Aula 05

Implementação do algoritmo **Naïve Bayes** para classificação de texto

O objetivo deste exercício é implementar o algoritmo Naïve Bayes para a classificação de textos. Especificamente, treinaremos um modelo capaz de determinar o voto mais provável de um deputado na votação do impeachment com base no seu discurso, considerando como possíveis classes "sim" ou "não" (foram desconsideradas as abstenções). A linguagem de programação a ser usada é de livre escolha do aluno, mas não poderão ser utilizadas funções para treinamento ou teste de Naïve Bayes disponibilizadas em pacotes ou bibliotecas.

Junto a este arquivo do enunciado, são fornecidos dois arquivos em formato texto: `nao_com_preProcess.txt` e `sim_com_preProcess.txt` contendo os dados de treinamento das classes "não" (negativa) e "sim" (positiva), respectivamente. Estes dados foram pré-processados com alguns passos simples a fim de facilitar o seu uso nesta tarefa, removendo pontuações, nomes ou filiações dos deputados, e as palavras “sim” e “não” dos discursos para não influenciar na classificação.

O formato destes arquivos é dado por uma sequência de discursos, cada discurso se refere a uma instância e uma linha vazia indica o término de uma instância. Por exemplo:

```
“O o meu voto é o prosseguimento do processo de impedimento da Sra Presidente da República”

“Pelo legado de Getúlio Jango e Brizola pela democracia e o Estado Democrático de Direito pelo Brasil o PDT vota”
```

Existem um total de 364 instâncias positivas (classe "Sim") e 133 instâncias negativas (classe "Não") no conjunto de treinamento fornecido.

1. Antes de iniciar a implementação do código, separe uma porção destes conjuntos de treinamento para utilizar como um conjunto de teste a fim de avaliar o desempenho do modelo treinado. Falaremos mais adiante na disciplina sobre técnicas de avaliação de desempenho de modelos, por isso neste exercício adotaremos uma abordagem muito simples: reserve em torno de 10% das instâncias de cada classe como dados de teste, não utilizando estas instâncias no treinamento do modelo (passos a seguir).

2. Processe os arquivos texto com as instâncias dos dados de treinamento (já excluídos os dados de teste) para ambas as classes a fim de gerar os atributos de cada instância seguindo um modelo chamado "*bag-of-word*s": cada documento é representado por uma lista de palavras e o número de vezes em que cada palavra ocorre no documento. No contexto da classificação, cada documento (referente a um discurso) é uma instância, as palavras contidas serão os atributos da instância, e cada atributo terá um valor inteiro correspondendo à contagem de quantas vezes foi observado em uma dada instância. Um conjunto de palavras ("dicionário") é gerado para todos os dados do treinamento, e usado a fim de uniformizar o conjunto de atributos entre todas as instâncias de treinamento. Observe o exemplo abaixo:

   Dadas as instâncias:

   (1) John likes to watch movies. Mary likes movies too.

   (2) John also likes to watch football games.

   Gera-se o dicionário como a união de todas as palavras em (1) e (2):

```
[
  "John",
  "likes",
  "to",
  "watch",
  "movies",
  "Mary",
  "too",
  "also",
  "football",
  "games"
]
```

As instâncias são então representadas como vetores de atributos, onde cada atributo se refere à frequência da palavra correspondente (seguindo a ordem no dicionário) na instância em questão, como segue:

(1) [1, 2, 1, 1, 2, 1, 1, 0, 0, 0]
(2) [1, 1, 1, 1, 0, 0, 0, 1, 1, 1]

3. A partir destas informações, é possível calcular as probabilidades condicionais e a priori envolvidas no classificador Naïve Bayes (abaixo). Relembrando os exemplos vistos em aula, a contagem da frequência de cada atributo (palavra) é realizada para cada classe (sim/não), e assim na classificação de instâncias desconhecidas (dados de teste) a classe que maximiza a probabilidade a posteriori é retornada como a classe predita. Perceba que esta etapa de extração de atributos realizada a partir do documento, com a contagem de frequência de palavras, já nos fornece a informação necessária para estimar o termo referente à probabilidade condicional, onde x_i se refere aos atributos (palavras) e y_i às classes (sim/não).

$$P(y_i|\mathbf{x}) = P(y_i)\prod_{j=1}^{d} P(x^j|y_i)$$

4. Ao realizar a predição para as instâncias nos dados de teste, você irá calcular a acurácia do modelo, uma medida de desempenho bastante simples (mas com limitações, como discutiremos na disciplina), que visa calcular a taxa de acerto do modelo, isto é, a porcentagem de instâncias corretamente classificadas. Para calcular a acurácia, você deve somar o número de instâncias de teste da classe "Sim" que foram preditas corretamente com o número de instâncias de teste da classe "Não" que foram preditas corretamente, e dividir este valor pelo número total de instâncias nos dados de teste.

5. Faça o treinamento e teste do modelo para os dados fornecidos, reportando a acurácia obtida para as seguintes variações:

a. Sem aplicar a Correção de Laplace, isto é, sem tratar o problema de frequência zero, usando a fórmula original do Naïve Bayes
b. Sem aplicar a Correção de Laplace, usando a fórmula do Naïve Bayes expressa como uma soma, utilizando logaritmos
c. Aplicando a Correção de Laplace (conforme visto em aula) combinada à fórmula do Naïve Bayes baseada em logaritmos.

Após a conclusão do exercício, você deverá entregar via Moodle:

- O código da implementação do Naïve Bayes
- Um breve relatório com o desempenho do algoritmo Naïve Bayes e análise dos resultados em cada uma das etapas acima

O prazo final de entrega deste exercício é dia **21 de setembro às 23:55h**. A entrega do exercício, mesmo que de forma incompleta, valerá a presença da aula do dia 14 de setembro, e a entrega completa poderá garantir ao aluno 1 ponto extra na média final.