NLP na Prática

APRENDIZADO DE MÁQUINA, DEEP LEARNING E ANÁLISE DE SIMILARIDADE DE TEXTOS APLICADA AO CONTROLE EXTERNO

Prof.Leonardo Vilela

- Mestrando em Informática na Pontifícia Universidade Católica de MG.
- Cientista de Dados no TCE-MG a serviço da Inova Tecnologia.
- Professor da Fundação Getúlio Vargas (CTS DIREITO RIO).
- Professor do Instituto de Gestão e Tecnologia da Informação (IGTI).
- Coordenador e Professor da Escola de Contas Prof. Pedro Aleixo.

Contatos: leonardo.leovilela@gmail.com – (31)975076957.

Luan Lisboa

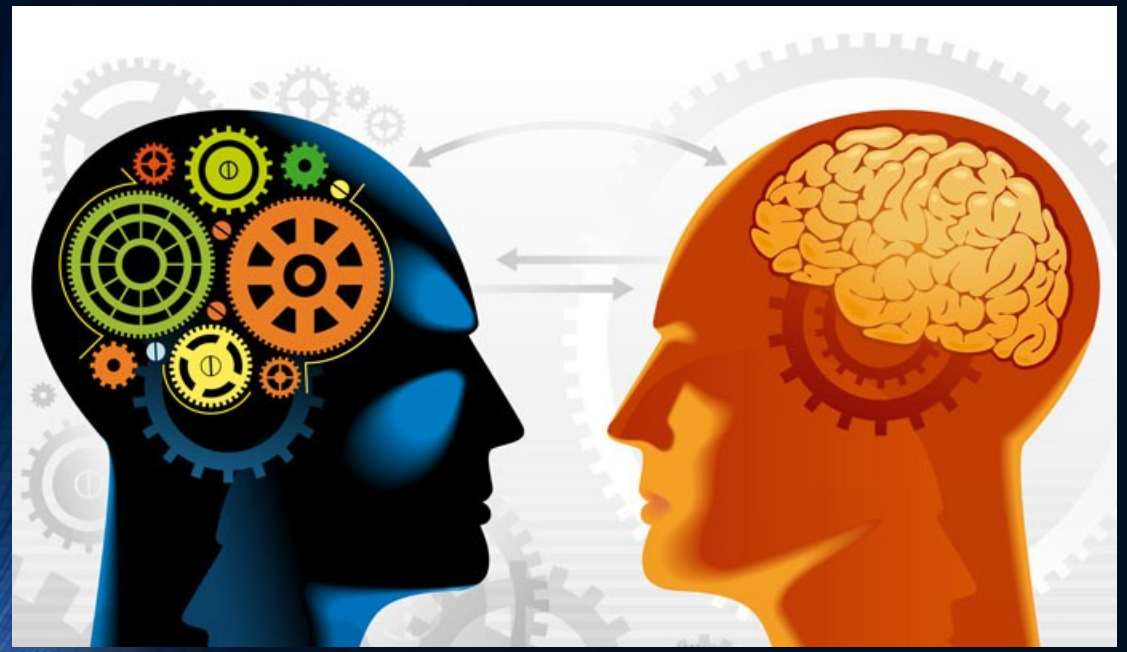
Graduando em Sistemas de Informação pela PUCMinas

Auxiliar no Laboratório de TI do Suricato

https://www.linkedin.com/in/luanslisboa/

Ígor Chagas Marques
Graduando em Engenharia da Computação pelo CEFET-MG
Auxiliar no Laboratório de TI do Suricato
https://www.linkedin.com/in/ígor-chagas-marques-193a81122/

O desafio da Natural Language Processing (NLP)



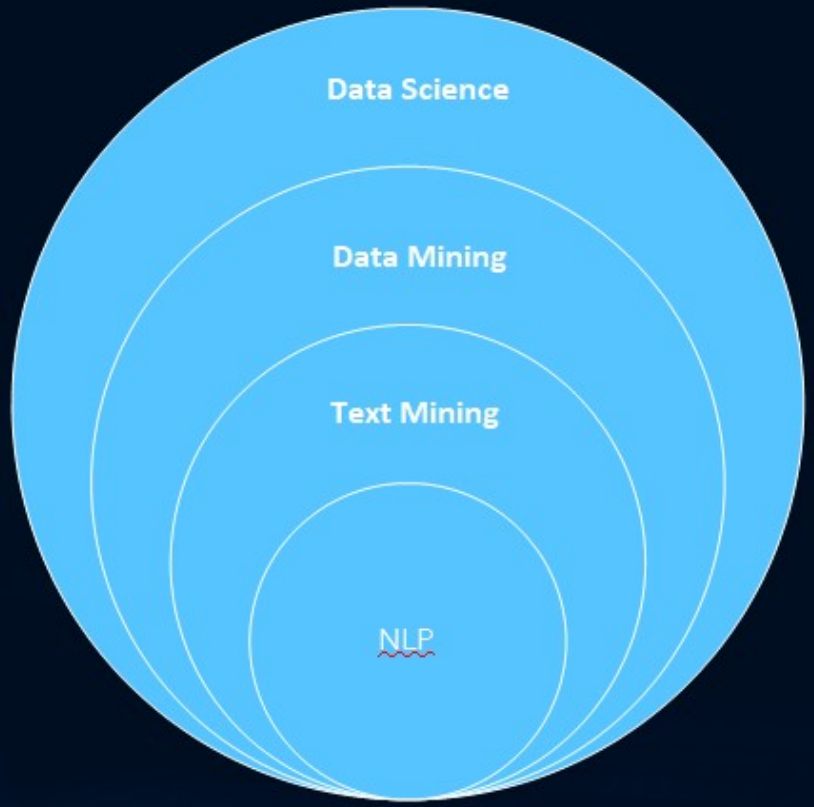
O Sonho de toda a comunidade da inteligência artificial é ter algoritmos capazes de automatizar a leitura e extrair conhecimento de textos vencendo os seguintes desafios:

- Ambiguidade de sentidos : não há regras gerais.
- Idioma fora do padrão : vc, bday, un, und, cx, cax.
- Neologismos.
- Entradas(inputs) com muitas imperfeições.
- Várias representações textuais com único sentido.

Aplicações da NLP no Controle Externo:

- Tratamento de “campos abertos” diversos.
- Recuperação de Informações em Editais e outros documentos para análise de legalidade.
- Agregação de documentos por similaridade.
- Construção de dados categóricos para agregar informações em algoritmos de Data Minning diversos.

Onde está a NLP no contexto do Data Science?



