

# Experimentos com o Text Mining Jurimétrico

Rafael B. Stern

March 31, 2014

Para rodar esse experimento, precisei de algumas bibliotecas padronizadas do R para text mining aliadas ao patenteado webcrawler da ABJ.

```
require(jurimetria)
require(SnowballC)
library(stringr)
require(tm)
```

O primeiro passo foi criar um pequeno banco de dados de algumas sentenças e salvá-las em forma de arquivo de texto. Para tal, fiz as seguintes adições ao crawler:

```
baixa_pagina <- function(search, pag_num, mydir) {
  cjpg <- crawler_cjpg(livre = search, pag = pag_num)
  sentencas <- cjpg[["txt"]]

  # Salvar sentencas em texto
  setwd(mydir)
  sentenca.write <- function(ii) {
    write(sentencas[ii], file = paste(search, as.character(10 * (pag_num -
      1) + ii), sep = ""))
  }
  sapply(1:length(sentencas), sentenca.write)
}

# Busca no sistema e-saj por sentencas ligadas a 'termo' e, em seguida,
# salva cada sentenca encontrada ate a pagina 'total_de_paginas' em um
# arquivo de texto separado em 'mydir'.
crawler <- function(termo, total_de_paginas, mydir) {
  baixa_termo <- function(num) {
    baixa_pagina(termo, num, mydir)
  }
  sapply(1:total_de_paginas, baixa_termo)
}
```

Escolhi para procurar pelos termos ‘usucapiao’ e ‘investigacao de paternidade’. Minha conjectura era que esses termos tinham boas propriedades:

1. Os termos são suficientemente precisos para que os resultados da busca encontrasse apenas um tipo de ação.
2. Os tipos de ação encontrados para cada termo são suficientemente diferentes. Mais explicitamente, as palavras que compõem as sentenças em cada tipo de ação são diferentes.

```
mydir <- "/home/rbstern/Desktop/papers/jurimetria/text mining/data"
crawler("investigacao de paternidade", 2, mydir)
crawler("usucapiao", 2, mydir)
```

Finalmente, comecei a limpar o banco de dados. Conforme mostrarei mais tarde, essa funcao de limpeza ainda nao está fazendo um bom serviço. Basicamente, removi algumas palavras frequentes e pouco informativas (sentença, processo, etc...) e realizei stemming.

```
# Carregar o Corpus
raw_docs <- Corpus(DirSource(mydir), readerControl = list(language = "portuguese"))

# Limpeza da sentenca
limpa_sentenca <- function(text) {
  text <- tolower(text)

  meses <- c("janeiro", "fevereiro", "abril", "maio", "junho", "julho", "agosto",
            "setembro", "outubro", "novembro", "dezembro")
  banned.words <- c(stopwords("portuguese"), meses, "sentena", "processo")
  text <- sentenca.raw <- removeWords(text, banned.words)

  text <- gsub("[:digit:][:punct:]", "", text)
  text <- stripWhitespace(text)

  text <- gsub("documento assinado digitalmente termos lei conforme impresso margem direita",
            "", text)
  text <- stemDocument(text, language = "portuguese")
  return(text)
}
clean_docs <- tm_map(raw_docs, limpa_sentenca)
clean_tm <- DocumentTermMatrix(clean_docs)
```

Realizando uma limpeza mínima, tentei prosseguir para a análise. Para tal, tentei usar o pacote “topicmodels”, que implementa uma função de clustering de textos bayesiana.

```

library(topicmodels)
test <- LDA(clean_tm, 2)
terms(test)

## Topic 1 Topic 2
## "fls" "imvel"

topics(test)

## investigacao de paternidade1 investigacao de paternidade10
## 2 1
## investigacao de paternidade11 investigacao de paternidade12
## 1 1
## investigacao de paternidade13 investigacao de paternidade14
## 1 1
## investigacao de paternidade15 investigacao de paternidade16
## 2 1
## investigacao de paternidade17 investigacao de paternidade18
## 1 1
## investigacao de paternidade19 investigacao de paternidade2
## 1 1
## investigacao de paternidade20 investigacao de paternidade3
## 1 1
## investigacao de paternidade4 investigacao de paternidade5
## 1 1
## investigacao de paternidade6 investigacao de paternidade7
## 1 1
## investigacao de paternidade8 investigacao de paternidade9
## 1 1
## usucapiao1 usucapiao10
## 2 2
## usucapiao11 usucapiao12
## 2 1
## usucapiao13 usucapiao14
## 1 1
## usucapiao15 usucapiao16
## 2 2
## usucapiao17 usucapiao18
## 1 2
## usucapiao19 usucapiao2
## 2 1
## usucapiao20 usucapiao3
## 2 1
## usucapiao4 usucapiao5
## 1 2
## usucapiao6 usucapiao7

```

##	2	1
##	usucapiao8	usucapiao9
##	1	1

O resultado parece confirmar que existem muitos termos frequentes em sentenças judiciais mas que são pouco úteis para a análise e classificação de sentenças. Por exemplo, “contratos”, “direito”, “fls”, “juiz”, .... Rodando esse arquivos diversas vezes, também notei que o algoritmo é instável para o tamanho de amostra selecionado.

