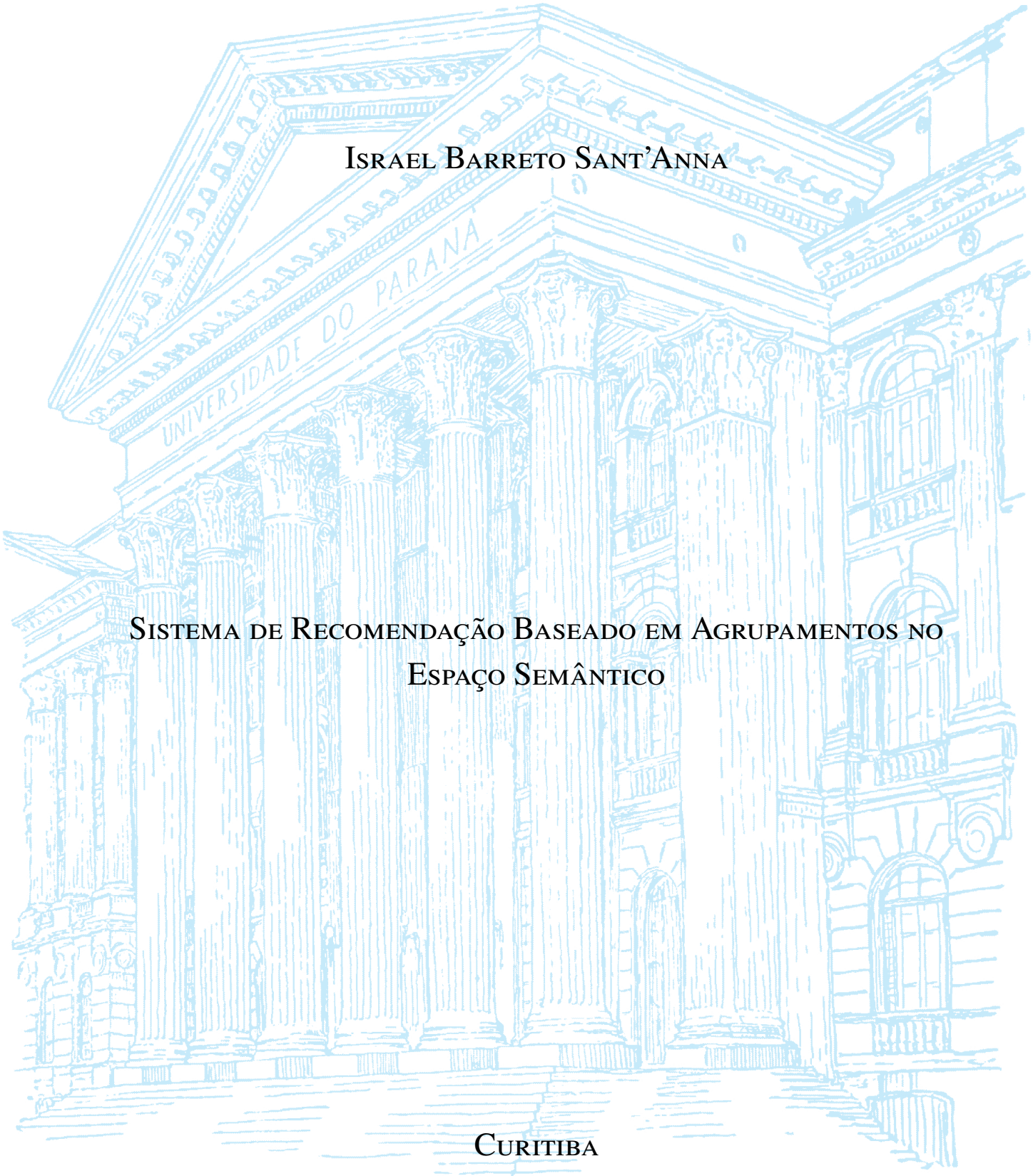
Universidade Federal Do Paraná

Israel Barreto Sant'Anna



Sistema de Recomendação Baseado em Agrupamentos no Espaço Semântico

Curitiba
2017

Israel Barreto Sant'Anna

Sistema de Recomendação Baseado em Agrupamentos no Espaço Semântico

Monografia de Trabalho de Graduação apresentada ao Programa de Bacharelado em Ciência da Computação, Departamento de Informática, Setor de Ciências Exatas, Universidade Federal Do Paraná..

Área de concentração: *Ciência da Computação*.

Orientador: Eduardo Jaques Spinosa.

Curitiba
2017

Agradecimentos

A Deus pelo seu infinito amor, sua fidelidade e seu cuidado durante toda minha vida, me guiando desde a escolha do curso até sua conclusão.

A minha mãe e irmã, que participaram ativamente da minha formação, estando sempre ao meu lado, tanto nos momentos de comemoração quanto nos momentos de dificuldade, me apoiando e motivando a concretizar meus sonhos.

Ao meu orientador, pela sua dedicação e prestatividade, sanando minhas diversas dúvidas e me ajudando a seguir os caminhos corretos na elaboração deste trabalho.

Aos meus amigos, pela disposição em ouvir minhas queixas e temores, sempre me consolando e motivando a seguir em frente.

Aos colegas de curso que me ajudaram a superar os desafios da graduação, seja através de dicas em trabalhos ou grupos de estudos.

E a todos que contribuíram direta ou indiretamente na minha formação.

Resumo

Sistemas de Recomendação servem para guiar os usuários em meio a enorme quantidade de conteúdos presente em serviços atualmente, trazendo a eles itens com maior probabilidade de despertar seus interesses sem que eles necessitem percorrer todos os itens existentes. Esse trabalho propõe um novo algoritmo de recomendação que inicialmente mapeia os itens para um espaço semântico, onde itens similares semanticamente se encontram mais próximos, e em seguida encontra regiões de interesse para cada usuário através de um algoritmo de agrupamento, descobrindo então usuários com gostos similares através das interseções entre essas regiões. Dessa forma, o sistema proposto é capaz de recomendar itens de regiões novas e promissoras para cada usuário, evitando recomendações triviais, como as de itens muito populares que seriam facilmente identificadas por algoritmos convencionais. Um amplo conjunto de métricas foi utilizado para avaliar experimentalmente diversos aspectos da abordagem proposta. Os resultados comprovam o potencial ainda inexplorado que existe na área.

Palavras-chave: Sistema de Recomendação, Espaço Semântico, Agrupamento de Dados.

Abstract

Recommendation Systems serve to guide users trough the overwhelming amount of content existent in services today, bringing to them items that are more likely to spark their interests, without the need for them to go through all existing items. This work proposes a new recommendation algorithm that initially maps the items to a semantic space, where semantically similar items are closer together, and then finds regions of interest for each user through a clustering algorithm, finally discovering users with similar tastes from the intersections between these regions. In this way, the proposed system is able to recommend items from promising new regions for each user, avoiding trivial recommendations, such as those of very popular items that would be easily identified by conventional algorithms. A broad set of metrics was used to experimentally evaluate several aspects of the proposed approach. The results show there's still unexplored potential in the area.

