma arquitetura em alto nível formada por três etapas principais, onde cada uma trata de uma necessidade em específico. São elas:

(i) **Analisador de Conteúdo (ou *Content Analyser*, em inglês):** esta etapa é responsável pelo pré-processamento dos dados não estruturados, tais como documentos, páginas *Web*, notícias ou descrição de produtos. Nesta etapa ocorre a transformação e tratamento dos dados a fim de gerar um modelo de representação, no qual, possa

ser analisado pela fase seguinte;

(ii) **Aprendizado de Perfil (ou *Profile Learner*, em inglês):** esta etapa é responsável por receber os dados que representam as preferências do usuário e criar um modelo generalizado, visando a formação do perfil de usuário. Esta tarefa ocorre por meio da aplicação de algoritmos de aprendizagem computacional sobre o conjunto de dados de *feedbacks* positivos e negativos enviados pelo usuário, no qual, os dados são treinados e é obtido um aprendizado sobre o perfil do usuário. Com base nesta análise, é possível predizer novas recomendações aos usuários;

(iii) **Filtragem de Componente (ou *Filtering Component*, em inglês):** esta etapa é responsável por cruzar e correlacionar as informações presentes nas representações de perfil de usuário com as de itens geradas nas fases anteriores e, com isso, sugerir, de fato, itens relevantes ao usuário.

A proposição de uma arquitetura é importante, pois serve de base para a implementação de sistemas. A figura 9 ilustra as etapas da arquitetura mencionada acima, apresentando as estruturas e componentes envolvidos.

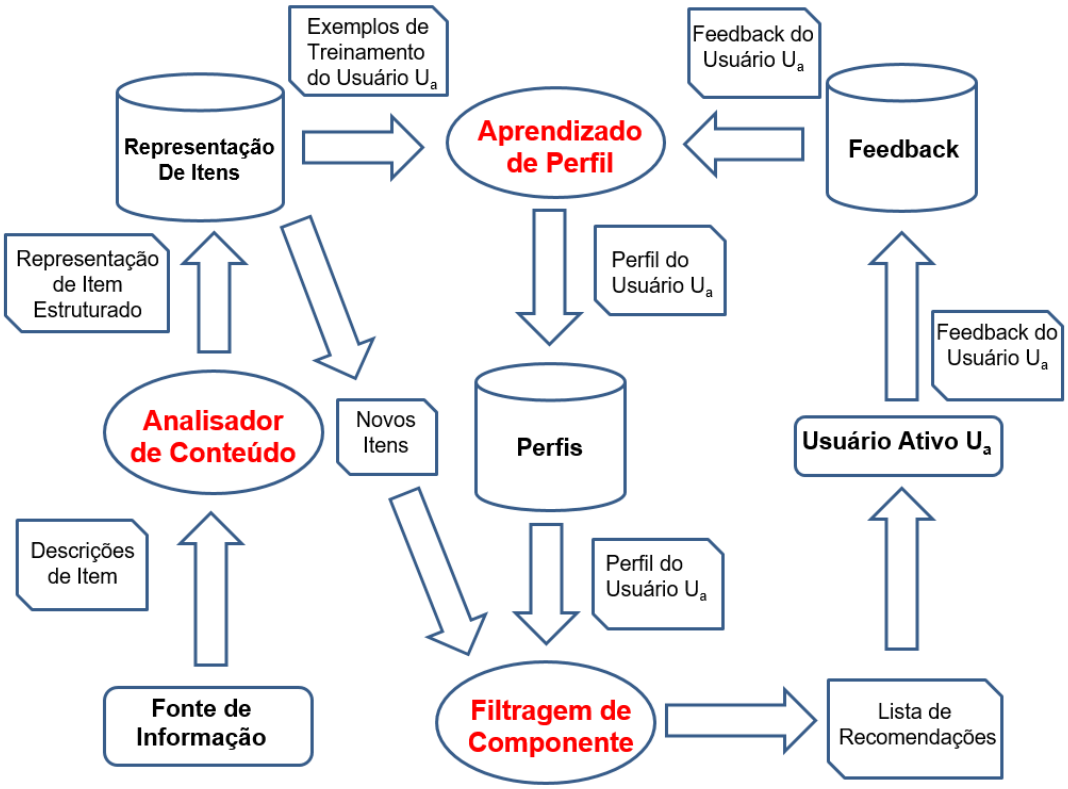


Figura 9 – Arquitetura em alto nível para a técnica CBF (adaptado de (RICCI et al., 2011))

Para a implementação desta arquitetura, é necessário o esclarecimento de algumas formas de representação e modelagem que se fazem necessárias para o desenvolvimento do processo de recomendação utilizando a técnica CBF.

Inicialmente, é necessário descobrir os interesses dos usuários. Estes interesses podem ser representados como documentos, os quais possuem a descrição de cada item presente na base de dados. Ou seja, existem dois conjuntos de documentos: aqueles que são relevantes ao usuário ($d+$) e aqueles que não possuem relevância (d-). Para a identificação de qual conjunto pertence o item ou documento, é possível utilizar alguma técnica de aprendizagem computacional, seja por meio de algoritmos de classificação ou modelos de recuperação de informação (LOPES, 2007; SILVA, 2014). Este trabalho propõe a utilização da segunda abordagem como forma de implementação.

Os modelos de RI possuem este nome graças ao surgimento antecessor dos sistemas de recuperação de informação, no qual, tinham o desafio de representar os documentos, de forma que pudessem ser buscados e recuperados. Os modelos clássicos de RI propõem a utilização de um conjunto de termos de indexação para a representação de cada documento, sendo cada termo independente e não possuindo uma correlação com os demais. Apesar disso nem sempre ser verdade na prática, é uma forma de simplificação adotada, a fim de facilitar a implementação dos modelos (LOPES, 2007). Segundo (YATES; NETO, 1999), existem três modelos de RI principais: modelo booleano, modelo de espaço vetorial e modelo probabilístico. Neste trabalho, será apresentado apenas o modelo vetorial, por ser um modelo adequado às necessidades do trabalho proposto.

O **Modelo de Espaço Vetorial** (**VSM**, ou ***Vector Space Model***, em inglês) é um modelo no qual os documentos e consultas são representados em forma de vetores de termos de indexação, sendo que para cada termo é atribuído um peso que representa a relevância entre os documentos e consultas específicas, podendo assumir um valor entre 0 e 1. Valores próximos a 1 representam um grau de relevância alto, enquanto os mais próximos a zero representam pouca ou nenhuma relevância. Estes vetores são dispostos e relacionados num espaço multidimensional ou n-dimensional, onde cada dimensão representa um termo específico e o peso de cada termo é representado por uma coordenada deste espaço. O princípio do VSM é constituído, basicamente, por relacionar a distância entre os vetores de termos no espaço com o grau de similaridade entre os documentos que eles representam (LOPES, 2007). Para calcular o grau de similaridade, podemos utilizar a equação da similaridade cosseno (equação 2.1), que é uma fórmula obtida pelo produto escalar entre dois vetores desejados, dividido pela multiplicação dos módulos desses vetores. A letra w representa o peso de cada termo/palavra no documento, k o termo/palavra atual e t a quantidade de termos no conjunto. Em geral, são relacionados dois vetores Q e D, representando, respectivamente, uma consulta ou query e um documento, no qual, se quer analisar a distância de um documento (D) com um perfil de usuário específico (Q).

Nesse caso, o perfil pode ser representado também como um vetor de termos de indexação, apontando os termos de interesse do usuário. Na figura 10 é apresentado um gráfico tridimensional com 3 vetores: uma consulta (Q) e dois documentos ($D1$ e $D2$). Cada vetor possui 3 coordenadas: $t1$, $t2$ e $t3$, representando os pesos de importância para cada termo. Neste caso, existem apenas 3 termos e por isso é possível representar num espaço tridimensional. Aplicando o cálculo de similaridade, é possível constatar que o segundo documento ($D2$) está mais próximo da consulta (Q) do que o primeiro ($D1$), ou seja, o documento $D2$ é mais similar à consulta Q do que $D1$. Esta proximidade maior de $D2$ também é possível visualizar graficamente na figura 10.

$$Sim\,(Q,\ D) = \frac{\sum_{k=1}^{|T|} w_{qk}\ .\ w_{dk}}{\sqrt{\sum_{k=1}^{|T|} (w_{qk})^2\ .\ \sum_{k=1}^{|T|} (w_{dk})^2}} \tag{2.1}$$

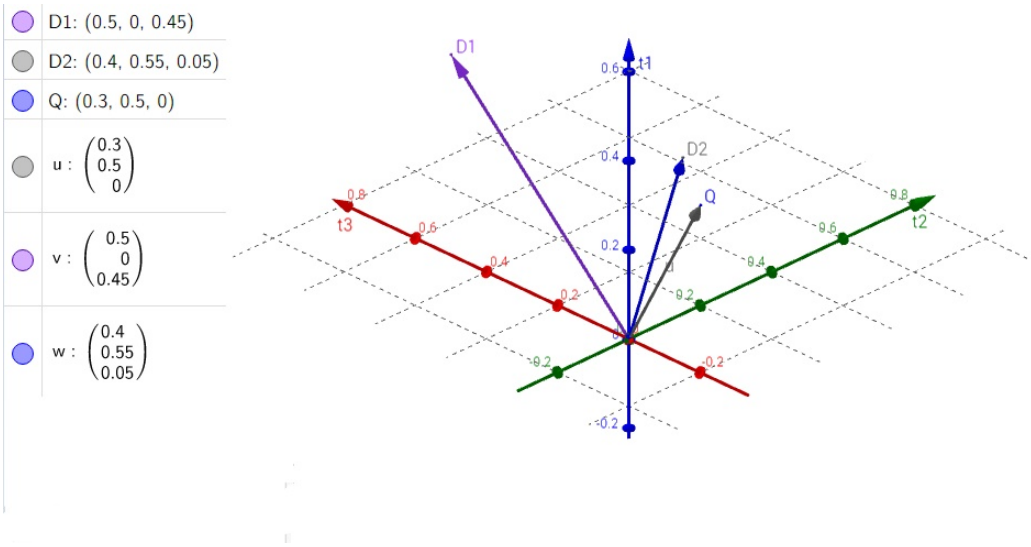


Figura 10 – Representação de um espaço vetorial tridimensional (adaptado de (LOPES, 2007) e obtido pelo simulador *online* do GeoGebra)

Para a definição dos pesos sobre os termos, é necessário construir uma representação para o conteúdo. A representação mais utilizada para textos ou dados não estruturados ocorre através de um conjunto de termos/palavras chamado de *Bag of Words* (**BOW**) que são representados por vetores de termos/palavras por documento, onde se verifica a frequência de cada termo em cada documento e, assim, pode ser proposto um grau de relevância do termo/palavra no documento (SILVA, 2014). Isso pode ser implementado de algumas formas, geralmente, por algoritmos de AM, tais como: TF-IDF, modelos probabilísticos com o classificador de Naïve Bayes, árvores de decisão, redes neurais, entre outros (ISINKAYE; FOLAJIMI; OJOKOH, 2015). Todos estes são capazes de atribuir pesos aos termos de diferentes documentos num conjunto e, assim, permitindo o relacionamento dos diversos itens presentes numa base de dados. Como o foco deste trabalho visa a implementação do algoritmo TF-IDF, apenas este será explicado com maiores detalhes.

