Desenvolvimento de um plugin para navegador, utilizando técnicas de filtragem colaborativa, para realizar recomendações no repositório Merlot

Gustavo Magalhães

Luan Einhardt

Marcelo Fay

### Sumário

- Filtragem Colaborativa
- Objetivo
- Recomendação baseada em usuário
- Busca de Similaridade
- Dificuldades e desafios da Filtragem Colaborativa
- Avaliação do recomendador
- Material e métodos

### Sumário

- Criação de um recomendador
- Cálculo de erros
- Criação de recomendação
- PHP, MySQL e Plugin
- Geração do arquivo de classificação
- Repositório Merlot com e sem o plugin
- Conclusões

## Filtragem Colaborativa

- Predições
- Colaboração
- Itens
- Histórico de avaliação dos itens
- Utilizado principalmente em catálogos online

### Filtragem Colaborativa

- Usuário que concordou no passado, concordará no futuro.
- Calcula similaridades entre usuários
- Retorna vizinhança do usuário ativo
- Para cada item não avaliado pelo usuário ativo, mas avaliado pelos vizinhos, calcula-se a preferência estimada do usuário ativo.

# Fluxo da Filtragem Colaborativa

- Classificação de um item por um usuário
- Exploração da base de dados em busca de pessoas com gostos similares
- Recomendação

## Objetivo

• Desenvolvimento de um plugin externo para o repositório Merlot cuja finalidade é fornecer recomendações gerais ou específicas para o usuário ativo.

## Recomendação baseada em usuário

- Baseada em dois passos:
  - Busca de usuários similares
  - Predição

### Busca de similaridade

- K-vizinhos mais próximos
- Correlação de Pearson Intervalo [-1,1]

### Dificuldades e desafios da Filtragem Colaborativa

- Início em frio
- Esparsidão da tabela usuário-item
- Escalabilidade e Clusterização
- Shilling Attacks
- Usuário ovelha negra
- Super-especialização

### Avaliação do recomendador

- Erro absoluto médio
- · Raíz do erro quadrático médio

• 0 = Recomendador perfeito para o data-set corrente.

### Material e métodos

- Base de dados previamente obtida
- Repositório Merlot
- Materiais e membros
- Java + Apache Mahout
- PHP e MySQL

### Criação de um recomendador

Código 1.1. Leitura da entrada e construção do recomendador

#### Cálculo de erros

```
// Avaliacao de erro para o recomendador
 / MAE = Erro absoluto medio
// RMS = Raiz do erro quadratico medio
   if (error) {
       RecommenderEvaluator evaluatorMAE = new
           AverageAbsoluteDifferenceRecommenderEvaluator();
       RecommenderEvaluator evaluatorRMS = new
           RMSRecommenderEvaluator();
       double scoreMAE = 0, scoreRMS = 0;
       int num = 50;
       for (int i = 0; i < num; i++) {
           scoreMAE += evaluatorMAE.evaluate(builder, null,
               model, 0.8, 1.0);
           scoreRMS += evaluatorRMS.evaluate(builder, null,
               model, 0.8, 1.0);
       scoreMAE /= num;
       scoreRMS /= num;
       result.put("mae", scoreMAE);
       result.put("rms", scoreRMS);
```

Código 1.2. Avaliacao do recomendador

### Criação da recomendação

```
//Gera as recomendacoes para o usuario
List<RecommendedItem> list = builder.buildRecommender(
    model).recommend(iduser, 10);
JSONArray objects = new JSONArray();
for(RecommendedItem rec : list){
    JSONObject object = new JSONObject();
    object.put("idobject", rec.getItemID());
    object.put("value", rec.getValue());
    objects.put(object);
}
result.put("objects", objects);
System.out.println(result);
```