Keywords: Recommender System, Semantic Space, Data Clustering.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Acrônimos

ILD	*Intra-List Distance*
IUF	*Inverse User Frequency*
PV	*Paragraph Vector*
PV-DBOW	*Paragraph Vector - Distributed Bag of Words*
PV-DM	*Paragraph Vector - Distributed Memory*

Capítulo 1

Introdução

Grande parte dos serviços mais conhecidos atualmente se baseiam no consumo de itens pelos usuários, sejam estes itens textos, vídeos, fotos, músicas, ou até mesmo produtos físicos. Para atender usuários com os mais variados gostos, os serviços oferecem uma quantidade enorme de itens. E, em alguns serviços, os itens são criados pelos próprios usuários, que chegam a gerar milhares de novos itens a cada segundo. Essa grande quantidade de itens acaba sendo prejudicial para o usuário, que se perde em meio a tanta informação e não encontra o que de fato seria de seu interesse (Ricci et al., 2015a).

Sistemas de Recomendação surgem como uma forma de guiar os usuários nesse mar de opções, aprendendo seus gostos de forma a apresentar os itens mais prováveis de despertar seu interesse, além de itens novos e diferentes que levem o usuário a descobrir novos interesses. Essas recomendações podem ser feitas tanto a partir de itens parecidos aos que o usuário gostou, quanto a partir de usuários com gostos parecidos. Ambas as opções apresentam vantagens e desvantagens, e a junção das duas acaba sendo uma boa opção para evitar as desvantagens de cada uma.

Uma forma de recomendar itens independentemente do tipo de conteúdo é através das suas descrições textuais, que podem ser mapeadas para um espaço onde a proximidade dos itens tem relação direta com a sua similaridade, chamado de Espaço Semântico. Aliando isso aos conceitos de Agrupamento de Dados, é possível encontrar regiões no espaço onde um usuário tem maior interesse. Este trabalho propõe um novo algoritmo para Sistemas de Recomendação que busca encontrar usuários com gostos similares através da descoberta dessas regiões de interesse, recomendando itens que usuários com regiões interseccionadas gostaram, mas o usuário atual ainda não interagiu, unindo dessa forma a estratégia de recomendação por itens e por usuários.

Esta monografia está organizada em 6 Capítulos. O Capítulo 2 define alguns conceitos de Sistemas de Recomendação, como Filtragem Colaborativa e Filtragem Baseada em Conteúdo. O Capítulo 3 explica o que são Espaço Semântico e Agrupamento de Dados e apresenta os algoritmos *Paragraph Vector* e *Density Peaks Clustering* que foram utilizados nesse trabalho. O Capítulo 4 traz a abordagem proposta para o algoritmo de recomendação utilizado neste trabalho. O Capítulo 5 apresenta os detalhes da implementação e dos experimentos realizados, assim como seus resultados. O Capítulo 6 apresenta as considerações finais e oportunidades para trabalhos futuros.

Capítulo 2

Sistemas de Recomendação

2.1 Introdução

De acordo com Ricci et al. (2015b), Sistemas de Recomendação são técnicas e ferramentas de software que proporcionam sugestões de itens que provavelmente são de interesse para um usuário em particular. Esses itens variam de acordo com o objetivo do software, podendo ser músicas, filmes, produtos comerciais, entre outros.

Como dito por Ricci et al. (2015a), a ideia dos Sistemas de Recomendação surgiu a partir da observação de que as pessoas geralmente se baseiam em recomendações de outras pessoas na hora de fazer decisões, seja através de cartas de recomendação num emprego, críticas de filmes, ou mesmo relatos de amigos sobre produtos.

Além disso, atualmente o volume de conteúdo na Web é enorme, sendo muitas vezes impossível para um usuário percorrer todos os itens de um site para encontrar os que mais lhe interessam. A grande variedade de opções acaba sobrecarregando o usuário ao invés de beneficiá-lo (Ricci et al., 2015a). Portanto, os Sistemas de Recomendação servem também para facilitar a navegação do usuário por meio desses diversos itens, trazendo os mais relevantes ao usuário sem a necessidade dele analisar diversos itens irrelevantes no caminho.

Existem diversas abordagens diferentes em relação as técnicas utilizadas para encontrar as recomendações, sendo as principais a Filtragem Colaborativa e a Filtragem Baseada em Conteúdo, cada uma com suas vantagens e desvantagens, sendo possível ainda a utilização delas em conjunto, buscando evitar algumas dessas desvantagens.

2.2 Filtragem Colaborativa

A Filtragem Colaborativa realiza as recomendações se baseando em itens que o usuário ativo ainda não interagiu, mas que outros usuários com gostos parecidos se interessaram (Ricci et al., 2015c)

Esse interesse geralmente é medido por alguma forma de avaliação do item, como uma nota alta dada pelo usuário, por exemplo, mas também podem ser utilizadas informações implícitas, como a simples visualização do item pelo usuário (Aggarwal, 2016a).

A similaridade entre dois usuários pode então ser calculada se baseando em avaliações parecidas aos mesmos itens (Ricci et al., 2015c). Eventualmente terão itens que um usuário avaliou mas o outro ainda não, a Filtragem Colaborativa então infere essa avaliação inexistente devido a alta correlação entre as avaliações dos outros usuários similares a este, e se essa inferência for positiva, recomenda o item ao usuário (Aggarwal, 2016b).

Apesar de efetiva, essa abordagem acaba apresentando problemas em casos onde o usuário realizou poucas ou nenhuma avaliação, pois para aumentar a precisão da similaridade é necessário uma quantidade significativa de dados, logo, nesses casos as recomendações acabam não sendo tão efetivas. Isso é bem presente em sistemas novos, onde a base de usuários está começando a ser criada, sendo conhecido como o problema de *cold start* (Aggarwal, 2016c).

Outro problema resultante dessa abordagem é que um item recém inserido no sistema não será recomendado, pois ainda não possui uma avaliação, e devido a grande quantidade de itens no sistema ele pode acabar não sendo encontrado pelos usuários, e consequentemente nunca será avaliado ou recomendado.

2.3 Filtragem Baseada em Conteúdo

De acordo com Aggarwal (2016d), em Sistemas de Recomendação com Filtragem Baseada em Conteúdo, os atributos descritivos dos itens são usados para fazer as recomendações.

Através desses atributos descritivos é calculada a similaridade entre os itens que o usuário já demonstrou interesse e os itens que ele ainda não interagiu, recomendando então os de maior similaridade, uma vez que o usuário provavelmente irá se interessar em conteúdos pelo qual ele já se interessou no passado.

Essa abordagem não possui os problemas citados na Filtragem Colaborativa pois pode recomendar itens sem a necessidade da avaliação dos usuários, logo, itens recém criados também serão recomendados, e usuários que não realizaram avaliações ainda podem receber recomendações de itens similares à algum que ele venha a visualizar.

Entretanto, possui outros problemas, como a realização de recomendações *óbvias* em muitos casos, pois só serão recomendados itens parecidos, enquanto que o usuário pode se interessar por itens de conteúdos diversificados que nunca lhe serão recomendados (Aggarwal, 2016d).