Antes de analisar a frequência dos termos nos documentos e calcular os seus pesos com o TF-IDF, é necessária a passagem por uma fase de pré-processamento. Nesta fase ocorrem duas pequenas tarefas: extração e seleção (SILVA, 2014). Na primeira, o texto é decomposto em elementos, os quais definem cada termo do texto e são filtrados por técnicas que descartam palavras que não possuam representatividade semântica no domínio do problema, também conhecidas como *stopwords*, e por técnicas que reduzam as palavras às suas raízes, também conhecidas como *stemming* (WANG; WANG, 2005). Na segunda, os termos extraídos são mapeados num vetor de frequências. No pré-processamento, geralmente, pode ocorrer também uma análise semântica a fim de verificar a possível presença de termos/palavras com duplo sentido ou sinônimos num texto (RICCI et al., 2011).

A técnica de **TF-IDF** (ou ***Term Frequency – Inverse Document Frequency***, em inglês) é utilizada, em geral, em mineração de textos e expressa a importância de um termo/palavra num conjunto de documentos ou *corpus* (TAKAHASHI, 2015). Conceitualmente, pode ser entendida como a atribuição de uma importância maior caso haja uma frequência alta do termo/palavra num documento (TF), porém, por outro lado, uma importância menor caso o mesmo termo/palavra apareça em vários documentos (IDF), gerando dessa forma uma classificação para cada documento, conforme seus termos/palavras. A fórmula de TF é obtida pela divisão do número de ocorrências do termo no documento pelo número total de termos no documento (equação 2.2). IDF é obtida pelo logaritmo da divisão do número total de documentos pelo número de documentos com o termo em questão (equação 2.3). Finalmente, TF-IDF é obtida pela multiplicação de TF com IDF (equação 2.4) (RICCI et al., 2011).

$$TF(d_i,\ t_j) = \frac{F_{i,j}}{\sum_{s=1}^{|T|} F_{i,s}} \tag{2.2}$$

$$IDF\,(d_i,\ t_j) = \log \frac{N}{n_j} \tag{2.3}$$

$$TF - IDF\,(d_i,\ t_j) = TF\,(d_i,\ t_j)\ .\ IDF(d_i,\ t_j) \tag{2.4}$$

Para o cálculo dos pesos, é necessário normalizar os valores obtidos pelo TF-IDF de cada termo em cada documento, pois deve-se levar em consideração tamanhos diferentes de documentos, fazendo-se necessária a aplicação de uma proporção por frequência encontrada em cada documento. A fórmula de normalização é obtida pela divisão do valor de TF-IDF pela raiz quadrada do somatório do quadrado dos valores TF-IDF dos termos na totalidade do documento. Esta técnica é conhecida como normalização de cosseno (equação 2.5) (RICCI et al., 2011).

$$w_{i,j} = \frac{TF - IDF\ (d_i,\ t_j)}{\sqrt{\sum_{s=1}^{|T|} TF - IDF\ (d_i,\ t_s)^2}} \tag{2.5}$$

Após a aplicação dessas técnicas para a implementação do CBF, é possível representar cada item e, dessa forma, cada perfil de usuário, fazendo-se necessário ainda levar em consideração o aprendizado com a avaliação do usuário sobre cada recomendação. Para isso, é possível contabilizar os valores de aprovações (ou *likes*, em inglês) e reprovações (ou *dislikes*, em inglês) retornados pelos usuários, representando os valores de "gostei" e "não gostei", respectivamente, e servindo de entrada para o processamento de novas recomendações àquele usuário especificamente.

Ainda na fase de avaliação e *feedback*, é possível obter também resultados independentemente de avaliação do usuário, o que (SILVA, 2014) chama de avaliação do sistema. De acordo com esta definição, existem dois modos de avaliar o desempenho de um SR ou qualquer técnica de AM: *online* e *off-line* (RICCI et al., 2011), o primeiro representando o contato direto com o usuário e o segundo uma forma de validar as classificações realizadas pelo SR de forma autônoma. Este último é possível através da aplicação de duas técnicas: precisão (ou *accuracy*, em inglês) e revocação (ou *recall*, em inglês) (RICCI et al., 2011). Estas são baseadas na matriz de confusão, obtida pelo relacionamento entre os valores de verdadeiro (acerto) e falso (erro) com os de positivo (recomendado) e negativo (não recomendado), conforme é mostrado na figura 11. Assim, a precisão é dada pela proporção entre a quantidade de itens recomendados que também eram de interesse do usuário (verdadeiro positivo) e todos os itens que foram recomendados (total de valores positivos), enquanto que a revocação é dada pela proporção entre a quantidade de itens recomendados que também eram de interesse do usuário (verdadeiro positivo) e todos os itens que são de seu real interesse (verdadeiro positivo e falso negativo) (SILVA, 2014). Os cálculos de cada técnica são apresentados nas fórmulas 2.6 e 2.7, respectivamente. Neste trabalho, aplicamos esta avaliação relacionando todas os *feedbacks* realizados por todos os usuários sobre um item em específico, ao qual é atribuído um peso, sendo diferente da forma de avaliação anterior que focava apenas num usuário em específico.

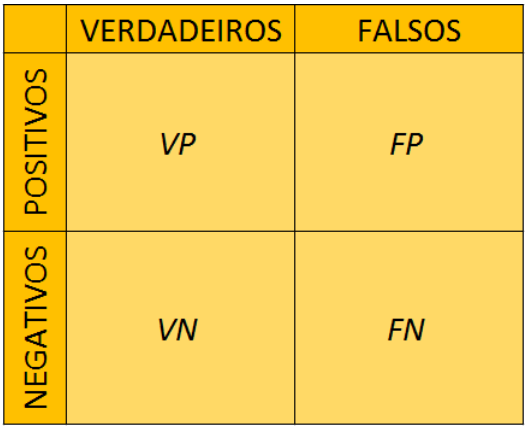


Figura 11 – Matriz de confusão para a fase de avaliação (adaptado de (SILVA, 2014))

$$P = \frac{\#vp}{\#vp + \#fp} \tag{2.6}$$

$$R = \frac{\#vp}{\#vp + \#fn} \tag{2.7}$$

2.5 Sistemas de Recomendação para Bibliotecas Universitárias

Uma biblioteca universitária possui uma série de documentações e produções, seja livros, artigos, revistas, entre outras. Estes são consultados, periodicamente por usuários que podem gerar transações, seja por meio de saída (empréstimos) ou entrada (devoluções). Em geral, há a presença de funcionários formados na área de biblioteconomia, chamados bibliotecários, que possuem o papel de gerir uma biblioteca. Mais especificamente, um bibliotecário de referência é um profissional que possui o papel de capacitar e ajudar o usuário a buscar e encontrar o conteúdo que esteja a procura com o menor tempo e menor custo possível (KREBS, 2013).

Uma questão de importância no contexto de uma biblioteca é o tipo de público que se lida, pois se faz necessário conhecer as necessidades e o contexto no qual o usuário se insere, podendo antecipar o atendimento com mais qualidade, alcançando melhores níveis de satisfação (KREBS, 2013). Num âmbito mais específico, em bibliotecas universitárias, existem, em geral, alunos, docentes e pesquisadores. Os primeiros buscam os livros que estão na bibliografia sugerida pelos professores, enquanto os docentes e pesquisadores necessitam de uma informação mais específica e mais refinada, servindo de apoio a buscas de conteúdos mais aprofundados, relacionados ao tema de pesquisa (MANGAS, 2007).

Além disso, um bibliotecário possui outro papel importante relacionado a Disseminação Seletiva da Informação (DSI), que pode ser definida como a entrega de um serviço

de qualidade para o usuário conforme a informação desejada, agregando os valores de imparcialidade e respeito no atendimento de uma biblioteca (FIGUEIREDO, 1992).

Estes papeis e valores que devem ser atribuídos a um bibliotecário também podem ser aplicados a um sistema de recomendação para sistemas em bibliotecas, transformando as atribuições de um funcionário em um processo automatizado, cujo alcance pode ser maior, chegando a usuários remotos que não são frequentes no ambiente físico da biblioteca, ou com um atendimento personalizado, de acordo com as necessidades e preferências de um usuário em específico.

Visando o esclarecimento do contexto de uma biblioteca, um banco de dados bibliotecários é composto, em geral, por três entidades: usuários, itens e movimentações. Cada uma representa uma tabela em específico. As duas primeiras, como o próprio nome sugere, são, respectivamente, referentes aos cadastros de pessoas que possam explorar e ter acesso aos documentos de uma biblioteca (possuindo atributos como id, nome, documento, data de nascimento, entre outros) e itens que estejam disponíveis para uso na mesma (possuindo atributos como id, título, descrição, tipo, autor, editora, ano de publicação, entre outros). As últimas descrevem o histórico de ocorrências e transações realizadas pelos usuários com os itens de uma biblioteca (possuindo atributos como id do usuário, id do item, data, tipo de ocorrência, entre outros), ou seja, representam as interações dos usuários com os itens.

Neste caso, um sistema de recomendação baseada em conteúdo entra em ação utilizando as movimentações, a fim de analisar os perfis de usuário e gerar um conjunto de interesses para cada perfil. Posteriormente, é possível analisar a descrição de cada item e classificá-lo, com base nos termos que cada um possui. Finalmente, é possível recomendar os itens que estejam mais próximos ao perfil de usuário, recebendo um tipo de avaliação e *feedback* deste e, dessa forma, rever o conjunto de necessidades e preferências dos perfis de usuário.

3 Desenvolvimento

Este capítulo apresenta uma breve descrição da proposta do sistema de recomendação desenvolvido, menciona todos as ferramentas e tecnologias utilizadas para o desenvolvimento da solução e fornece uma descrição detalhada das etapas de desenvolvimento, implementação e fluxo do sistema.

