**IA para reconhecimento de leis fiscais**

Antonio Marcos Ferreira

Alberto Messias da Costa Souza

**Resumo**

Este artigo tem como objetivo criar um Data Wirehouse contendo dados das leis governamentais, com o intuito de fácilitar analizes de dados e aplicação de algoritimos de inteligência artificial e machine learning. Trata-se apenas de um esperimento, no qual notamos várias particularidades relacionadas as leis bem como os benefícios de ter estes dados estruturados. O foco principal é o processo de ETL e as dificuldades encontradas na preparação dos dados. Já a aplicação de aprendizado de máquina e processamento de linguagem natural são objetivos secundários que são sitados aqui mas devem fazer parte de outro artigo.

**Palavras-chave**: data wirehouse de leis, preparar leis fiscais para inteligencia artificial, processo de ETL para leis governamentais

**ABSTRACT**

This article aims to create a Data Wirehouse containing data from government laws, in order to facilitate data analysis and application of artificial intelligence and machine learning algorithms. It is just a hope, in which we noticed several peculiarities related to the laws as well as the benefits of having these data extracts. The main focus is the ETL process and the difficulties encountered in preparing the data. The application of machine learning and natural language processing, on the other hand, are secondary objectives that are listed here but should be part of another article.

**Keywords**:   
wirehouse data laws, preparing tax laws for artificial intelligence, ETL process for government laws

# iNTRODUÇÃO

O thema **IA para Reconhecimento de Leis Fiscais** foi a primeira resposta a pergunta “qual é o tema do artigo?”, pois havia uma utopia em conseguir calcular uma nota fiscal simplesmente olhando para as leis governamentais disponíveis usando Inteligência Artificial. Porém assim que foi iniciado o projeto, as pesquisas mostraram um escopo gigantesco, um verdadeiro mar de informações, tornando esta idéia praticamente impossível de ser contemplada. Principalmente em curto praso e com poucas pessoas envolvidas. Além disso os envolvidos não são especialistas em leis fiscais ou coisa do tipo. Mas isso tudo levantou outra questão, como podemos facilitar, ou preparar o caminho para em um futuro breve aplicarmos Inteligência Artificial neste contexto? Com base nesta questão foi possível dar sequencia a este projeto de estudo.

De modo geral este estudo tem por objetivo ajudar a interpretação das leis por meio de inteligencia artificial, com isto teriamos avanços em diversos setores da sociedade, como por exemplo a área júridica, sivil, criminal e também na área tributária. Tais avanços seriam simplesmente o fato de ter respostas mais rápidas e previsões mais certeiras oriundas de uma fonte confiável. Tais previsões são muito bem vindas em empresas no que diz respeito a compliance fiscal, cálculos tributários e geração de cenários de compra e venda sendo usados como ferramenta competiva.

A interpretação da lei ainda não é um problema, porque a mesma é interpretada e aplicada a maioria das vezes com justiça. Já a interpretação da lei de maneira rápida isso sim é um problema, em muitos casos não é possível ter respostas rápidas a respeito de uma decisão. Outro fato, mas que ainda não é um problema atual, é a dificuldade de fazer previsões com boa velocidade e veracidade, considerando as leis cabiveis a uma determinada ação. Algumas empresas de compliance fiscal oferecem ferramentas que ajudam muito nesta questão, porém exige grande mão de obra para chegar a um resultado final bem sucedido, além de que é passível de erros humanos. Basta que um texto na lei seja mau interpretado ou inserido erroneamente em um banco de dados.

Com todo o poder computacional existente hoje, e que provavelmente será maior no futuro parece ser propício que um máquina possa ser treinada e gerar respostas rápidas, principalmente levando em conta que as leis estarão todas disponíveis no memso local, pacíveis de consulta e cruzamento. Isto não é para substituir o homem e seu intelecto mas sim para servir como ferramenta de auxícilo em tomadas de decisão. Criar um DW com estas informações é necessário para garantir a qualidade e veracidade dos dados. Feito isso a aplicação de machine learning fica eficiente. Contudo sabemos que as leis são inumeras e um dos desafios é como fazer o proceço de ETL. Várias perguntas surgem, como por exemplo qual será a fonte dos dados? Como será a primeira carga? E as atualizações? Quais tecnologias serão utilizadas? É realmente um caso para Big Data?

# **2. o Trabalho**

Para inicio de estudo foi avaliado se este trabalho era necessário e se de fato vai agregar valor a sociedade. Sabemos que as leis são inumeras, existem a muito tempo, e vão continuar existindo. Devido a quantidade, fica claro a necessidade de arquivar isto de forma mais simples, passível por exemplo de um cruzamento de dados. Para embasar isto, basta ver o livro “Patria Amada” escrito por Vinícios Leôncio contendo 41 mil páginas somente a respeito da legislação tributária.