Capítulo 3

Espaço Semântico e Agrupamento

3.1 Espaço Semântico

A semântica é um ramo da linguística que estuda o significado das palavras e suas mudanças de sentido ao longo do tempo (Weiszflog, 2017). Aliando isso à observação de Firth (1957) de que você conhece uma palavra pela companhia que ela mantém, o Espaço Semântico surge como um espaço euclidiano onde as palavras são organizadas de acordo com a similaridade de seu significado num contexto, sendo os eixos desse espaço definidos pela quantidade de vezes que as palavras ocorrem juntas (Lowe, 2001).

A ideia é que a compreensão do significado de uma palavra depende da frase em que ela está presente, sendo possível uma palavra ter significados diferentes em frases diferentes, ou palavras diferentes terem significado semelhante numa frase. Ao contar as co-ocorrências de uma palavra com outras d palavras, é possível a inserção dessas palavras em um espaço de d dimensões, fazendo com que a distância de uma palavra a outra nesse espaço represente quão similares elas são (Lowe, 2001). Logo, palavras próximas no Espaço Semântico podem ser substituídas entre si sem perda de significado no contexto.

Esse mesmo conceito pode ser usado também para textos, ou seja, é possível mapear textos em um Espaço Semântico de forma que os textos mais similares em significado num contexto estejam posicionados mais próximos entre si. Um algoritmo que realiza esse mapeamento é o *Paragraph Vector*.

3.1.1 *Paragraph Vector*

Paragraph Vector (vetor do parágrafo - PV) (Le e Mikolov, 2014) é um algoritmo não supervisionado que aprende representações de tamanho fixo a partir de pedaços de textos de tamanho variável, como sentenças, parágrafos e documentos.

Nele cada parágrafo é mapeado para um vetor único, representado por uma coluna numa matriz D, e cada palavra também é mapeada para um vetor único, representado por uma coluna numa matriz W. A matriz de palavras W é compartilhada por todos os parágrafos, e o vetor do parágrafo é compartilhado por todos os contextos gerados desse parágrafo, onde contextos são janelas de tamanho fixo que deslizam pelo parágrafo.

Para realizar o treinamento dos vetores existem dois modelos diferentes: o modelo de memória distribuída, *Paragraph Vector - Distributed Memory* (PV-DM), e o modelo de saco de palavras distribuído, *Paragraph Vector - Distributed Bag of Words* (PV-DBOW). Ambos utilizam os métodos de descida estocástica de gradiente (*Stochastic Gradient Descent* - SGD) e retropropagação (*backpropagation*).

No primeiro modelo, os vetores do parágrafo e das palavras de um contexto são usados em conjunto, através da média ou concatenação, para prever a palavra seguinte ao contexto, como mostrado na Figura 3.1. Dessa forma, o vetor do parágrafo pode ser pensado como uma outra palavra que age como uma memória, lembrando o tópico do parágrafo e ajudando a achar o que está faltando no contexto (Le e Mikolov, 2014).

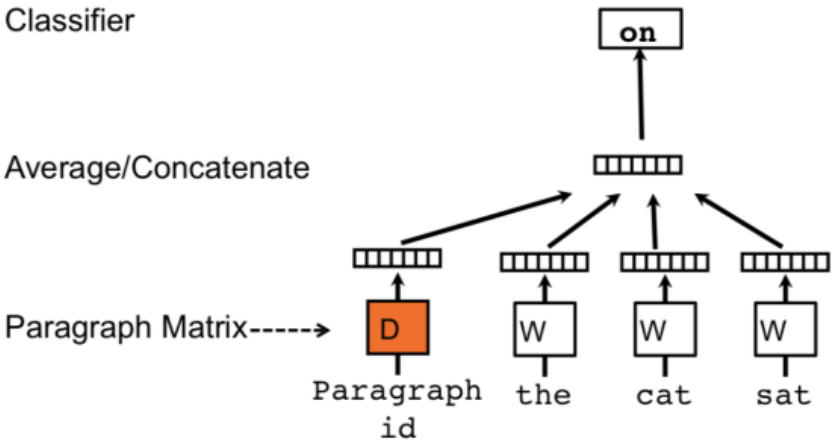


Figura 3.1: Estrutura do modelo PV-DM. FONTE: Le e Mikolov (2014).

Já no PV-DBOW, o contexto é ignorado e o modelo é forçado a prever palavras do parágrafo selecionadas aleatoriamente (Le e Mikolov, 2014). Ou seja, nesse modelo, o parágrafo é treinado para prever as palavras que ele contém, como visto na Figura 3.2.

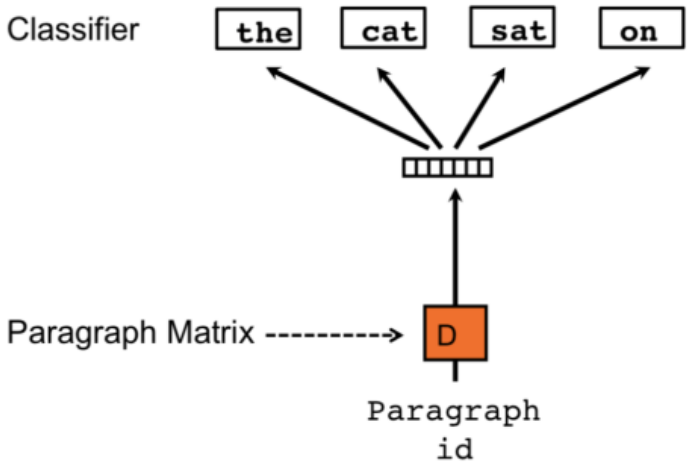


Figura 3.2: Estrutura do modelo PV-DBOW. FONTE: Le e Mikolov (2014).

3.2 Agrupamento de Dados

O problema de agrupamento pode ser definido basicamente como: a partir de um conjunto de dados, particionar esse conjunto em vários grupos (*clusters*) de forma que os dados de um mesmo grupo sejam os mais similares possíveis (Aggarwal e Reddy, 2013a).

Como dito por MacKay (2005), o cérebro humano é bom em encontrar uniformidade nos dados. Uma forma de expressar essa uniformidade é separando um conjunto de objetos em grupos que são similares entre si. Essa operação de colocar objetos similares juntos é chamada agrupamento (*clustering*).

Há vários motivos para agrupar dados, sendo um dos principais o seu poder preditivo. Como consequência de suas similaridades, objetos de um mesmo grupo tendem a possuir o mesmo comportamento. Logo, ao encontrar um novo objeto e agrupá-lo de acordo com alguma característica, é possível realizar previsões sobre outras características não presentes inicialmente, mas que outros objetos do grupo possuem. Dessa forma, o agrupamento é útil pois leva a uma melhor descrição dos dados, ajudando a fazer melhores escolhas sobre os objetos no futuro (MacKay, 2005).

Existem diversos métodos diferentes para agrupar os dados, sendo possível separá-los em 3 categorias principais: métodos de particionamento, métodos hierárquicos e métodos baseados em densidade. Os métodos de particionamento criam partições sem sobreposição, com base em métricas de distância entre os objetos. Os métodos hierárquicos particionam os dados em diferentes níveis, de forma que haja algum tipo de hierarquia entre eles, facilitando o resumo e visualização dos dados. Por último, os métodos baseados em densidade criam grupos em volta de objetos que possuem uma alta densidade de dados ao seu redor, e baixa densidade entre este objeto e objetos de outros grupos. Eles são interessantes pois conseguem capturar agrupamentos com formatos arbitrários, como um S, por exemplo (Aggarwal e Reddy, 2013b).

