

Introdução

Transcrição

Olá pessoal, neste curso vamos estudar **Machine Learning** para recomendações, ou seja, construir sistemas de recomendação para o usuário. Estes sistemas podem recomendar qualquer coisa desde comida, livros, produtos até lugares, etc.

Por exemplo, você precisa estudar para a prova do ENEM, que exercícios podem ser recomendados para você fazer? O aluno vai tirar uma certificação? Que áreas podem ser estudadas para realizar a prova? Vai sair com os familiares para algum restaurante e deseja comer algo? Que restaurante e pratos podemos recomendar para que todos comam algo que gostem?

Estes sistemas ajudam os usuários dentro de um determinado tema, pois eles têm disponíveis para si centenas e milhares de opções. Dentre elas ajudamos o usuário a não perder tempo, pois recomendamos um produto, item, curso, restaurante ou qualquer outra coisa.

O que você vai fazer no próximo fim de semana a noite? Ver um filme? Como fará a escolha desse filme? Existem sistemas como a [Netflix](http://www.netflix.com.br) (<http://www.netflix.com.br>) e o [MovieLens](https://movielens.org) (<https://movielens.org>) que recomendam filmes baseados em gostos pessoais. É isso que iremos aprender neste curso: criar recomendadores baseados em Machine Learning.

Neste curso utilizaremos o Java como linguagem e por isso saber o básico dessa linguagem é essencial. Existem também outros cursos de Machine Learning que utilizam **Python** como linguagem, mas estes são cursos de [introdução a classificação] <https://cursos.alura.com.br/course/introducao-a-machine-learning-com-classificacao> (<https://cursos.alura.com.br/course/introducao-a-machine-learning-com-classificacao>) e [Avançando com tipos diferentes de classificação](https://cursos.alura.com.br/course/machine-learning-classificacao-ii) (<https://cursos.alura.com.br/course/machine-learning-classificacao-ii>) que ajudam a classificar itens, por exemplo: se um e-mail é ou não um spam. Isso é uma classificação.

Este curso focará nas recomendações e utilizará um algoritmo conhecido como **User Based Similarity**, ou seja, baseado nas similaridade dos usuários. Nós utilizaremos inicialmente um conjunto de dados irreais e depois utilizaremos dados concretos e com isso testaremos o algoritmo, otimizaremos o código e também validaremos o quão bom está nosso algoritmo. Tudo isso utilizando uma biblioteca Java chamada Mahout.

Vamos começar?