

Mineração de Texto

Prof. Rodrigo Macedo

Escopo do Curso

- Mineração de Texto.
- Metodologias de Mineração de Texto.
- Recuperação de Informações.
- Classificação e relevância de termos.
- Processamento de Linguagem Natural.
- Questões de concursos



Mineração de Texto - Conceito

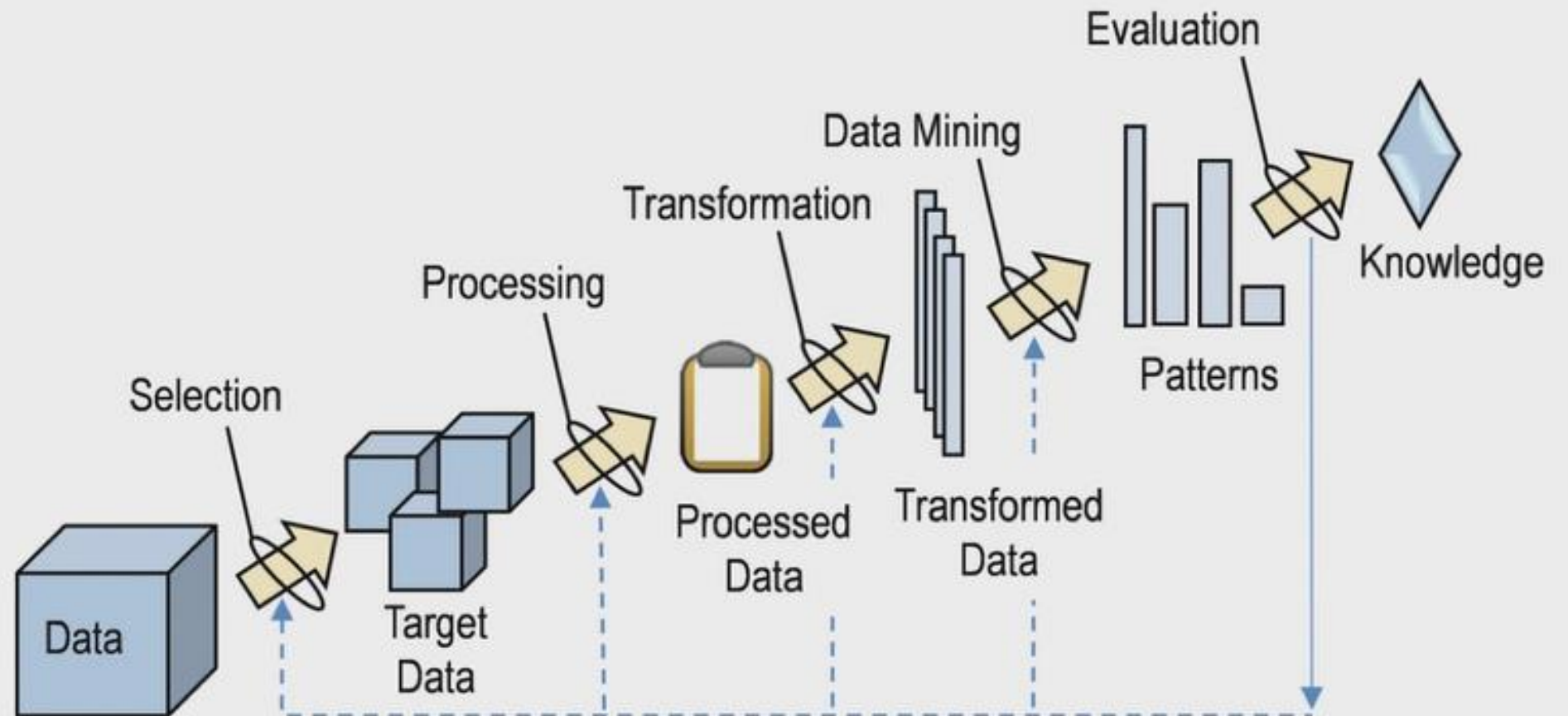
- Mineração de textos, também chamado de mineração de dados textuais ou descoberta de conhecimento de bases de dados textuais é um campo novo e multidisciplinar que inclui conhecimentos de áreas como Informática, Estatística, Linguística e Ciência Cognitiva.
- Mineração de textos consiste em extrair regularidades, padrões ou tendências de grandes volumes de textos em linguagem natural, normalmente, para objetivos específicos.
- Inspirado pelo data mining ou mineração de dados, que procura descobrir padrões emergentes de banco de dados estruturados, a mineração de textos pretende extrair conhecimentos úteis de dados não estruturados ou semi-estruturados.

