



Rafael Rodrigues Marquesi <rafael.4marcos.nt@gmail.com>

#7DaysOfCode - Ciência de Dados 4/7: 🧑🏻 Sistemas de recomendação

Paulo Vasconcellos <paulo.vasconcellos@7daysofcode.io>

Mon, Apr 25, 2022 at 7:04 AM

Reply-To: reply-caelum57945.activehosted.1376.8451.1043955@d32a.emsend3.com

To: Rafael Rodrigues Marquesi <rafael.4marcos.nt@gmail.com>



Olá, Rafael Rodrigues Marquesi.

Olha só, você está na **metade do #7DaysOfCode** de Dados!
Espero que esteja aproveitando muito e desenvolvendo novas habilidades.

Hoje vou começar falando sobre **Machine Learning**.

Este conceito tem sido utilizado para resolver diversos problemas em todos os setores da indústria.

Se você já usou o Google Maps, Spotify, Youtube, ou Amazon, é certo que já esbarrou em algum sistema utilizando aprendizado de máquina para **melhorar sua experiência e criar novos produtos**.

Uma das aplicações que mais interessam empresas de todos os setores são os sistemas de recomendação:
aplicações que aprendem, a partir de dados históricos, como sugerir os melhores itens para clientes a fim de maximizar alguma métrica de negócio, como faturamento, taxa de conversão, dentre outras.

E é exatamente sobre isso que eu vou falar no desafio de hoje.

O MovieLens é um dataset clássico usado em problemas de sistema de recomendação. Ele é usado até em artigos científicos para validar novos tipos de algoritmos de recomendação.

Eu gosto desse conjunto de dados porque ele está muito próximo do mercado.

Afinal, quantas empresas não se beneficiariam de um sistema de recomendações?

Minha proposta para este desafio é que você crie um sistema de recomendações a partir do MovieLens.

Pense em como uma empresa como a Netflix utilizaria um sistema desse jeito. **Você consegue criar um sistema que recomenda 5 filmes** para o usuário com base em comportamentos passados?

DICA

Embora Machine Learning seja uma abordagem que você pode usar para resolver este problema, **existem versões mais simples e funcionais de um sistema de recomendação** que você pode usar para criar, testar, e iterar com soluções mais complexas.

Existem desde abordagens aleatórias (isso mesmo, recomendar itens de forma randômica, sem nenhum outro critério), de popularidade (pegar os filmes mais famosos e recomendar para usuários), e até baseadas em grafos.

Será que elas funcionam bem neste caso?

Mesmo que criar um sistema de recomendação demore mais de um dia no mundo real, esse exercício tem o objetivo de te motivar a conhecer mais sobre esse amplo campo.

Aproveite este momento para conhecer as diferentes abordagens de sistemas de recomendação e suas vantagens e desvantagens, como filtragem colaborativa, filtragem baseada em conteúdo e aprendizado por reforço.

Não esqueça de serializar seu modelo ao fim do processo, isso é, gerar um arquivo com seu modelo salvo para que possa ser carregado depois.

Duas bibliotecas muito famosas no Python para isso são a Joblib e a Pickle. Para salvar seu modelo usando o Joblib, por exemplo, é só adicionar a linha de código `joblib.dump(variavel_do_modelo, nome_do_arquivo.sav)`. Esse modelo vai ser útil para as próximas etapas.

EXTRA

Esse post no blog do Data Hackers mostra uma aplicação de sistemas de recomendação usando Python, que você pode se basear para esse projeto.

Já esse post do Analytics Vidhya traz uma visão completa sobre as abordagens mais comuns de sistemas de recomendação, suas vantagens e desvantagens.

Uma das abordagens que eu mencionei é criar um recomendador baseado em popularidade.

Enquanto essa abordagem funciona, é muito importante você entender um problema que ela pode trazer: **o viés da popularidade**, que basicamente ocorre quando recomendamos filmes muito populares, enquanto esquecemos de filmes não tão populares, mas que podem ser ótimas recomendações também.

Para entender mais sobre esse efeito, [recomendo a leitura desse paper](#) que mostra um estudo feito sobre o impacto que esse tipo de viés pode trazer para os negócios.

Quando eu preciso criar um "recomendador", **uma abordagem que adoto é começar da forma mais simples para a mais complexa**. Então, testar abordagens aleatórias e de popularidade pode ser uma boa forma de começar antes de passar para modelos mais complexos.

O importante é ter em mente uma métrica para comparar a performance dos modelos.

É hora de colocar a mão na massa.

Até amanhã!

Paulo Vasconcellos

Cientista de Dados e Alura Star

alura

Enviado para: rafael.4marcos.nt@gmail.com

[Não quero mais receber os Desafios](#)

Alura, Rua Vergueiro 3185, São Paulo - SP, 04101-300, Brasil