3.2.1 *Density Peaks Clustering*

O algoritmo de agrupamento *Density Peaks Clustering* utiliza tanto a distância entre os objetos como a densidade deles, conseguindo assim encontrar agrupamentos em formatos não esféricos e sem precisar de um parâmetro definindo o número de agrupamentos presente no conjunto de dados (Rodriguez e Laio, 2014).

O algoritmo se baseia na suposição de que os centros dos grupos são cercados por vizinhos com menor densidade local, e estão afastados de qualquer ponto com uma alta densidade local. Ou seja, o objeto que está no centro de um grupo possui vários objetos próximos a ele e está distante de outro objeto que também possui vários vizinhos próximos, pois este outro objeto é o centro de outro grupo.

Para cada objeto i são calculadas a sua densidade local ρ_i e a sua distância δ_i de objetos de maior densidade que a dele. A Equação 3.1 apresenta o cálculo de ρ_i, onde $\chi(x) = 1$ se $x < 0$ e $\chi(x) = 0$ caso contrário, d_{ij} é a distância entre o objeto i e um objeto j e d_c é uma distância de corte. Ela basicamente conta quantos objetos são mais próximos de i do que de d_c. O cálculo de δ_i é feito pela Equação 3.2, que retorna a menor distância para os objetos de maior ρ que i (Rodriguez e Laio, 2014).

$$\rho_i = \sum_j \chi(d_{ij} - d_c) \tag{3.1}$$

$$\delta_i = \min_{j:\rho_j > \rho_i} (d_{ij}) \tag{3.2}$$

Dessa forma, como demonstra a Figura 3.3, os centros dos grupos terão valores altos para ρ e δ, enquanto os outros membros do grupo terão um valor alto para ρ mas baixo para δ, e os objetos que estão isolados (*outliers*), ou seja, que não pertencem a nenhum grupo, terão um valor alto para δ e baixo para ρ (Rodriguez e Laio, 2014).

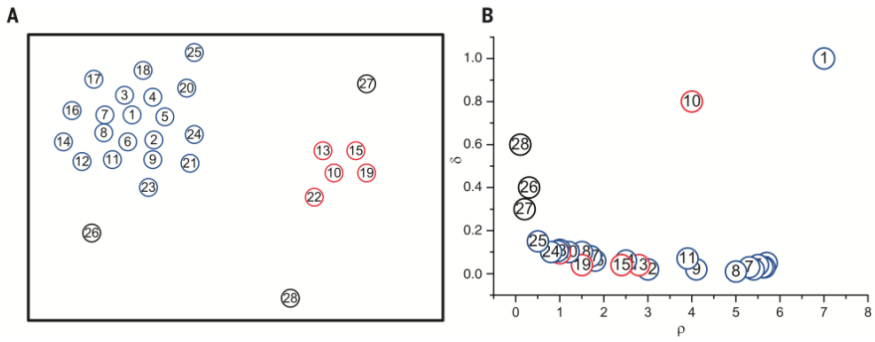


Figura 3.3: Exemplo do algoritmo *Density Peaks Clustering*. (A) Distribuição dos objetos ranqueados em ordem decrescente de densidade. (B) Gráfico de dispersão de δ como uma função de ρ para os dados em (A). Diferentes cores correspondem a diferentes grupos. FONTE: Rodriguez e Laio (2014).

Capítulo 4

Abordagem Proposta

4.1 Introdução

Tendo em vista as vantagens e desvantagens das diferentes abordagens dos Sistemas de Recomendação (Capítulo 2), foi elaborada uma estratégia que utilizasse tanto Filtragem Baseada em Conteúdo como Filtragem Colaborativa. Para isso, foi criada uma representação dos itens a partir das suas descrições e informações textuais, de forma a deixar o algoritmo mais genérico, não o limitando a apenas recomendações de imagens ou músicas, por exemplo.

A partir dessa representação, um algoritmo de agrupamento é utilizado para definir os gostos do usuário, possibilitando a detecção de usuários de gostos parecidos e a realização de recomendações colaborativas de maior diversidade.

4.2 Representação dos Itens

Para a representação dos itens foi utilizado o algoritmo PV-DBOW (Seção 3.1.1), devido a sua capacidade de mapear as informações textuais dos itens em vetores multidimensionais, permitindo a criação de um Espaço Semântico (Capítulo 3) onde a distância entre os vetores determina a similaridade dos itens.

Com essa representação itens parecidos são posicionados próximos entre si, o que possibilita recomendações baseadas na distribuição desses itens no espaço.

A Tabela 4.1 demonstra como os itens mais parecidos ficam próximos entre si, mostrando os 10 filmes mais similares a *Toy Story* no espaço semântico.

Tabela 4.1: Exemplo de filmes similares a *Toy Story*

Filme	**Similaridade**
Toy Story 2	0,708
E.T. the Extra-Terrestrial	0,496
A Bug's Life	0,495
Big	0,491
Gremlins	0,480
Pinocchio	0,464
Small Soldiers	0,456
The Transformers: The Movie	0,450
The Tigger Movie	0,447
House Arrest	0,440

4.3 Descoberta das Regiões de Interesse dos Usuários

Uma vez que temos a distribuição dos itens no espaço semântico, podemos descobrir regiões desse espaço em que um certo usuário tem maior interesse através da posição dos itens que ele avaliou positivamente.

Se o sistema possuir uma avaliação de itens não binária, a binarização é feita através do cálculo da média total das avaliações do usuário, considerando então as avaliações iguais ou acima dessa média como avaliações positivas, e as avaliações abaixo da média como negativas.

Uma vez que a forma como os itens avaliados positivamente se distribuem no espaço não é conhecida, nem a quantidade de agrupamentos possíveis desses itens, utilizou-se o algoritmo *Density Peaks Clustering* (Seção 3.2.1) para encontrar esses agrupamentos, pois se comporta bem nessas condições, encontrando agrupamentos de formas variadas, sem deixar a alta dimensionalidade do espaço semântico afetar seus resultados.

Em cada agrupamento encontrado, o item central tem sua distância calculada em relação aos outros itens do agrupamento. A partir da maior distância encontrada, é então criada uma hiperesfera que representa a região de interesse do usuário.

4.4 Recomendações para Usuários com Interesses em Comum

Para encontrar usuários de gostos similares, cada usuário tem suas hiperesferas que representam suas regiões de interesse comparadas com a de todos os outros usuários. Se uma interseção é encontrada, significa que os dois usuários avaliaram positivamente itens similares, e portanto possuem uma região de interesse em comum.

Além disso, uma vez que eles possuem um interesse em comum, é provável que um usuário possua alguma outra região de interesse que também seja interessante para o outro usuário, mas que este ainda não teve contato.

Portanto, a lista de recomendações para um usuário é composta pelos itens presentes nas regiões de interesse de cada usuário com quem ele possui uma interseção, com exceção dos itens já avaliados pelo usuário.