Tarefas comuns de NLP

• Chunking (uso de frases e nomes compostos) • Tag de gramática (Part-of-Speech Tagging). • Reconhecimento de Entidades Nomeadas. • Detecção de SPAM • Thesaurus – Dicionário de antônimos e sinônimos. • Syntactic Parsing – análise sintática de sentenças.	• Análise de Sentimentos. • Recuperação de informação. • Tradução automática. • Geração de Textos • Indexação e resumo automático • Resposta automática a questões (chat bot) • Word Sense Disambiguation – Desambiguação de Nomes.

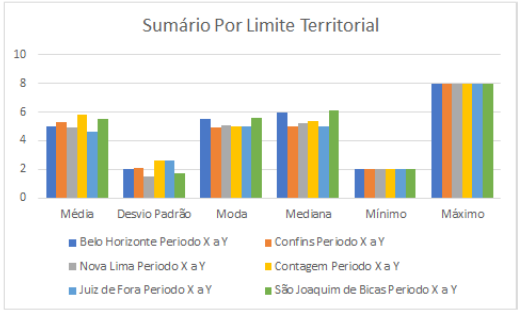
Esta Oficina:

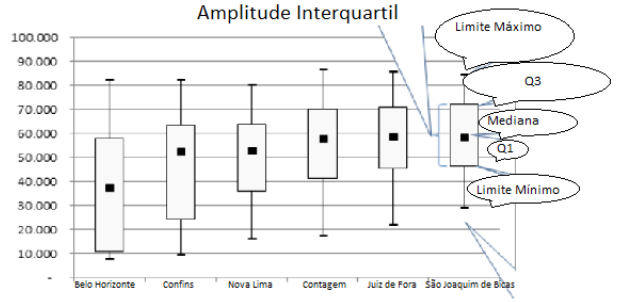
- Clusterização de textos – Banco de Preços baseado na NFE-MG – TCEMG – Suricato.
- Reconhecimento de Entidades Nomeadas através de Aprendizado de Máquina com CRF (ML) e BILSTMCRF - Grupo de Pesquisas da Escola de Contas do TCE-MG.
- Sistemas de Recomendação – Indicação de documentos para auditoria baseado em similaridade com documentos fraudulentos. PUC Minas.

Clusterização de Textos

O BANCO DE PREÇOS BASEADO NA NFE PARA O TCEMG
LABORATÓRIO DE TECNOLOGIA DA INFORMAÇÃO DO SURICATO – FISCALIZAÇÃO INTEGRADA E INTELIGENCIA TCE-MG

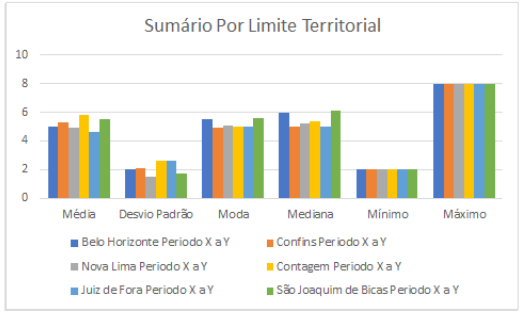
A clusterização de textos pode possibilitar consolidações inicialmente inviáveis em função da utilização de linguagem natural em campos nominativos.

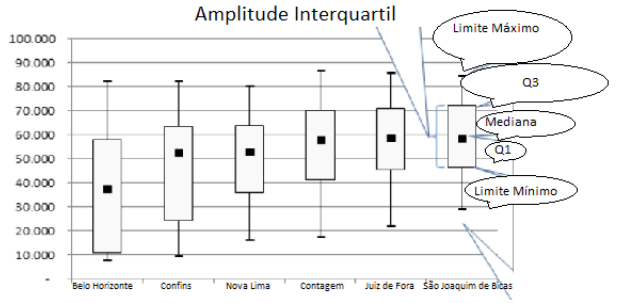




		Sumário						Amplitude Interquartil				
Limite Territorial	Tempo	Média	Desvio Padrão	Moda	Mediana	Mínimo	Máximo	Limite Máximo	Q3	IIQ	Q1	Liminte Mínimo
Belo Horizonte	Periodo X a Y	37597	5	0	39873	0	0	82269	57840	0	10968	7653
Confins	Periodo X a Y	52645		0	51755	0	0	82275	63619	0	24103	9044
Nova Lima	Periodo X a Y	52798	0	0	51897	0	0	80110	63849	0	35792	15721
Contagem	Periodo X a Y	58059	0	0	55717	0	0	86423	700048	0	41168	17137
Juiz de Fora	Periodo X a Y	58800	0	0	56748	0	0	85654	71021	0	45467	21789
São Joaquim de Bicas	Periodo X a Y	58456	0	0	53992	0	0	84300	72200	0	46346	28956

Comportamento do Produto no Estado de Minas Gerais:





		Sumário						Amplitude Interquartil				
Limite Territorial	Tempo	Média	Desvio Padrão	Moda	Mediana	Mínimo	Máximo	Limite Máximo	Q3	IIQ	Q1	Liminte Mínimo
MINAS GERAIS	EXERCICIO - 1	37597	5	0	39873	0	0	82269	57840	0	10968	7653

A Nota Fiscal Eletrônica e as Notas de Empenho: desafio para os órgãos de controle em virtude do preenchimento em linguagem natural:

ÁGUA	MINERAL	INGÁ	COM GÁS	500	ML
ÁGUA	MINERAL	COM GÁS	500	ML	INGÁ
ÁGUA	MINERAL	500	ML	INGÁ	COM GÁS

Algoritmos para Clusterização Profunda

Clusterização: encontrar grupos de valores diferentes que podem ser representações alternativas da mesma coisa, no caso de textos então encontrar diversas escritas diferentes, que significam a mesma coisa.

https://github.com/OpenRefine/OpenRefine/wiki/Clustering-In-Depth

Métodos de Key Collision

Os métodos "Key Collision" baseiam-se na ideia de criar uma representação alternativa de um valor (uma "chave") que contenha apenas a parte mais valiosa ou significativa da string e juntar diferentes cadeias de caracteres com base no fato de que sua chave é a mesma (daí o nome "colisão de chave") :

Impressão digital – Fingerprint

O método de impressão digital é rápido e simples, funciona relativamente bem em vários contextos e é o menos provável de produzir falsos positivos, e é por isso que é o método de primeira escolha.