Mesmo colocando em um livro todas as leis ficaria difícil encontrálas e muito mais difícil ainda cruzar informações.

Para tira prova, precisamos entender se estes dados contemplas todos os requisitos necessários para clássificá-lo como um projeto de Big Data. Conforme observam Zikopoulos e Eaton (2011), a IBM caracteriza e define a tecnologia de Big Data por três aspectos conjuntos: pelo seu **volume**, sua **velocidade** e a **variedade** de dados. Além destes três V inicialmente propostos para caracterizar a tecnologia de Big Data, foi adicionado o **valor** e a **veracidade** dos dados.

Então fazemos um checklist:

1. Volume

A quantidade de leis existentes provam que o volume é grande, principalmente se levar em conta todos os níveis federal, estadual e municipal.

1. Velocidade

A cada dia provavelmente uma ou mais normas, leis ou emendas são criadas. Isto é necessário pois muitos casos precisam de normas e requerimentos para serem aplicados.

1. Variedade

As leis são variadas, por exemplo a nível federal, temos: Leis Complementares, Leis Delegadas, Leis do Império, Leis Ordinárias, Decretos, Decreto não numerado, Decreto - Lei, Decreto do Poder Legislativo, Decreto do Conselho de Ministros, Decreto do Império, Medidas Provisórias, Códigos e outros isso pode ser visto no site do governo: <http://www.planalto.gov.br/>

1. Valor

O principal valor agregado é o tempo de resposta para consultas e cruzamento de dados. Com tanta variedade de leis, decretos, emendas fica complicado procurar um texto que ajude em um processo seja uma ajuda incremental ou anulativa.

