04

Criando um dicionário

Criando um dicionário

Até agora, vimos diversos problemas que podemos ter no nosso dia a dia, e também, aprendemos como resolvê-los. Entretanto, iremos nos deparar com algo novo, isto é, uma situação que ainda não vimos. Vejamos então o problema que precisamos resolver:

- Recebi alguns e-mails novos dentro da página de contato do meu site com os seguintes conteúdos:
  - Se eu comprar cinco anos antecipados, eu ganho algum desconto?
  - O exercício 15 do curso de Java 1 está com a resposta errada. Pode conferir por favor?
  - Existe algum curso para cuidar do marketing da minha empresa?
  - Já trabalho como designer e queria aprender mais de UX, quais cursos devo fazer?

Observe que cada um dos e-mail refere-se a um assunto específico, por exemplo, o primeiro e-mail refere-se ao setor comercial/financeiro da minha empresa. Já o segundo e-mail, refere-se a um assunto totalmente diferente do primeiro, pois no primeiro, o remetente está interessado em comprar, porém, no segundo, ele está relatando um problema técnico. E o terceiro e-mail? Também é diferente, pois o remetente está em dúvida, ou seja, está precisando de sugestões para o seu objetivo. Por fim, o quarto e-mail, é bem similar ao terceiro, pois o remetente está em dúvida e precisa de dicas/sugestões sobre quais cursos ele poderia fazer. Então podemos classificar esses e-mails da seguinte forma:

- 1º venda ou comercial.
- 2º problemas técnicos.
- 3º e 4º dúvidas e sugestões de carreira.

Dentro da minha empresa, cada um desses tipos de e-mails são tratados e respondidos por diferentes setores, por exemplo, para e-mails de venda ou comercial, são respondidos pelo pessoal de vendas, porém, para os e-mails de problemas técnicos, serão respondidos pelo pessoal técnico, por fim, para e-mails relacionados à carreira, serão respondidos por pessoas que possuem uma certa experiência em sua carreira. Portanto, concluímos que dentro da minha empresa existem diversas sessões, logo, cada e-mail que chega pelo formulário de contato precisa ser redirecionado à sessão que irá atendê-lo. Como poderíamos resolver esse problema? Uma das formas bem comum de resolver esse tipo de situação seria adicionar um *combo box* (https://en.wikipedia.org/wiki/Combo_box) permitindo o usuário escolher entre as três opções:

- Comercial.
- Técnico.
- Carreira.

Considerando apenas o cenário que vimos até agora, resolveria. Entretanto, suponhamos que a minha empresa cresceu, ou seja, surgiram sessões diferentes, como por exemplo o financeiro, então adicionaríamos dentro do combo box:

- Comercial.
- Financeiro.
- Técnico.
- Carreira.

Se tivéssemos que mandar um e-mail que tanto o comercial quanto o financeiro resolvessem, qual dentre essas sessões escolheríamos? Vejamos outro exemplo adicionando a sessão de conteúdo:

- Comercial.
- Financeiro.
- Técnico.
- Conteúdo.
- Carreira.

Se tivéssemos que enviar o segundo e-mail que trata de um problema técnico, porém, também refere-se ao conteúdo, para qual sessão deveríamos enviar? E para o terceiro que refere-se tanto a conteúdo quanto à carreira, para qual dessas sessões enviaríamos? Percebe que está começando a ficar difícil para o usuário final escolher para qual sessão ele precisa enviar? Como podemos lidar com esse problema? Primeiramente, vamos permitir apenas que os nossos e-mails sejam classificados pelas três categorias que vimos anteriormente:

- Comercial.
- Técnico.
- Carreira.

Nesse instante, você pode estar pensando que já nos deparamos com um problema bem similar a esse que dado um conjunto de dados **numéricos**, classificávamos em três ou mais categorias distintas, como por exemplo, o algoritmo `OneVsRest` e também a validação utilizando o k-fold. Mas tudo isso estava baseado em conjuntos numérico, ou seja, mesmo que tivéssemos categorias textuais, transformávamos esses dados em números novamente. Em outras palavras, dessa vez, ao invés de rodar um algoritmo com números, queremos rodar o algoritmo com textos, ou melhor, fazer com que o nosso algoritmo analise e classifique um texto. Como podemos fazer isso? Lembre-se, da mesma forma que vimos anteriormente, quando temos um problema grande, podemos reduzi-lo em problemas menores que saibamos resolver. Então quais são os problemas que já sabemos resolver? Até o momento, sabemos resolver o seguinte problema:

- Classificar números em uma categoria.

Então qual é o nosso grande desafio? É justamente transformar essas palavras, ou melhor, essas sequências de palavras, em uma sequência de números. E mais, mesmo os textos possuindo quantidade de caracteres diferentes, como por exemplo o 1º e o 2º e-mail, elas precisam possuir o mesmo tamanho. Portanto, todas as sequência de palavras precisam conter a mesma quantidade de colunas, como por exemplo o arquivo `situacao_do_cliente.csv` :

```
recencia,frequencia,semanas_de_inscricao,situacao
1,4,4,2
2,1,2,1
1,4,2,2
1,3,8,1
2,2,1,1
...
4,1,6,0
```

Note que as colunas desse arquivo são fixas para todos os dados, ou seja, precisamos fazer o mesmo com cada sequência de texto. Consegue imaginar como podemos fazer isso? Podemos começar pela primeira sequência de texto:

- Se eu comprar cinco anos antecipados, eu ganho algum desconto?

