Universidade Federal de Ouro Preto - UFOP

Instituto de Ciências Exatas e Biológicas - ICEB

Departamento de Computação - DECOM

# ADICIONANDO ESCALABILIDADE AO *FRAMEWORK* DE RECOMENDAÇÃO *IRF*

Aluno: Alex Amorim Dutra
Matricula: 07.1.4149

Orientador: Álvaro Rodrigues Pereira Júnior
Co-Orientador: Felipe Santiago Martins Coimbra de Melo

Ouro Preto
15 de setembro de 2011

Universidade Federal de Ouro Preto - UFOP

Instituto de Ciências Exatas e Biológicas - ICEB

Departamento de Computação - DECOM

# ADICIONANDO ESCALABILIDADE AO *FRAMEWORK* DE RECOMENDAÇÃO *IRF*

Proposta de monografia apresentada ao curso de Bacharelado em Ciência da Computação, Universidade Federal de Ouro Preto, como requisito parcial para a conclusão da disciplina Monografia II (BCC391).

Aluno: Alex Amorim Dutra
Matricula: 07.1.4149

Orientador: Álvaro Rodrigues Pereira Júnior
Co-Orientador: Felipe Santiago Martins Coimbra de Melo

Ouro Preto
15 de setembro de 2011

# Resumo

*Palavras-chave:* Escalabilidade. Sistemas de Recomendação. *Idealize Recommendation Framework (IRF). Hadoop. HBase*

Desde a antiguidade o homem faz recomendações às outras pessoas com as quais se relaciona. Na *Web*, sistemas de recomendação têm a cada dia deixado de ser uma novidade e se tornado uma necessidade para os usuários, devido ao grande volume de dados disponíveis. Estes dados tendem a crescer cada vez mais, o que poderá ocasionar uma perda de tempo considerável pelo usuário ao realizar buscas manualmente para encontrar conteúdos relevantes na Web. Sistemas de recomendação têm a finalidade de levar conteúdo relevante a seus utilizadores de forma personalizada. Para isto são utilizados algoritmos de aprendizagem de máquina e outras técnicas de recomendação, tais como clusterização de usuários, semelhança entre itens. Para atuarem de maneira eficiente, os algoritmos de recomendação precisam manipular grandes volumes de dados, o que torna necessário tanto o armazenamento quanto o processamento destes dados de maneira distribuída. Ainda, é desejado que a distribuição tanto do armazenamento quanto do processamento sejam escaláveis, ou seja, é desejado que mais capacidade de armazenamento e processamento possam ser acrescentados à medida que forem necessários. Como trabalho de conclusão de curso tratarei da escalabilidade no *Framework de Recomendação Idealize (IRF)* . Para tornar o *IRF* escalável utilizarei o *framework Hadoop*, pelo fato de ser *open-source* e estar presente em grandes sistemas que utilizam computação distribuída e escalável.

# Sumário

# Lista de Figuras



# Lista de Tabelas

# 1 Introdução

Com o crescimento da produção de dados, principalmente na Web [5], temos ao alcance informações relevantes em diversas áreas. Algumas vezes quando estamos realizando buscas sobre determinado assunto, produto, ou qualquer outro item acabamos não encontrando o que desejamos, devido à grande quantidade de dados existentes e a dificuldade de realização de buscas manuais sobre estes dados. Sistemas de recomendação têm a finalidade de levar ao usuário o que realmente é relevante para ele. O *Framework de Recomendação Idealize (IRF)* foi desenvolvido para suportar qualquer estratégia de recomendação [6]. As aplicações de recomendação desenvolvidas sobre o *IRF* até o momento possuem as seguintes abordagens: baseada em conteúdo [3], filtragem colaborativa [6], dados de uso [1] e híbrida [1]. Em resumo recomendações baseadas em conteúdo é realizada com base na descrição dos itens mais similares ao item sendo acessado, ou recomendam itens que possuem características similares as definidas no perfil do usuário [1]. Recomendações por filtragem colaborativa tem sua origem na mineração de dados [2] e constituem o processo de filtragem ou avaliação dos itens através de múltiplos usuários [1, 7, 8], muitas vezes formando grupos de usuários que possuem características similares. Recomendações baseadas em dados de uso, levam em consideração as ações realizadas por seus usuários [4], por exemplo, a sequência de links clicados por um usuário quando navega em um site de compras. A recomendação híbrida é interessante, pois possibilita que as limitações de cada técnica sejam supridas por características das demais [1]. Inicialmente a abordagem de recomendação escolhida para se tornar distribuída é a recomendação por filtragem colaborativa. Assim sendo, implementarei módulos e classes de forma a tornar esta abordagem de recomendação distribuída e escalável, sendo capaz de processar grandes volumes de dados.