Isso pode ser formalizado pela Equação 4.1, sendo R_i a lista de recomendações de um usuário i, U_i o conjunto de usuários com que um usuário i possui uma intercessão, I_u o conjunto de itens dentro das regiões de interesse de um usuário u, e A_i o conjunto de itens avaliados por um usuário i.

$$R_i = \bigcup_{u \in U_i} I_u - A_i \tag{4.1}$$

A Figura 4.1 traz um exemplo visual de como a recomendação entre dois usuários funciona no espaço semântico.

Caso o algoritmo de agrupamento não tenha encontrado regiões de interesse para o usuário, ou não exista nenhuma intercessão entre suas regiões de interesse e a dos demais usuários, serão recomendados os itens de maior similaridade aos itens avaliados positivamente por esse usuário.

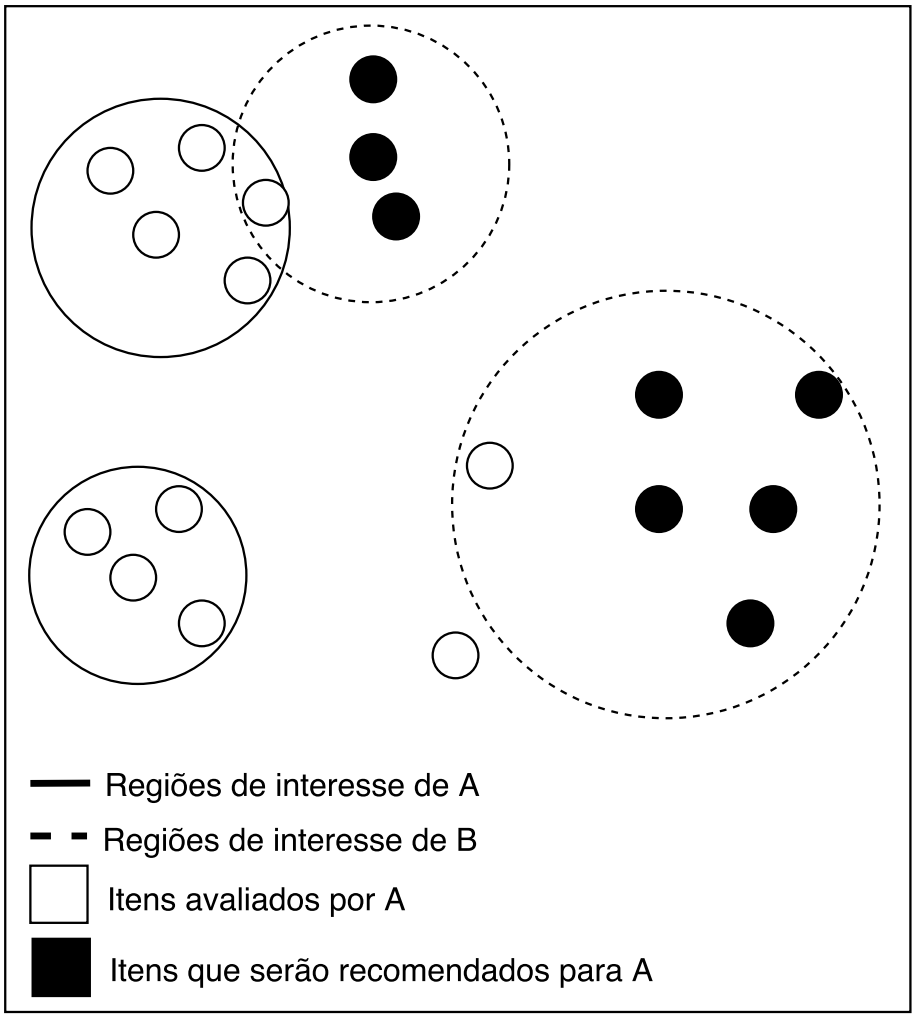


Figura 4.1: Exemplo de recomendações através de regiões de interesse no espaço semântico

Capítulo 5

Experimentos

5.1 Base de Dados

Para realizar os experimentos foi utilizada a base de dados MovieLens 1M, criada pelo GroupLens (2003), devido ao seu fácil acesso e ampla utilização na comunidade científica. Ela possui originalmente 1 milhão de avaliações feitas por 6.040 usuários diferentes a 3.883 filmes.

A partir dessa lista de filmes foi então realizada uma extração de dados sobre eles, de forma a adquirir informações descritivas como sinopse, gêneros e críticas, para alimentar o algoritmo PV (Seção 3.1.1). Essas informações foram extraídas do site IMDb (1990), utilizando a biblioteca Python ImdbPie (O'Dwyer, 2013). Infelizmente alguns filmes não foram encontrados pela biblioteca, e portanto tiveram que ser retirados dos experimentos, assim como suas avaliações.

Desse modo, a base de dados sobre qual os experimentos foram realizados contém 3.427 filmes, 6.040 usuários e 943.644 avaliações. Sendo as informações dos filmes compostas por nome, sinopse, gêneros e textos de 20 críticas feitas por usuários do IMDb para cada filme.

5.2 Pré-Processamento

Em alguns testes iniciais com os *Paragraph Vectors* verificou-se que em alguns casos, apesar dos filmes mais próximos serem parecidos em contexto, eram de gêneros opostos. *Toy Story*, por exemplo, um filme infantil sobre brinquedos que vivem aventuras enquanto os humanos não estão olhando, estava próximo de Brinquedo Assassino, filme de terror em que um brinquedo ganha vida e se torna um psicopata.

Levando em conta o processo de treinamento do PV (Seção 3.1.1), foi levantada a hipótese de que isso ocorria devido a quantidade de palavras contendo os gêneros dos filmes ser muito pequena comparada ao resto dos conteúdos. Desse modo, foram então colocados pesos nas entradas, de forma que o texto referente a um atributo i (gênero, sinopse ou crítica) era copiado uma quantidade x_i de vezes, sendo x_i o peso atribuído ao atributo i. Esse novo texto referente ao atributo é então concatenado com os textos dos outros atributos e passado como entrada para o PV.

Outro pré-processamento necessário é a separação dos filmes entre gostados ou não pelo usuário. Como as avaliações contidas na base MovieLens possuem um valor entre 1 e 5, para poder identificar os filmes de que o usuário gostou, levando em conta os efeitos de ancoragem descritos por Koren (2010), onde a avaliação de um usuário deve ser considerada relativa a suas outras avaliações, a média dos valores de todas as avaliações realizadas por este usuário é calculada, e então os filmes que receberam avaliações acima dessa média pelo usuário são

escolhidos para realizar o agrupamento. Antes disso, esses filmes avaliados positivamente são separados entre treino e teste, sendo 20% para teste e o resto para treino. Isso é feito para cada usuário.

Na fase de Descoberta das Regiões de Interesse de um Usuário (Seção 4.3), o algoritmo de *Density Peaks Clustering* (Seção 3.2.1) classifica os itens de acordo com sua densidade local δ e distância aos itens de maior densidade ρ, sendo necessária então a escolha de um valor de densidade e distância de corte para definir quais serão os centros dos agrupamentos. Para poder automatizar esse processo, os valores de δ e ρ, que variam de acordo com a distribuição dos itens de cada usuário, são normalizados entre 0 e 1 utilizando a função de normalização min-max, e então um valor de corte é definido para δ e ρ, conforme experimentos relatados na Seção 5.11.

