

FATEC Rubens Lara

# **Similaridade do Cosseno para Recomendação de Presentes no Stardew Valley**

Guilherme de Souza Teodoro

2º Ciclo de Ciência de Dados

# 1. Introdução

Este trabalho foi desenvolvido com o objetivo de aplicar conceitos de Álgebra Linear e Processamento de Linguagem Natural (PLN) em um projeto simples, porém prático e divertido. A ideia foi criar um sistema que recomenda personagens do jogo Stardew Valley com base em um presente que o jogador deseja oferecer.

Ao invés de o usuário buscar diretamente pelo nome do personagem, o sistema recebe como entrada o nome de um presente (como “Daffodil” ou “Pufferfish”) e, a partir disso, retorna quais personagens amam, gostam ou odeiam esse item. A resposta também traz outras informações relevantes, como o aniversário do personagem, local onde vive, com quem é relacionado e se é possível se casar com ele.

A escolha desse projeto foi pensada como uma forma de aplicar na prática os conhecimentos que temos adquirido em Álgebra Linear e PLN dentro do curso de Ciência de Dados. Mesmo sendo um exemplo descontraído e baseado em um jogo, ele mostra como conceitos matemáticos e computacionais podem ser combinados para resolver problemas reais de recomendação e análise semântica.

## 2. Objetivo

O principal objetivo deste trabalho é desenvolver um sistema capaz de identificar, com base em um presente escolhido pelo usuário, quais personagens do jogo *Stardew Valley* reagem positivamente ou negativamente a esse item, utilizando técnicas de Processamento de Linguagem Natural (PLN) e conceitos de Álgebra Linear, como a similaridade do cosseno.

A ideia é que, ao digitar o nome de um presente (como ele aparece no jogo, em inglês), o sistema consiga analisar o grau de afinidade desse presente com os gostos de cada personagem, classificando a reação como: “ama”, “gosta”, “neutro”, “não gosta” ou “odeia”. Além disso, o sistema também mostra informações complementares do personagem, como aniversário, local onde mora, vínculos familiares e possibilidade de casamento.

Com isso, o trabalho pretende demonstrar na prática como a vetorização de textos e o cálculo de similaridade podem ser usados para construir um sistema inteligente de recomendação. O projeto também busca reforçar o entendimento de como PLN e Álgebra Linear se conectam dentro do contexto da Ciência de Dados.

### **3. Similaridade do Cosseno**

A similaridade do cosseno é uma técnica matemática muito usada em Processamento de Linguagem Natural (PLN) para medir o quanto dois textos (ou conjuntos de palavras) são parecidos entre si.

No projeto, usamos a similaridade do cosseno para comparar o presente digitado pelo usuário com os gostos de cada personagem do jogo. Cada personagem teve seus presentes convertidos em um vetor com a técnica de TF-IDF, que leva em consideração a frequência e a importância de cada presente dentro do conjunto de dados.

Depois, o nome do presente que o usuário digita também é convertido em vetor. Em seguida, usamos a função *cosine\_similarity*, da biblioteca *Scikit-Learn*, para comparar esse vetor com os vetores de cada personagem. O resultado é uma pontuação que mostra o quão próximo aquele personagem está do presente dado.

## 4. Estrutura

O arquivo `characters.csv` inclui colunas como:

- **Name:** nome do personagem
- **Birthday Day e Birthday Season:** data e estação do aniversário
- **Lives In e Address:** cidade e local onde vive
- **Marriage:** indica se é possível se casar com o personagem
- **Family:** lista com membros da família
- **Loved Gifts, Liked Gifts, Neutral Gifts, Disliked Gifts, Hated Gifts:** listas com os presentes que o personagem ama, gosta, é neutro, desgosta ou odeia

Cada personagem pode ter múltiplos presentes associados a essas categorias.

## 5. Funcionamento do Algoritmo

- **Leitura e limpeza dos dados:**  
O código carrega o arquivo characters.csv e transforma as colunas de listas (como Loved Gifts) em listas Python reais. Também normaliza os textos (sem acento, letras minúsculas).
- **Unificação dos presentes:**  
Todos os presentes (independente da categoria) são

reunidos em uma única string por personagem, usada para comparação.

- **Vetorização com TF-IDF:**

Cada conjunto de presentes por personagem é transformado em vetor com o *TfidfVectorizer*.

- **Entrada do usuário:**

O presente digitado também é convertido em vetor.

- **Cálculo da similaridade do cosseno:**

Com os vetores prontos, o sistema calcula o quanto o presente se aproxima dos gostos de cada personagem.

- **Classificação por reação:**

Com base na presença do presente nas categorias (loved, liked etc.), o personagem recebe um rótulo: **Ama**, **Gosta**, **Neutro**, **Não gosta** ou **Odeia**.

- **Exibição do resultado:**

O sistema mostra os personagens que reagiriam ao presente, com informações como nome, aniversário, local onde vive, família e se é casável.

## 6. Resultados

O sistema funcionou corretamente, retornando personagens de forma coerente com o universo do jogo. Por exemplo:

- Ao digitar “**Pumpkin**”, o sistema retornou **Abigail** e **Krobus** como personagens que **amam** esse presente.
- Para “**Daffodil**”, o resultado foi **Sandy** (ama), e outros personagens que gostam ou aceitam o presente.
- Presentes não encontrados foram tratados com mensagens apropriadas, evitando erros.

Além da classificação por reação, o sistema também calcula uma pontuação de afinidade (de 0 a 1) com base na similaridade do cosseno, ordenando os personagens do mais compatível para o menos compatível.