Código 1.3. Saida do recomendador java

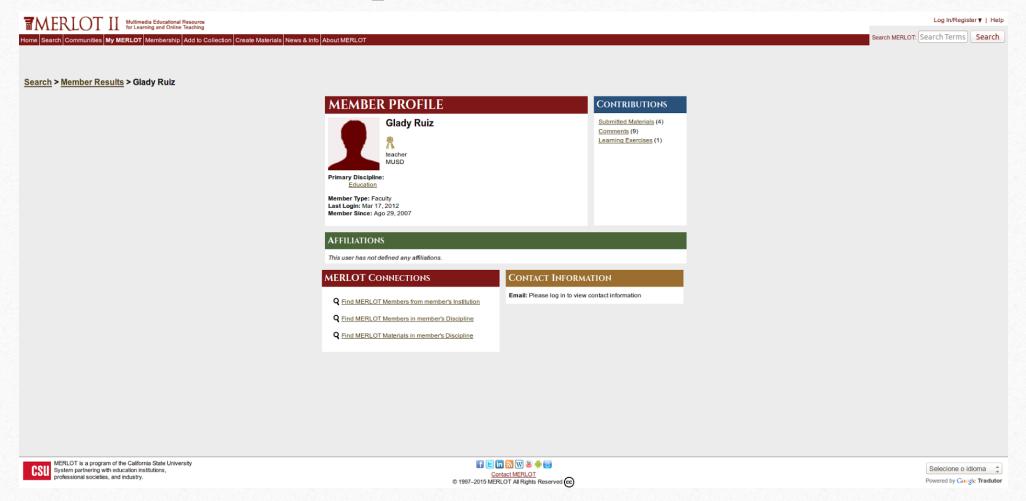
# PHP, MySQL e Plugin

- Chrome e Firefox
  - Manifest.json no Chrome, package.json no Firefox (além do add-on SDK)
    - Permissões e javaScript
- Extração de ID do usuário na URL
- Servidor aguarda requisições
- Servidor recebe tipo de recomendação e ID do usuário
- Servidor retorna erro ou os dados da recomendação
- Montagem de duas colunas para recomendação (JS + CSS)

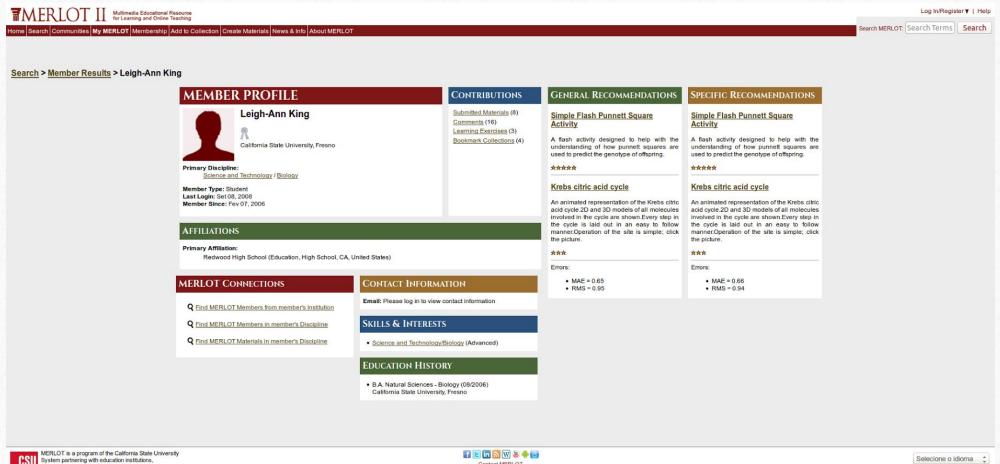
## Geração do arquivo de classificação

Código 1.4. Geracao de arquivo de classificacao

## Repositório Merlot



### Repositório Merlot + Plugin



rofessional societies, and industry.

Contact MERLOT
© 1997–2015 MERLOT All Rights Reserved (cc)

Powered by Google Tradutor

#### Conclusões

- Filtragem colaborativa requer grande número de usuários ativos para responder de forma adequada para cada usuário do repositório.
- É possível, além de simples relativamente, o desenvolvimento e implantação de sistemas de filtragem colaborativa através de plugin externo.

Desenvolvimento de um plugin para navegador, utilizando técnicas de filtragem colaborativa, para realizar recomendações no repositório Merlot

Gustavo Magalhães

Luan Einhardt

Marcelo Fay