Algoritmo genérico da impressão digital:

- remover espaço em branco à esquerda e à direita
- mudar todos os caracteres para sua representação em minúscula
- remover todos os caracteres de pontuação e controle
- normalize caracteres ocidentais estendidos para sua representação ASCII (por exemplo, "gödel" → "godel")
- dividir a string em tokens separados por espaços
- classificar os tokens e remover duplicatas
- juntar os tokens juntos
- Agregar itens de mesma impressão digital em um mesmo cluster

Ngran - Fingerprint

- O método de impressão digital n-gram evolui os tokens separados por espaços em branco para o conceito de n-grams, que é um token especial onde n (ou o tamanho em caracteres do token) pode ser especificado pelo usuário.

Algoritmo Genérico para Impresssão Digital N-Gram

- mudar todos os caracteres para sua representação em minúscula
- remover todos os caracteres de pontuação, espaço em branco e caracteres de controle
- obter toda a corda n-gramas
- ordenar os n-gramas e remover duplicados
- juntar os n-gramas classificados juntos novamente
- normalize caracteres ocidentais estendidos para sua representação ASCII
- Agregar itens de mesma impressão digital em um mesmo cluster

Vilela Marques - Fingerprint

- O método de impressão digital Vilela Marques evolui das n-gramas para uma ordenação caracter por caracter dentro do texto, alfabeticamente e numericamente.

Algoritmo Genérico para Impresssão Digital Vilela Marques

- mudar todos os caracteres para sua representação em maiúscula
- remover todos os caracteres de pontuação, espaço em branco e caracteres de controle
- Ordenar os caracteres alfabeticamente e numericamente
- Agregar itens de mesma impressão digital em um mesmo cluster

```
def vilela_marques(s):
 s = s.translate(s.maketrans('', '', '',.;:#$%&@!')).replace(" ","")
 s = "".join(sorted(s))
 return s
```

Métodos de Key Collision - **Impressão digital fonética**

Um terceiro método de codificação usa uma impressão digital fonética (especificamente, o método Metaphone3 para o inglês e o keyer fonético de Cologne para o alemão), que é uma maneira de transformar os tokens na maneira como são pronunciados. Isso é útil para detectar erros que são causados por mal-entendidos das pessoas ou por não saberem a ortografia de uma palavra depois de ouvi-la. A ideia é que palavras sonoras semelhantes acabarão compartilhando a mesma chave e sendo, assim, colocadas no mesmo cluster.

Por exemplo, "Reuben Gevorkiantz" e "Ruben Gevorkyants" compartilham a mesma impressão digital fonética para a pronunciação em inglês, mas possuem impressões digitais diferentes para os métodos de impressão digital regular e n-gram, independentemente do tamanho do n-grama.

Métodos de KNN – K Nearest Neighbor (Vizinho mais Próximo)

Embora os métodos de colisão de chaves sejam muito rápidos, eles tendem a ser muito rigorosos ou muito frouxos, sem meios de ajustar a diferença entre as sequências que estamos dispostos a tolerar.

Os métodos vizinho mais próximo (também conhecido como kNN), por outro lado, fornecem um parâmetro (o raio, ou k) que representa um limite de distância: qualquer par de seqüências de caracteres que esteja mais próximo que um certo valor será agrupado em conjunto.

Infelizmente, dado n strings, existem n (n-1) / 2 pares de strings (e distâncias relativas) que precisam ser comparados e isso acaba sendo muito lento mesmo para conjuntos de dados pequenos (um conjunto de dados com 3.000 linhas requer 4.5 milhões cálculos de distância!)

Distancia de Edição - A distância de Levenshtein (também conhecida como "distância de edição") é provavelmente a função de distância mais simples e intuitiva entre as orações e é muito eficaz devido à sua aplicabilidade geral. Ela mede o número mínimo de 'operações de edição' que são necessárias para alterar uma cadeia na outra.

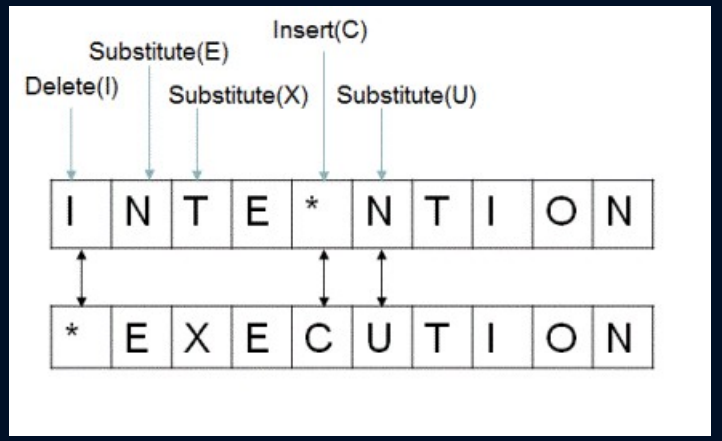
Por exemplo, "Paris" e "paris" têm uma distância de edição de 1, pois a alteração de P em p é a única operação necessária. "New York" e "newyork" tem editar as substituições de distância 3: 2 e 1 remoção. "Al Pacino" e "Albert Pacino" têm uma distância de edição de 4 porque requer 4 inserções.

Distancia de Edição - Implementação genérica em Python

```
import numpy as np

def levenshtein(seq1, seq2):
    size_x = len(seq1) + 1
    size_y = len(seq2) + 1
    matrix = np.zeros ((size_x, size_y))
    for x in xrange(size_x):
        matrix [x, 0] = x
    for y in xrange(size_y):
        matrix [0, y] = y

    for x in xrange(1, size_x):
        for y in xrange(1, size_y):
            if seq1[x-1] == seq2[y-1]:
                matrix [x,y] = min(
                    matrix[x-1, y] + 1,
                    matrix[x-1, y-1],
                    matrix[x, y-1] + 1
                )
            else:
                matrix [x,y] = min(
                    matrix[x-1,y] + 1,
                    matrix[x-1,y-1] + 1,
                    matrix[x,y-1] + 1
                )
    print (matrix)
    return (matrix[size_x - 1, size_y - 1])
```



Métodos de KNN – K Nearest Neighbor (Vizinho mais Próximo)

PPM - Prediction by Partial Matching

Esta distância é uma implementação de um artigo seminal sobre o uso da complexidade de Kolmogorov para estimar 'similaridade' entre cadeias de caracteres e tem sido amplamente aplicado na comparação de cadeias originadas de sequenciamento de DNA.

A ideia é que, como os compressores de texto funcionam estimando o conteúdo da informação de uma string, se duas strings A e B são idênticas, compactar A ou A + B (concatenar as strings) deve resultar em pouca diferença (idealmente, um único bit extra para indicar a presença da informação redundante). Por outro lado, se A e B são muito diferentes, comprimir A e comprimir A + B deve produzir diferenças dramáticas no comprimento.Apresentamos uma versão normalizada do algoritmo, onde a distância entre A e B é dada por:

d (A, B) = comp (A + B) + comp (B + A) / (comp (A + A) + comp (B + B));

Onde comp (s) é o comprimento de bytes da seqüência compactada da string se + é o operador append. Isso é usado para explicar o desvio na otimização dos compressores fornecidos.