5.3 Métricas

Os resultados dos experimentos foram avaliados em 3 quesitos: exatidão, diversidade e novidade.

Como o algoritmo proposto não prediz uma nota para os itens, apenas escolhe itens que podem ser de interesse para o usuário, as métricas escolhidas para avaliar a exatidão foram *Precision*, *Recall* e F_1Score, que é a média harmônica das duas anteriores. *Precision* avalia a quantidade de recomendações corretas dentre todas as recomendações, enquanto *Recall* avalia a quantidade de recomendações corretas dentre toda a base de teste. Seus cálculos são feitos através das Equações 5.1 e 5.2 respectivamente, sendo R o conjunto de itens recomendados ao usuário e T o conjunto de itens da base de teste do usuário. A Equação 5.3 é referente ao cálculo do F_1Score.

$$Precision = \frac{|R \cap T|}{|R|} \tag{5.1}$$

$$Recall = \frac{|R \cap T|}{|T|} \tag{5.2}$$

$$F_1Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{5.3}$$

Ao analisar os resultados da avaliação de exatidão do algoritmo é importante considerar que a base utilizada não possui a interação dos usuários, tendo que assumir portanto que um usuário não gosta dos filmes que não estão presentes na sua base de teste, sendo que é possível que o usuário tenha interesse pelos filmes mas ainda não os avaliou, ou simplesmente não saiba da existência desses filmes, vindo a descobrir um interesse por eles devido as recomendações (Ricci et al., 2015d).

Para avaliar a diversidade foi utilizada a métrica *Intra-List Distance* (ILD), que avalia quão similares são os itens presentes nas recomendações de um usuário, tendo portanto uma diversidade maior ou menor devido a essa similaridade. Seu valor é calculado através da Equação 5.4, onde R é o conjunto de itens recomendados ao usuário e $d(i, j)$ é uma medida de distância entre i e j, que no caso destes experimentos foi definida conforme relatado na Seção 5.10.

$$ILD = \frac{1}{|R| \times |R-1|} \times \sum_{i \in R} \sum_{j \in R} d(i, j) \tag{5.4}$$

A avaliação de novidade foi realizada através das métricas *Unexpectedness* e *Inverse User Frequency* (IUF). A primeira avalia quão diferente os itens recomendados são em relação

aos itens que um usuário já avaliou, tendo seu valor calculado através da Equação 5.5, onde I é o conjunto de itens avaliados pelo usuário, e R e $d(i, j)$ tem as mesmas definições da métrica ILD (Ricci et al., 2015e). A segunda avalia a popularidade dos itens recomendados, partindo da ideia de que itens que tiveram um número menor de interações com os usuários, ou seja, um item menos popular, traz maior novidade para uma recomendação. Seu cálculo é feito através da Equação 5.6, sendo R o conjunto de itens recomendados ao usuário, U o conjunto total de usuários e U_i o conjunto de usuários que avaliaram o item i.

$$Unexpectedness = \frac{1}{|R| \times |I|} \times \sum_{i \in R} \sum_{j \in I} d(i, j) \tag{5.5}$$

$$IUF = \frac{1}{|R|} \times \sum_{i \in R} \log_2 \frac{|U|}{|U_i|} \tag{5.6}$$

Outro fator importante na avaliação de novidade é quão relevante esses novos itens são, fator conhecido pelo termo *serendipity* (Ricci et al., 2015f). Para realizar esta avaliação são utilizadas as Equações 5.5 e 5.6. Elas são basicamente iguais as Equações 5.5 e 5.6, modificando apenas o uso de R para $R \cap T$, onde T tem a mesma definição das métricas de *Recall* e *Precision*, avaliando então a novidade apenas das recomendações corretas.

$$S\_Unexp = \frac{1}{|R \cap T| \times |I|} \times \sum_{i \in R \cap T} \sum_{j \in I} d(i, j) \tag{5.7}$$

$$S\_IUF = \frac{1}{|R \cap T|} \times \sum_{i \in R \cap T} \log_2 \frac{|U|}{|U_i|} \tag{5.8}$$

Os valores das métricas apresentados nos resultados dos experimentos são referentes a média de todos os usuários, ou seja, a cada experimento eles foram calculados para cada usuário, somados e depois divididos pelo número total de usuários. Esses valores tem distribuição entre 0 e 1, exceto no caso da IUF, que fica entre 0 e $\log_2 |U|$, onde $|U|$ é igual a 6040 na base de dados dos experimentos (Seção 5.1), portanto, o valor máximo para IUF é aproximadamente $12,56$. Quanto maiores os valores, maior a exatidão, diversidade ou novidade do experimento.

5.4 Metodologia Experimental

Os experimentos foram realizados em uma máquina virtual equipada com 32 GiB de memória e processador "Intel(R) Xeon(R) CPU E5-2690 v2"com 16 núcleos de 3GHz. O código do sistema de recomendação implementado neste trabalho foi escrito na linguagem Python e está disponível em `https://github.com/israel-santanna/semantic-clustering`.

Para otimização dos resultados do algoritmo foram realizados diversos experimentos com configurações diferentes. Primeiramente os experimentos buscaram definir o modelo do *Paragraph Vector* (Seção 3.1.1) a ser utilizado, em seguida os valores dos pesos das entradas, a medida de distância e os valores de corte para o algoritmo de *Density Peaks Clustering* (Seção 3.2.1). Cada experimento foi realizado de forma independente, com o tamanho da base de teste equivalente a 20% do total, os pesos das entradas fixados em 1, o valor de corte do *Density Peaks Clustering* em 0,2 e utilizando o coeficiente de correlação de Pearson como medida de distância, exceto nos experimentos onde foi necessário variar esses valores.

Foram feitas 10 recomendações para cada usuário, sendo que para isso os filmes foram ranqueados de acordo com o valor médio das avaliações recebidas, recomendando os 10 primeiros itens desse ranking.

As seções a seguir apresentam os resultados desses experimentos, sendo que as métricas que apresentam um S na frente são referentes aos valores de *serendipity* das métricas de novidade, e Unexp se refere a métrica *Unexpectedness*. Os melhores resultados estão apresentados em negrito.

5.5 Comparação entre Modelos do *Paragraph Vector*

Para medir o impacto do modelo do *Paragraph Vector* foram comparados os modelos PV-DBOW e PV-DM descritos na Seção 3.1.1, sendo que este último ainda teve comparada a diferença entre a concatenação e a média dos vetores de palavra e parágrafo na hora de prever a próxima palavra do contexto durante o treinamento. Os parâmetros utilizados em todos os modelos foram definidos baseado no código de Mikolov (2014), com vetores de 100 dimensões e janela de contexto contendo 10 palavras. Além disso, os pesos das entradas foram fixados em 1 e o valor de corte do *Density Peaks Clustering* em 0,2.

Analisando os resultados apresentados na Tabela 5.1 podemos observar que o modelo PV-DM com concatenação se sobressaiu entre os demais, apresentando os melhores valores de exatidão, diversidade e novidade.

