1766-2016
RUTGERS
250

Prioritizing and Classifying 8-K Information

Mauricio Codesso
Jamie Freiman

- Downloading Data
- Extract Text from PDFs
- Cleaning Process
- Exploratory Data Analysis
- Model Development

Downloading Data EmpresaNet

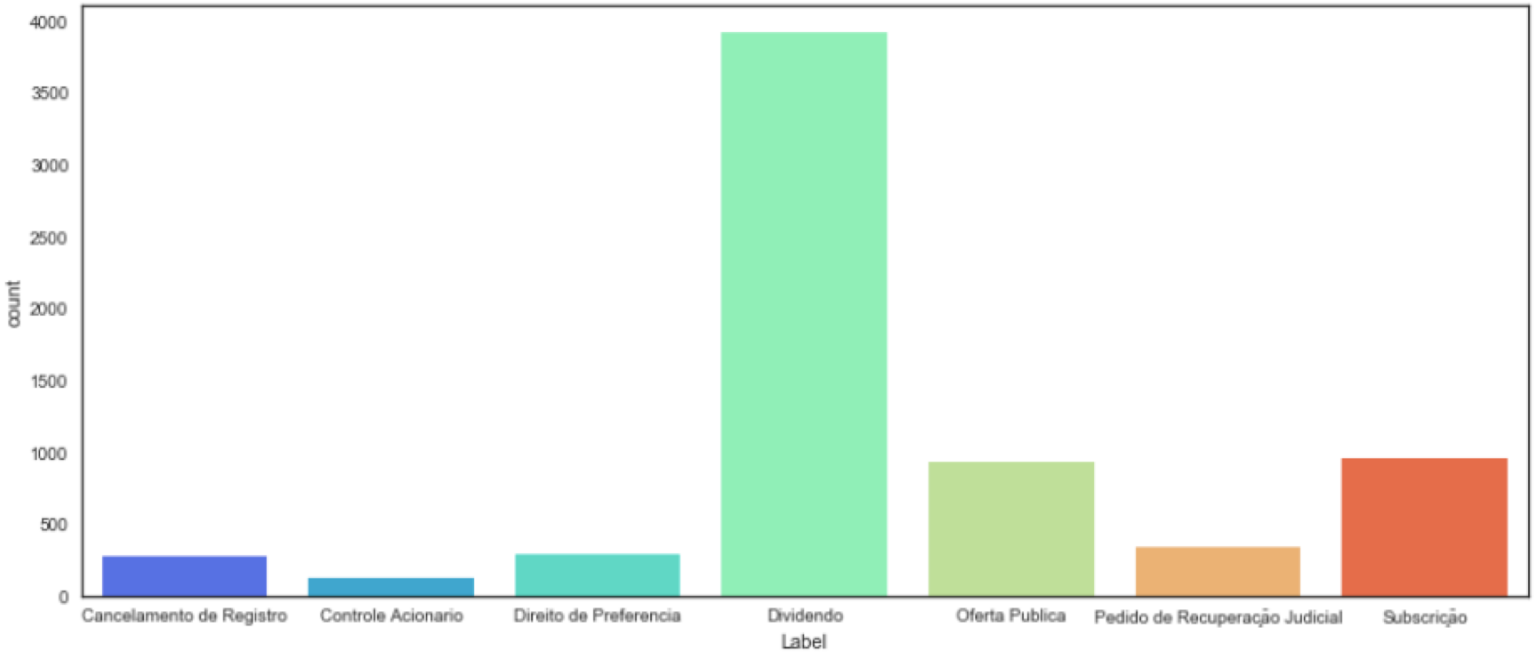
- Download 8500 files from January 2010 to March 2017
- 7 Categories:
  - Cancelamento de Registro / Cancelations
  - Stock Control
  - Direito de Preferência / Right of first refusal
  - Dividends
  - Public Offering
  - Request for Judicial Recovery /
  - bankruptcy
  - Share subscription