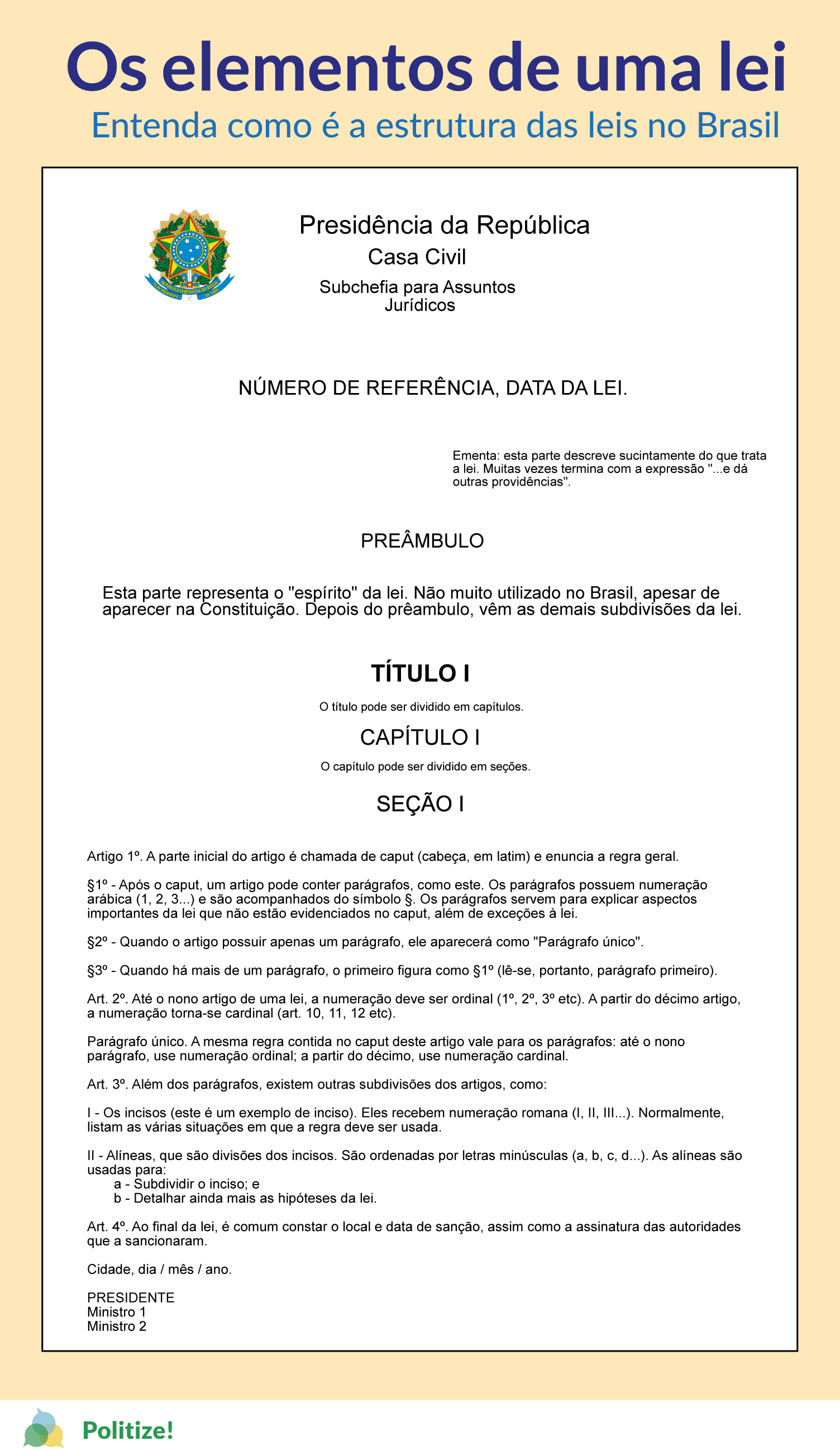
1. Veracidade

As leis por sí só são documentos tidos como válidos, o importante aqui é a fonte original dos dados. Sabemos que estes documentos precisão ser publicados para chegar ao conhecimento da população. Assim temos como principal fonte de dados, os sites publicos responsáveis pelas publicação das leis.

Com tantos documentos para serem analizados, precisamos filtrar alguns casos específicos, queremos neste momento provar o conceito fazendo testes. Precisamos também definir quais tecnologias serão utilizadas no projeto pois hoje existem diversos meios para lidar com Big Data, projetos de ETL e BI.

Para este teste será utilizado apenas os Decretos, Leis Complementares e Lei-Ordinarias, todos extraidos do site do governo “[planalto.gov.br](http://www.planalto.gov.br/)” respeitando a Portaria 1492, <http://www.planalto.gov.br/ccivil_03/Portaria/P1492-11-ccivil.htm>.

Depois de definir a origem dos dados e quais informações vamos lidar, precisavamos entender como é formada uma lei, saber se existe algum tipo de padrão nos ajuda a preparar o processo de extração dos dados. Para entender sobre a composição das leis, alguns sites forma consultados, então seguimos o padrão:

  
Figure 1: Imagem do site Politize! representando a estrutura de um documento de lei.

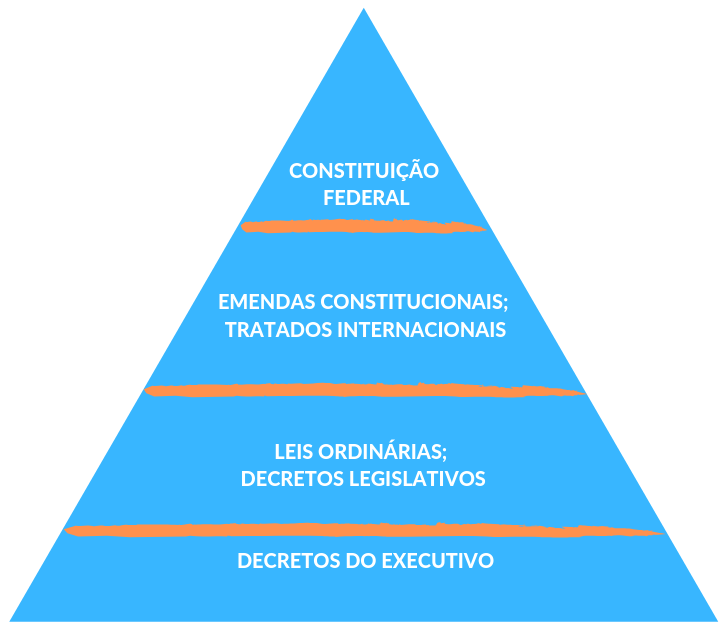
**A escolha da tecnologia utilizada para fazer o processo de ETL.**

Foi escolhido a linguagem de programação **Python** para criação dos scripts de scraping, crawling, tratativa, manipulação e carga dos dados. Python é muito utilizado em ciencia de dados, possui muitas libs que facilitam a implementação deste tipo de projeto.

Para armazenamento dos dados está sendo utilizado estrutura de pastas simples para salvar os arquivos baixados da internet, e o banco de dados NoSQL **MongoDB**. A escolha deste banco de dados é simplesmente pelo fato do projeto ser um teste e não sabiamos exatamente se iriamos ter muitas surpresas a respeito do esquema de dados. Ficar atualizando esquema de dados tomaria muito tempo do projeto, neste caso a definição do esquema não deveria ser um problema para nós.