3.1 Proposta da Solução

A proposta para a construção da solução consistiu no desenvolvimento de um sistema de recomendação baseada em conteúdo para bibliotecas universitárias, utilizando técnicas baseadas em mineração de dados ou, mais especificamente, mineração de textos, e aprendizagem computacional, apresentadas com mais detalhes no capítulo anterior.

Este sistema acessa uma base de dados e o usuário pode consultar livros que são de seu interesse, adicionando-os a uma lista de seleção. Ao final, cada livro desta lista recebe um número de recomendações e é solicitado ao usuário que faça uma avaliação sobre a efetividade de cada livro recomendado.

3.2 Técnicas Utilizadas

Seguem as soluções escolhidas para cada uma das áreas aplicadas neste trabalho:

- Em sistemas de recomendação, foi escolhida a técnica de filtragem baseada em conteúdo;
- Em mineração de dados e pré-processamento de dados textuais, foram escolhidas a extração e seleção com slugify e stopwords;
- Em aprendizagem computacional, foi escolhida a técnica de modelos de RI com espaço vetorial (com a fórmula de similaridade cosseno) para a representação de documentos e perfis de usuário. Para a representação de conteúdo textual foi escolhida a técnica de TF–IDF (com a fórmula de normalização cosseno);
- Nas fases de um sistema de recomendação, foram escolhidas na coleta de informações, a identificação e a coleta explícita, a predição e recomendação e a avaliação e *feedback* explícitos.

Na figura 12 é possível visualizar todas as técnicas citadas neste trabalho e as que foram, de fato, escolhidas.

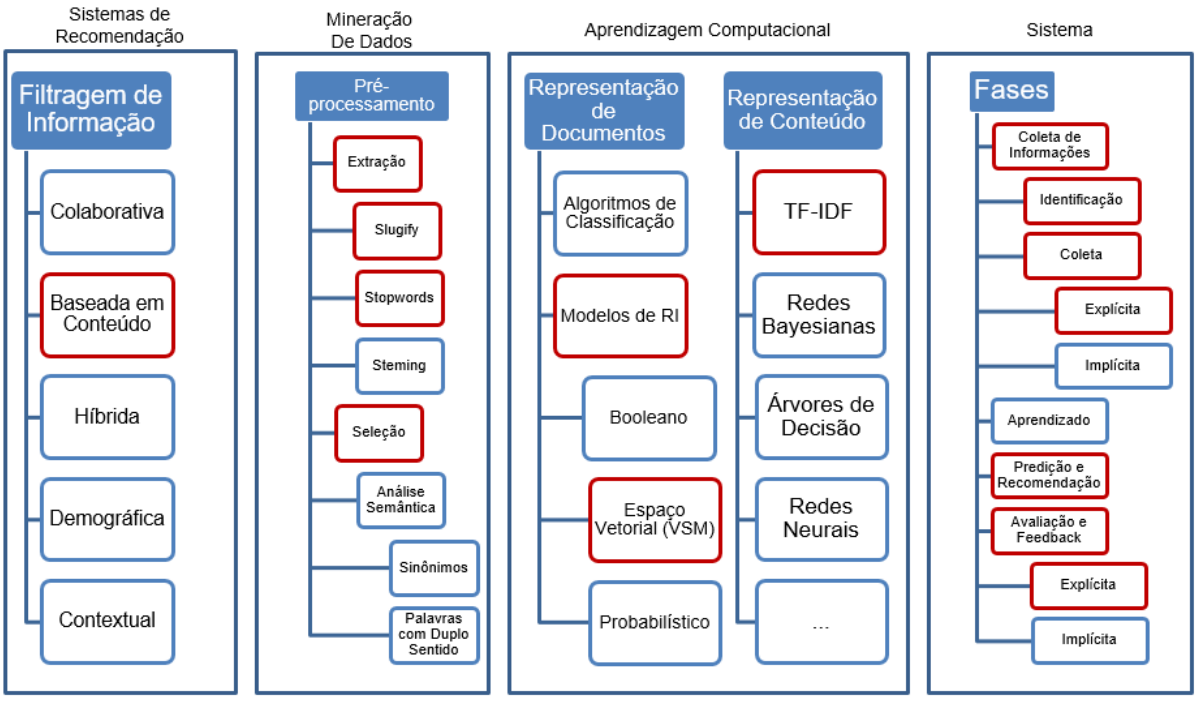


Figura 12 – Técnicas existentes para a solução de sistemas de recomendação

3.3 Conjunto de Dados

Por motivos de restrição de acesso, questões éticas e relacionadas à confidencialidade da informação, os dados usados para o desenvolvimento deste trabalho abrangeram apenas os dados de produções literárias ou obras conhecidas, coletadas por arquivos de dados de livre acesso, sites de livre acesso ou ainda por API's de acesso a dados livres, sem a presença de dados de usuários envolvidos originalmente.

O conjunto de dados utilizado para este trabalho foi baseado nos dados da ETEC e na API do *Google Books*. Segue uma breve descrição dos dados originais.

3.3.1 Base de Livros da ETEC

Os dados referentes aos livros são provenientes do acervo da biblioteca da ETEC, obtido pelo site institucional [1]. Esta base é formada por dados de livros, cadastro de trabalhos e TCC's, DVD's e revistas. Foram utilizados para este trabalho apenas os dados de livros e estes são formados, originalmente, pelos seguintes campos: autor(es), responsável(is)/colaboadores, instituto responsável, título, série – coleção, local de publicação, editora, ano de publicação, volume, edição, impressão/tiragem, língua, ISBN, assunto 1, assunto 2, assunto 3, assunto 4, classificação, cutter, exemplar, data do registro, notas e observação. Destes, os campos utilizados para o projeto foram os seguintes: ISBN, título, autor, editora e ano de publicação.

[1] *http://www.etecfran.com.br/biblioteca.html*

3.3.2 API do *Google Books*

Com o intuito de compor os dados dos livros, foi utilizado o *Google Books* para explorar a descrição de cada obra literária. A consulta aos dados foi feita acessando a API do *Google Client* [2].

Para permitir a utilização desta API, foi necessário acessar o site do *Google* [3], se autenticar com o *login* e senha do usuário corrente, entrar na Plataforma de Nuvem do *Google*, criar um novo projeto e selecioná-lo, acessar as API's e Serviços, selecionar uma API específica e, finalmente, ativá-la para consulta de aplicações externas. Após isto feito, foi fornecida uma chave de acesso, na qual foi utilizada, obrigatoriamente, no código para acessar as requisições da API. No site [4] é fornecida uma documentação completa descrevendo todo o processo de configuração para diversos tipos de linguagem de programação e diversas soluções disponíveis de API que a empresa oferece. Além disso, o *Google* disponibiliza um painel com indicadores de acessos às suas API's.

A escolha por essa forma de acesso e coleta dos dados de descrição dos livros se deu devido a necessidade de simular um cenário onde não está disponível um registro das descrições, resumos ou sinopses das obras literárias no banco de dados da biblioteca a ser analisado, além do próprio fato de não haver o cadastro deste atributo na ETEC, o que realmente exigiu um meio alternativo para obter acesso a este tipo de informação.

3.4 Materiais

Para a implementação do sistema proposto, foi utilizada a linguagem de programação *Python*, na versão 3.6.2, obtida no site oficial da linguagem [5], devido ao fato de ser uma linguagem muito usual para a implementação de algoritmos de ciência de dados e aprendizagem computacional e, por isso, há uma série de bibliotecas e ferramentas que estão sendo desenvolvidas por uma grande comunidade ativa na *Web*. Além disso, a linguagem oferece um desempenho e performance similares ao de outras de mais baixo nível, pois existem diversas bibliotecas implementadas em camadas mais baixas, como em *C* e *Fortran*. Este é o caso da biblioteca *NumPy* (RASCHKA, 2016), utilizada para o processamento do algoritmo deste trabalho.

Visando a implantação em plataforma *Web*, foi utilizado o *framework Django*, na versão 1.11, obtido no site oficial [6], pois é um conjunto de bibliotecas que viabiliza a construção de sites e facilita na comunicação de classes e objetos em Python com as tabelas relacionais do banco de dados. Além disso, a ferramenta oferece uma área de administração

[2] *https://developers.google.com/api-client-library/*
[3] *https://www.google.com.br/*
[4] *https://developers.google.com/api-client-library/*
[5] *https://www.python.org/*
[6] *https://www.djangoproject.com/*

para controlar os objetos relacionais e executar operações de CRUD nas tabelas do banco de dados.

Para a estruturação e manipulação do banco de dados, foi usado o *PostgreSQL*, obtido no site oficial [7], pois é uma tecnologia de SGBD que, em geral, se comunica bem com o *Python* e, especificamente, com o *Django*. Para a comunicação entre o banco e a linguagem, foi necessária a instalação do conector *psycopg*, obtido no site oficial [8].

Como bibliotecas adicionais, foi necessária a instalação das seguintes: *google-api-python-client*, *requests*, *stop-words* e *numpy*. Estes podem ser instalados via comando *pip* no terminal (para *Linux* ou *Mac-OS*) ou prompt de comando (para *Windows*), conforme instruções no site oficial [9]. Além disso, as instalações do *Django* e do *psycopg* também podem ser realizadas por intermédio deste comando.

Para fins de controle de versionamento de código, foi escolhido o *GitHub*, onde os projetos podem ser facilmente manipulados via linha de comando nos sistemas operacionais *Windows*, *Linux* e *Mac-OS*, desde que seja instalado o programa previamente. O mesmo pode ser obtido pelo site oficial [10]. Além disso, o programa possui integração com diversos editores de código, tais como *Atom* e *Visual Studio Code*.

3.5 Metodologia

O cronograma para o segundo semestre de 2017 foi dividido pelas seguintes etapas principais: coleta de dados, levantamento de requisitos, implementação, testes e validação, testes com usuário, análise dos resultados, elaboração do texto e entrega.