Emmbora muitos compressores diferentes possam ser usados, quanto mais próximos da otimalidade de Kolmogorov (ou seja, quanto melhor eles codificam), mais efetivo será o resultado.

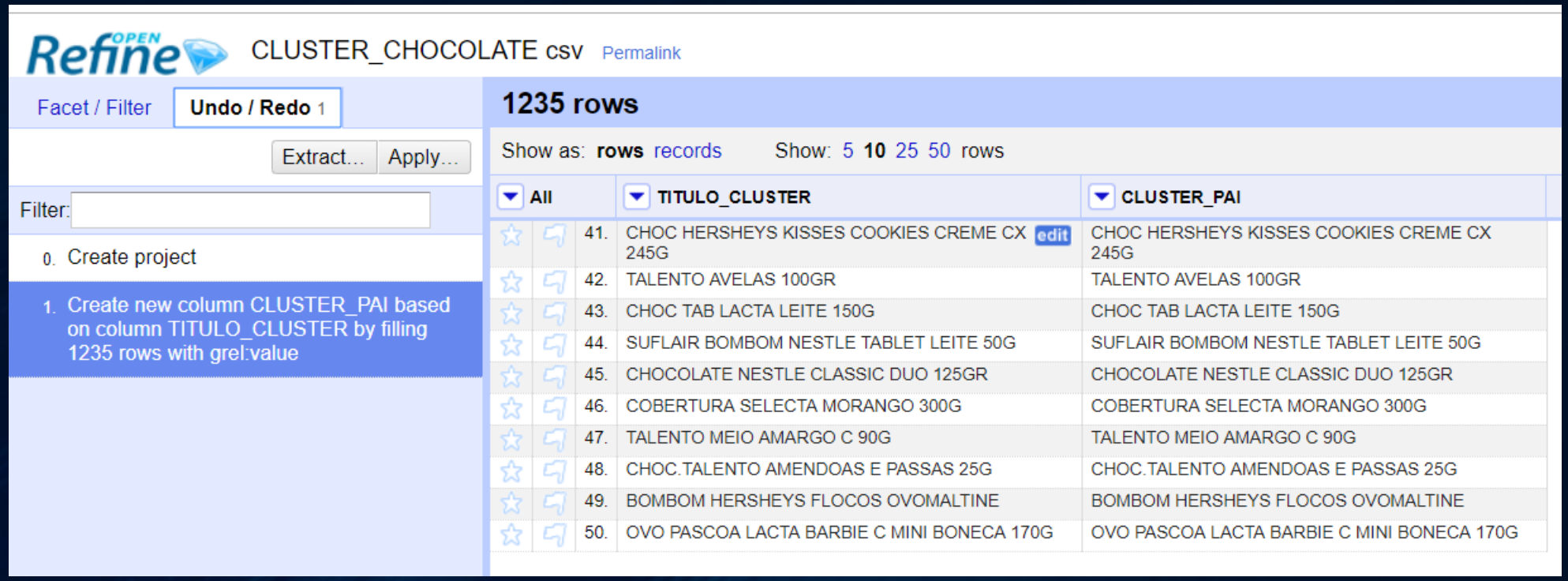
O pulo do gato

The cat's pump.

OpenRefine

A free, open source, powerful tool for working with messy data

O projeto Open Refine foi uma iniciativa google nos anos finais da década passada e início da atual, que para as demandas do Google de NLP foi substituido por soluções melhores ao longo desta década e que foi então disponibilizado para a sociedade na forma de software gratuito.



Clusterização Profunda Com Open Refine.

Pré tratamento de dados:

- Remoção de Stops Words.
- Conversão de CASE.

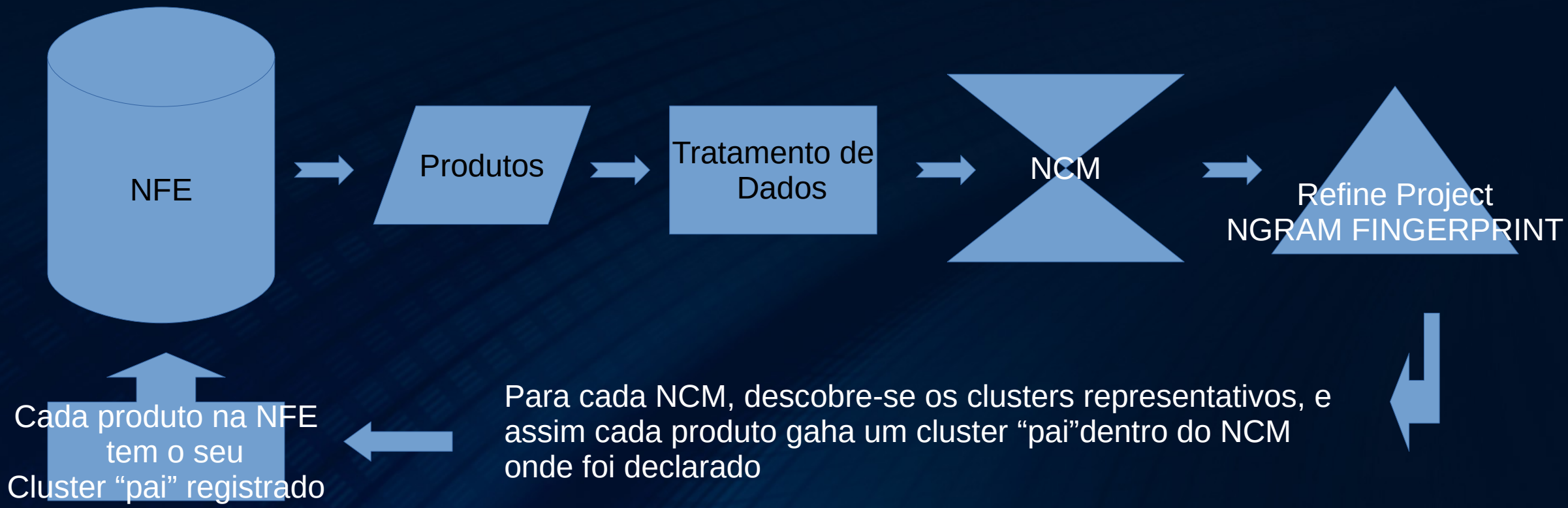
Delimitação do Corpus:

- Você precisa ter uma classificação prévia genérica.
- Não se joga tudo junto com tudo para clusterizar textos.