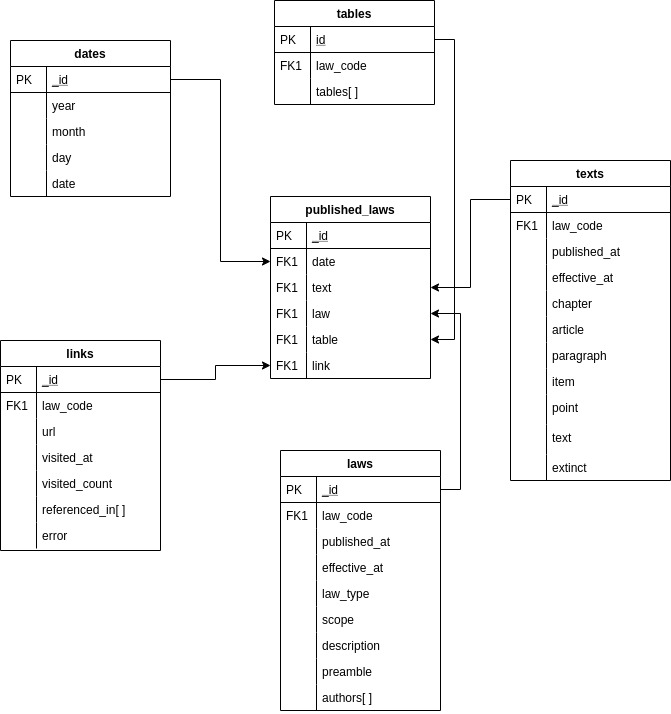
O projeto foi criado pensando na forma mais simples de execução, por este motivo ignoramos vários outros recursos que poderiam ser utilizados para melhorar a performance. Como por exemplo poderiamos usar o Hadoop para salvar as páginas baixadas, ao invés de simplesmente salvar em uma pasta normal do sistema. Poderiamos também escolher uma infra estrutura na nuvem, como por exemplo a AWS. Usando o recurso AWS Lambyda podemos criar serviços para baixar as páginas e salvá-las no S3, criando um lago de dados, depois utilizando o AWS Athena os dados poderiam ser mapeados e preparados para IA. No entanto o foco aqui é simplesmente experimental, vendo desta forma achamos melhor não envolver custos de infra ou muitas tecnologias agregadas. O importante aqui é ver o que é possível fazer apenas com estes poucos recursos, este projeto pode ser executado por uma máquina simples conectada a internet. Apenas considerando que deve ser utilizado para efeito de teste e estudo, limitando-se a uma extração reduzida de dados das fontes originais, por exemplo não é necessário baixar 100 mil documetos do governo para testar. Se for usar em produção deve ser reproduzido em uma estrutura mais robusta e sofisticada, requerendo um valor alto de investimento, também deve ser asegurado de forma juridica respeitando a lei LGPD e as politicas de acesso a dados públicos do governo.

Além de entender a estrutura de uma lei, precisamos também saber que existe hierarquia, leis que são sobrepõe outras:

  
Figure 2: Pirâmide com a hierarquia das leis (Simplificada). Fonte: Politize!. www.politize.com.br/decretos-presidenciais/ . Acessado em: 22 mar. 2020.

Com toda esta bagagem, precisavamos entender mais sobre as leis principalmente no que diz respeito a aplicação dela, quando ela entra em vigor e se ela ainda está em vigorando. Certamente esta é uma parte crítica quando se diz respeito as leis, pois analizando um documento legal, percebe-se que em alguns casos a lei está em vigor mas um determinado paragrafo está revogado, e no mesmo texto pode ter paragrafos que ainda não entraram em vigor. Isto nos da o entendimento de que cada parágrafo do texto legal tem uma data de vigor. Sendo assim a data de publicação é uma para todo o texto legal, mas a data de vigor de cada paragrafo pode ser diferente. Com base nisso, foi decidido que o ponto principal da análise dos dados é o paragrafo da lei e não o documento como um todo. Para sabermos se o paragrafo esta em vigor precisamos usar NLP e criar um modelo treinado para descobrir e classificar como em vigor ou fora de vigor. A partir daqui podemos criar nossa estrutura de dados, definindo como será a staging área e o exquema de dados.

Com base nas informações retiradas de uma lei chegamos ao seguinte esquema de dados, sabemos que isto mode ser melhorado mas precisamos ter um início.

  
Figure 3: Esquema de dados utilizado no DW. Banco de dados MongoDB.