Para melhorar os resultados decidi:

1. Aumentar o tamanho da amostra para 100 de cada tipo de caso.
2. Retirar mais termos que são genéricos e frequentemente usados em sentenças.

Aumentar a amostra é a parte fácil ...

```
mydir <- "/home/rbstern/Desktop/papers/jurimetria/text mining/data2"
crawler("investigacao de paternidade", 10, mydir)
crawler("usucapiao", 10, mydir)
```

A seguir, após algumas iterações de testes, decidi usar a seguinte função de limpeza. A principal modificação são as “banned.words.direito” ao final.

```
raw_docs <- Corpus(DirSource(mydir), readerControl = list(language = "portuguese"))
limpa_sentenca <- function(text) {
  text <- tolower(text)

  meses <- c("janeiro", "fevereiro", "maro", "abril", "maio", "junho", "julho",
    "agosto", "setembro", "outubro", "novembro", "dezembro")
  banned.words <- c(stopwords("portuguese"), meses)
  text <- sentenca.raw <- removeWords(text, banned.words)

  text <- gsub("[[:digit:]][:punct:]]", "", text)
  text <- stripWhitespace(text)

  text <- gsub("documento assinado digitalmente termos lei conforme impresso margem direita",
    "", text)
  text <- stemDocument(text, language = "portuguese")

  banned.words.direito <- c("a", "ajuiz", "aleg", "art", "artig", "autor",
    "cdig", "direit", "fls", "inicial", "julg", "juiz", "juz", "justic",
    "lei", "peti", "process", "ru", "sentenc", "vist", "vot")
  text <- sentenca.raw <- removeWords(text, banned.words.direito)
  text <- stripWhitespace(text)
```

```

    return(text)
}
clean_docs <- tm_map(raw_docs, limpa_sentenca)
clean_tm <- DocumentTermMatrix(clean_docs)

```

Para encontrar as palavras banidas do direito, eu fiz sucessivas iterações procurando por termos comuns nos documentos. Para tal, utilizei o comando<sup>1</sup>:

```
inspect(removeSparseTerms(clean_tm, 0.5))
```

Finalmente, podemos testar novamente o clustering ...

```

test <- LDA(clean_tm, 2)
terms(test)

## Topic 1 Topic 2
## "patern" "imvel"

topics(test)

##   investigacao de paternidade1  investigacao de paternidade10
##                               2                               1
## investigacao de paternidade100  investigacao de paternidade11
##                               1                               1
## investigacao de paternidade12  investigacao de paternidade13
##                               1                               1
## investigacao de paternidade14  investigacao de paternidade15
##                               1                               1
## investigacao de paternidade16  investigacao de paternidade17
##                               1                               1
## investigacao de paternidade18  investigacao de paternidade19
##                               1                               1
##   investigacao de paternidade2  investigacao de paternidade20
##                               1                               1
## investigacao de paternidade21  investigacao de paternidade22
##                               1                               2
## investigacao de paternidade23  investigacao de paternidade24
##                               2                               2
## investigacao de paternidade25  investigacao de paternidade26
##                               1                               1
## investigacao de paternidade27  investigacao de paternidade28
##                               1                               1
## investigacao de paternidade29  investigacao de paternidade3
##                               1                               2

```

---

<sup>1</sup>Para não ocupar muito espaço, decidi não exibir os resultados.

##	investigacao de paternidade30	investigacao de paternidade31
##	2	1
##	investigacao de paternidade32	investigacao de paternidade33
##	1	1
##	investigacao de paternidade34	investigacao de paternidade35
##	1	1
##	investigacao de paternidade36	investigacao de paternidade37
##	1	1
##	investigacao de paternidade38	investigacao de paternidade39
##	2	1
##	investigacao de paternidade4	investigacao de paternidade40
##	2	1
##	investigacao de paternidade41	investigacao de paternidade42
##	1	1
##	investigacao de paternidade43	investigacao de paternidade44
##	1	1
##	investigacao de paternidade45	investigacao de paternidade46
##	1	1
##	investigacao de paternidade47	investigacao de paternidade48
##	2	1
##	investigacao de paternidade49	investigacao de paternidade5
##	1	1
##	investigacao de paternidade50	investigacao de paternidade51
##	1	1
##	investigacao de paternidade52	investigacao de paternidade53
##	1	2
##	investigacao de paternidade54	investigacao de paternidade55
##	2	1
##	investigacao de paternidade56	investigacao de paternidade57
##	1	1
##	investigacao de paternidade58	investigacao de paternidade59
##	1	1
##	investigacao de paternidade6	investigacao de paternidade60
##	2	1
##	investigacao de paternidade61	investigacao de paternidade62
##	1	1
##	investigacao de paternidade63	investigacao de paternidade64
##	1	1
##	investigacao de paternidade65	investigacao de paternidade66
##	1	1
##	investigacao de paternidade67	investigacao de paternidade68
##	1	2
##	investigacao de paternidade69	investigacao de paternidade7
##	1	1
##	investigacao de paternidade70	investigacao de paternidade71