Seguem, em detalhes, todas as tarefas (ou *tasks*, em inglês) que fizeram parte destas etapas: busca por dados de livre acesso; esboço do algoritmo dos cálculos de frequência, TF–IDF, pesos e similaridades de livros; criação do modelo conceitual (MER) e relacional (DER) do banco de dados com as tabelas, campos e relacionamentos; criação e estruturação do projeto em *Django* com todos os componentes necessários para o seu funcionamento, conforme modelo do BD; implementação do módulo de pré-processamento de livros; implementação do módulo de processamento de usuários; definição da arquitetura e fluxo de dados do sistema; execução de testes e validação das principais funcionalidades envolvendo livros e usuários; execução de testes com usuários; coleta de dados para a análise dos resultados; elaboração do texto (referente aos novos capítulos propostos no TCC II); revisão da introdução e referencial teórico (produzidos no TCC I); elaboração do resumo, *abstract* e conclusão; elaboração da apresentação (*slides*); elaboração do pôster;

[7] *https://www.postgresql.org/*
[8] *http://initd.org/psycopg/*
[9] *https://pip.pypa.io/en/stable/*
[10] *https://github.com/*

apresentação prévia ao professor da disciplina; apresentação prévia ao orientador; entrega do TCC; e, finalmente, a defesa do TCC para a banca avaliadora.

É importante ressaltar que estas *tasks* ocorreram paralelamente e não, necessariamente, estavam em ordem cronológica.

3.6 Especificações Técnicas

3.6.1 Banco de Dados

Para o desenvolvimento deste trabalho, foi criado um banco de dados chamado bibinterativa, com as seguintes tabelas: livro, termo, *stopword*, peso, similaridade, usuario, pesquisa, pesquisa palavra chave, pesquisa livro selecionado e pesquisa recomendacao. Além destas, foram criadas tabelas referentes ao armazenamento de controle do *Django*.

As tabelas foram estruturadas com os seguintes campos:

- **Livro:** id, isbn, titulo, autor, editora, ano_publicacao, conteudo_processado;
- **Termo:** id, nome;
- **Peso:** id, livro, termo, valor;
- **Similaridade:** id, livro_i, livro_j, valor;
- **Stopword:** id, nome;
- **Usuario:** id, username, first_name, last_name, email, password, is_verified, is_staff, is_active, date_joined;
- **Pesquisa:** id, data, usuario;
- **Pesquisa palavra chave:** id, nome, pesquisa;
- **Pesquisa livro selecionado:** id, livro, pesquisa;
- **Pesquisa recomendacao:** id, selecionado, recomendado, rating, data.

Para a modelagem das tabelas, foram criados os modelos conceitual (MER) e relacional (DER), conforme as figuras 13 e 14, respectivamente.

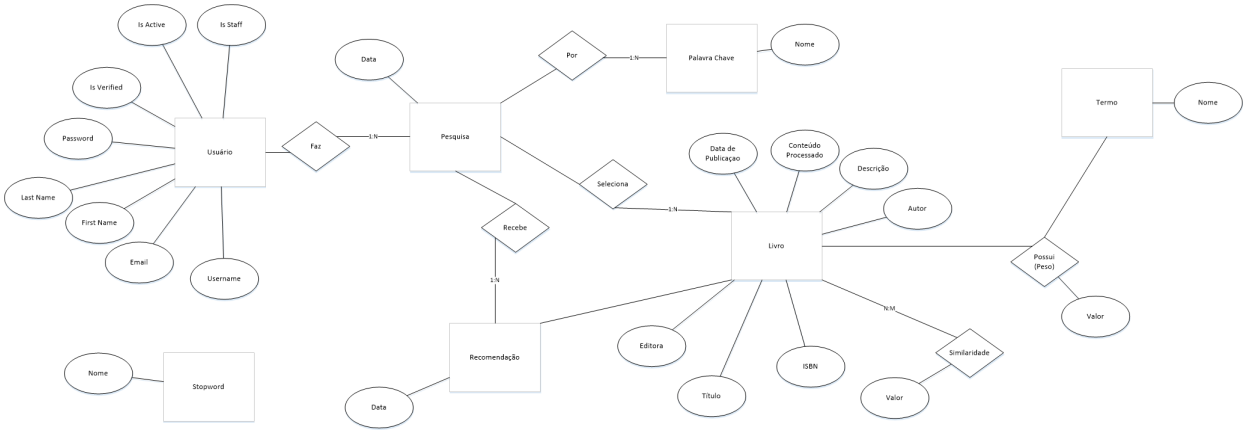


Figura 13 – Modelo Entidade Relacionamento do Banco de Dados

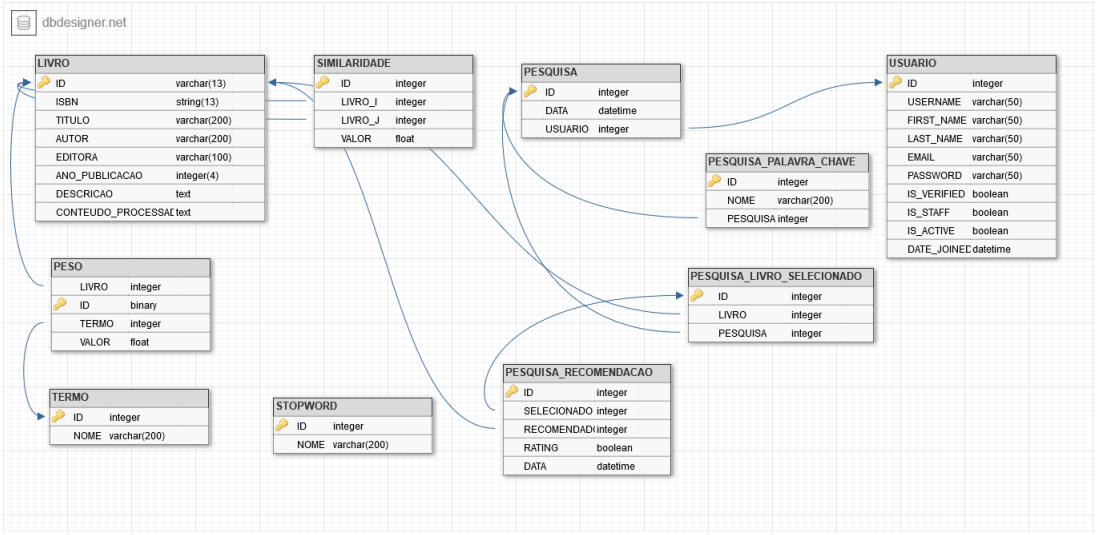


Figura 14 – Diagrama Entidade Relacionamento do Banco de Dados

3.6.2 Estrutura do *Django*

O *Django*, por padrão, é formado por uma estrutura hierárquica em um ambiente com um ou mais projetos que possuem um ou mais aplicativos. Cada aplicativo possui seus próprios arquivos de migração, modelos, controladores, campos de formulários, páginas *Web*, administração, comandos, testes automatizados e mapeamento de urls. Isto auxilia na organização das informações e na comunicação entre cada componente do sistema como um todo, tornando o código portável e facilitando a manutenção dos desenvolvedores, caso haja alguma alteração de escopo no projeto.

Um projeto possui também por padrão um arquivo de configuração chamado `settings.py`, onde são definidas todas as configurações do ambiente no qual o mesmo é aplicado. Além deste, existe um arquivo de inicialização chamado `manage.py`, no qual executa o método principal (`main`) e permite acessar comandos pré-definidos ou personali-

zados por desenvolvedores via linha de comando. Graças a este último arquivo é possivel ativar o servidor, por intermédio do comando ***runserver***, e, além disso, ter acesso a um console do *python* com o projeto já implantado, onde é possível simular e testar comandos válidos dentro do projeto *Django*, denominado ***shell***.

Como é comum em outros *frameworks Web*, o *Django* também trabalha com o padrão *Models – View – Controller* (MVC), porém com diferenças sutis, pois o padrão utilizado é o *Model – Template – View* (MTV), na qual, os *models* possuem a função de gerenciar e integrar os objetos relacionais com o banco de dados (*models* do MVC), os *templates* gerenciam as páginas e formulários (*views* do MVC) e, finalmente, as *views* estabelecem a comunicação entre os objetos relacionais e as visualizações (*controllers* no MVC).

Os arquivos de migração são formados por comandos que determinam a estrutura de cada tabela no banco de dados, conforme a definição das classes e os tipos de atributos declarados nos modelos. Caso haja uma mudança estrutural nos modelos, o mesmo é refletido no banco de dados, com a criação de novas versões do arquivo de migração. Estes arquivos são gerados através do comando ***makemigrations*** e os mesmos migrados para o banco de dados pelo comando ***migrate***.

Uma vez criados e migrados os modelos com as classes e objetos relacionais, é possível acessá-los por intermédio de uma API que é disponibilizada pelo *framework* para consultas (ou *queries*, em inglês) e recuperação de informações pertinentes aos dados dos mesmos para executar alterações, exclusões ou novas inclusões na base, privando assim o desenvolvedor de saber a configuração interna dos objetos do banco de dados e outras questões estruturais, para utilizar comandos do T–SQL. A API disponibliza uma série de comandos, tais como relacionamento, filtragem, agregação, agrupamento, entre outros.

Outra facilidade que é oferecida por este *framework* é a disponibilização automática de uma área de administração que permite gerenciar os registros de cada modelo gravados no banco de dados e executar operações de criação, consulta, atualização e exclusão (CRUD) nos mesmos. É exigido apenas que seja configurado o arquivo `admin.py` para cada aplicativo e, desssa forma, permitir a visualização dos modelos, seus respectivos campos, além da configuração do campo de busca para pesquisar em seus registros. Na figura 15 é possível visualizar um exemplo de área de administração do *Django*.

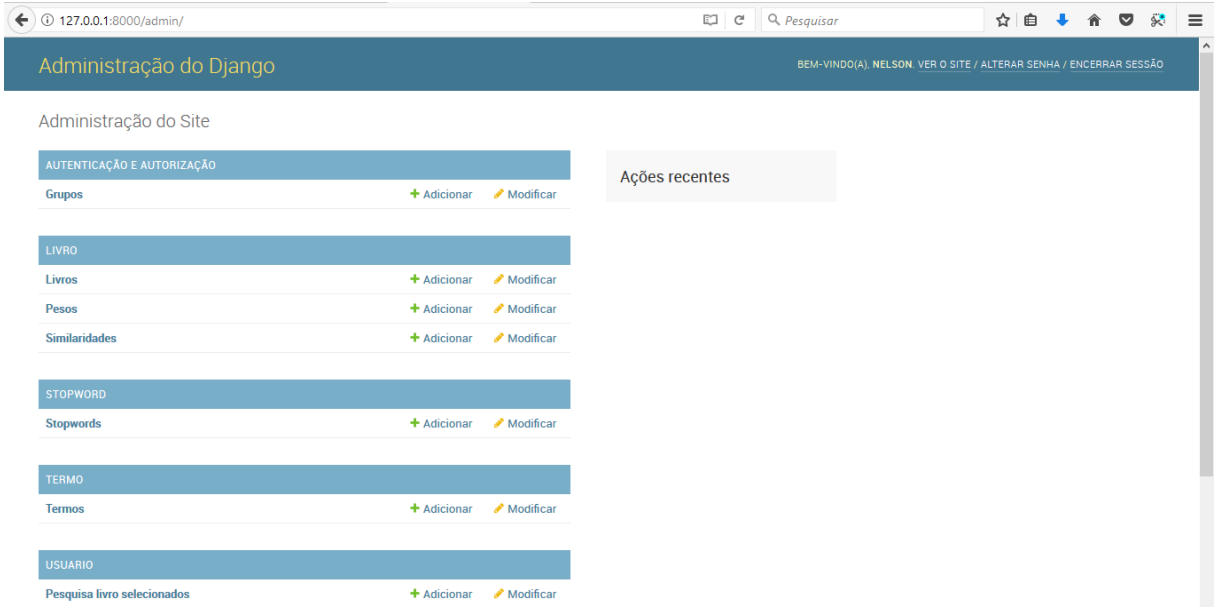


Figura 15 – Área de administração do *Django*

Todas as informações referentes a configuração, comandos e personalização do *Django* estão descritos em detalhes no site oficial [11].