Neste esquema todos os nomes estão em minusculo separados por underline, as entidades recebem uma identificação única chamada **\_id** e estão representadas com o formato estrela, onde o fato principal é a **published\_laws** que contem as dimenções *laws*, *texts*, *links*, *tables* e *dates*. Os nomes estão em inglês simplesmente para manter compatibilidade com o código fonte do projeto.

O projeto foi batisado com o nome **ialeis**, a junção da cigla IA de Inteligencia Artificial junto com a palavra LEIS. Todo o projeto é estruturado de forma simples e intuitivo, tendo a seguinte estruturas de pasta:

* laws
  + Contém as páginas das leis baixadas, dentro desta pasta tem a pasta **errors** onde ficam as leis que tiveram erros no processo de ETL. E a pasta **filtered** que contém as leis que forma lidas com sucesso.
  + Esta pasta funciona como uma staging área para pré-processo dos dados. Não está inserida no git.
* laws\_ai
  + Esta pasta contém os principais scripts Python responsáveis pelos processos de limpesa de dados, extração, carga de dados, entre outros. Estam nesta pasta justamente para torna-la um módulo.
* Scripts
  + Esta pasta contém os scripts gatilhos, estes scripts podem ser tranformados em serviços para manter o DW atualizado.
  + A pasta notbooks contém scripts utilizados para testes.
* Imgs
  + Comtém imagens extraidas durante o estudo dos casos.

# 3. DISCUSSÃO/ANÁLISES DOS RESULTADOS

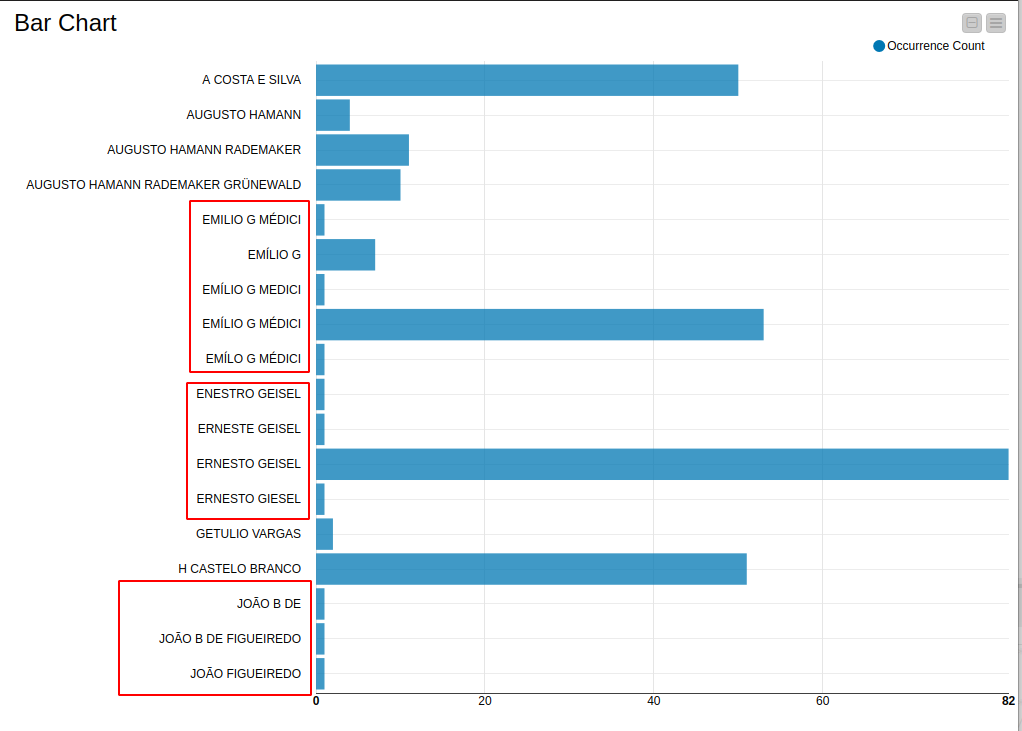
O processo todo de ETL foi realizado com um pequena parte dos dados estimados conforme os demosntrativos abaixo:

|  |  |  |
| --- | --- | --- |
| **Processo** | **Descrição** | **Totais** |
| crawling | Captura de links das leis | 41.425 |
| scraping | Links sorteados e feito o download da página htm da lei para pasta laws. | 4.830 |

