# Inteligência artificial aplicada

## Arquitetura de dados

### Trabalho da disciplina - Questão 1

Nomes: Paulo Sergio Herval Silva Junior, Pedro de Sousa Alves Graça, Matheus Eduardo de Arazão e Anderson Felipe de Paiva

# Identificador automático de idioma

**Problema**: Dados um texto de entrada, é possível identificar em qual língua o texto está escrito?

Entrada: "texto qualquer"
Saída: português ou inglês ou francês ou italiano ou...

# O processo de Reconhecimento de Padrões

O objetivo desse trabalho é demonstrar o processo de "construção de atributos" e como ele é fundamental para o **Reconhecimento de Padrões (RP)**.

Primeiro um conjunto de "amostras" previamente conhecido (classificado)

In [ ]:
```
#
# amostras de texto em diferentes línguas
#
ingles = [
"Hello, how are you?",
"I love to read books.",
"The weather is nice today.",
"Where is the nearest restaurant?",
"What time is it?",
"I enjoy playing soccer.",
"Can you help me with this?",
"I'm going to the movies tonight.",
"This is a beautiful place.",
"I like listening to music.",
"Do you speak English?",
"What is your favorite color?",
"I'm learning to play the guitar.",
"Have a great day!",
"I need to buy some groceries.",
```

```
"Let's go for a walk.",
"How was your weekend?",
"I'm excited for the concert.",
"Could you pass me the salt, please?",
"I have a meeting at 2 PM.",
"I'm planning a vacation.",
"She sings beautifully.",
"The cat is sleeping.",
"I want to learn French.",
"I enjoy going to the beach.",
"Where can I find a taxi?",
"I'm sorry for the inconvenience.",
"I'm studying for my exams.",
"I like to cook dinner at home.",
"Do you have any recommendations for restaurants?",
]

espanhol = [
"Hola, ¿cómo estás?",
"Me encanta leer libros.",
"El clima está agradable hoy.",
"¿Dónde está el restaurante más cercano?",
"¿Qué hora es?",
"Voy al parque todos los días.",
"¿Puedes ayudarme con esto?",
"Me gustaría ir de vacaciones.",
"Este es mi libro favorito.",
"Me gusta bailar salsa.",
"¿Hablas español?",
"¿Cuál es tu comida favorita?",
"Estoy aprendiendo a tocar el piano.",
"¡Que tengas un buen día!",
"Necesito comprar algunas frutas.",
"Vamos a dar un paseo.",
"¿Cómo estuvo tu fin de semana?",
"Estoy emocionado por el concierto.",
"¿Me pasas la sal, por favor?",
"Tengo una reunión a las 2 PM.",
"Estoy planeando unas vacaciones.",
"Ella canta hermosamente.",
"El perro está jugando.",
"Quiero aprender italiano.",
"Disfruto ir a la playa.",
"¿Dónde puedo encontrar un taxi?",
"Lamento las molestias.",
"Estoy estudiando para mis exámenes.",
"Me gusta cocinar la cena en casa.",
"¿Tienes alguna recomendación de restaurantes?",
]

portugues = [
"Estou indo para o trabalho agora.",
"Adoro passar tempo com minha família.",
"Preciso comprar leite e pão.",
"Vamos ao cinema no sábado.",
"Gosto de praticar esportes ao ar livre.",
```

```
"O trânsito está terrível hoje.",
"A comida estava deliciosa!",
"Você já visitou o Rio de Janeiro?",
"Tenho uma reunião importante amanhã.",
"A festa começa às 20h.",
"Estou cansado depois de um longo dia de trabalho.",
"Vamos fazer um churrasco no final de semana.",
"O livro que estou lendo é muito interessante.",
"Estou aprendendo a cozinhar pratos novos.",
"Preciso fazer exercícios físicos regularmente.",
"Vou viajar para o exterior nas férias.",
"Você gosta de dançar?",
"Hoje é meu aniversário!",
"Gosto de ouvir música clássica.",
"Estou estudando para o vestibular.",
"Meu time de futebol favorito ganhou o jogo.",
"Quero aprender a tocar violão.",
"Vamos fazer uma viagem de carro.",
"O parque fica cheio aos finais de semana.",
"O filme que assisti ontem foi ótimo.",
"Preciso resolver esse problema o mais rápido possível.",
"Adoro explorar novos lugares.",
"Vou visitar meus avós no domingo.",
"Estou ansioso para as férias de verão.",
"Gosto de fazer caminhadas na natureza.",
"O restaurante tem uma vista incrível.",
"Vamos sair para jantar no sábado.",
]
```