Tabela 5.1: Comparação entre Modelos do *Paragraph Vector*

Métrica	PV-DBOW	PV-DM Média	PV-DM Concatenação
Precision	0,0217	0,0244	**0,0272**
Recall	0,0124	0,0122	**0,0130**
F_1 Score	0,0127	0,0130	**0,0141**
ILD	0,1246	0,0907	**0,1584**
Unexp	0,2264	0,1418	**0,2920**
IUF	**6,2717**	5,2878	4,8347
S_Unexp	0,0054	0,0041	**0,0084**
S_IUF	0,0441	0,0508	**0,0574**

5.6 Comparação entre Atributos dos Filmes

Definido o modelo, foi então verificado quão bem os atributos extraídos do IMDb conseguem representar os filmes, treinando o PV-DM com apenas a sinopse, apenas as críticas, e finalmente com uma junção da sinopse, críticas e gêneros.

Pelos resultados da Tabela 5.2 é possível verificar que as críticas sozinhas apresentam a melhor exatidão, mas a junção dos atributos apresenta quase a mesma exatidão e maior diversidade e novidade.

Tabela 5.2: Comparação entre Atributos dos Filmes

Métrica	Sinopse	Críticas	Todos
Precision	0,0144	**0,0277**	0,0272
Recall	0,0075	**0,0132**	0,0130
F_1 Score	0,0080	**0,0143**	0,0141
ILD	0,0270	0,1537	**0,1584**
Unexp	0,0476	0,2842	**0,2920**
IUF	**7,3635**	4,7248	4,8347
S_Unexp	0,0005	0,0083	**0,0084**
S_IUF	0,0281	**0,0586**	0,0574

5.7 Avaliação do Impacto do Gênero nos Resultados

Após definido um modelo, o impacto dos pesos nas entradas do *Paragraph Vector* foi avaliado, começando pelo gênero. Como seu conteúdo é menor, os valores testados para o peso foram mais altos que dos outros atributos dos filmes.

Analisando a Tabela 5.3 é possível observar que o aumento no peso de fato afetou positivamente a exatidão do algoritmo, até o peso 50, decaindo no 100. Entretanto, os valores de novidade e diversidade se mostraram melhores nos pesos 1 e 10, pois conforme o peso aumenta, os filmes passam a ter uma quantidade maior de palavras iguais, tornando-se mais parecidos e por consequência perdendo a diversidade.

Tabela 5.3: Avaliação do Impacto do Gênero nos Resultados

Métrica	1	10	25	50	100
Precision	0,0272	0,0302	0,0273	**0,0367**	0,0304
Recall	0,0130	0,0145	0,0154	**0,0186**	0,0177
F_1 Score	0,0141	0,0157	0,0157	**0,0199**	0,0180
ILD	**0,1584**	0,1532	0,1246	0,1175	0,1124
Unexp	**0,2920**	0,2742	0,2067	0,2084	0,1889
IUF	4,8347	3,9058	**5,0265**	3,6863	4,0892
S_Unexp	0,0084	**0,0088**	0,0061	0,0081	0,0063
S_IUF	0,0574	0,0650	0,0617	**0,0830**	0,0699

5.8 Avaliação do Impacto da Sinopse nos Resultados

O próximo peso a ser avaliado foi o da sinopse, tendo seus resultados apresentados na Tabela 5.4, mostrando que apesar da sinopse ser importante, uma vez que quando é retirada os resultados caem, multiplicar seu conteúdo acaba diminuindo a exatidão, mas em contrapartida melhora a novidade e a diversidade. Uma hipótese para este comportamento é que, diferente do gênero, a quantidade de palavras na sinopse é grande e nem todas são importantes para a descrição do filme, logo, ao multiplicá-las o algoritmo do PV vai se focar mais nessas palavras e deixar o vetor do filme muito específico, e portanto, mais distante dos outros filmes. Isso torna mais difícil encontrar agrupamentos, resultando em uma maior quantidade de filmes *outliers* que não serão recomendados, e possibilitando a subida de filmes mais diversos no ranqueamento feito para escolher os 10 filmes que serão recomendados.

Tabela 5.4: Avaliação do Impacto da Sinopse nos Resultados

Métrica	**0**	**1**	**2**	**5**	**10**
Precision	0,0201	**0,0272**	0,0219	0,0205	0,0204
Recall	0,0105	**0,0130**	0,0111	0,0109	0,0109
F_1 Score	0,0111	**0,0141**	0,0118	0,0114	0,0114
ILD	0,1529	0,1584	**0,1645**	0,1623	0,1403
Unexp	0,2731	0,2920	**0,2930**	0,2906	0,2613
IUF	5,8250	4,8347	5,8122	**5,9586**	5,6493
S_Unexp	0,0059	**0,0084**	0,0068	0,0064	0,0057
S_IUF	0,0402	**0,0574**	0,0442	0,0410	0,0408

5.9 Avaliação do Impacto das Críticas nos Resultados

Como mostrado na Tabela 5.5, o aumento do peso das críticas piorou a exatidão do algoritmo enquanto trouxe uma melhora na novidade e diversidade. Este comportamento pode ser explicado da mesma forma que o comportamento do peso da sinopse, descrito na Seção 5.8. No caso das críticas ele é agravado pois elas possuem uma quantidade muito maior de palavras, tornando os vetores dos filmes mais específicos ainda.

Tabela 5.5: Avaliação do Impacto das Críticas nos Resultados

Métrica	**0**	**1**	**2**	**5**	**10**
Precision	0,0169	**0,0272**	0,0177	0,0144	0,0117
Recall	0,0090	**0,0130**	0,0091	0,0075	0,0060
F_1 Score	0,0095	**0,0141**	0,0098	0,0080	0,0063
ILD	0,0137	0,1584	**0,1875**	0,1836	0,1831
Unexp	0,0444	0,2920	0,3504	0,3592	**0,3624**
IUF	6,5569	4,8347	6,5922	7,4267	**8,2341**
S_Unexp	0,0011	**0,0084**	0,0067	0,0056	0,0047
S_IUF	0,0324	**0,0574**	0,0349	0,0283	0,0229

5.10 Avaliação do Impacto da Medida de Distância nos Resultados

No algoritmo proposto existem alguns passos onde é necessário calcular a distância entre os itens. Para avaliar o impacto que a medida de distância tem nos resultados, a medida de similaridade de cossenos foi comparada com o coeficiente de correlação de Pearson, pois são duas medidas que se comportam bem em espaços de alta dimensão. A primeira calcula o cosseno do ângulo entre dois vetores, indicando se são ortogonais, opostos ou idênticos (Singhal, 2001), enquanto a segunda indica o grau de intensidade da correlação linear entre duas variáveis e o sentido, positivo ou negativo, dessa correlação (Rocha, 2014).

Analisando os resultados da Tabela 5.6 é possível perceber que o coeficiente de correlação de Pearson se mostrou superior na maioria das métricas.