Mineração de Texto - Motivação

- A mineração de textos surgiu a partir da necessidade de se descobrir, de forma automática, informações (padrões e anomalias) em textos. O uso dessa tecnologia permite recuperar informações, extrair dados, resumir documentos, descobrir padrões, associações e regras e realizar análises qualitativas ou quantitativas em documentos de texto.
- O crescimento do armazenamento de dados não estruturados, devido ao avanço da mídia digital, propiciou o desenvolvimento das técnicas de mineração de textos. Normalmente, os documentos onde são aplicadas as técnicas de mineração de textos incluem: emails, textos livres obtidos por resultados de pesquisas, arquivos eletrônicos gerados por editores de textos, páginas da Web, campos textuais em bancos de dados, documentos eletrônicos, digitalizados a partir de papéis.

Descoberta de Conhecimento

KD
Process

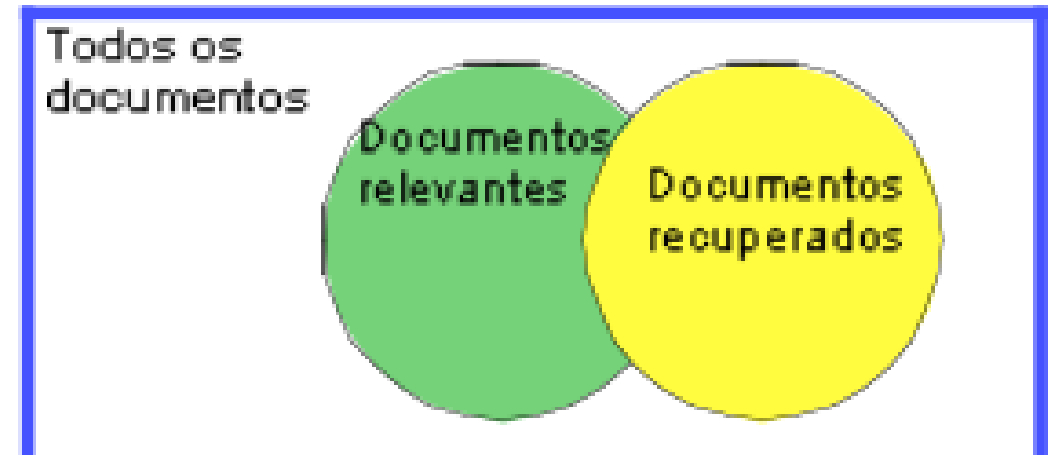


Descoberta de Conhecimento em Texto



Mineração de Texto - Avaliação

- Os critérios mais comumente utilizados na literatura são acurácia e abrangência, ou, precision e recall. Para calculá-los é necessário ter uma base de treinamento e teste apontando o certo e o errado, o relevante ou o irrelevante.



$$\text{Precisão} = \frac{\text{Número de documentos relevantes recuperados}}{\text{Número total de documentos recuperados}}$$

$$\text{Eficiência (Recall)} = \frac{\text{Número de documentos relevantes recuperados}}{\text{Número total de documentos relevantes}}$$

Mineração de Texto - Etapas

- O processo de mineração de texto é composto das seguintes etapas:

1. Coleta.
2. Pré-processamento.
3. Indexação.
4. Mineração.
5. Análise.



Mineração de Texto - Coleta

- Formatação da base de documentos ou Corpus. (Robôs de Crawling atuando em qualquer ambiente).
- Corpus é o conjunto de textos escritos e registros orais em uma determinada língua e que serve como base de análise.
- Corpora é o plural de Corpus. O termo dataset é sinônimo de Corpus.