A "amostras" de texto precisa ser "transformada" em **padrões**

Um padrão é um conjunto de características, geralmente representado por um vetor e um conjunto de padrões no formato de tabela. Onde cada linha é um padrão e as colunas as características e, geralmente, na última coluna a **classe**

In [ ]:
```python
import random

pre_padroes = []
for frase in ingles:
  pre_padroes.append( [frase, 'inglês'])

for frase in espanhol:
  pre_padroes.append( [frase, 'espanhol'])

for frase in portugues:
  pre_padroes.append( [frase, 'português'])

random.shuffle(pre_padroes)
print(pre_padroes)
```

[['O restaurante tem uma vista incrível.', 'português'], ['Gosto de fazer caminhadas na natureza.', 'português'], ['O filme que assisti ontem foi ótimo.', 'português'], ['Você gosta de dançar?', 'português'], ['Estoy aprendiendo a tocar el piano.', 'espanhol'], ['Preciso resolver esse problema o mais rápido possível.', 'português'], ['Tenho uma reunião importante amanhã.', 'português'], ['What is your favorite color?', 'inglês'], ['O livro que estou lendo é muito interessante.', 'português'], ['Preciso comprar leite e pão.', 'português'], ["Let's go for a walk.", 'inglês'], ['Adoro explorar novos lugares.', 'português'], ['Estou cansado depois de um longo dia de trabalho.', 'português'], ['I enjoy playing soccer.', 'inglês'], ['O parque fica cheio aos finais de semana.', 'português'], ['Ella canta hermosamente.', 'espanhol'], ['Hello, how are you?', 'inglês'], ["I'm excited for the concert.", 'inglês'], ['Vamos fazer um churrasco no final de semana.', 'português'], ['Este es mi libro favorito.', 'espanhol'], ['The weather is nice today.', 'inglês'], ['I love to read books.', 'inglês'], ['Me gusta cocinar la cena en casa.', 'espanhol'], ['Estou aprendendo a cozinhar pratos novos.', 'português'], ['I need to buy some groceries.', 'inglês'], ['She sings beautifully.', 'inglês'], ['Where is the nearest restaurant?', 'inglês'], ['A festa começa às 20h.', 'português'], ['Estoy estudiando para mis exámenes.', 'espanhol'], ['¿Me pasas la sal, por favor?', 'espanhol'], ['Vou viajar para o exterior nas férias.', 'português'], ['Where can I find a taxi?', 'inglês'], ['I like to cook dinner at home.', 'inglês'], ['Can you help me with this?', 'inglês'], ['O trânsito está terrível hoje.', 'português'], ['¿Puedes ayudarme con esto?', 'espanhol'], ['Necesito comprar algunas frutas.', 'espanhol'], ['Have a great day!', 'inglês'], ['¿Cómo estuvo tu fin de semana?', 'espanhol'], ['¿Dónde está el restaurante más cercano?', 'espanhol'], ['Me gusta bailar salsa.', 'espanhol'], ['El perro está jugando.', 'espanhol'], ['Gosto de praticar esportes ao ar livre.', 'português'], ['¿Hablas español?', 'espanhol'], ['How was your weekend?', 'inglês'], ['Vamos fazer uma viagem de carro.', 'português'], ['Me gustaría ir de vacaciones.', 'espanhol'], ['Estou ansioso para as férias de verão.', 'português'], ['Tengo una reunión a las 2 PM.', 'espanhol'], ['I have a meeting at 2 PM.', 'inglês'], ["I'm studying for my exams.", 'inglês'], ['This is a beautiful place.', 'inglês'], ['Vamos sair para jantar no sábado.', 'português'], ['Vamos a dar un paseo.', 'espanhol'], ['¿Dónde puedo encontrar un taxi?', 'espanhol'], ['Você já visitou o Rio de Janeiro?', 'português'], ['Disfruto ir a la playa.', 'espanhol'], ["I'm going to the movies tonight.", 'inglês'], ['¿Cuál es tu comida favorita?', 'espanhol'], ['Lamento las molestias.', 'espanhol'], ['Quero aprender a tocar violão.', 'português'], ['Do you speak English?', 'inglês'], ["I'm learning to play the guitar.", 'inglês'], ['Gosto de ouvir música clássica.', 'português'], ['Estoy planeando unas vacaciones.', 'espanhol'], ['Hoje é meu aniversário!', 'português'], ['Estoy emocionado por el concierto.', 'espanhol'], ['¿Qué hora es?', 'espanhol'], ["I'm sorry for the inconvenience.", 'inglês'], ['Could you pass me the salt, please?', 'inglês'], ['Hola, ¿cómo estás?', 'espanhol'], ['Meu time de futebol favorito ganhou o jogo.', 'português'], ['Vou visitar meus avós no domingo.', 'português'], ['Preciso fazer exercícios físicos regularmente.', 'português'], ['Estou indo para o trabalho agora.', 'português'], ['Estou estudando para o vestibular.', 'português'], ['¿Tienes alguna recomendación de restaurantes?', 'espanhol'], ['El clima está agradable hoy.', 'espanhol'], ['I enjoy going to the beach.', 'inglês'], ['¡Que tengas un buen día!', 'espanhol'], ['I like listening to music.', 'inglês'], ['Adoro passar tempo com minha família.', 'português'], ['Do you have any recommendations for restaurants?', 'inglês'], ["I'm planning a vacation.", 'inglês'], ['What time is it?', 'inglês'], ['Quiero aprender italiano.', 'espanhol'], ['The cat is sleeping.', 'inglês'], ['Me encanta leer libros.', 'espanhol'], ['Voy al parque todos los días.', 'espanhol'], ['A comida estava deliciosa!', 'português'], ['I want to learn French.', 'inglês'], ['Vamos ao cinema no sábado.', 'português']]