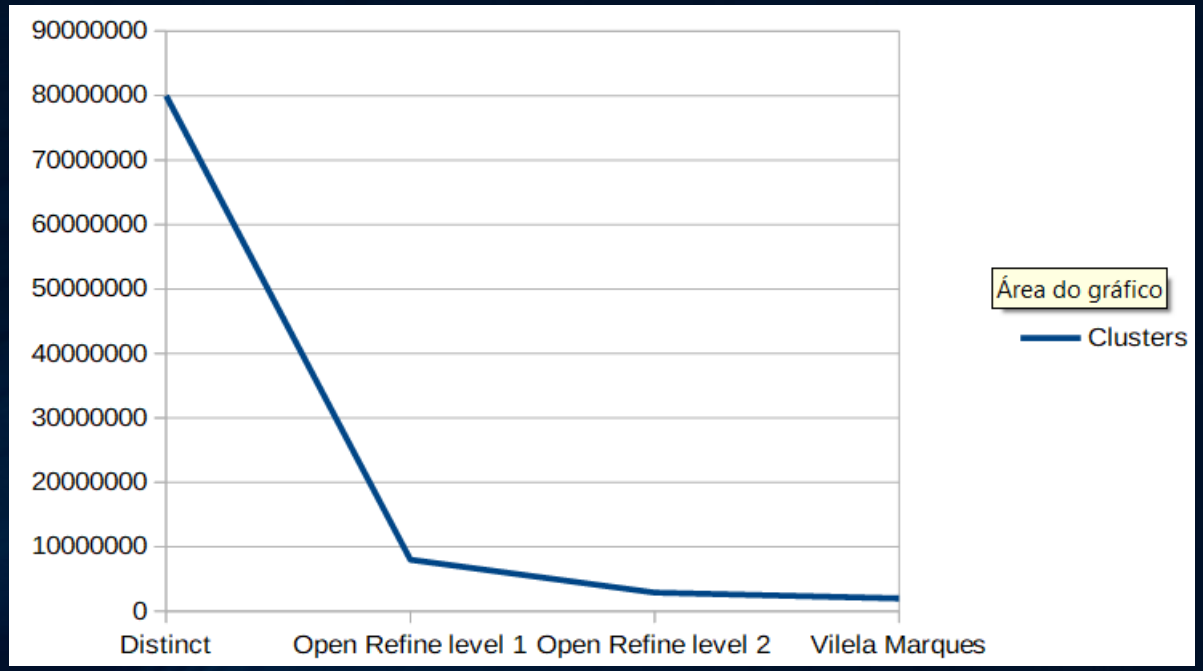
Plataforma:

- Servidor Java Open Refine
- Servidor de Aplicação para processar os posts e gets para o Open Refine (no caso Python)
- Servidor de Banco de Dados que armazena o resultado

Qual foi o Fluxo de Trabalho com o Open Refine?



Resultado da aplicação da Clusterização Profunda : de oitenta milhões a dois milhões em 3 etapas.

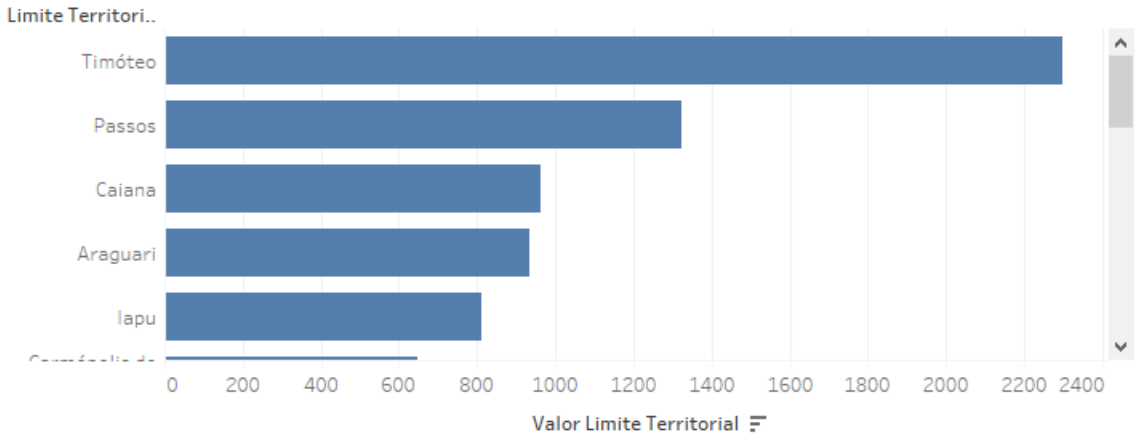


Painel História Análise Mapa Formatar Servidor Janela Ajuda

Exibição inteira

Mostre-me

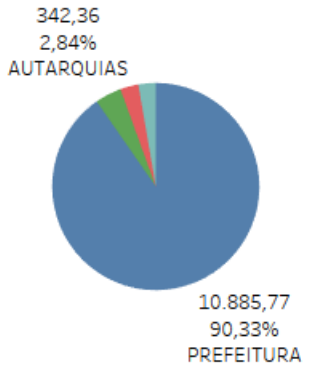
[MUN] 1 Barras: Maiores Limites Territoriais - Município - Nível 1 - 0 a 10% da Mediana - Materialidade - Em R$



Nível
Nível 1 - 0 a 10% da Mediana

Tipo Jurisdicionado
PREFEITURA
FUNDOS
AUTARQUIAS
CÂMARA

[MUN] 1 Pizza: Maiores Tipos de Jurisdicionado - Nível 1 - 0 a 10% da Mediana - Materialidade - Em R$



Valor de Medida
Materialidade - Em ...

Nível
Nível 1 - 0 a 10% da ...