Mineração de Texto - Pré-processamento

- O objetivo desta etapa é a eliminação de dados irrelevantes e a padronização de termos utilizando algoritmos específicos, de modo que, a etapa de recuperação de informação tenha uma melhor performance.
- Textos obtidos na web usando os crawlers são carregados desses termos (palavras) irrelevantes considerando que são intrínsecas à qualquer texto e, portanto, não discriminam o domínio.
- É importante que essas palavras sejam eliminadas para não sobrecarregarem o processo e/ou para não interferirem na análise das informações.



Download from
Dreamstime.com

This watermarked comp image is for previewing purposes only.



ID 20120603

© Alexander Limbach | Dreamstime.com

Mineração de Texto - Indexação

- Objetivo acesso rápido, busca. (Recuperação de Informação [IR]).
- Permite a recuperação da informação minerada.



Mineração de Texto - Mineração

- Cálculos, inferências e extração de conhecimento.
- Utilizar algoritmos para classificar, agrupar textos.
- Análise de sentimentos.



Mineração de Texto - Análise

- Análise humana. Navegação. (Leitura e Interpretação dos dados).
- Etapa em que envolve a tomada de decisão, com base na mineração de texto realizado anteriormente.



Mineração de Texto - Aplicações

Prevenção de Crimes: Como a Internet é anônima e a maioria dos softwares de comunicação que operam através dela, muitos criminosos planejam e se comunicam usando esses métodos. No entanto, você pode entender que milhões de pessoas normais também usam esses meios de comunicação, e é uma tarefa difícil identificar mensagens que podem ser consideradas uma ameaça. Isso é feito facilmente, usando um software avançado de análise de texto que verifica as fontes de comunicação em tempo real e soa diferentes níveis de alerta de ameaça ao encontrar diferentes tipos de texto.



Mineração de Texto - Aplicações

Gerenciamento de Risco: Todo setor quer tomar consciência dos riscos que está enfrentando ou daqueles que poderá enfrentar em um futuro próximo. Por esse motivo, os analistas de risco estão em alta demanda nos últimos anos. Muitos no setor financeiro, como bancos, instituições de microfinanças e outros, agora dependem de softwares de gerenciamento de risco que podem passar por documentos e perfis, a fim de decidir sobre em que empresa investir, em quais pessoas emprestar empréstimos. As tecnologias de mineração de texto usadas por esse software de ponta absorvem petabytes de dados e apresentam informações em um formato consumível. Isso ajuda na mitigação de riscos. Esse software está ajudando instituições financeiras em todo o mundo a diminuir sua porcentagem de ativos com desempenho insatisfatório.



Mineração de Texto - Aplicações

Publicidade personalizada:

Lembra quando você viu anúncios do mesmo celular no Facebook que estava visualizando na Amazon? Não, isso não é uma coincidência. A publicidade digital foi revolucionada pela mineração de texto e web. Os dados de texto relacionados a tudo o que você digita, visualiza ou faz on-line são armazenados por gigantes da tecnologia ou vendidos a outras empresas para mostrar anúncios em que você tem maior probabilidade de clicar e que têm maior probabilidade de serem convertidos em uma venda. Este é um dos aplicativos mais recentes e mais usados de análise de texto e mineração.



Mineração de Texto - Aplicações

Enriquecimento de conteúdo: Escrever conteúdo para blogs é algo que um bot criado artificialmente trabalhando com análise de texto ainda não pode fazer. No entanto, ele pode coletar várias informações relacionadas ao tópico que você precisa, juntamente com as notícias mais recentes e os artigos mais vistos sobre o assunto, para ajudá-lo a fazer uma estimativa calculada de como formar seu artigo e quais subtópicos serão adicionados a ele. Isso faz uma diferença significativa ao escrever sobre tópicos com grandes volumes de dados preexistentes na Internet. Isso ajuda a tornar seu conteúdo informativo e a conectar-se a artigos e estudos anteriores no mesmo campo.