O DataFrame do pandas facilita a visualização.

In [ ]:
```
import pandas as pd
dados = pd.DataFrame(pre_padroes)
dados
```

Out[ ]:

| | 0 | 1 |
|---|---|---|
| **0** | O restaurante tem uma vista incrível. | português |
| **1** | Gosto de fazer caminhadas na natureza. | português |
| **2** | O filme que assisti ontem foi ótimo. | português |
| **3** | Você gosta de dançar? | português |
| **4** | Estoy aprendiendo a tocar el piano. | espanhol |
| **...** | ... | ... |
| **87** | Me encanta leer libros. | espanhol |
| **88** | Voy al parque todos los días. | espanhol |
| **89** | A comida estava deliciosa! | português |
| **90** | I want to learn French. | inglês |
| **91** | Vamos ao cinema no sábado. | português |

92 rows × 2 columns

# Construção dos atributos

Esse é o coração desse trabalho e que deverá ser desenvolvido por vocês. Pensem em como podemos "medir" cadas frase/sentença e extrair características que melhorem o resultado do processo de identificação.

Após a criação de cada novo atributo, execute as etapas seguintes e registre as métricas da matriz de confusão. Principalmente acurácia e a precisão.

In [ ]:
```
import re
import nltk
from nltk import ngrams
from nltk.corpus import stopwords
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer

nltk.download('punkt')
nltk.download('stopwords')
nltk.download('corpus')

# a entrada é o vetor pre_padroes e a saída desse passo deverá ser "padrões"

def tamanhoMedioFrases(texto):
  pattern_regex = re.compile('[^\w+]', re.UNICODE)
```

```
  texto = re.sub(pattern_regex,' ',texto)
  palavras = re.split("\s",texto)
  #print(palavras)
  tamanhos = [len(s) for s in palavras if len(s)>0]
  #print(tamanhos)
  soma = 0
  for t in tamanhos:
    soma=soma+t

  return soma / len(tamanhos)

def obter_frquencia_ngrans(text, n):
  palavras = nltk.word_tokenize(text)

  ngrams_list = list(ngrams(palavras, n))

  freq = nltk.FreqDist(ngrams_list)

  return freq.N()

def obter_especial_chars(text):
  chars_especiais = len(re.findall(r"[^\w\s\.\,\u00C0-\u017F]", text))
  letras_com_acento = len(re.findall(r'[áéíóúãõâêîôûàèìòùäëïöüçñÁÉÍÓÚÃÕÂÊÎÔÛÀÈÌÒÙÄË
  return chars_especiais + letras_com_acento

def obter_media_ngrans(text, n):
  palavras = nltk.word_tokenize(text)

  ngrams_list = list(ngrams(palavras, n))

  media = len(ngrams_list) / len(palavras)

  return media

def obter_letras_ngrams(text, n):
  palavras = nltk.word_tokenize(text)

  ngrams_list = list(ngrams(palavras, n))

  return ngrams_list

def count_stop_words(text, language):

  lang = ''

  if language == 'inglês':
    lang = 'english'
  elif language == 'espanhol':
    lang = 'spanish'
  else:
    lang = 'portuguese'

  stop_words = set(stopwords.words(lang))
  words = nltk.word_tokenize(text)
  count = sum(1 for word in words if word.lower() in stop_words)
  return count
```

```
def media_tamanho_palavras(text):
  words = nltk.word_tokenize(text)
  word_lengths = [len(word) for word in words]
  return sum(word_lengths) / len(word_lengths)

def diversidade_caracteres(text):
  return len(set(text)) / len(text)

def extraiCaracteristicas(frase):
  # frase é um vetor [ 'texto', 'lingua' ]
  texto = frase[0]
  texto = texto.lower()
  #print(texto)
  caracteristica1=tamanhoMedioFrases(texto)
  caracteristica2=obter_frquencia_ngrans(texto, 3)
  caracteristica3=obter_especial_chars(texto)
  caracteristica4=obter_media_ngrans(texto, 3)
  caracteristica5=count_stop_words(texto, frase[1])
  caracteristica6=media_tamanho_palavras(texto)
  caracteristica7=diversidade_caracteres(texto)
  # acrescente as suas funcoes no vetor padrao
  padrao = [caracteristica1, caracteristica2, caracteristica3, caracteristica4, car
  return padrao

def geraPadroes(frases):
  padroes = []
  for frase in frases:
    padrao = extraiCaracteristicas(frase)
    padroes.append(padrao)

  return padroes

# converte o formato [frase classe] em
# [caracteristica_1, caracteristica_2,... caracteristica n, classe]
padroes = geraPadroes(pre_padroes)

#
# apenas para visualizacao
print(padroes)

dados = pd.DataFrame(padroes)
dados
```

