# Projeto 3

TI327 - Tópicos em Inteligência Artificial

Prof. Guilherme Macedo

COTUCA - Unicamp

Quando os motores de busca como o Google exibem os resultados de uma pesquisa, eles fazem isso colocando páginas mais "importantes" e de maior qualidade em posições mais altas nos resultados de pesquisa do que páginas menos importantes. Mas como o motor de busca sabe quais páginas são mais importantes do que outras?

Uma heurística pode ser que uma página "importante" é aquela para a qual muitas outras páginas criam links, já que é razoável imaginar que mais sites irão vincular a uma página da web de maior qualidade do que a uma página da web de menor qualidade. Portanto, poderíamos imaginar um sistema onde cada página recebe uma classificação de acordo com o número de links que recebe de outras páginas, e classificações mais altas indicariam maior importância.

Mas essa definição não é perfeita: se alguém quiser fazer sua página parecer mais importante, então, sob esse sistema, poderia simplesmente criar muitas outras páginas que linkam para sua página desejada para inflacionar artificialmente sua classificação.

Por essa razão, o algoritmo PageRank foi criado pelos cofundadores do Google (incluindo Larry Page, a quem o algoritmo foi nomeado). No algoritmo PageRank, um site é mais importante se for vinculado por outros sites importantes, e os links de sites menos importantes têm seu peso reduzido. Essa definição parece um pouco circular, mas resulta que existem várias estratégias para calcular essas classificações.

### Modelo do Navegador Aleatório

Uma maneira de pensar sobre o PageRank é com o modelo do navegador aleatório, que considera o comportamento de um navegador hipotético na internet que clica em links aleatoriamente. Considere o conjunto de páginas da web abaixo, onde uma seta entre duas páginas indica um link de uma página para outra.

O modelo do navegador aleatório imagina um navegador que começa com uma página da web aleatoriamente e, em seguida, escolhe links para seguir aleatoriamente. Se o navegador estiver na Página 2, por exemplo, ele escolheria aleatoriamente entre a Página 1 e a Página 3 para visitar a seguir (links duplicados na mesma página são tratados como um único link, e links de uma página para si mesma também são ignorados). Se ele escolhesse a Página 3, o navegador então escolheria aleatoriamente entre a Página 2 e a Página 4 para visitar a seguir.

O PageRank de uma página, então, pode ser descrito como a probabilidade de que um navegador aleatório esteja nessa página em um dado momento. Afinal, se houver mais links para uma determinada página, então é mais provável que um navegador aleatório acabe nessa página. Além disso, um

link de um site mais importante é mais provável de ser clicado do que um link de um site menos importante para o qual menos páginas linkam, então esse modelo lida com a ponderação dos links por sua importância também.

Uma maneira de interpretar esse modelo é como uma Cadeia de Markov, onde cada página representa um estado, e cada página tem um modelo de transição que escolhe entre seus links aleatoriamente. A cada passo de tempo, o estado muda para uma das páginas vinculadas pelo estado atual.

Ao amostrar estados aleatoriamente da Cadeia de Markov, podemos obter uma estimativa para o PageRank de cada página. Podemos começar escolhendo uma página aleatoriamente e, em seguida, continuar seguindo links aleatoriamente, mantendo o controle de quantas vezes visitamos cada página. Depois de reunir todas as nossas amostras (com base em um número escolhido antecipadamente), a proporção de tempo que estivemos em cada página pode ser uma estimativa para a classificação dessa página.

No entanto, essa definição de PageRank se mostra ligeiramente problemática, se considerarmos uma rede de páginas como a abaixo.

Imagine que começamos aleatoriamente amostrando a Página 5. Então não teríamos escolha a não ser ir para a Página 6, e depois não teríamos escolha a não ser voltar para a Página 5, e então Página 6 novamente, e assim por diante. Acabaríamos com uma estimativa de 0,5 para o PageRank das Páginas 5 e 6, e uma estimativa de 0 para o PageRank de todas as outras páginas, já que passamos todo o nosso tempo nas Páginas 5 e 6 e nunca visitamos nenhuma das outras páginas.

Para garantir que possamos sempre chegar a outro lugar no conjunto de páginas da web, vamos introduzir um fator de amortecimento d em nosso

modelo. Com probabilidade d (onde d é geralmente definido em torno de $0,85$), o navegador aleatório escolherá um dos links na página atual aleatoriamente. Mas, caso contrário (com probabilidade $1-d$), o navegador aleatório escolhe uma página aleatoriamente entre todas as páginas do conjunto (incluindo a página em que está atualmente).

Nosso navegador aleatório agora começa escolhendo uma página aleatoriamente e, então, para cada amostra adicional que gostaríamos de gerar, escolhe um link da página atual aleatoriamente com probabilidade d e escolhe qualquer página aleatoriamente com probabilidade $1-d$. Se mantivermos o controle de quantas vezes cada página apareceu como uma amostra, podemos tratar a proporção de estados que estavam em uma determinada página como seu PageRank.

### 0.0.1 Algoritmo Iterativo

Também podemos definir o PageRank de uma página usando uma expressão matemática recursiva. Seja $P(p)$ o PageRank de uma determinada página p: a probabilidade de que um navegador aleatório acabe nessa página. Como definimos PR(p)? Bem, sabemos que há duas maneiras pelas quais um navegador aleatório poderia acabar na página:

- com probabilidade $1-d$, o navegador escolheu uma página aleatoriamente e acabou na página $p$.
- com probabilidade $d$, o navegador seguiu um link de uma página $i$ para a página $p$.

A primeira condição é bastante direta de expressar matematicamente: é $1-d$

dividido por N, onde N é o número total de páginas em todo o conjunto. Isso ocorre porque a probabilidade de $1-d$ de escolher uma página aleatoriamente é dividida igualmente entre todas as $N$ páginas possíveis.

Para a segunda condição, precisamos considerar cada página possível i que vincula à página p. Para cada uma dessas páginas de entrada, seja $NumLinks(i)$ o número de links na página i. Cada página i que vincula a p tem seu próprio PageRank, PR(i), representando a probabilidade de estarmos na página i em um dado momento. E como, a partir da página i, viajamos para qualquer um dos links dessa página com probabilidade igual, dividimos PR(i) pelo número de links $NumLinks(i)$ para obter a probabilidade de que estivemos na página i e escolhemos o link para a página p.

Isso nos dá a seguinte definição para o PageRank de uma página p.

Nesta fórmula, d é o fator de amortecimento, N é o número total de páginas no conjunto, i varia entre todas as páginas que vinculam à página p, e $NumLinks(i)$ é o número de links presentes na página i.

Como calcularíamos os valores de PageRank para cada página, então? Podemos fazer isso por iteração: comece assumindo que o PageRank de cada página é 1 / N (ou seja, igualmente provável de estar em qualquer página). Em seguida, use a fórmula acima para calcular novos valores de PageRank para cada página, com base nos valores anteriores de PageRank. Se continuarmos repetindo esse processo, calculando um novo conjunto de valores de PageRank para cada página com base no conjunto anterior de valores de PageRank, eventualmente os valores de PageRank convergirão (ou seja, não mudarão mais do que um pequeno limite a cada iteração).

Neste projeto, você implementará ambas as abordagens para calcular PageR-

ank - calculando tanto amostrando páginas de um navegador aleatório da Cadeia de Markov quanto aplicando iterativamente a fórmula de PageRank.