*Números referentes ao processo de busca das leis em sites do governo.*

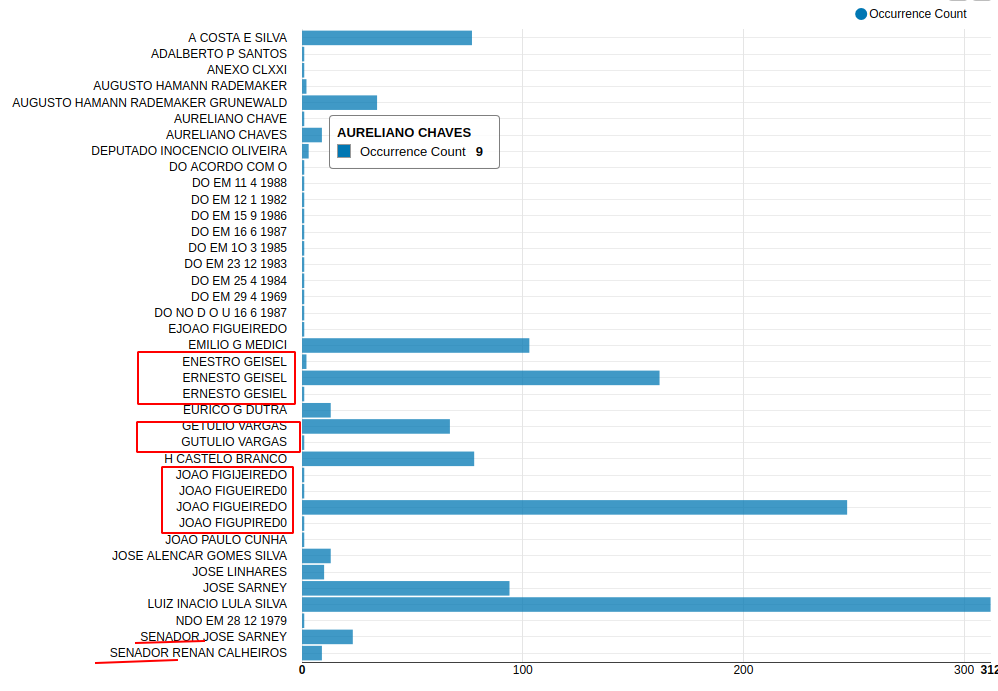
Processo ETL depois de baixado os 4.830 arquivos foram submetidos ao processo de captura, limpesa e carga no banco de dados. Esperavamos que os documentos tivessem um padrão maior, mas tivemos muitos erros neste processo. A primeira execussão retornou valores bem ruins como é mostrado no gráfico abaixo:

Aqui descobrimos que mesmo sendo documentos oficiais, erros humanos grotescos estão presentes, como pode ser observado no nome ERNESTO GEISEL.

  
Figure 4: Gráfico de leis por autores. Gerado no início dos testes.

O script foi ajustado para contornar estes problemas, mas ao trazer mais informações, mais erros parecidos forma encontrados.

O processo de extração e limpesa de dados foi aprimorado, agora além de extrair e limpar precisamos melhorar estes dados.

  
Figure 5: Gráfico de leis por autores. Gerado depois de melhorias

Mesmo com muita melhoria feita no script ainda tivemos muitos casos discrepantes, devido a falta de padrão nos dados recebidos. A tabela abaixo mostra o total de páginas processadas 4.830 separadas entre sucesso e erro. No caso de erro o script deve ser ajustado e a página deve ser reprocessada.

|  |  |  |
| --- | --- | --- |
| **Processo** | **Descrição** | **Totais** |
| Mine ok | Mineração de dados realisada com sucesso | 2.727 |
| Mine erro | Mineração de dados realisada com algum tipo de erro | 2.103 |

*Números de sucesso e erro no processo de mineração de dados.*

Com as leituras consideradas bem sucedidas obtemos os seguintes números:

* 2.539 tabelas extraidas

A maioria delas contém conteúdo que não agrega valor. Certamente cabe aqui uma melhoria no script de extração dos dados para ignorar este tipo de tabela e considerar apenas as tabelas que agregam valores.