[[5.166666666666667, 5, 1, 0.7142857142857143, 3, 4.571428571428571, 0.43243243243243246, 'português'], [5.333333333333333, 5, 0, 0.7142857142857143, 2, 4.714285714285714, 0.47368421052631576, 'português'], [4.142857142857143, 6, 1, 0.75, 3, 3.75, 0.4166666666666667, 'português'], [4.25, 3, 3, 0.6, 2, 3.6, 0.7142857142857143, 'português'], [4.833333333333333, 5, 0, 0.7142857142857143, 3, 4.285714285714286, 0.42857142857142855, 'espanhol'], [5.75, 7, 2, 0.7777777777777778, 3, 5.222222222222222, 0.3148148148148148, 'português'], [6.2, 4, 2, 0.6666666666666666, 2, 5.333333333333333, 0.3888888888888889, 'português'], [4.6, 4, 1, 0.6666666666666666, 3, 4.0, 0.6071428571428571, 'inglês'], [4.625, 7, 1, 0.7777777777777778, 5, 4.222222222222222, 0.3777777777777777, 'português'], [4.6, 4, 1, 0.6666666666666666, 1, 4.0, 0.5, 'português'], [2.3333333333333335, 5, 1, 0.7142857142857143, 2, 2.2857142857142856, 0.7, 'inglês'], [6.25, 3, 0, 0.6, 0, 5.2, 0.5172413793103449, 'português'], [4.444444444444445, 8, 0, 0.8, 5, 4.1, 0.3877551020408163, 'português'], [4.75, 3, 0, 0.6, 1, 4.0, 0.6521739130434783, 'inglês'], [4.125, 7, 0, 0.7777777777777778, 3, 3.7777777777777777, 0.4146341463414634, 'português'], [7.0, 2, 0, 0.5, 1, 5.5, 0.5416666666666666, 'espanhol'], [3.5, 4, 1, 0.6666666666666666, 3, 2.6666666666666665, 0.631578947368421, 'inglês'], [3.6666666666666665, 5, 1, 0.7142857142857143, 3, 3.4285714285714284, 0.5357142857142857, 'inglês'], [4.5, 7, 0, 0.7777777777777778, 3, 4.111111111111111, 0.4090909090909091, 'português'], [4.2, 4, 0, 0.6666666666666666, 3, 3.6666666666666665, 0.5384615384615384, 'espanhol'], [4.2, 4, 0, 0.6666666666666666, 2, 3.6666666666666665, 0.5769230769230769, 'inglês'], [3.2, 4, 0, 0.6666666666666666, 2, 2.8333333333333335, 0.6666666666666666, 'inglês'], [3.7142857142857144, 6, 0, 0.75, 3, 3.375, 0.45454545454545453, 'espanhol'], [5.833333333333333, 5, 0, 0.7142857142857143, 2, 5.142857142857143, 0.4146341463414634, 'português'], [3.8333333333333335, 5, 0, 0.7142857142857143, 3, 3.4285714285714284, 0.5517241379310345, 'inglês'], [6.333333333333333, 2, 0, 0.5, 1, 5.0, 0.6818181818181818, 'inglês'], [5.4, 4, 1, 0.6666666666666666, 3, 4.666666666666667, 0.375, 'inglês'], [3.4, 4, 2, 0.6666666666666666, 2, 3.0, 0.6818181818181818, 'português'], [6.0, 4, 1, 0.6666666666666666, 3, 5.166666666666667, 0.4857142857142857, 'espanhol'], [3.3333333333333335, 6, 2, 0.75, 