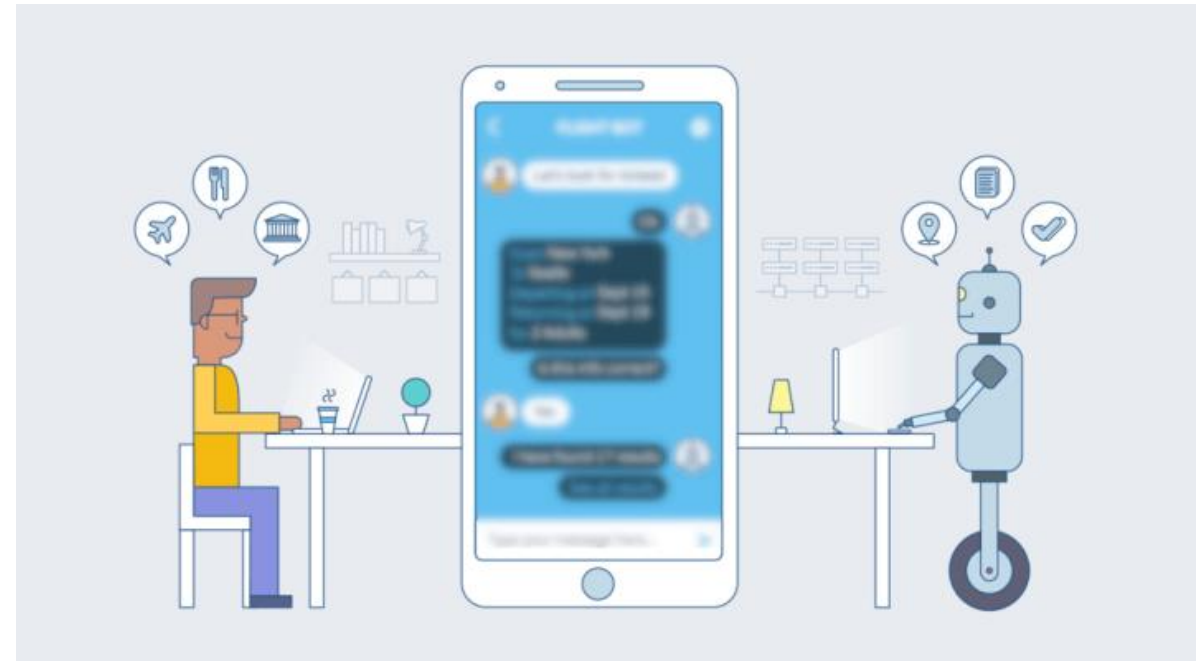
Mineração de Texto - Aplicações

Filtro de Spam: Os e-mails ainda são considerados a forma mais oficial de comunicação na maioria das organizações. Mas tem um lado sombrio que só aumentou no século XXI - o spam. De cada dez emails na minha caixa de correio, pelo menos nove são spam. Os spams não apenas ocupam espaço, mas também servem como ponto de entrada para vírus, golpes e muito mais. As empresas estão se esforçando para filtrar cada vez mais spam usando análises de texto inteligentes em comparação com a correspondência de palavras-chave usada anteriormente, para filtrar mais emails de spam e oferecer ao usuário uma experiência mais saudável.



Mineração de Texto - Aplicações

Atendimento ao cliente: A mineração de texto e o processamento de linguagem natural são frequentemente usados nos serviços de atendimento ao cliente, seja por bate-papo ou chamada de voz. O formato "pressione um para recarregar, pressione dois para ..." foi alterado para "diga sim para encerramento de conta ou não para cancelamento ..." em muitos lugares para fazer com que o sistema pareça mais humano. A maioria dos bancos e empresas de comércio eletrônico está usando bots de bate-papo baseados no processamento de linguagem natural que tentam imitar um funcionário de atendimento ao cliente quando conversam com um cliente.



Recuperação de Informações

- Os dados textuais são desestruturados, ao contrário dos dados rigidamente estruturados nos bancos de dados relacionais.
- O termo recuperação de informações geralmente se refere à consulta de dados textuais não estruturados.
- Os sistemas de recuperação de informações tem associação aos sistemas de banco de dados, no que tange a armazenamento e recuperação de dados.
- Porém, a ênfase nos sistemas de recuperação concentra-se em questões como consulta baseada em **palavra-chave**, **relevância** de documentos à consulta, análise, **classificação** e **indexação** do documento.