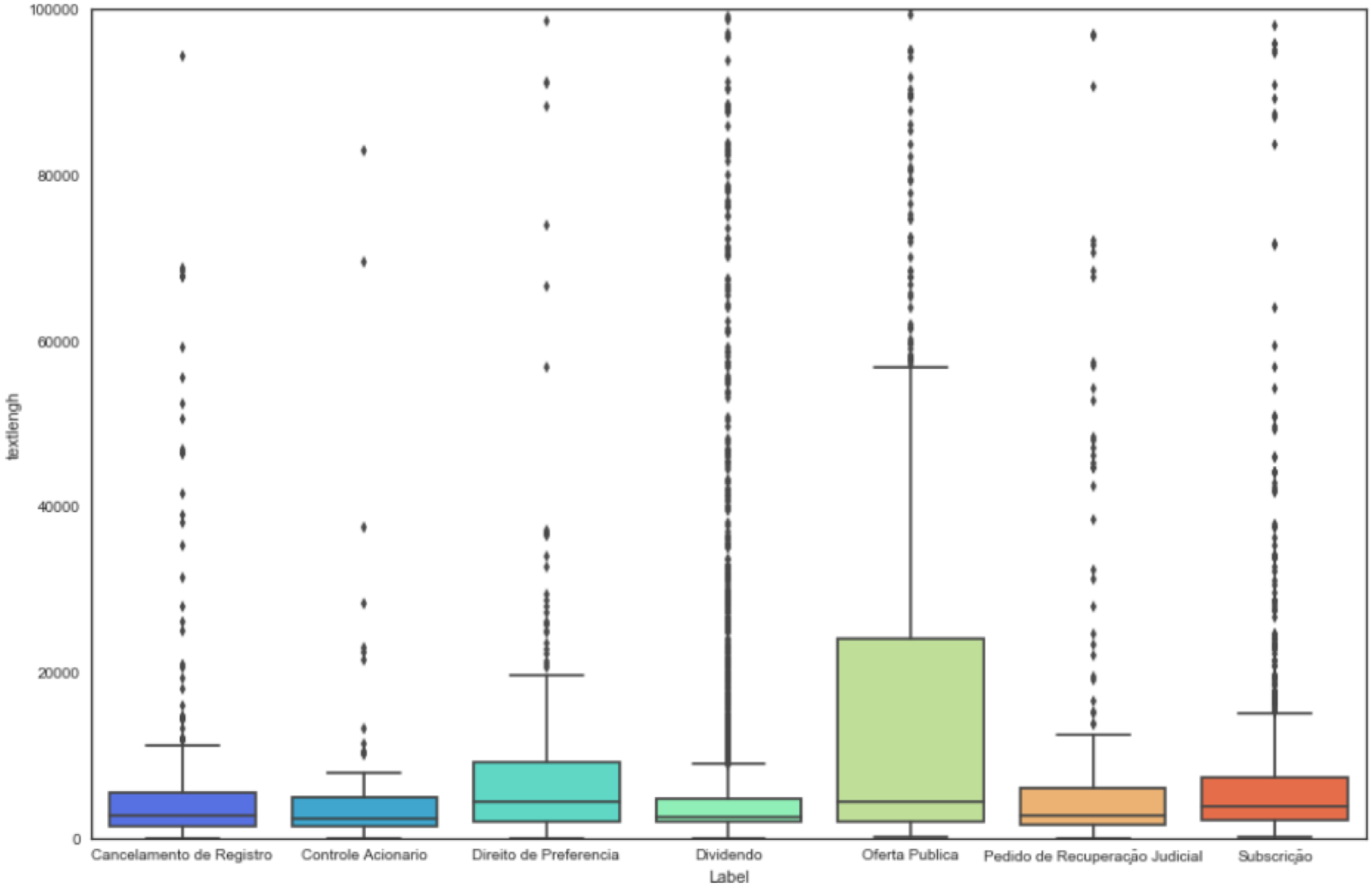
Extract and Cleaning Data

- Development of a script for extracting and processing the texts of the files in PDF
- Deletion of encrypted and corrupted files
- Final files after extraction and cleaning: 6837