3.6.3 Estrutura do Projeto

Para o desenvolvimento deste trabalho, foi criado um projeto chamado `bibinterativa` e, posteriormente, criados os aplicativos correspondentes às entidades que estão em destaque no sistema. São eles: `livro`, `termo`, `stopword` e `usuario`. Os três primeiros manipulam informações pertinentes aos livros, sejam dados de detalhes dos mesmos ou resultantes de processamentos paralelos, os quais são necessários para o processo de recomendação. O último armazena todas as informações pertinentes aos usuários e suas interações com o sistema. Para cada aplicativo foi editado o arquivo `models.py` para incluir as classes relacionais e o arquivo `admin.py` para habilitar a área de administração de modelo. Houve também a edição do arquivo de configurações `settings.py` para se conectar ao banco *PostgreSQL*, a tradução do sistema para o idioma português e a definição de fuso horário.

No aplicativo de `livro`, estão presentes os modelos com as seguintes classes relacionais: `Livro`, `Peso` e `Similaridade`, as quais guardam os valores do pré–processamento de livros. Em `termo`, está presente a classe `Termo` que guarda todos os termos presentes na descrição dos livros. Analogamente, em ***`stopword`*** está presente a classe `Stopword` que guarda todas as palavras que devem ser ignoradas da descrição de cada livro. E, por fim, no caso do aplicativo `usuario`, estão presentes as seguintes classes: `Usuario`, `Pesquisa`, `PesquisaPalavraChave`, `PesquisaLivroSelecionado`, `PesquisaRecomendacao`, as quais armazenam os dados das interações realizadas pelos usuários no sistema. Para cada classe

[11] *https://www.djangoproject.com/*

foram criados os métodos `__str__` para uma breve visualização do conteúdo de seus registros e os índices para cada atributo, a fim de aumentar o desempenho das buscas nos dados que as compõe.

Nos aplicativos de livro e usuario foram criados os arquivos `libs.py` que implementam a classe que manipula o processamento necessário para funcionamento do sistema e uma pasta management e commands dentro desta que possuem os arquivos `processar_livros.py` de pré-processamento dos livros e `recomendacao.py` com a execução do fluxo do sistema, respectivamente. Estes últimos arquivos acessam as `libs.py` de cada aplicativo com as classes `ProcessamentoLivros` e `ProcessamentoUsuarios`, respectivamente, a fim de utilizar seus atributos e métodos para a manipulação de seus modelos correspondentes.

Além destes, foram criados os seguintes arquivos na raiz do projeto: `shell.py` que fornece os comandos–chave e de teste para serem executados no console do projeto e `docstring.py` que fornece uma documentação com os principais comandos utilizados para a execução do sistema.

Externamente à raiz do projeto estão os arquivos com os dados do Acervo da ETEC em formato *xls* e *csv*, backup e modelagem do banco de dados e arquivos com comandos e tutorial aplicáveis ao projeto.

A figura 16 ilustra a estrutura do projeto criado para este trabalho, com seus diretórios e arquivos.

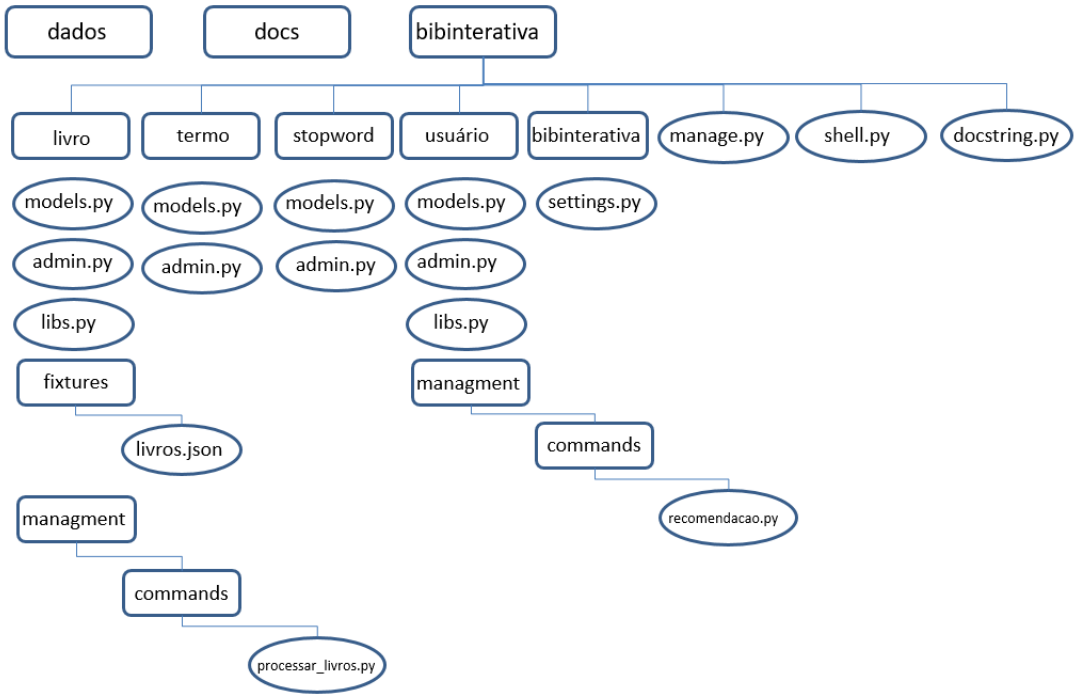


Figura 16 – Estrutura do projeto desenvolvido com o *Django*

3.6.4 Processamento

O sistema desenvolvido para este trabaho foi composto por dois módulos principais: o de pré–processamento dos dados referentes aos livros e o de interação do usuário com o sistema para a execução de pesquisas de livros, a criação de perfis de usuário e o processamento e resposta da recomendação acompanhada de uma avaliação por parte do usuário.

3.6.4.1 Pré–Processamento de Livros

Na primeira etapa ocorreu a execução de uma rotina de carga e tratamento dos dados dos livros presentes no cadastro do acervo da biblioteca da ETEC, onde houve a necessidade de analisar os mesmos por intermédio do processo de mineração de textos e CBF.

Inicialmente, os dados referentes às obras literárias presentes no arquivo chamado `Acervo da Biblioteca - ETEC Emilio Hernandes Aguilar.xls` foram tratados e limpos, gerando um novo arquivo em formato csv, chamado `livros.csv`. Dessa forma, os livros que não possuiam ISBN e título foram excluídos e os que não apresentavam ISBN foram preenchidos manualmente com o seu respectivo ISBN, conforme pesquisado na *Web*. Destes, foram escolhidos apenas os que estavam no idioma português. Além disso, os campos que não eram usuais para o trabalho foram excluídos. Ao final, a base ficou apenas com os seguintes campos: ISBN, título, autor, editora e ano de publicação. Este procedimento foi necessário para manter apenas os dados relevantes ao processamento da rotina dos livros.

O processo de mineração de textos e filtragem baseada em conteúdo consistiu na seguinte sequência de passos: os dados dos livros foram carregados do arquivo *`livros.csv`* para o arquivo `livros.json`, no qual, os dados foram estruturados de forma que o *Django* permitisse armazená–los na classe Livro do modelo livro. Isto pode ser feito através da função de carga do *Django* chamada *loaddata*, onde um arquivo em formato json, localizado dentro de uma pasta *fixtures* no projeto, é carregado automaticamente para o BD. Posteriormente, ocorreu a carga da descrição dos mesmos via *API* do *Google Books*. É importante mencionar que os livros que não foram encontrados na base do *Google*, foram excluídos do BD, pelo fato de que os cálculos foram efetuados com base nos dados de descrição dos livros. Após esta etapa, os dados de descrição de cada livro foram extraídos da base, passando por um processo de remoção de *stopwords* e transformação *slugify*, ou seja, o descarte de caracteres especiais e a transformação de acentuação e letras maiúsculas por minúsculas não acentuadas, muito utilizada para a composição de URL's, e, posteriormente, a inclusão do texto no campo `conteudo_processado` da tabela de livros e de cada termo separadamente na tabela de termos. Nesta última, é armazenado apenas termos não repetidos, a fim de manter uma indexação para todas as palavras existentes na descrição

dos livros. O campo `conteudo_processado` representa cada documento a ser analisado. Após a carga de documento e termos no BD, foram criadas 5 matrizes de dimensões M x N, onde M representa a quantidade total de documentos e N a quantidade total de termos, servindo para armazenar os valores obtidos dos cálculos de frequência, TF, IDF, TF-IDF e pesos para cada termo em cada documento, respectivamente. Além disso, foram criados os vetores de ID's dos livros e termos para guardar uma referência das posições i e j das matrizes para a recuperação dos respectivos livros e termos do banco de dados, os vetores de quantidade total de termos em cada livro e quantidade total de livros com o mesmo termo para os cálculos de TF e IDF, respectivamente, e um vetor de M posições para armazenar as médias dos termos de cada documento, servindo para o cálculo posterior dos pesos. Após a matriz de pesos ser obtida, a mesma foi gravada na sua respectiva tabela no BD, com exceção de valores zerados, evitando assim um gasto de armazenamento desnecessário ao banco. Por fim, foi criada uma matriz de dimensões M x M para armazenar os valores obtidos no cálculo de similaridades entre os documentos. Para este processamento, a cada iteração, foram calculados os valores dos três somatórios da fórmula de similaridade cosseno. Finalmente, a matriz de similaridades foi armazenada na sua respectiva tabela no BD, para servir ao procedimento de recomendação posteriormente. É importante observar também que, a partir da inclusão nas tabelas de peso e/ou similaridade, não é necessária a execução novamente deste pré-processamento por completo, sendo suficiente apenas a carga dos valores do BD numa matriz de pesos e/ou similaridades.