Recuperação de Informações

- O campo de recuperação de informações desenvolveu em paralelo com o campo de banco de dados.
- No modelo tradicional usado no campo de recuperação de informações, a informação é organizada em documentos e considera-se que exista uma grande quantidade de documentos.
- Os dados contidos no documento são não estruturado e não dispõem de qualquer esquema associado.
- O processo de recuperação de informações consiste em localizar documentos relevantes, com base na entrada do usuário, como palavras-chave ou documentos de exemplo.

Exemplo



Gerenciamento de documentos online



Catálogos de biblioteca online

Os documentos intencionados normalmente são descritivos por um conjunto de **palavras-chave** - por exemplo, as palavras chaves “sistema de BD”, podem ser usadas para localizar livros sobre sistemas de BD e as palavras “ações” e “escândalo” podem ser usadas para localizar artigos sobre escândalos no mercado de ações.

Recuperação de Informações

- A recuperação de informações baseadas em palavra-chaves pode ser usada não apenas para recuperação de dados textuais, mas também para a recuperação de outros tipos de dados, como dados de vídeo e áudio, que possuem palavras-chave descritivas associadas.
- Por exemplo, um filme de vídeo pode ter, associado a ele, palavras-chaves como seu título, diretor, atores e tipo - **metadado**.



Sistemas de Banco de Dados x Recuperação de Informações

Sistemas de BD	Recuperação de Informação
Informações estruturadas.	Informações não estruturadas.
Atualização e requisito transacional associados ao controle de concorrência e durabilidade.	Não prioriza controle de concorrência e durabilidade.
Modelo de dados relativamente complexo (modelo relacional ou orientado a objetos)	Modelo de dados simples, em que a organização é organizada em uma coleção de documentos.
Consultas realizadas com SQL	Consultas por palavras-chave e classificação de documentos em graus de relevância a consulta.

Consultas Adicionais

- Os sistemas de recuperação de informação normalmente permitem expressões de consulta formadas usando palavras-chave e os conectivos lógicos and, or e not.

Ex1: “motocicleta and manutenção” - motorcycle e manutenção.

Ex2: “computador or processador” - computador ou processador.

Ex3: “computador but not database” - computador mas não database.



Termo

- Na recuperação de texto completo, todas as palavras em cada documento são consideradas como sendo palavras-chave.
- Usamos a palavra **termo** para nos referir às palavras em um documento, pois todas as palavras são palavras-chave.
- Sistemas mais sofisticados estimam a relevância dos documentos a uma consulta, de modo que os documentos possam ser mostrados na ordem de relevância estimada.



Aplicabilidade

- Hoje, os mecanismos de busca visam satisfazer as necessidades de informação dos usuários, avaliando a que tópico uma consulta se refere, e mostrar não somente páginas Web julgadas como relevantes, mas também outros tipos de informação a respeito do tópico.
- Ex: Em resposta a uma consulta com as palavras “Nova York”, um mecanismo de busca pode mostrar o mapa da cidade de Nova York, suas imagens, além de páginas Web relacionadas com a cidade.



Classificação de relevância usando termos

- Na recuperação da informação, cada documento pode conter muitos termos, e até mesmo termos que são apenas mencionados de passagem são equivalente a documentos em que o termo é realmente relevante.
- Os sistemas de recuperação de informações, nesse caso, estimam a relevância de documentos a uma consulta e retornam apenas documentos com avaliação alta como resposta.
- A classificação da relevância não é uma ciência exata, mas existem algumas técnicas bem aceitas.

Classificação com TF-IDF

- A primeira questão a considerar, dado um termo t em particular, é quão relevante é um documento em particular d ao termo.
- Uma técnica é usar o número de ocorrências do termo no documento como uma medida de sua relevância, supondo que os termos relevantes provavelmente serão mencionados muitas vezes em um documento.
- Apesar disso, há algumas restrições: primeiro, o número de ocorrências depende do tamanho do documento e, segundo, um documento contendo 10 ocorrências de um termo pode não ser 10 vezes mais relevante do que um documento contendo uma ocorrência.

Classificação com TF-IDF

- Uma forma de medir $TF(d,t)$, a relevância de um documento d a um termo t , é:

$$TF(d, t) = \log \left(1 + \frac{n(d, t)}{n(d)} \right)$$

CS Scanned with CamScanner

- Onde $n(d)$ indica o número de termos no documento e $n(d,t)$ indica o número de ocorrências do termo t no documento d .
- A relevância aumenta com mais ocorrências de um termo no documento, embora não seja diretamente proporcional ao número de ocorrências.