Tabela 5.6: Avaliação do Impacto da Medida de Distância nos Resultados

Métrica	Cosseno	Pearson
Precision	0,0245	**0,0272**
Recall	0,0118	**0,0130**
F_1 Score	0,0127	**0,0141**
ILD	**0,1727**	0,1584
Unexp	0,2912	**0,2920**
IUF	**5,2428**	4,8347
S_Unexp	0,0075	**0,0084**
S_IUF	0,0513	**0,0574**

5.11 Avaliação do Impacto dos Valores de Densidade e Distância nos Resultados

Como mencionado na Seção 5.2, para a execução do algoritmo *Density Peaks Clustering* é necessário a definição de um valor de corte para a densidade local e a distância dos itens que serão considerados centros dos grupos. A Tabela 5.7 mostra os resultados das variações desses valores, sendo que o valor de corte no cabeçalho das colunas foi atribuído tanto a densidade quanto a distância em cada experimento.

Os resultados indicam uma melhora na exatidão conforme o valor de corte aumenta, atingindo o ápice em 0,5, enquanto que os valores de novidade e diversidade apresentam melhores resultados em cortes mais baixos. Isso acontece pois quanto menor o valor de corte, mais agrupamentos são encontrados e consequentemente mais intercessões acontecem e filmes mais diferenciados são recomendados.

Tabela 5.7: Avaliação do Impacto dos Valores de Densidade e Distância nos Resultados

Métrica	0,1	0,2	0,3	0,4	0,5	0,6
Precision	0,014768	0,027201	0,032268	0,036092	**0,040480**	0,040463
Recall	0,007565	0,013051	0,015407	0,016429	**0,017864**	0,017862
F_1 Score	0,008073	0,014122	0,016725	0,018077	**0,019915**	0,019912
ILD	**0,165652**	0,158424	0,165466	0,157864	0,156443	0,156442
Unexp	0,290205	0,292086	**0,297710**	0,293317	0,292535	0,292533
IUF	**7,107985**	4,834744	3,662071	3,606755	3,440297	3,440405
S_Unexp	0,004591	0,008454	0,010078	0,011157	**0,012480**	0,012475
S_IUF	0,029425	0,057481	0,068371	0,074850	**0,082735**	0,082699

Capítulo 6

Conclusão

Este trabalho apresentou uma nova abordagem para Sistemas de Recomendação, encontrando usuários com gostos similares através de interseções de regiões de interesse descobertas por meio do Agrupamento de Dados em um Espaço Semântico criado a partir de descrições textuais dos itens. Um algoritmo foi implementado para a realização de experimentos, utilizando os algoritmos *Paragraph Vector* e *Density Peaks Clustering* e a base de dados MovieLens 1M.

Diferentes parâmetros do algoritmo implementado, como pesos das entradas, medidas de distância e valores de corte para o algoritmo de agrupamento foram testados, buscando otimizar os resultados dos experimentos em relação as diversas métricas utilizadas para analisar a exatidão, diversidade e novidade das recomendações.

A partir da análise dos resultados verificou-se que o sistema proposto de fato produz recomendações diversas e não triviais, recomendando itens diferentes dos já avaliados pelo usuário e também itens que não foram avaliados por muitos usuários e portanto dificilmente apareceriam em algoritmos de recomendação convencionais. Entretanto, essa novidade e diversidade vem em custa da exatidão, sendo necessária uma concessão entre as diferentes métricas para atingir bons resultados em todos os aspectos.

Em trabalhos futuros cabe uma análise da importância da data das avaliações de um usuário, pois os gostos dos usuários tendem a mudar com o tempo, logo, suas regiões de interesse também devem se deslocar pelo espaço semântico. Além disso, outra análise interessante a ser feita é em relação ao formato das regiões de interesse, buscando trazê-lo o mais próximo possível do formato do agrupamento encontrado, ao invés do formato hiperesférico utilizado neste trabalho.

Referências Bibliográficas

Aggarwal, C. C. (2016a). *Recommender Systems: The Textbook*, página 1. Springer International Publishing.

Aggarwal, C. C. (2016b). *Recommender Systems: The Textbook*, página 8. Springer International Publishing.

Aggarwal, C. C. (2016c). *Recommender Systems: The Textbook*, página 24. Springer International Publishing.

Aggarwal, C. C. (2016d). *Recommender Systems: The Textbook*, páginas 14–15. Springer International Publishing.

Aggarwal, C. C. e Reddy, C. K. (2013a). *Data Clustering: Algorithms and Applications*, página 1. Chapman and Hall/CRC.

Aggarwal, C. C. e Reddy, C. K. (2013b). *Data Clustering: Algorithms and Applications*, página 32. Chapman and Hall/CRC.

Firth, J. R. (1957). *A Synopsis of Linguistic Theory*.

GroupLens (2003). Movielens 1m dataset. `http://grouplens.org/datasets/movielens/1m/`. Acessado em 28/11/2017.

IMDb (1990). Imdb - movies, tv and celebrities. `http://www.imdb.com/`. Acessado em 28/11/2017.

Koren, Y. (2010). Collaborative filtering with temporal dynamics. *Communications of the ACM*, 53(4):89–97.

Le, Q. V. e Mikolov, T. (2014). Distributed representations of sentences and documents. *CoRR*, abs/1405.4053.

Lowe, W. (2001). Towards a theory of semantic space. Em *Proceedings of the Twenty-Third Annual Conference of the Cognitive Science Society*, página 576–581.

MacKay, D. J. C. (2005). *Information Theory, Inference, and Learning Algorithms*, página 284. Cambridge University Press.

Mikolov, T. (2014). "distributed representations of sentences and documents"code? `https://groups.google.com/forum/#!msg/word2vec-toolkit/Q49FIrNOQRo/J6KG8mUj45sJ`. Acessado em 28/11/2017.

O'Dwyer, R. (2013). Imdbpie. `https://github.com/richardasaurus/imdb-pie`. Acessado em 28/11/2017.

Ricci, F., Rokach, L., Shapira, B. e Kantor, P. B. (2015a). *Recommender Systems Handbook*, página 2. Springer US.

Ricci, F., Rokach, L., Shapira, B. e Kantor, P. B. (2015b). *Recommender Systems Handbook*, página 1. Springer US.

Ricci, F., Rokach, L., Shapira, B. e Kantor, P. B. (2015c). *Recommender Systems Handbook*, página 10. Springer US.

Ricci, F., Rokach, L., Shapira, B. e Kantor, P. B. (2015d). *Recommender Systems Handbook*, página 283. Springer US.

Ricci, F., Rokach, L., Shapira, B. e Kantor, P. B. (2015e). *Recommender Systems Handbook*, páginas 888–889. Springer US.

Ricci, F., Rokach, L., Shapira, B. e Kantor, P. B. (2015f). *Recommender Systems Handbook*, páginas 892–893. Springer US.

Rocha, A. V. (2014). *Probabilidade e Estatística*, página 119. Editora da UFPB.

Rodriguez, A. e Laio, A. (2014). Clustering by fast search and find of density peaks. *Science*, 344(6191):1492–1496.

Singhal, A. (2001). Modern information retrieval: A brief overview. 24:35–43.

Weiszflog, W. (2017). Michaelis dicionário brasileiro da língua portuguesa. `http://michaelis.uol.com.br/moderno-portugues/busca/portugues-brasileiro/sem%C3%A2ntica/`. Acessado em 28/11/2017.