* 70.955 paragrafos

sendo estes classificados como capítulo, artigo, paragrafo, item e aliena

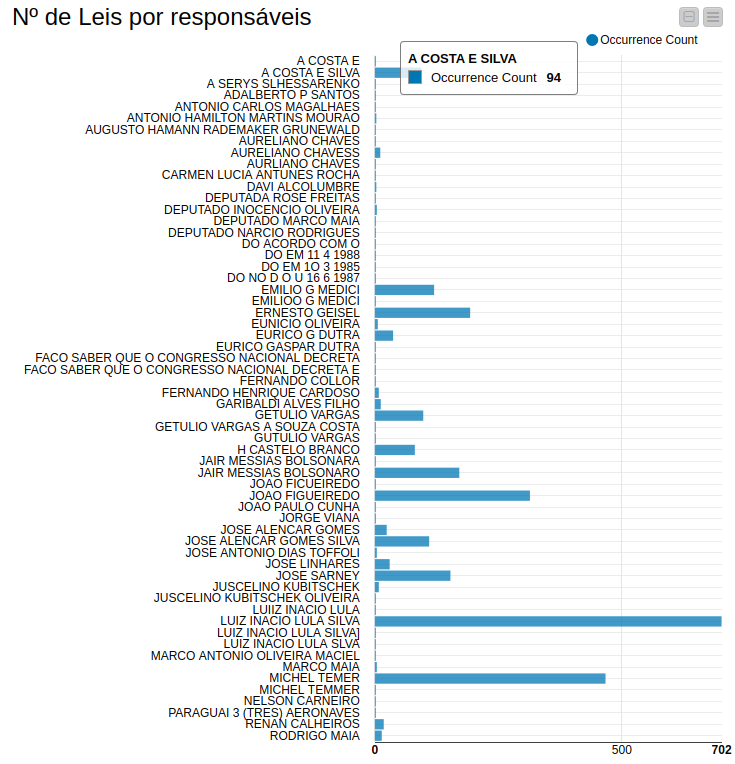
* 1363 datas de publicação diferentes

Depois de melhorias no script tivemos resultados mais satisfatórios, mesmo assim sabemos que o script ainda precisa de melhoria. E isso é esperado, conforme novos erros forem aparecendo ajustes vão sendo feitos e o script vai ficando cada vez melhor.

A estração dos dados do arquivo htm não é algo simples, devido a falta de padrão e diversos problemas encontrados como por exemplo quebras de linhas no meio dos nomes, textos que esperávamos que estivesem em tags de paragrafo porém em alguns arquivos ficam soltos na estrutura do arquivo htm alguns dentro de tabelas. Enfim, é um trabalho tenso.

Foram realisados dois testes tentando pegar todo o texto do documento removendo toda a estrutura de tags html e com o testo limpo aplicar Processamento de Linguagem Natura. Mas não conseguimos bons resultados.

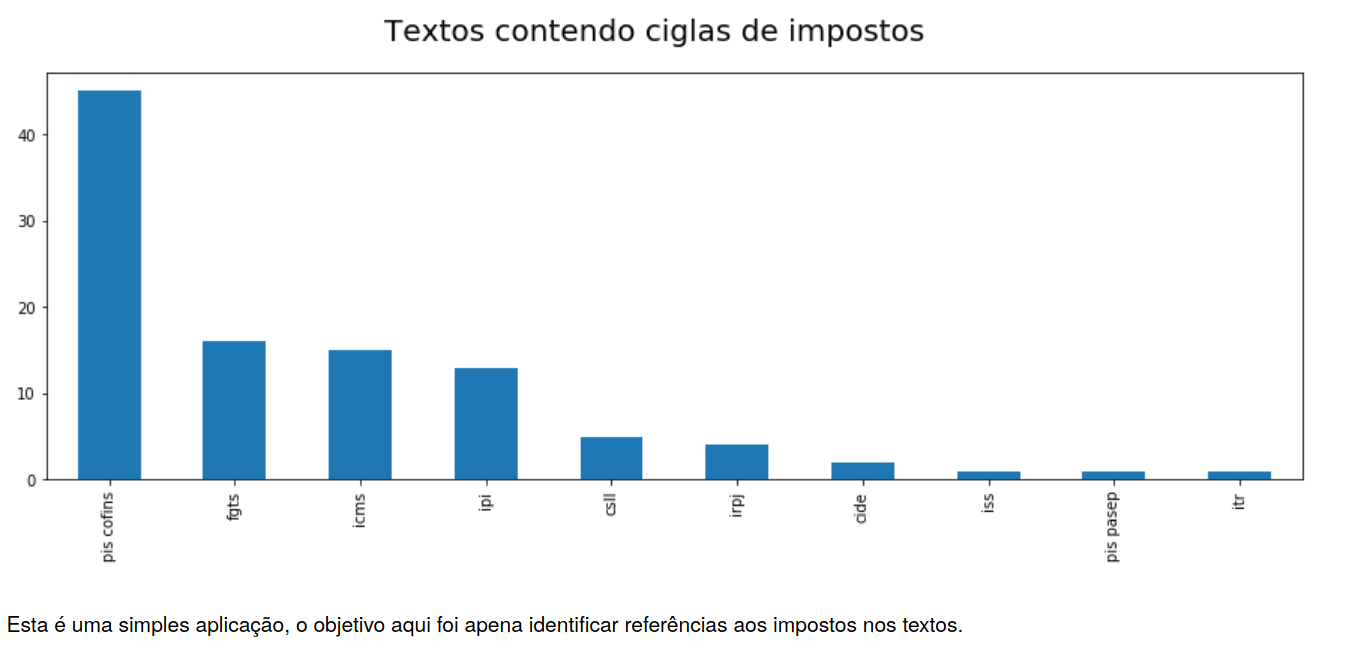
Depois de muita melhoria chegamos na imagem abaixo no que diz respeito a nomes de responsáveis pelas leis.