3.6.4.2 Processamento de Usuários

A segunda etapa foi formada pela interação do usuário com o sistema, incluindo as seguintes ações: autenticação, pesquisa, exploração, seleção, recomendação e avaliação. Para cada uma destas foram criados comandos específicos.

O processo de autenticação do usuário consistiu em receber o seu *login* e verificar se este está cadastrado na tabela `usuario` no BD. Caso existir, o sistema o identifica, caso contrário, é realizado um novo cadastro no BD com os seus dados pessoais de nome, sobrenome e *e–mail*. Neste caso, estes campos também devem ser recebidos pelo usuário.

O processo de pesquisa por palavra–chave consistiu na seguinte sequência de passos: é cadastrada um nova pesquisa na tabela `pesquisa` no BD e as palavras–chave digitadas passam por um procedimento de extração e seleção, assim como ocorreu na etapa de pré–processamento dos livros, porém, neste caso, para cada palavra resultante, além de ser armazenada na tabela de `pesquisa palavra chave`, são gerados conjuntos de livros, nos quais, cada palavra ocorre em seu título. Ao fim deste processo, é recebida a intersecção desses conjuntos com uma lista de livros recuperados e são apresentados apenas os primeiros 10 livros.

O processo de exploração e seleção consistiram, respectivamente, na consulta de

um livro específico por ID e na manipulação de uma lista, onde foi permitida a inclusão e exclusão de IDs de livros recuperados anteriormente, por intermédio de uma lista de IDs, separados por vírgula, e a limpeza total da mesma.

O processo de recomendação e avaliação consistiram na seguinte sequência de passos: o sistema carrega os dados de similaridades de todos os livros presentes na lista de seleção. Com base nisso, ocorre a recomendação dos livros que estão mais próximos daqueles interagidos pelo usuário, gerando um novo cadastro no BD, na tabela `pesquisa livro selecionado` para cada livro da lista de seleção e na tabela `pesquisa recomendacao` para cada obra recomendada. Ao final, é requisitado do usuário uma avaliação para cada livro recomendado, a fim de validar a precisão do processamento de CBF e mineração de textos realizadas anteriormente. A avaliação é enviada através de uma lista de valores 1 ou 0, separados por vírgula, na ordem dos livros recomendados e o campo ***`rating`*** da tabela de recomendação é atualizado no BD. Os *feedbacks* positivos são representados pelo valor 1 e os negativos pelo valor 0. Por padrão, os dados de avaliação são armazenados no BD com valor 1.

3.7 Arquitetura do Sistema

A arquitetura do sistema consistia no desenvolvimento de uma interface amigável para que o usuário pudesse interagir com o sistema e receber recomendações sobre os livros selecionados. Porém, por questões de cronograma e prazos, foi definido que o sistema fosse desenvolvido pelo *prompt* de comando do *Windows* e fosse composto por um fluxo de entrada e saída de dados.

O fluxo do sistema, como um todo, foi formado pelas seguintes etapas: login, pesquisa, seleção, recomendação, exploração e avaliação. Segue a descrição funcional do fluxo do sistema:

O usuário acessa o sistema via login e, após ser identificado ou cadastrado com os demais dados pessoais (nome, sobrenome e e–mail), em caso de novo usuário, deve entrar com os livros já lidos ou que são de seu interesse. Para isso, o mesmo deve buscar os livros com a entrada de palavras–chave que contenham ou ocorram no título. Caso estiver cadastrado na base, o livro é apresentado em tela como resultado da pesquisa e é oferecida a opção de incluir a uma lista de seleção. Esta lista é apresentada ao usuário, à medida que cada livro é selecionado. Enquanto o usuário não enviar a lista para a recomendação, o mesmo pode seguir pesquisando na base e selecionando novos itens. Ao final, o mesmo clica em enviar e o sistema apresenta dois livros recomendados para cada selecionado e é solicitado ao usuário uma avaliação para cada obra recomendada com valores 1 (gostei) ou 0 (não gostei). Para isso, é oferecida a opção de explorar os livros recomendados. É importante informar que os itens são selecionados e explorados pelo ID e a avaliação deve

ser passada com uma lista de valores binários (1 ou 0), separados por vírgula, na mesma ordem e correspondendo às recomendações que foram apresentadas anteriormente.

O fluxo segue o mesmo conceito do funil de conversão do *marketing* digital usado para *sites* de compras *online*, pois a cada etapa que o usuário avança, os dados resultante se tornam mais específicos e mais condizentes com o seu perfil (TAKAHASHI, 2015). Segue uma ilustração deste fluxo na figura 17.

Figura 17 – Funil de conversão do sistema recomendação

4 Resultados e Discussões

Este capítulo apresenta a descrição dos testes executados com os usuários, analisa criticamente os resultados obtidos e discute sobre os mesmos.

4.1 Método de Avaliação

Com o intuito de verificar o trabalho desenvolvido, foi necessária a escolha de uma metodologia de avaliação. Por este trabalho envolver apenas o desenvolvimento de uma única técnica de recomendação, a filtragem baseada em conteúdo, foi determinado junto ao professor e orientador da disciplina que fosse aplicada uma pesquisa de campo, envolvendo um cenário onde os usuários pudessem avaliar as recomendações e, consequentemente, a precisão do algoritmo empregado e implementado. Esta pesquisa foi feita por meio de testes onde o usuário pudesse acessar o sistema, selecionar livros, receber recomendações dos mesmos e, finalmente, fornecer um *feedback*.

4.2 Pré-Requisitos dos Testes

Para que a etapa de testes com o usuário final fosse realizada, foi exigido como pré-requisito que o usuário tivesse acesso a um computador e que, por intermédio deste, acessasse remotamente a máquina do desenvolvedor, pois o projeto estava implantado apenas em sua máquina local. Para isso, foi usada a ferramenta *TeamViewer*, na sua versão 12, obtida por meio do site oficial [1].

Após a instalação da ferramenta, o usuário teve que entrar com os dados de acesso do *TeamViewer* do desenvolvedor, compostos por ID e senha. Dessa forma, o usuário obteve acesso à máquina remota e pode iniciar os testes.

4.3 Execução dos Testes

Os testes consistiram no acesso dos usuários ao sistema, informando, inicialmente, o login e, possivelmente, outros dados pessoais, solicitados pelo sistema, como, nome, sobrenome e *e–mail*. Posteriormente, os usuários faziam pesquisas, obtinham resultados e, eventualmente, selecionavam livros, adicionando–os a uma lista. Opcionalmente, os mesmos podiam realizar novas pesquisas e adicionar novos livros à lista de seleção. Ao final, a lista era enviada para recomendação e o sistema apresentava as possíveis recomendações para

[1] *https://www.teamviewer.com/pt/*

cada livro selecionado. Nesse momento, cada usuário podia explorar os livros recomendados através do seu ID ou seguir com o envio da avaliação sobre todos os recomendados. A avaliação consistia no fato de confirmar (com o valor 1) ou não (com o valor 0) a relação de cada livro recomendado com o seu respectivo livro selecionado. Dessa forma, os dados de *feedback* do usuário eram gravados no banco de dados para agregar aos resultados deste trabalho.

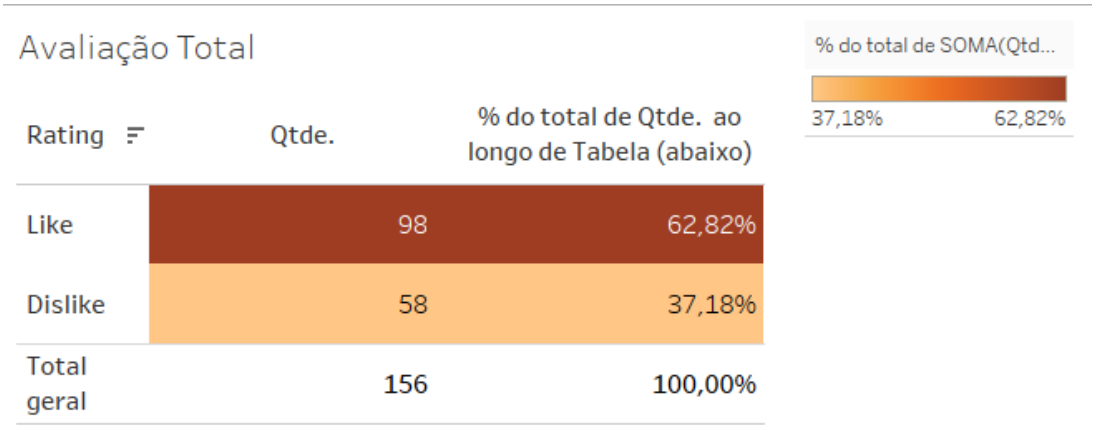
4.4 Resultados

Para o desenvolvimento deste trabalho, foram usados os dados de livros da base da ETEC em conjunto com as descrições dos livros carregados da API do *Google Books*, conforme mencionado no capítulo anterior. Além disso, o pré–processamento sobre os livros foi aplicado apenas sobre o campo de descrição de cada livro. Por isso, os livros que não possuiam o mesmo campo preenchido após a integração com a API do *Google* foram descartados e excluídos da base. Assim, originalmente, haviam 2881 livros no BD e, após a carga da API, restaram apenas 313, estes já com a descrição preenchida.

Por se tratar de livros de uma escola com ensino técnico, a maioria dos livros presentes na base e, assim, extraídas para o sistema desenvolvido para este trabalho são livros técnicos e acadêmicos, restando uma pequena minoria de livros de literatura geral.

Assim, após a realização dos testes com, aproximadamente, 30 usuários, foram obtidos os seguintes resultados:

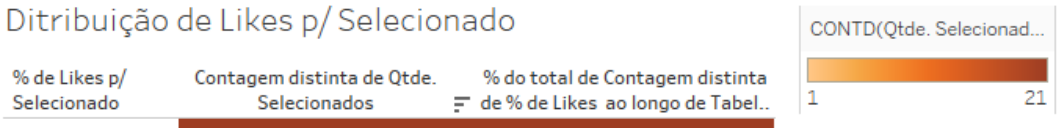
- De acordo com uma visão geral, foram recomendados 156 livros e destes, apenas 98 obtiveram 100% de aprovação por parte dos usuários, representando 62,82% dos recomendados. Já, os livros reprovados representaram 37,18%, equivalente a 58 livros dos recomendados. O resultado negativo se deu pelo fato de o acervo da ETEC ter sido reduzido e pela presença maior de livros técnicos, conforme citado anteriormente, e, além disso, houve usuários que selecionaram livros de literatura geral ou de temas mais específicos, ocasionando recomendações não satisfatórias e influenciando nos resultados finais. Na figura 18 é apresentada uma tabela com os percentuais, onde *like* representa os livros recomendados que receberam aprovação dos usuários e *dislike* representa os reprovados.