2, 2.875, 0.5, 'espanhol'], [4.428571428571429, 6, 1, 0.75, 3, 4.0, 0.4473684210526316, 'português'], [3.0, 5, 1, 0.7142857142857143, 4, 2.7142857142857144, 0.5833333333333334, 'inglês'], [3.2857142857142856, 6, 0, 0.75, 3, 3.0, 0.5, 'inglês'], [3.3333333333333335, 5, 1, 0.7142857142857143, 5, 3.0, 0.6538461538461539, 'inglês'], [5.0, 4, 3, 0.6666666666666666, 2, 4.333333333333333, 0.5333333333333333, 'português'], [5.25, 3, 2, 0.6, 2, 4.6, 0.6153846153846154, 'espanhol'], [7.0, 3, 0, 0.6, 1, 5.8, 0.53125, 'espanhol'], [3.25, 3, 1, 0.6, 2, 2.8, 0.6470588235294118, 'inglês'], [3.8333333333333335, 5, 3, 0.7142857142857143, 3, 3.5714285714285716, 0.5666666666666667, 'espanhol'], [5.333333333333333, 5, 5, 0.7142857142857143, 3, 4.857142857142857, 0.4358974358974359, 'espanhol'], [4.5, 3, 0, 0.6, 1, 3.8, 0.5909090909090909, 'espanhol'], [4.5, 3, 1, 0.6, 2, 3.8, 0.7272727272727273, 'espanhol'], [4.571428571428571, 6, 0, 0.75, 2, 4.125, 0.38461538461538464, 'português'], [6.5, 1, 3, 0.3333333333333333, 0, 5.0, 0.75, 'espanhol'], [4.25, 3, 1, 0.6, 3, 3.6, 0.6666666666666666, 'inglês'], [4.333333333333333, 5, 0, 0.7142857142857143, 2, 3.857142857142857, 0.5, 'português'], [4.8, 4, 1, 0.6666666666666666, 2, 4.166666666666667, 0.5862068965517241, 'espanhol'], [4.428571428571429, 6, 2, 0.75, 4, 4.0, 0.4473684210526316, 'português'], [3.142857142857143, 6, 1, 0.75, 4, 2.875, 0.5862068965517241, 'espanhol'], [2.5714285714285716, 6, 0, 0.75, 4, 2.375, 0.52, 'inglês'], [3.3333333333333335, 5, 1, 0.7142857142857143, 3, 3.142857142857143, 0.6923076923076923, 'inglês'], [4.2, 4, 0, 0.6666666666666666, 3, 3.6666666666666665, 0.5384615384615384, 'inglês'], [4.5, 5, 1, 0.7142857142857143, 2, 4.0, 0.48484848484848486, 'português'], [3.2, 4, 0, 0.6666666666666666, 2, 2.8333333333333335, 0.6190476190476191, 'espanhol'], [5.0, 4, 3, 0.6666666666666666, 1, 4.5, 0.5161290322580645, 'espanhol'], [3.7142857142857144, 6, 3, 0.75, 4, 3.375, 0.5151515151515151, 'português'], [3.6, 4, 0, 0.6666666666666666, 2, 3.1666666666666665, 0.6086956521739131, 'espanhol'], [3.5714285714285716, 6, 1, 0.75, 3, 3.375, 0.40625, 'inglês'], [4.4, 4, 3, 0.6666666666666666, 2, 4.0, 0.6428571428571429, 'espanhol'], [6.333333333333333, 2, 0, 0.5, 1, 5.0, 0.5, 'espanhol'], [5.0, 4, 1, 0.6666666

```
666666666, 1, 4.333333333333333, 0.5666666666666667, 'português'], [4.25, 3, 1, 0.6,
2, 3.6, 0.7619047619047619, 'inglês'], [3.5714285714285716, 6, 1, 0.75, 3, 3.375, 0.