# 2 Justificativa

Além da importância de sistemas de recomendação por levarem o que realmente interessa aos utilizadores, existem outros fatores que justificam a criação de um *framework* e a realização de armazenamento e processamento distribuído e escalável.

A importância de um *framework* deve se à necessidade de prover uma solução para uma família de problemas semelhantes, usando um conjunto em geral de classes abstratas e interfaces que mostram como decompor a família de problemas, e como objetos dessas classes colaboram para cumprir suas responsabilidades. O conjunto de classes deve ser flexível e extensível para permitir a construção de aplicações diferentes dentro do mesmo domínio mais rapidamente, sendo necessário implementar apenas as particularidades de cada aplicação. Em um *framework*, as classes extensíveis são chamadas de *hot spots* [6]. O importante é que exista um modelo a ser seguido para a criação de novas aplicações de recomendação, e definir a interface de comunicação entre os *hot spots* desse modelo. As classes que definem a comunicação entre os *hot spots* não são extensíveis e são chamadas de *frozen spots* [6], pois constituem as decisões de *design* já tomadas dentro do domínio ao qual o *framework* se aplica.

A importância da escalabilidade está relacionada ao grande volume de dados que devem ser processados. Grandes empresas como *Facebook, Yahoo!, Google, Twitter, Amazon* entre outras armazenam volumes de dados da ordem de petabytes[1], de onde podem ser extraídas informações relevantes. Assim, a escalabilidade tem a função principal de distribuir os componentes e serviços de forma a aumentar o desempenho, diminuindo o tempo de processamento das recomendações à medida que mais recursos (ex.: *hardwares*) são acrescentados. Na maior parte dos casos uma única máquina não é capaz de armazenar e manipular grandes volumes de dados, logo, faz-se necessário a utilização da computação distribuída. A medida que se deseja aumentar a capacidade processamento e armazenamento, têm-se a necessidade de utilizar a computação escalável.

[1] http://escalabilidade.com/2010/05/18/

# 3 Objetivos

## 3.1 Objetivo geral

- O objetivo ao final da disciplina Monografia II (BCC391) é apresentar o *Framework de Recomendação Idealize* juntamente com os componentes que facilitam a construção de aplicações distribuídas de recomendação. A figura 1 ilustra a distribuição física do sistema em seu modelo de produção e como será a arquitetura ao final do curso. Os setores de *Input*, *Batch* e *Cache*, o local de armazenamento de dados e o *cluster* (responsável pelo armazenamento e processamento distribuído) serão detalhados no relatório final.