Rating	Qtde.	% do total de Qtde. ao longo de Tabela (abaixo)
Like	98	62,82%
Dislike	58	37,18%
Total geral	156	100,00%

Figura 18 – Visão geral do total de avaliações realizadas

- De acordo com uma visão detalhada, é possível observar que, ao todo, foram selecionados 50 livros e destes, os que obtiveram 100% de aprovação foram 15 livros, representando 30% dos selecionados, e os que obtiveram 100% de reprovação foram apenas 6 livros, representando 12% dos selecionados. Além disso, vale observar que os livros selecionados que obtiveram mais de 0% de aprovações por recomendação pelos usuários representaram 88% dos selecionados ou 44 livros, em comparação aos que obtiveram 100% de reprovação (12% dos selecionados ou 6 livros). Na figura 19 é apresentada a distribuição da porcentagem de livros aprovados por recomendação (*likes*) pela quantidade de livros selecionados. Já, nas figuras 20 e 21, é dada uma relação dos livros selecionados que obtiveram 100% de aprovação (*likes*) sobre os livros que obtiveram 100% de reprovação (ou 0% de aprovação).



% de Likes p/ Selecionado	Contagem distinta de Qtde. Selecionados	% do total de Contagem distinta de % de Likes ao longo de Tabel..
50,00%	21	42,00%
100,00%	15	30,00%
0,00%	6	12,00%
75,00%	3	6,00%
25,00%	2	4,00%
87,50%	2	4,00%
90,00%	1	2,00%
Total geral	50	100,00%

Figura 19 – Distribuição dos livros aprovados por cada livro selecionado

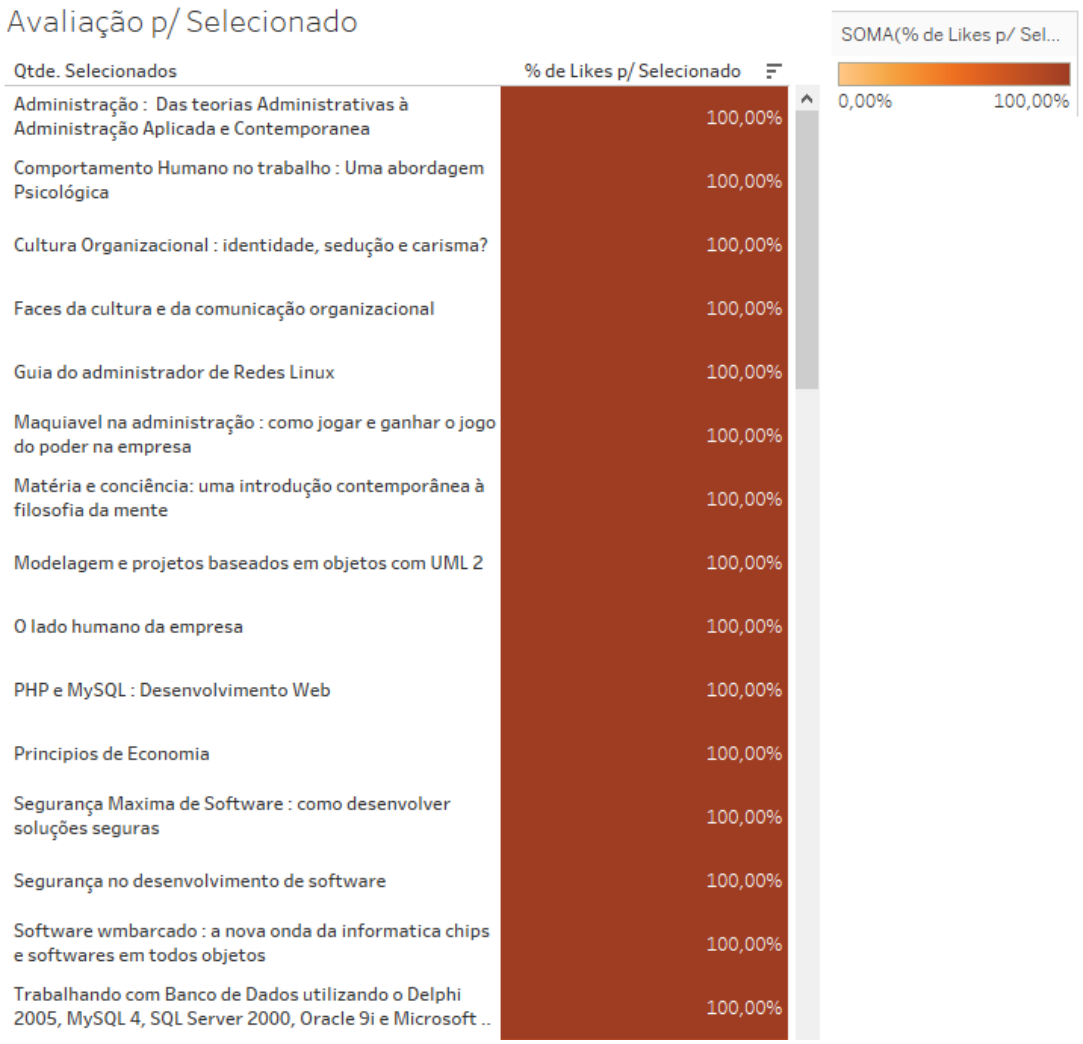


Figura 20 – Visão por livros selecionados com 100% de avaliações positivas

Figura 21 – Visão por livros selecionados com 0% de avaliações positivas

4.5 Análise e Discussões

Com base nos resultados apresentados, na visão geral foram obtidas mais de 60% de aprovações dos livros recomendados aos usuários. Porém, aproximadamente, 37% das recomendações foram reprovadas, fazendo com que a precisão do algoritmo caísse mais de 30%.

Porém, pelos resultados obtidos conforme a visão detalhada, foi possível perceber que 84% dos livros selecionados obtiveram uma avaliação com 50% ou mais de aprovação nas suas respectivas recomendações, o que aumentou em mais de 20% a precisão do algoritmo utilizado para este trabalho.

Conforme a apresentação dos títulos dos livros selecionados nas figuras 20 e 21 e a descrição da categoria dos livros existentes na base foi possível perceber que os livros que obtiveram 100% de reprovação corresponderam àqueles cujos temas são referentes a literatura geral ou mais específicos, existentes na base da ETEC, enquanto que os que obtiveram 100% de aprovação corresponderam a temas comuns à composição da base de dados. Isso ocorreu devido ao algoritmo utilizar o cálculo de similaridade entre cada um dos livros presentes na base de dados com valor no intervalo entre 0 e 1, selecionando os mais próximos de 1 para a recomendação, mesmo que estes estejam também próximos a 0, graças a ausência de temas em comum.

Além disso, o algoritmo não tratou da redução por raízes (ou *stemming*, em inglês) e, tampouco, de retirar sinônimos e palavras com duplo sentido do conteúdo das descrições dos livros presentes na base, fato este que também pode influenciar nos cálculos de similaridade.

5 Conclusões e Trabalhos Futuros

Com este trabalho, foi possível apresentar uma proposta para a recomendação de itens aos usuários de uma biblioteca universitária, e comprovar que a técnica de filtragem baseada em conteúdo e os algoritmos de TF–IDF, normalização cosseno e similaridade cosseno em modelos de espaços vetoriais empregados obtiveram resultados satisfatórios, atingindo uma maioria de avaliações e *feedbacks* positivos, principalmente no que diz respeito a disponibilidade de livros sobre um mesmo tema ou temas correlatos. Além disso, o sistema desenvolvido proporcionou um serviço personalizado para cada usuário.

Após os experimentos, é possível afirmar que este sistema pode ser integrado e aplicado ao cenário de bibliotecas universitárias, no qual exista a interação dos usuários com diversos livros por intermédio de locações e, dessa forma, poder sugerir novas obras literárias para os estudantes, assim como ocorreu nos testes executados, com as ações de seleção e recomendação de livros.

5.1 Sugestões para Trabalhos Futuros

Como trabalhos futuros, vislumbra-se as seguintes melhorias para o sistema desenvolvido neste trabalho:

- A criação de uma interface amigável em plataforma *Web*, aproveitando que o *Django* oferece todo o suporte necessário para este fim, ou também, a extensão para outras plataformas, tais como *desktop* e *mobile*;
- A definição de um valor limitante para a tomada de decisão de recomendação de forma que, caso existam apenas livros muito distantes na base, com um valor de similaridade mais próximo de 0, o sistema deverá retornar que não foram encontrados livros para recomendação. Este limitante pode ser definido de acordo com o problema a ser resolvido e com os dados presentes na base;
- A implementação de um processo de aprendizado, conforme as avaliações recebidas pelos usuários e conforme pesos para avaliações pessoais e globais, definindo assim se os livros previamente recomendados deverão ou não ser apresentados novamente ou em quais casos os *feedbacks* negativos deverão ser levados em consideração;
- A integração com outras API's de livros que disponibilizem conteúdo e contemplem uma quantidade maior de obras literárias. Neste caso, poderia também ser desenvolvido um robô (*crawler*) que coletasse dados de livre acesso sobre livros de *sites* que não tivessem uma API disponível;

- A criação e incorporação de técnicas de filtragem colaborativa ao sistema de recomendação para que o algoritmo levasse em consideração também as ações de usuários com interesses em comum;
- O desenvolvimento de técnicas de redução de raízes (ou *stemming*, em inglês) e retirada de sinônimos e palavras com duplo sentido, visando compor o pré–processamento dos livros.

Referências

ADOMAVICIUS, G.; TUZHILIN, A. *Context-Aware Recommender Systems*. [S.l.], 2010. Disponível em: <http://ids.csom.umn.edu/faculty/gedas/nsfcareer/CARS-chapter-2010.pdf>. Acesso em: 01/06/2017. Citado na página 35.

AMARAL, F. *Introdução a Ciência de Dados - Mineração de Dados e Big Data*. 1. ed. Rio de Janeiro, RJ: Editora Alta Books, 2016. Citado 12 vezes nas páginas 8, 13, 15, 17, 18, 20, 21, 22, 23, 24, 27 e 29.