  
Figure 6: Gráfico de número de leis por autores. Depois da 3º rodada de ajustes no código ETL.

Conseguimos muitas melhorias mas muito ainda deve ser feito. Em uma tentativa tentamos utilizar NLP para identificar os nomes, até conseguimos separar os nomes, mas idendificar os nomes e sobrenomes com precisão foi muito complicado, chegamos até a iniciar um modelo de machine learning para treinar mas vimos que nosso modelo poderia precisar de muitas atualizações. Principalmente porque os nomes contem erros terriveis. Como pode ser visto na imagem “Michel Temmer” ou “Luiz Inacio Lula Slva”

Além da limpesa de dados, precisamos verificar se de fato já temos algum benefício com este banco de dados. O gráfico abaixo mostra a quantidade de textos que contêm sitações diretas ligadas a nomes de impostos:

Para chegar a este reultado acima foi utilizado NLP (Natural Language Processing) nos textos das leis, cada texto passou pelos processos básicos de tokenização e remoção de stop words.

  
Figure 7: Gráfico de número de parágrafos por siglas de impostos

# 4. CONSIDERAÇÕES FINAIS

Valeu a pena todo esforço, embora o script de ETL, ainda precise ser melhorado para reconhecer os documentos que apontaram erros, os ajustes podem ser feitos fácilmente devido a modularização do projeto. Outro ponto é que com os dados organizados, encontrar informações nas leis ficou mais fácil. Muitos estudos podem ser feitos nestes dados, graficos de quantidade de leis por período, autor, tipo entre outros. O objetivo de ter uma fonte de dados que facilite a aplicação de machine learning está iniciado e um pequeno passo foi dado. Melhorias como adicionar reconhecimento de mais tipos de leis e também estender ao nivel estadual e depois municipal. A muito a ser feito.

Todo código utilizado no projeto estão disponíveis no repositório web juntamente com este documento e os demais relacionados. O projeto pode ser baixado e reutilizado por meio de licença MIT.

**5. Referências**

Zikopoulos, paul; eaton, chris. **Understanding big data: analytics for enterprise class hadoop and streaming data**. [s.l.]: MCGRAW-HILL Osborne media, 2011.

CASAS L. L.; **Em protesto a complexidade de leis, advogado lança livro de 7,5 toneladas**. G1. MG. 24 mar. 2014. Disponível em: <http://g1.globo.com/minas-gerais/noticia/2014/03/em-protesto-complexidade-de-leis-advogado-lanca-livro-de-75-toneladas.html>. Acesso em: 22 mar. 2020.

SOARES P. H.; **Como são feitas as leis.** Senado Federal. 22 mar. 2020. Disponível em: <https://www12.senado.leg.br/jovemsenador/home/paginas/como-sao-feitas-as-leis>. Acesso em: 22 mar. 2020.

Glassman G.**;** **A estrutura das leis. Politize**. 01 fev. 2017. Disponível em: <https://www.politize.com.br/estrutura-das-leis-entenda/>. Acessado em 22 mar. 2020.

Jurafsky D. and Martin J. H. **Speech and Language Processing (3rd ed. Draft)**. 16 out. 2019. Disponível em: <https://web.stanford.edu/~jurafsky/slp3/>. Acessado em 22 mar. 2020.

POLI, GABRIEL Antônio; BARROS, CANDIANI Guilherme. **Business intelligence aplicado a um data warehouse**. Franca, 2010. 65 p. Graduação - CIÊNCIA DA COMPUTAÇÃO.

BALLARD Chuck, FARRELL Daniel, GUPTA Amit , MAZUELA Carlos, VOHNIK Carlos; **Dimensional Modeling: In a Business Intelligence Environment**; An IBM Redbooks publication; 2012 Chapter 1. Introduction; Chapter 2. Business Intelligence: The destination.

FERREIRA Marcos, **Repositório do projeto IA Leis.** Tietê – SP. 13 mar. 2020. Disponível em:<https://github.com/antoniomarcosferreira/ialeis>. Acesso em: 22 mar. 2020.