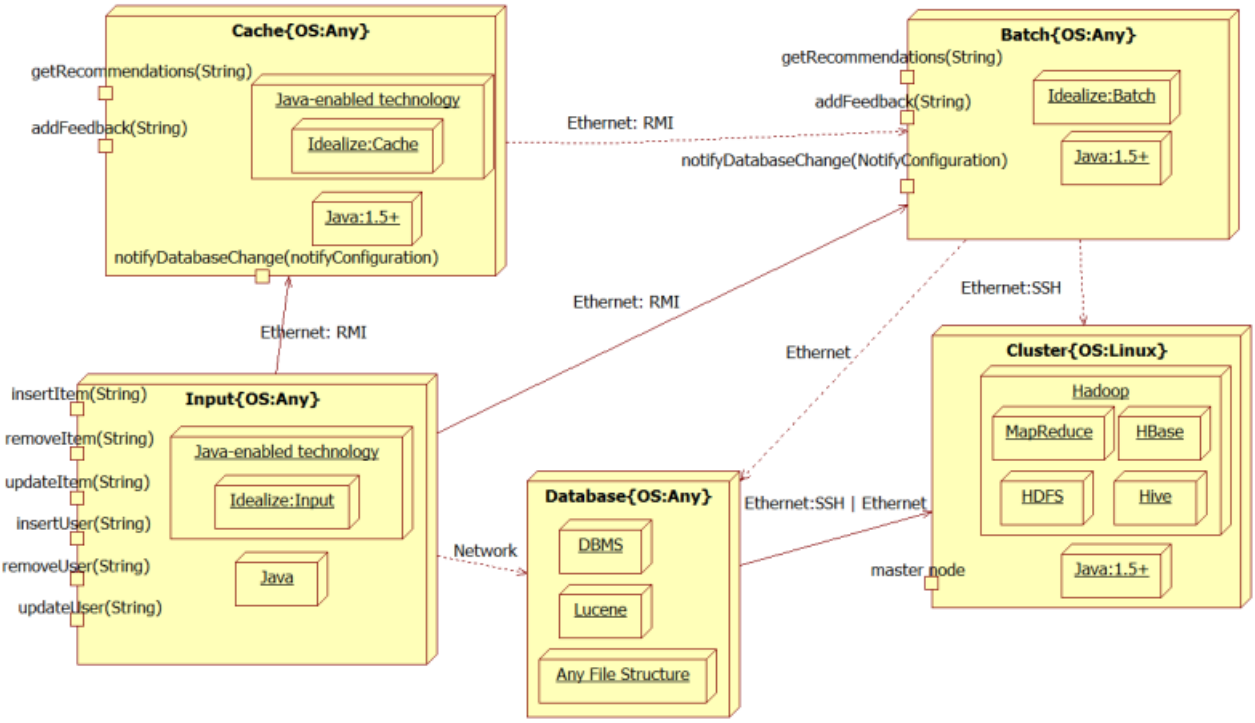


Figura 1: Arquitetura final

## 3.2 Objetivos específicos

- O primeiro objetivo é desenvolver uma aplicação de recomendação baseada em filtragem colaborativa utilizando o *IRF*, de forma que as recomendações possam ser processadas de forma distribuída. Para tal tarefa é necessária a construção de um *cluster*[2] e o estudo de tecnologias tais como *Hadoop* e *HBase*.

- O segundo objetivo é agrupar componentes que sejam comuns aos diversos métodos de recomendação distribuídos, e a partir daí derivar classes que se tornarão *hot spots* do *IRF*.

[2]Um cluster é formado por um conjunto de computadores, que realizam processamento em paralelo e distribuído.

- O terceiro objetivo é a realização de testes, análise dos resultados e melhorias nas implementações da aplicação de recomendação distribuída.

# 4 Metodologia

A metodologia aqui descrita abrange o que será apresentado na disciplina Monografia II (BCC391). Este trabalho é de caráter exploratório, onde deseja-se melhorar o poder de processamento de grandes volumes de dados em sistemas de recomendação. O trabalho será dividido em três fases.

A primeira fase consiste na implementação de métodos de recomendação distribuídos. Os métodos de recomendação serão escolhidos de acordo com os melhores resultados de acurácia obtidos em suas implementações sequenciais.

A segunda fase será a derivação de classes de modo que estas se tornem *hot spots*. Desta forma os novos métodos e aplicações distribuídas terão um modelo a ser seguido, facilitando a criação de novas aplicações.

E por fim a terceira fase será a realização dos experimentos e análise dos resultados encontrados. Os experimentos serão realizados levando em consideração a quantidade de máquinas do *cluster*, variando a quantidade de máquinas para processamento.

# 5 Cronograma de atividades

Na Tabela 1, segue o cronograma das atividades a serem realizadas.

| **Atividades** | **Ago** | **Set** | **Out** | **Nov** | **Dez** |
|---|---|---|---|---|---|
| Estudo de escalabilidade | X | | | | |
| Implementações dos módulos distribuídos | X | X | X | X | |
| Testes e redigir a monografia | | | X | X | X |
| Apresentação do trabalho realizado | | | | | X |

Tabela 1: Cronograma de Atividades.

# Referências


[1] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Eng.*, 17:734–749, June 2005.

[2] Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1999.

[3] Marko Balabanovic' and Yoav Shoham. Content-based, collaborative recommendation. *Communications of the ACM*, 40:66–72, 1997.

[4] Carlos Castillo, Debora Donato, Ranieri Baraglia, Franco Maria Nardini, Raffaele Perego, and Fabrizio Silvestri. Aging effects on query flow graphs for query suggestion.

[5] John F. Gantz, Christopher Chute, Alex Manfrediz, Stephen Minton, David Reinsel, Wolfgang Schlichting, and Anna Toncheva. The diverse and exploding digital universe, 2008.

[6] Felipe Martins Melo and Álvaro R. Pereira Jr. Idealize recommendation framework - an open-source framework for general-purpose recommender systems. In *14th international ACM Sigsoft symposium on Component based software engineering*, pages 67–72, June 2011.

[7] J Ben Schafer, Dan Frankowski, Jon Herlocker, and Shilad Sen. Collaborative filtering recommender systems. In *The adaptive web: methods and strategies of web personalization*, volume 4321, pages 291–324. Springer, 2007.

[8] Jun Wang, Arjen P. De Vries, and Marcel J. T. Reinders. Unifying user-based and item-based collaborative filtering approaches by similarity fusion. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 501–508. ACM Press, 2006.