BALABANOVIC, M.; SHOHAM, Y. Fab: Content-based, collaborative recommendation. *Communications of the ACM*, 1997. Disponível em: <https://www.ischool.utexas.edu/~i385q/readings/Balabanovic_Shoham-1997-Fab.pdf>. Acesso em: 06/06/2017. Citado 2 vezes nas páginas 31 e 34.

BAUDISCH, A. Ciência de dados é explorar o big data para fazer perguntas para prever o futuro. *Medium*, 2016. Disponível em: <https://medium.com/@AlfredBaudisch/o-que-%C3%A9-ci%C3%AAncia-de-dados-data-science-7af5bdac101a>. Acesso em: 03/05/2017. Citado 2 vezes nas páginas 8 e 16.

BRAGA, L. P. V. *Introdução à Mineração de Dados*. 2. ed. [S.l.]: Editora e-papers, 2005. Citado 4 vezes nas páginas 23, 24, 25 e 26.

CARVALHO, H. M. *Aprendizado de Máquina voltado para Mineração de Dados: Árvores de Decisão*. Tese (Doutorado) — Universidade de Brasília (UNB), Brasília, DF, 2014. Disponível em: <https://fga.unb.br/articles/0000/5556/TCC_Hialo_Muniz.pdf>. Acesso em: 08/03/2017. Citado 6 vezes nas páginas 8, 23, 25, 27, 28 e 29.

DOMINGOS, P. A few useful things to know about machine learning. *Communications of the ACM*, 2012. Disponível em: <https://homes.cs.washington.edu/~pedrod/papers/cacm12.pdf>. Acesso em: 05/06/2017. Citado na página 28.

FIGUEIREDO, N. M. A modernidade das cinco leis de ranganathan. *IBICT*, 1992. Disponível em: <http://revista.ibict.br/ciinf/article/viewFile/430/430>. Acesso em: 11/06/2017. Citado na página 44.

FINNEGAN, M. Boeing 787s to create half a terabyte of data per flight, says virgin atlantic. *Computer World UK - from IDG*, 2013. Disponível em: <http://www.computerworlduk.com/data/boeing-787s-create-half-terabyte-of-data-per-flight-says-virgin-atlantic-3433595>. Acesso em: 21/05/2017. Citado na página 21.

GOLDSCHMIDT, R.; BEZERRA, E. Exemplos de aplicações de data mining no mercado brasileiro. *Computer World*, 2016. Disponível em: <http://computerworld.com.br/exemplos-de-aplicacoes-de-data-mining-no-mercado-brasileiro>. Acesso em: 14/03/2017. Citado na página 26.

HAN, J.; KAMBER, M. *Data Mining: Concepts and Techniques*. 2. ed. San Francisco, CA: Editora Morgan Kaufmann, 2001. Disponível em: <http://ccs1.hnue.edu.vn/hungtd/DM2012/DataMining_BOOK.pdf>. Acesso em: 01/06/2017. Citado na página 23.

HAND, D.; MANNILA, H.; SMYTH, P. *Principles of Data Mining*. Cambridge, MA: Editora MIT Press, 2001. Citado na página 23.

HERLOCKER, J. L.; KONSTAN, J. A. *Understanding and Improving Automated Collaborative Filtering Systems*. Tese (Doutorado) — University of Minnesota, 2000. Disponível em: <https://www.researchgate.net/publication/34479898_Understanding_and_improving_automated_collaborative_filtering_systems>. Acesso em: 01/06/2017. Citado 3 vezes nas páginas 32, 33 e 36.

ISINKAYE, F. O.; FOLAJIMI, Y. O.; OJOKOH, B. A. Recommendation systems: Principles, methods and evaluation. *Egyptian Information Journal*, 2015. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1110866515000341>. Acesso em: 17/04/2017. Citado 5 vezes nas páginas 8, 31, 33, 36 e 40.

KREBS, L. M. *Sistemas de Recomendação para Bibliotecas Universitárias*. Tese (Doutorado) — Universidade Federal do Rio Grande do Sul (UFRGS), Porto Alegre, RS, 2013. Disponível em: <http://www.lume.ufrgs.br/bitstream/handle/10183/78367/000899325.pdf?sequen>. Acesso em: 20/03/2017. Citado 2 vezes nas páginas 13 e 43.

LOPES, G. R. *Sistemas de Recomendação para Bibliotecas Digitais sob a Perspectiva da Web Semântica*. Tese (Doutorado) — Universidade Federal do Rio Grande do Sul (UFRGS), Porto Alegre, RS, 2007. Disponível em: <http://www.lume.ufrgs.br/bitstream/handle/10183/10747/000601051.pdf?sequence=1>. Acesso em: 10/04/2017. Citado 6 vezes nas páginas 8, 31, 33, 36, 39 e 40.

MANGAS, S. F. Como planificar e gerir um serviço de referência. *Biblios*, 2007. Disponível em: <http://sisbib.unmsm.edu.pe/bibvirtualdata/publicaciones/biblios/n28/a02n28.pdf>. Acesso em: 11/06/2017. Citado na página 43.

MITCHELL, T. M. *Machine Learning*. New York, NY: Editora McGraw-Hill International Editions, 1997. Citado na página 27.

MONTANER, M.; LóPEZ, B.; ROSA, J. L. d. l. A taxonomy of recommender agents on the internet. In: _____. Netherlands: Editora Kluwer Academic Publishers, 2003. p. 285–330. Disponível em: <https://pdfs.semanticscholar.org/f381/f58e6921a372ecf5740fd9394ec6bfa145c8.pdf>. Acesso em: 06/06/2017. Citado na página 35.

NORVIG, P.; RUSSEL, S. *Artificial Intelligence: A Modern Approach*. [S.l.]: Editora Prentice Hall, 2010. Acesso em: 07/06/2017. Citado na página 28.

PORTO, F.; ZIVIANI, A. *Ciência de Dados*. Petrópolis, RJ, 2014. Disponível em: <http://www.lncc.br/~ziviani/papers/III-Desafios-SBC2014-CiD.pdf>. Acesso em: 07/03/2017. Citado 5 vezes nas páginas 8, 16, 18, 19 e 20.

PRESSMAN, R. S. *Engenharia de Software: Uma Abordagem Profissional*. 7. ed. São Paulo, SP: Editora Pearson Makron Books, 2011. Citado na página 15.

RASCHKA, S. *Python Machine Learning: Unlock deeper insights into Machine Leaning with this vital guide to cutting-edge predictive analytics*. [S.l.]: Editora Packt Publishing, 2016. Citado na página 47.

REATEGUI, E. B.; CAZARELLA, S. C. *Sistemas de Recomendação*. São Leopoldo, RS, 2005. Disponível em: <http://www.lbd.dcc.ufmg.br/colecoes/enia/2005/0100.pdf>. Acesso em: 13/03/2017. Citado 8 vezes nas páginas 8, 13, 31, 32, 33, 34, 35 e 36.

REZENDE, S. O. *Sistemas Inteligentes: Fundamentos e Aplicações*. Barueri, SP: Editora Manole, 2003. Citado na página 28.

RICCI, F. et al. *Recommender Systems Handbook*. Editora Springer, 2011. Disponível em: <http://www.cs.ubbcluj.ro/~gabis/DocDiplome/SistemeDeRecomandare/Recommender_systems_handbook.pdf>. Acesso em: 10/04/2017. Citado 8 vezes nas páginas 8, 31, 32, 36, 37, 38, 41 e 42.

SAS. Machine learning – o que é e por que é importante? *SAS The Power to Know*, 2016. Disponível em: <https://www.sas.com/pt_br/insights/analytics/machine-learning.html>. Acesso em: 10/04/2016. Citado 4 vezes nas páginas 27, 28, 29 e 30.

SILVA, R. G. N. *Sistema de Recomendação baseado em conteúdo textual: avaliação e comparação*. Tese (Doutorado) — Universidade Estadual de Feira de Santana (UEFS) and Universidade Federal da Bahia (UFB), Salvador, BA, 2014. Disponível em: <https://repositorio.ufba.br/ri/bitstream/ri/19281/1/dissertacao_mestrado_ciencia_computacao_rafael_glauber.pdf>. Acesso em: 09/05/2017. Citado 6 vezes nas páginas 8, 39, 40, 41, 42 e 43.

SINGH, N. Difference between data mining and machine learning. *All Round Experts – Deep Search to Score you More*, 2014. Disponível em: <http://allroundexpert.blogspot.com.br/2014/05/difference-between-data-mining-and.html>. Acesso em: 15/05/2017. Citado na página 30.

TAKAHASHI, M. M. *Estudo Comparativo de Algoritmos de Recomendação*. Tese (Doutorado) — Instituto Matemática e Estatística – Universidade de São Paulo (IME-USP), São Paulo, SP, 2015. Disponível em: <https://linux.ime.usp.br/~marcost/mac0499/monografia_final.pdf>. Acesso em: 06/03/ 2017. Citado 8 vezes nas páginas 13, 27, 29, 31, 33, 34, 41 e 57.

TAN, A.-H. *Text Mining: The state of the art and the challenges*. Singapore, 2017. Disponível em: <https://pdfs.semanticscholar.org/9a80/ec16880ae43dc20c792ea3734862d85ba4d7.pdf>. Acesso em: 04/06/2017. Citado na página 27.

TUAN, D. Recommender systems - how they works and their impacts 2012 - content-based filtering. 2012. Disponível em: <http://findoutyourfavorite.blogspot.com.br/2012/04/content-based-filtering.html>. Acesso em: 09/06/2017. Citado 2 vezes nas páginas 8 e 37.

WANG, Y.; WANG, X.-J. A new approach to feature selection in text classification. *Proceedings of 2005 International Conference on*, 2005. Disponível em: <http://www.ijritcc.org/download/1413085566.pdf>. Acesso em: 06/06/2017. Citado na página 41.

YATES, R. B.; NETO, B. R. *Modern Information Retrieval*. New York, NY: Editora ACM Press, 1999. Citado 2 vezes nas páginas 31 e 39.

ZAIDAN, F. H. *Processo de Desenvolvimento de Sistemas de Informação como Forma de Retenção do Conhecimento Organizacional para Aplicação Estratégica: estudo de múltiplos casos*. Tese (Doutorado) — Universidade FUMEC, Belo Horizonte, MG, 2008. Disponível

em: <http://www.fumec.br/anexos/cursos/mestrado/dissertacoes/completa/fernando_hadad_zaidan.pdf>. Acesso em: 15/05/2017. Citado na página 20.