Lote Principal
(Tudo)

Produto
(Tudo)

Selecione Limite Territo..
Município

Limite Territorial
(Tudo)

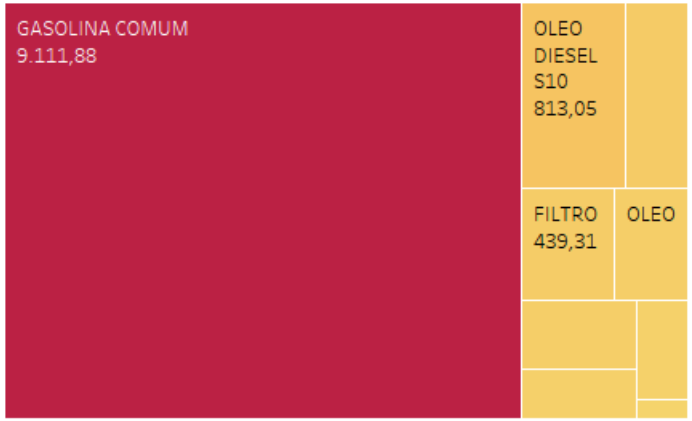
Fornecedor
(Tudo)

Tipo Jurisdicionado
(Tudo)

Jurisdicionado
(Tudo)

Unidade Comercial
(Tudo)

[MUN] 1 Treemap: Maiores Produtos - Nível 1 - 0 a 10% da Mediana - Materialidade - Em R$



[MUN] 1 Bolhas: Maiores Fornecedores - Nível 1 - 0 a 10% da Mediana - Materialidade - Em R$

[MUN] 1 Bolhas: Maiores Jurisdicionados - Nível 1 - 0 a 10% da Mediana - Materialidade - Em R$

Reconhecimento de Entidades Nomeadas através de Aprendizado de Máquina com CRF (ML) e BILSTMCRF -

RECONHECIMENTO DE PADRÕES COM APRENDIZADO SUPERVISIONADO – GRUPO DE PESQUISAS EM NLP DA ESCOLA DE CONTAS PROF. PEDRO ALEIXO DO TCE-MG

O que é Aprendizado de Máquina?

Aprendizado de Máquina é uma área da Inteligência Artificial cujo objetivo é o desenvolvimento de técnicas computacionais sobre o aprendizado bem com a construção de sistemas capazes de adquirir conhecimento de uma forma automática.

Um sistema de aprendizado é um programa de computador que toma decisões baseado em experiências acumuladas através da solução bem sucedida de problemas anteriores

De que se trata o Reconhecimento de Entidades Nomeadas?

Entidades Nomeadas [EN] compreendem-se como termos que apresentam um ou mais designadores rígidos, num determinado texto. Alguns dos tipos mais comuns de entidades são substantivos próprios, tais como nomes de pessoas, organizações e entidades locais; temporais como datas, tempo, dia, ano e mês; entidades numéricas, tais como medições, percentagens e valores monetários. (Daniela Amaral,2013)

O Reconhecimento de Entidades Nomeadas [REN] pode ser entendido com um processamento computacional de aprendizado de máquina cujo objetivo é identificar as entidades nomeadas e executar sua classificação, atribuindo uma categoria semântica para essas entidades:

Produto [PD] – Palavras como ÁGUA MINERAL, CANETA ESFEROGRÁFICA, BOMBOM.
Detalhe de Produto [DP] – Expressões como COM GÁS, PRETA DE PONTA FINA, SORTIDO.
Marca/Fabricante [M] – Palavras como INGÁ, NESTLÉ, BIC, GAROTO.
Detalhe da Marca/Fabricante [DM] – Palavras S/A, CIA, LTDA.
Modelo [M] – Palavras como SERENATA, PILOT, J5.
Detalhe do Modelo [DM] – Expressões como DE AMOR, 0.7 MM, PLUS.
Apresentação Sigla [S] – Siglas como KG, CX, PCT, UN, ML, LT, GL.
Apresentação Número [U] – Números como 1 (FK), 200 (ML).
Apresentação Sigla+Número [SCN] – Expressões com 200ML, 1KG, 20MT.
Lixo [LX] – Qualquer elemento do texto que não possa ser classificado como uma das entidades acima.

As Hierarquias de Aprendizado de Máquina:

Até Então...

Clusterização Profunda:

Aprendizagem não supervisionada

Não houve setup prévio do algoritmo com o objetivo de induzir o processamento de acordo com algum padrão

Reconhecimento de Entidades Nomeadas:

Aprendizagem Supervisionada

É oferecido ao algoritmo uma base chamada “treino”, onde uma amostra da classificação é oferecida ao algoritmo para que o mesmo “aprenda”sobre as classes que ele deve reconhecer.

Conditional Random Fields

Considere uma situação onde um determinado padrão desconhecido, porém vigente, é utilizado para criar uma ordem qualquer de elementos. Suponhamos que esse padrão possa ser descoberto ou replicado. Em NLP e REN consideramos que reconhecer uma entidade nomeada depende da palavra x que representa essa entidade e também das outras palavras que antecedem (y) ou sucedem (z) a palavra x, estabelecendo então um padrão de sucessão e antecedência. Em diversas situações reais das linguagens, este padrão também pode ser observado. No corpus de trabalho, encontramos diversos padrões para elementos de mesmo sentido:

ÁGUA	MINERAL	INGÁ	COM GÁS	500	ML
ÁGUA	MINERAL	COM GÁS	500	ML	INGÁ
ÁGUA	MINERAL	500	ML	INGÁ	COM GÁS

Conditional Random Fields é um modelo matemático probabilístico que tem o objetivo de etiquetar e segmentar dados sequenciais, seguindo um padrão condicional identificado através de Modelos de Markov. O CRF está contido nas técnicas de Aprendizado de Máquina Supervisionado, e pode ser aplicado nos seguintes passos:

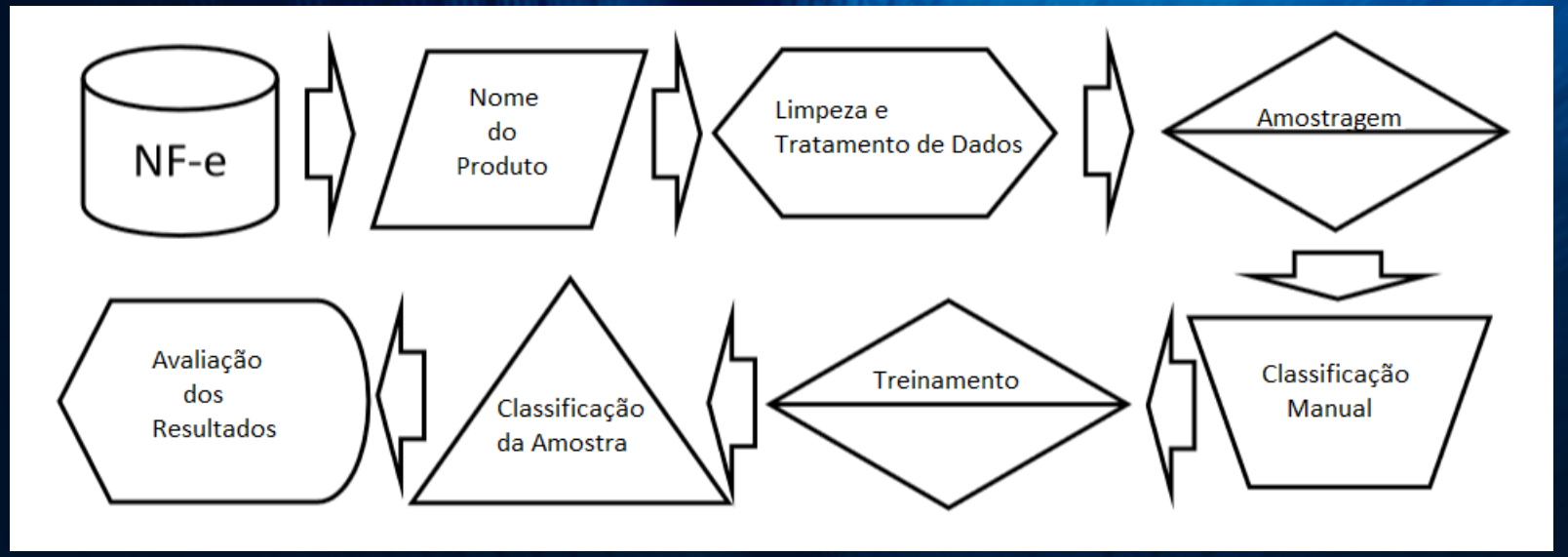
Aprendizado: Dado uma tomada de observações onde as entidades estão classificadas, reconheça os padrões que determinam a classe de uma entidade (palavra) através da observação da sequência de caracteres que a compõe e sua relação com outras entidades (palavras).

Inferência: Utiliza os padrões reconhecidos para classificar novas entidades em uma outra tomada de observações onde as entidades não estão classificadas.

Para realizar o aprendizado, o modelo CRF possibilita a configuração de features que vão direcionar a análise do padrão procurado no corpus de aprendizado. Observamos que a feature é *uma função de confiança para uma sequência de caracteres, para ajudar o modelo a estimar a probabilidade de uma determinada palavra ser de uma classe*

Feature	Valor
palavra	UNIVERSAL
palavra - 1	SANFONADO
palavra - 2	SIFAO
palavra + 1	ASTRA
palavra + 2	NULO
palavra contém número?	NÃO
palavra - 1 contém número?	NÃO
palavra + 1 contém número?	NÃO
palavra + 2 contém número?	NÃO
palavra - 2 contém número?	NÃO

METODOLOGIA



Classificação Manual

Cada conjunto de produtos retirado como amostra, passou então por uma subdivisão onde parte do conjunto foi encaminhada para que técnicos classificassem as palavras que os descreviam, atribuindo a cada uma delas uma entidade pré-determinada como as descritas anteriormente.

IDX	ENTIDADE	TOKEN
0	PD	ACHOCOLATADO
0	S	CX
0	SCN	2KG
0	MA	TODDY
1	PD	CHOCOLATE
2	PD	CHOKITO
3	PD	ACHOC
3	MA	TODDY
3	SCN	2KG

- Resultados:

Palavra	sit	Palavra	sit	Palavra	sit
joelho PD PD	OK	luva PD PD	OK	interna DP DP	OK
90 DP DP	OK	sold DP DP	OK	3 SCN SCN	OK
sold DP DP	OK	20mm DP SCN	ERRO	4 SCN SCN	OK
50mm SCN SCN	OK	krona SCN DP	ERRO	te PD PD	OK
joelho PD PD	OK	luva PD PD	OK	sold DP DP	OK
90 DP DP	OK	esgoto DP DP	OK	25mm SCN SCN	OK
soldavel DP DP	OK	100mm SCN SCN	OK	cap PD PD	OK
50mm SCN SCN	OK	luva PD PD	OK	sold DP DP	OK
joelho PD PD	OK	soldavel DP DP	OK	20mm SCN SCN	OK
90 DP DP	OK	20 SCN SCN	OK	te PD PD	OK
sold DP DP	OK	mm SCN SCN	OK	soldavel DP DP	OK
20mm SCN SCN	OK	luva PD PD	OK	25 SCN SCN	OK
luva PD PD	OK	sold DP DP	OK	mm SCN SCN	OK
soldavel DP DP	OK	32mm SCN SCN	OK	luva PD PD	OK
60mm SCN SCN	OK	uniao PD PD	OK	esgoto DP DP	OK
				50mm SCN SCN	OK

Corpus	Acurácia
Tubos e Conexões	96,00%
Chocolates	91,00%
Produtos de Limpeza	100,00%
Carne de Frango	80,00%
Aleatório	52,00%

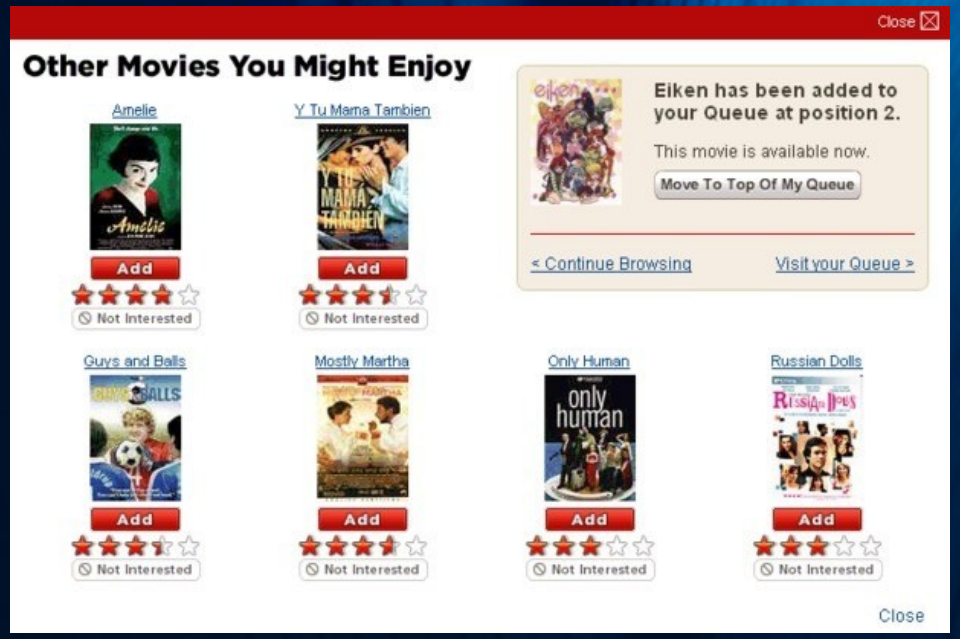
A Recommender System for Suspect Biding Contracts