53125, 'inglês'], [5.2, 4, 2, 0.6666666666666666, 1, 4.5, 0.5806451612903226, 'portu
guês'], [7.0, 3, 0, 0.6, 1, 5.8, 0.5, 'espanhol'], [4.75, 3, 3, 0.6, 2, 4.0, 0.69565
21739130435, 'português'], [5.8, 4, 0, 0.6666666666666666, 3, 5.0, 0.470588235294117
64, 'espanhol'], [3.0, 2, 3, 0.5, 1, 2.75, 0.9230769230769231, 'espanhol'], [4.33333
3333333333, 5, 1, 0.7142857142857143, 3, 4.0, 0.5, 'inglês'], [3.857142857142857, 7,
1, 0.7777777777777778, 3, 3.2222222222222223, 0.45714285714285713, 'inglês'], [4.333
333333333333, 3, 4, 0.6, 1, 3.2, 0.8333333333333334, 'espanhol'], [4.375, 7, 0, 0.77
77777777777778, 3, 4.0, 0.4418604651162791, 'português'], [4.5, 5, 1, 0.714285714285
7143, 2, 4.0, 0.48484848484848486, 'português'], [8.2, 4, 2, 0.6666666666666666, 0,
7.0, 0.43478260869565216, 'português'], [4.5, 5, 0, 0.7142857142857143, 3, 4.0, 0.51
51515151515151, 'português'], [5.8, 4, 0, 0.6666666666666666, 3, 5.0, 0.470588235294
11764, 'português'], [7.8, 4, 3, 0.6666666666666666, 1, 6.833333333333333, 0.4, 'esp
anhol'], [4.6, 4, 1, 0.6666666666666666, 2, 4.0, 0.6428571428571429, 'espanhol'],
[3.5, 5, 0, 0.7142857142857143, 3, 3.142857142857143, 0.5185185185185185, 'inglês'],
[3.6, 4, 3, 0.6666666666666666, 2, 3.3333333333333335, 0.5833333333333334, 'espanho
l'], [4.2, 4, 0, 0.6666666666666666, 2, 3.6666666666666665, 0.5384615384615384, 'ing
lês'], [5.166666666666667, 5, 1, 0.7142857142857143, 2, 4.571428571428571, 0.4864864
864864865, 'português'], [5.857142857142857, 6, 1, 0.75, 5, 5.25, 0.375, 'inglês'],
[3.8, 4, 1, 0.6666666666666666, 2, 3.5, 0.5833333333333334, 'inglês'], [3.0, 3, 1,
0.6, 3, 2.6, 0.625, 'inglês'], [7.333333333333333, 2, 0, 0.5, 0, 5.75, 0.56, 'espanh
ol'], [4.0, 3, 0, 0.6, 2, 3.4, 0.65, 'inglês'], [4.75, 3, 0, 0.6, 1, 4.0, 0.60869565
21739131, 'espanhol'], [3.8333333333333335, 5, 1, 0.7142857142857143, 3, 3.428571428
5714284, 0.5517241379310345, 'espanhol'], [5.5, 3, 1, 0.6, 2, 4.6, 0.5, 'portuguê
s'], [3.6, 4, 0, 0.6666666666666666, 2, 3.1666666666666665, 0.6086956521739131, 'ing
lês'], [4.2, 4, 1, 0.6666666666666666, 2, 3.6666666666666665, 0.5384615384615384, 'p
ortuguês']]
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\matheus_arazao\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\matheus_arazao\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Error loading corpus: Package 'corpus' not found in index
```