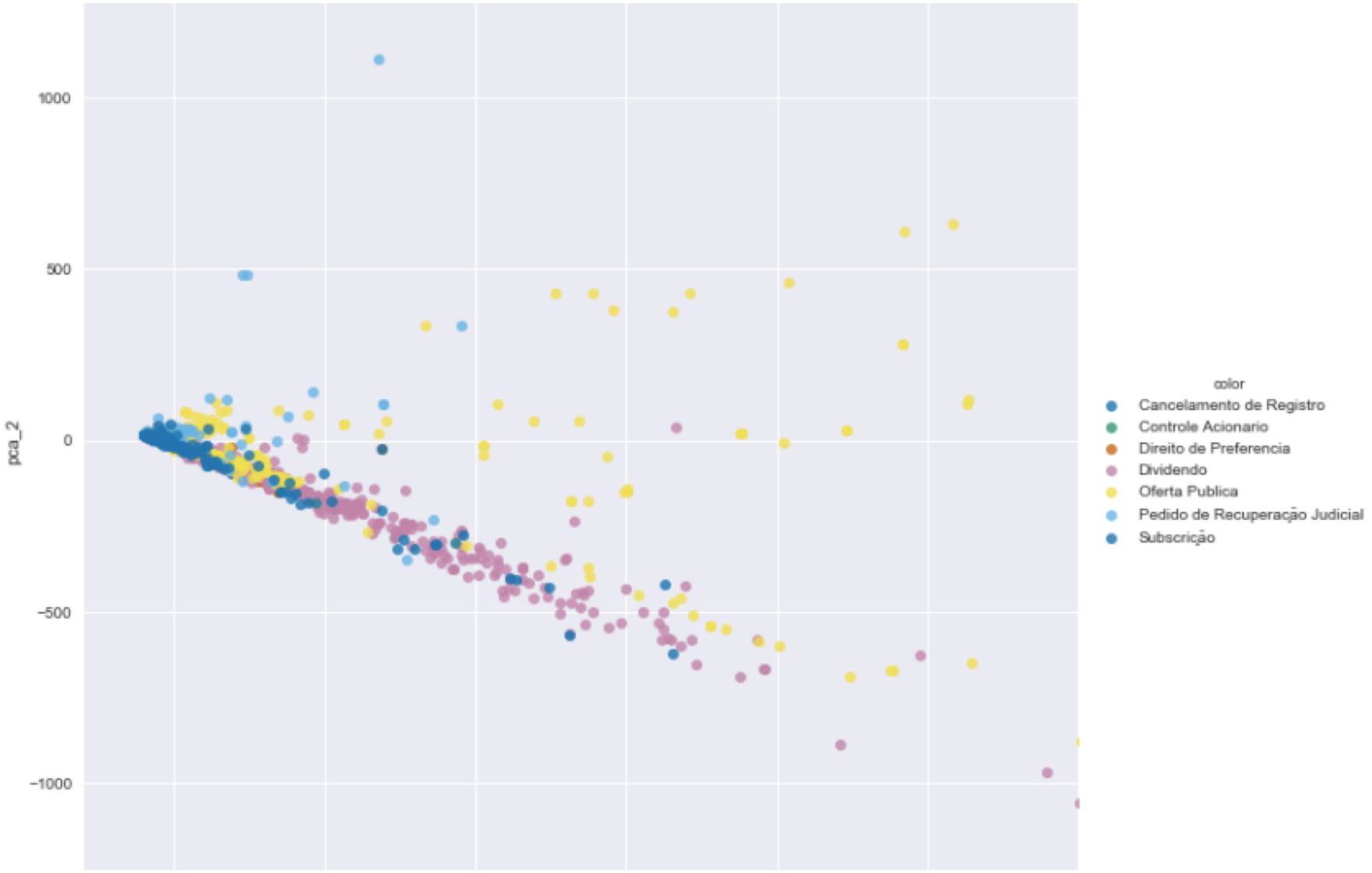
Exploratory Analysis

Categories	Files
Cancelamento de Registro	282
Stock Control	128
Direito de Preferência / Right of first refusal	292
Dividends	3919
Public Offering	928
Request for Judicial Recovery / bankruptcy	333
Share subscription	955




100000
80000
60000
40000
20000
0
textlengh
Cancelamento de Registro
Controle Acionario
Direito de Preferencia
Dividendo
Oferta Publica
Pedido de Recuperação Judicial
Subscrição
Label

PCA Analysis



Cleaning Process

```
def review_to_words( raw_review ):
    # Function to convert a raw review to a string of words
    # The input is a single string (a raw movie review), and
    # the output is a single string
    #
    # 1. Remove HTML
    review_text = BeautifulSoup(raw_review,"lxml").get_text()
    #
    # 2. Remove non-letters
    letters_only = re.sub("[^a-zA-Z]", " ", review_text)
    #
    # 3. Convert to lower case, split into individual words
    words = letters_only.lower().split()
    #
    # 4. In Python, searching a set is much faster than searching
    #   a list, so convert the stop words to a set
    stops = set(stopwords.words("Portuguese"))
    #
    # 5. Remove stop words
    meaningful_words = [w for w in words if not w in stops]
    #
    # 6. Join the words back into one string separated by space,
    # and return the result.
    return( " ".join( meaningful_words ))
```