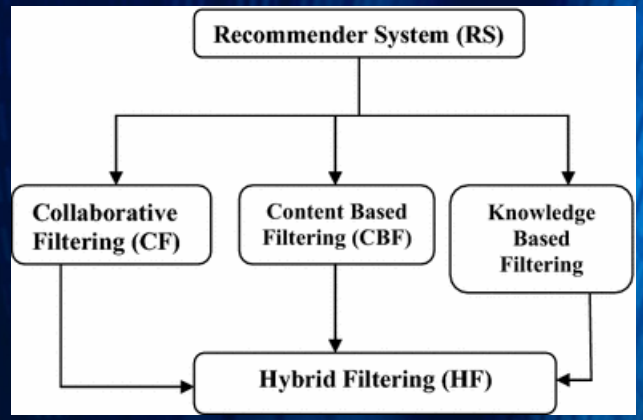
Puc Minas – Master's Degree Program

Information Retrieval

Advisor: Phd Wladmir Cardoso Brandão

Recommender System is a set of algorithms that use Machine Learning (ML) and Information Retrieval (IR) techniques to generate recommendations based on some type of filtering, the most common being: collaborative (considers the experience of all users) , based on content (considers the target user experience) and hybrid (both approaches are considered).





We are investigating if Recommendation Systems can be useful to indicate an irregular contract suspect in a set of contracts, taking a group of previous irregular contract rated as irregular by a user comparing with a group of contract not analyzed yet.

Content-Based Filtering

They make suggestions for items that are similar to what the user has shown interest in the past, and / or about user preference settings. Thus, the recommendations are customized for each user.
For example, a user who has fully analyses a contract rating it as suspect and wants to inspect more similar contracts. A content-based recommender may recommend new similar contracts based on the user's historical inspection behavior.

The algorithm works with the analysis of data through the similarity between terms and/or expressions. Given a dataset, the algorithm checks with the input of a parameter, what information is present in this dataset, is similar to the term/expression passed as a parameter. The irrelevant words for the result set to be displayed are disregarded. These words are known as Stop Words. As the user provides more inputs or performs actions on the recommendations, the mechanism becomes more accurate.

In order to achieve similarity calculations, we are based on the concept of the term frequency (TF) and inverse document frequency (IDF) that are used in Information Retrieval Systems and also in content-based filtering mechanisms. They are used to determine the relative importance of a document, article, news, film, among others.

TF-IDF MATRIX

TF is simply the frequency of a word in a document. The IDF is the inverse of the document's frequency between the entire corpus of documents.
The term frequency score is calculated by taking the frequency of a token in an article. The inverse document frequency score is calculated by the logarithm of dividing the total number of articles by the number of articles in which the token occurs. When multiplying these two scores, a value is obtained that is high for features that occur frequently in a small number of articles and is low for features that occur often in many articles.

TF-IDF Score

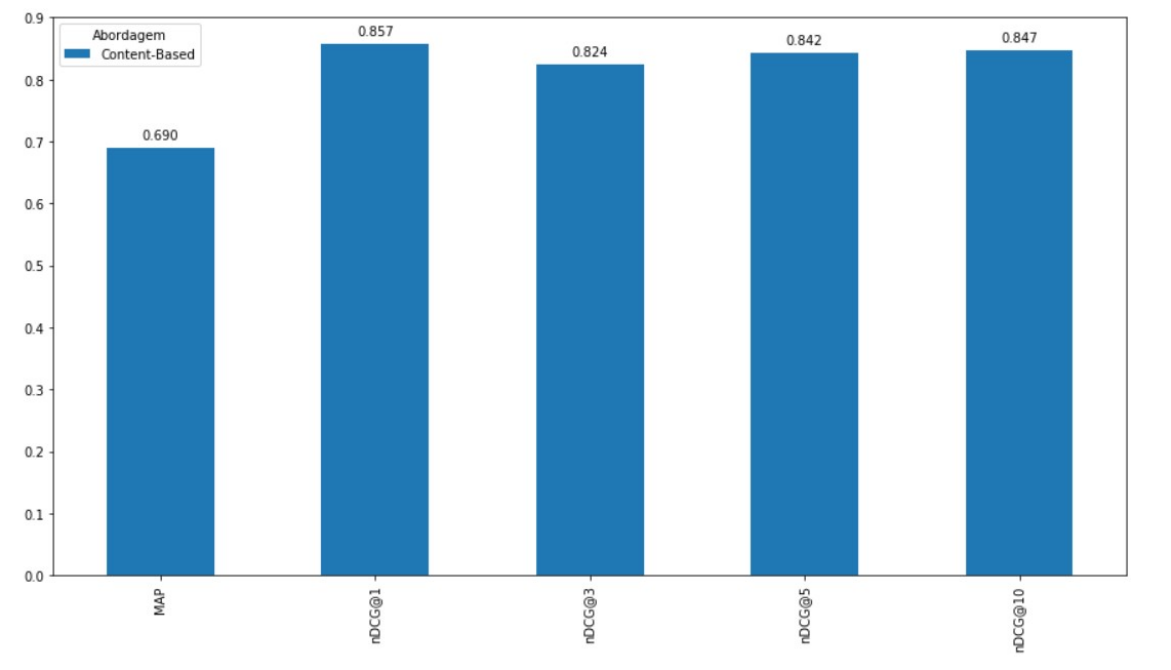
$$TF - IDF\ Score = TF_{x,y} * IDF = TF_{x,y} * \log\frac{N}{df} \ldots\ldots (1)$$

, where $TF_{x,y}$ is the frequency of keyphrase X in the article Y,
N is the total number of documents in the corpus.
df is the number of documents containing keyphrase X

The result extracted in the execution of the algorithm is ranked, that is, after the verification of the similarity between the data sets, those records that are with the ranking nearest to one, have a greater similarity compared to the Parameter used.

Method and Results:

For evaluate the hypothesis experiments were carried out what involved calculate a similarity between contracts; assess whether the recommendations generated were interesting to users, as well as applying precision (precision) and scope (recall) metrics with the objective of evaluating the technique. The experiments were performed using a sample for convenience (not probabilistic) composed of 22504 Contracts with inrequirement of bidding extracted from the Open data of the TCE/MG. These contracts were offered to 7 technicians to ascertain indications of irregularity, at the end 405 of these contracts contained something strange in their textual description (object, justification and reason) Totaling 32160 interactions. Once the evaluation of these Technical for the Contracts selected, it was possible to start the process of recommendation by content analysis. Subsequently, the precision and scope measures were used to assess the result.



https://github.com/LeoVilelaRibeiro