Out[ ]:

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| **0** | 5.166667 | 5 | 1 | 0.714286 | 3 | 4.571429 | 0.432432 | português |
| **1** | 5.333333 | 5 | 0 | 0.714286 | 2 | 4.714286 | 0.473684 | português |
| **2** | 4.142857 | 6 | 1 | 0.750000 | 3 | 3.750000 | 0.416667 | português |
| **3** | 4.250000 | 3 | 3 | 0.600000 | 2 | 3.600000 | 0.714286 | português |
| **4** | 4.833333 | 5 | 0 | 0.714286 | 3 | 4.285714 | 0.428571 | espanhol |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... |
| **87** | 4.750000 | 3 | 0 | 0.600000 | 1 | 4.000000 | 0.608696 | espanhol |
| **88** | 3.833333 | 5 | 1 | 0.714286 | 3 | 3.428571 | 0.551724 | espanhol |
| **89** | 5.500000 | 3 | 1 | 0.600000 | 2 | 4.600000 | 0.500000 | português |
| **90** | 3.600000 | 4 | 0 | 0.666667 | 2 | 3.166667 | 0.608696 | inglês |
| **91** | 4.200000 | 4 | 1 | 0.666667 | 2 | 3.666667 | 0.538462 | português |

92 rows × 8 columns

# Treinando o modelo com SVM
# Separando o conjunto de treinamento do conjunto de testes

In [ ]:
```
from sklearn.model_selection import train_test_split
import numpy as np

#from sklearn.metrics import confusion_matrix

vet = np.array(padroes)
classes = vet[:,-1]         # classes = [p[-1] for p in padroes]
#print(classes)
padroes_sem_classe = vet[:,0:-1]
#print(padroes_sem_classe)
X_train, X_test, y_train, y_test = train_test_split(padroes_sem_classe, classes, te
```

Com os conjuntos separados, podemos "treinar" o modelo usando a SVM.

In [ ]:
```
from sklearn import svm
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report

treinador = svm.SVC()  #algoritmo escolhido
modelo = treinador.fit(X_train, y_train)

#
# score com os dados de treinamento
```

```python
acuracia = modelo.score(X_train, y_train)
print("Acurácia nos dados de treinamento: {:.2f}%".format(acuracia * 100))

#
# melhor avaliar com a matriz de confusão
y_pred = modelo.predict(X_train)
cm = confusion_matrix(y_train, y_pred)
print(cm)
print(classification_report(y_train, y_pred))

#
# com dados de teste que não foram usados no treinamento
print('métricas mais confiáveis')
y_pred2 = modelo.predict(X_test)
cm = confusion_matrix(y_test, y_pred2)
print(cm)
print(classification_report(y_test, y_pred2))
```

```
Acurácia nos dados de treinamento: 71.01%
[[12  5  5]
 [ 3 18  2]
 [ 4  1 19]]
              precision    recall  f1-score   support

    espanhol       0.63      0.55      0.59        22
      inglês       0.75      0.78      0.77        23
   português       0.73      0.79      0.76        24

    accuracy                           0.71        69
   macro avg       0.70      0.71      0.70        69
weighted avg       0.71      0.71      0.71        69

métricas mais confiáveis
[[1 1 6]
 [0 5 2]
 [1 2 5]]
              precision    recall  f1-score   support

    espanhol       0.50      0.12      0.20         8
      inglês       0.62      0.71      0.67         7
   português       0.38      0.62      0.48         8

    accuracy                           0.48        23
   macro avg       0.50      0.49      0.45        23
weighted avg       0.50      0.48      0.44        23
```