Tentaremos primeiro analisar essa frase. Mas como faremos isso? Precisamos realizar essa analise para cada uma dessas palavras. Entretanto, o nosso algoritmo já viu alguma palavra na vida? A resposta é não! Portanto, precisaremos armazenar quaisquer palavras existentes no texto, vejamos como seria esse armazenamento:

- Se eu comprar cinco anos antecipados, eu ganho algum desconto?

```
[Se, eu, comprar, cinco, anos, antecipados, ganho, algum, desconto]
```

Observe que criamos um array adicionando as palavras do texto, mas perceba que existem 10 palavras nesse texto, porém, palavras distintas são 9, pois a palavra "eu" repete uma vez. Esse processo de armazenamento das palavras contidas no texto, é justamente a identificação de todas palavras distintas. Então qual é o nosso próximo passo? Atualmente conhecemos todas as palavras contidas no array, porém, quantas vezes essa mesma palavra aparece nesse texto? Ou seja, quantas vezes a palavra "se" aparece? E a palavra "eu"? Vejamos o resultado:

- Se eu comprar cinco anos antecipados, eu ganho algum desconto?

```
[Se, eu, comprar, cinco, anos, antecipados, ganho, algum, desconto]
[1, 2, 1, 1, 1, 1, 1, 1, 1]
```

Repare que agora temos 2 arrays, um deles indica todas as palavras distintas e o outro a quantidade que cada uma dessas palavras se repetem dentro do texto. Considerando esse primeiro exemplo, suponhamos que estamos analisando um outro texto que contém as mesmas palavras que armazenamos no nosso array de palavras distintas, então o array de quantidade de palavras distintas para esse texto resulta em:

```
[Se, eu, comprar, cinco, anos, antecipados, ganho, algum, desconto]
[1, 2, 1, 1, 1, 1, 1, 1, 1]
[2, 1, 0, 3, 1, 3, 4, 2, 1]
```

Observe que mesmo sendo textos diferentes, conseguimos reprensentar todos os nossos textos por meio de arrays que sempre terão o mesmo tamanho! Além disso, se de repente, no texto que estamos analisando, aparecer palavras como: "carreira", "curso", "novos cursos" entre outras palavras referentes à carreira, provavelmente, refere-se a um texto destinado à carreira. Caso no texto apareça as palavras: "preço", "desconto", "valor", "pagamento", provavelmente é do comercial. Mas somos nós que devemos dizer isso? Não, o próprio algoritmo terá a capacidade de analisar todos esses aspectos e classificar em qual sessão o texto se encaixa.

O detalhe importante nesse instante é que conseguimos transformar um texto em um array de tamanho fixo, ou seja, qualquer texto que contenha essas mesmas palavras que armazenamos, poderão ser representados com arrays diferentes, porém de tamanhos fixos! Vejamos um exemplo:

- Se eu comprar cinco anos antecipados, eu ganho algum desconto?
- Eu ganho desconto se comprar cinco anos antecipados?

Note que esse exemplo trata-se de um texto com palavras que vimos até agora. Então como ficaria os arrays para cada uma dessas palavras?

```
[Se, eu, comprar, cinco, anos, antecipados, ganho, algum, desconto]
```

- Se eu comprar cinco anos antecipados, eu ganho algum desconto?
  - [1, 2, 1, 1, 1, 1, 1, 1, 1]
- Eu ganho desconto se comprar cinco anos antecipados?
  - [1, 1, 1, 1, 1, 1, 1, 0, 1]

Veja que o primeiro texto é representado com o array `[1, 2, 1, 1, 1, 1, 1, 1]` e o segundo com o seguinte array `[1, 1, 1, 1, 1, 1, 0, 1]`. Vamos verificar mais uma frase:

- Se eu comprar cinco anos antecipados, eu ganho algum desconto?
- Eu ganho desconto se comprar cinco anos antecipados?
- Ao terminar um curso, eu ganho um certificado?

Observe que dessa vez, nem todas as palavras dessa frase nova estão contidas no nosso array de palavras distintas, isto é, o nosso dicionário. Portanto, precisamos criar um único array que dê suporte para todas as palavras dos textos que temos, ou seja, com todas as palavras distintas contidas nos textos, em seguida, criamos o array para cada uma das frases indicando a quantidade de vezes que uma dessas palavras se repetem. Para esse exemplo, o que precisamos fazer? Simplesmente adicionar ao nosso array de palavras distintas, isto é, todas as palavras que não estão contidas dentro dele, vejamos como fica o nosso array de palavras distintas:

```
[Se, eu, comprar, cinco, anos, antecipados, ganho, algum, desconto, Ao, terminar, um, curso, certifi
```

Veja que agora temos um array de palavras distintas, que da suporte para cada uma das frases que vimos. Então qual que é o próximo passo? É justamente gerar o array que indica a quantidade de vezes que essas palavras se repentem. Começaremos pela primeira frase:

- Se eu comprar cinco anos antecipados, eu ganho algum desconto?
  - [1, 2, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0]

Vejamos agora a próxima frase, nesse caso utilizaremos a frase nova:

- Ao terminar um curso, eu ganho um certificado?
  - [0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 2, 1, 1]

Observe que dessa vez fomos capazes de representar em arrays do mesmo tamanho, todas as palavras do nosso universo, ou seja, todas as palavras contidas no array de palavras distintas. Portanto, agora conseguimos trabalhar esses dados dentro dos nossos algoritmos de classificação. Vale lembrar que teremos que realizar alguns ajustes para realizar esse tipo de classificação dentro do nosso algoritmo, porém, aprendemos que, quando temos um conjunto de texto e precisamos transformá-lo em um conjunto de números, precisamos criar um dicionário.