How to improve the pdfs text extraction?

```
clean_document = review_to_words( X[0] )
print(clean_document)
```

tivit terceiriza processos servi tecnologia s cnpj mf nire edital segunda convoca assembleias gerais extraordin riasficam senhores acionistas tivit terceiriza processos servi ose tecnologia s convocados reunir assembleias geraisextraordin rias companhia ser realizadas segunda convoca julho sede social companhia localizada avenidaprefeito carlos ferreira lopes n cidade mogi cruzes estado s paulo s hs hs seguintes ordens dia assembleia geral extraordin ria ser realizada s hs aprova proposta plano op compra es emiss oda companhia termos minuta disponibilizada acionistas aconseq ente altera artigos item h estatuto social dacompanhia b aumento valor global remunera administradores assembleia geral extraordin ria ser realizada s hs aprova sa companhia novo mercado segmento especial denegocia bm fbovespa b sele acionistas n controladores companhia detentores dea es circula companhia conforme definido regulamento novomercado dentre seguintes institui es especializadas sercontratada elabora termos legisla regulamenta oaplic veis laudo avalia es companhia fins darealiza acionista dethalas empreendimentos participa es s oferta p blica sa novo mercado cancelamento

Dictionaries

```
# Take a look at the words in the vocabulary
vocab = vectorizer.get_feature_names()
print(vocab[:100])
```

```
['aa', 'aaa', 'aadministra', 'aapl', 'aarrttiiggoo', 'ab', 'abaixo', 'abalancete', 'abandono', 'abas
tecimento', 'abate', 'abatimentos', 'abbett', 'abbott', 'abc', 'aberdeen', 'aberta', 'abertas', 'abe
rto', 'abertos', 'abertura', 'abilio', 'abl', 'abn', 'abono', 'abordagem', 'abordando', 'abordar',
'abott', 'about', 'abr', 'abrang', 'abrange', 'abrangem', 'abrangendo', 'abrangente', 'abrangentes',
'abrangentespatrim', 'abranger', 'abrangidas', 'abrasce', 'abreu', 'abril', 'abrilpar', 'abrir', 'ab
riu', 'absoluta', 'absor', 'absorver', 'absorvido', 'absorvidos', 'absten', 'abster', 'abstiveram',
'abu', 'abyara', 'ac', 'acabados', 'acabou', 'acad', 'acadian', 'acarretando', 'acarretar', 'acarret
aria', 'acatar', 'acautelar', 'acc', 'accenture', 'acciona', 'acciones', 'accionistas', 'accountin
g', 'aceit', 'aceita', 'aceitado', 'aceitantes', 'aceitar', 'aceitas', 'aceite', 'aceito', 'aceito
s', 'acelera', 'acelerar', 'acerca', 'acervo', 'acess', 'acessada', 'acessado', 'acessar', 'acesse',
'acesso', 'acessos', 'achada', 'acidente', 'acidentes', 'acima', 'acion', 'acionista', 'acionistas',
'acionistasque']
```

Models

- We tested four machine learning models

• Naïve Bayes	84% accuracy
• Randon Forest	88% accuracy
• **XGBBoost**	**91%** accuracy
• Deep Learning (DNN)	89% accuracy

XGBoost

	precision	recall	f1-score	support
Cancelamento de Registro	0.81	0.54	0.65	56
Stock Control	0.87	0.74	0.80	27
Direito de Preferência / Right of first refusal	0.84	0.42	0.56	65
Dividends	0.96	1.00	0.98	783
Public Offering	0.79	0.88	0.83	190
Request for Judicial Recovery / bankruptcy	1.00	0.88	0.93	72
Share subscription	0.81	0.87	0.84	175
avg / total	0.91	0.91	0.90	1368

```
[[ 30   0   0   2  23   0   1]
 [  0  20   0   1   6   0   0]
 [  0   0  27   3   7   0  28]
 [  0   0   0 780   2   0   1]
 [  7   1   1   9 168   0   4]
 [  0   0   0   7   1  63   1]
 [  0   2   4  11   6   0 152]]
```

Next Steps

- The results were obtained only with public data available at EmpresaNet
- Metadata analysis
- Financial Data indexes
- Knowledge Engineering
- Topic Modeling – LDA
- Word Embeddings

- Readability Score – *FOG*
- Sentimental Analysis
- Passive Voice for formal documents.

1766-2016
RUTGERS
250

Thank you!