##	1	2
##	investigacao de paternidade72	investigacao de paternidade73
##	1	1
##	investigacao de paternidade74	investigacao de paternidade75
##	1	1
##	investigacao de paternidade76	investigacao de paternidade77
##	1	1
##	investigacao de paternidade78	investigacao de paternidade79
##	1	1
##	investigacao de paternidade8	investigacao de paternidade80
##	1	1
##	investigacao de paternidade81	investigacao de paternidade82
##	2	1
##	investigacao de paternidade83	investigacao de paternidade84
##	1	1
##	investigacao de paternidade85	investigacao de paternidade86
##	1	1
##	investigacao de paternidade87	investigacao de paternidade88
##	2	1
##	investigacao de paternidade89	investigacao de paternidade9
##	1	2
##	investigacao de paternidade90	investigacao de paternidade91
##	1	1
##	investigacao de paternidade92	investigacao de paternidade93
##	1	1
##	investigacao de paternidade94	investigacao de paternidade95
##	2	1
##	investigacao de paternidade96	investigacao de paternidade97
##	1	1
##	investigacao de paternidade98	investigacao de paternidade99
##	1	1
##	usucapiao1	usucapiao10
##	2	2
##	usucapiao100	usucapiao11
##	2	2
##	usucapiao12	usucapiao13
##	2	2
##	usucapiao14	usucapiao15
##	2	2
##	usucapiao16	usucapiao17
##	2	2
##	usucapiao18	usucapiao19
##	2	2
##	usucapiao2	usucapiao20
##	2	2

##	usucapiao21	usucapiao22
##	2	2
##	usucapiao23	usucapiao24
##	2	2
##	usucapiao25	usucapiao26
##	2	2
##	usucapiao27	usucapiao28
##	2	2
##	usucapiao29	usucapiao3
##	2	1
##	usucapiao30	usucapiao31
##	2	2
##	usucapiao32	usucapiao33
##	2	2
##	usucapiao34	usucapiao35
##	2	2
##	usucapiao36	usucapiao37
##	2	2
##	usucapiao38	usucapiao39
##	2	2
##	usucapiao4	usucapiao40
##	2	2
##	usucapiao41	usucapiao42
##	2	2
##	usucapiao43	usucapiao44
##	2	2
##	usucapiao45	usucapiao46
##	2	2
##	usucapiao47	usucapiao48
##	2	2
##	usucapiao49	usucapiao5
##	2	2
##	usucapiao50	usucapiao51
##	2	2
##	usucapiao52	usucapiao53
##	2	2
##	usucapiao54	usucapiao55
##	2	2
##	usucapiao56	usucapiao57
##	2	2
##	usucapiao58	usucapiao59
##	2	2
##	usucapiao6	usucapiao60
##	2	2
##	usucapiao61	usucapiao62



##	2	2
##	usucapiao63	usucapiao64
##	2	2
##	usucapiao65	usucapiao66
##	2	2
##	usucapiao67	usucapiao68
##	2	2
##	usucapiao69	usucapiao7
##	2	2
##	usucapiao70	usucapiao71
##	2	2
##	usucapiao72	usucapiao73
##	1	2
##	usucapiao74	usucapiao75
##	2	2
##	usucapiao76	usucapiao77
##	2	2
##	usucapiao78	usucapiao79
##	2	2
##	usucapiao8	usucapiao80
##	2	2
##	usucapiao81	usucapiao82
##	2	2
##	usucapiao83	usucapiao84
##	2	1
##	usucapiao85	usucapiao86
##	2	2
##	usucapiao87	usucapiao88
##	2	2
##	usucapiao89	usucapiao9
##	2	2
##	usucapiao90	usucapiao91
##	2	2
##	usucapiao92	usucapiao93
##	2	2
##	usucapiao94	usucapiao95
##	2	2
##	usucapiao96	usucapiao97
##	2	2
##	usucapiao98	usucapiao99
##	2	2

Os resultados ainda são um pouco instáveis. Contudo, na maioria das vezes em que rodo esse código um dos clusters é centrado na palavra “imóvel” e concentra as ações de usucapiao. O outro é centrado na palavra “paternid” ou “aliment”. Os resultados parece mais promissores agora!