Classificação com TF-IDF


- Muitos sistemas refinam essa métrica usando outras informações. Por exemplo, se o termo ocorre no título ou na lista de autores, ou no resumo, o documento é considerado mais relevante ao termo.
- De modo similar, se a primeira ocorrência de um termo estiver muito longe do início do documento, este pode ser considerado menos relevante do que se a primeira ocorrência estivesse no início do documento.
- Na comunidade de recuperação de informações, a relevância de um documento a um termo é conhecida como **frequência do termo (TF - term frequency)**.

Classificação com TF-IDF

- Suponha que uma consulta use dois termos, um dos quais ocorre com frequência, como “banco”, e outro que é menos frequente, como “Silberchatz”.
- Um documento contendo “Silberchatz”, mas não “banco”, deverá ter uma classificação mais alta do que um documento contendo o termo “banco”, mas não “Silberchatz”.
- Por isso, utiliza-se a **frequência de documento inversa (IDF - Inverse Document Frequency)** que tem como fórmula:

Onde $n(t)$ indica o número de documentos (entre aqueles indexados pelo sistema) que contêm o termo t .

$$IDF(t) = \frac{1}{n(t)}$$

 Scanned with CamScanner

Classificação com TF-IDF

- TF-IDF é usado para ponderar as palavras de acordo com a importância delas. Palavras que são usadas com frequência em muitos documentos terão uma ponderação mais baixa, enquanto as menos frequentes terão uma ponderação mais alta.
- TF-IDF é uma técnica de recuperação de informações que pesa a frequência de um termo (TF) e sua frequência inversa no documento (IDF).
- Cada palavra ou termo tem sua respectiva pontuação TF e IDF. O produto das pontuações TF e IDF de um termo é chamado de peso TF * IDF desse termo.

Bag of Word

- O modelo Bag of Word é uma representação simplificada utilizada no processamento de linguagem natural e na recuperação de informações. Neste modelo, o texto (uma frase ou documento) é representado como um multiconjunto de suas palavras (o "saco"), desconsiderando a estrutura gramatical e até mesmo a ordenação delas, mas mantendo sua multiplicidade.
- O modelo Bag of Word é frequentemente utilizado em métodos de classificação de documentos, onde a frequência de ocorrência de cada palavra é vista como uma característica utilizada para treinar o classificador. No entanto, já foram registrados usos do modelo em estudos na área de visão computacional.



Bag of Word - Exemplo

- Documento de texto 1: John gosta de assistir ao futebol. Chris gosta de futebol também.
- Documento de texto 2: John também gosta de assistir a filmes.
- Com base nesses dois documentos de texto, você pode gerar a seguinte lista:
- Lista de palavras= ["John", "gosta", "de", "assistir", "futebol", "Chris", "também", "filmes"]
- Esta lista é chamada BOW (Bag of Words). Aqui, não estamos considerando a gramática das sentenças. Nós também não estamos incomodados com a ordem das palavras.

Processamento de Linguagem Natural

- Processamento de língua natural (PLN) é uma subárea da ciência da computação, inteligência artificial e da linguística que estuda os problemas da geração e compreensão automática de línguas humanas naturais.
- Sistemas de geração de língua natural convertem informação de bancos de dados de computadores em linguagem compreensível ao ser humano e sistemas de compreensão de língua natural convertem ocorrências de linguagem humana em representações mais formais, mais facilmente manipuláveis por programas de computador.
- Alguns desafios do PLN são compreensão de língua natural, fazer com que computadores extraiam sentido de linguagem humana ou natural e geração de língua natural.

Processamento de Linguagem Natural - Objetivo

- O objetivo do PLN é fornecer aos computadores a capacidade de entender e compor textos.
- “Entender” um texto significa reconhecer o contexto, fazer análise sintática, semântica, léxica e morfológica, criar resumos, extrair informação, interpretar os sentidos, analisar sentimentos e até aprender conceitos com os textos processados.



Processamento de Linguagem Natural



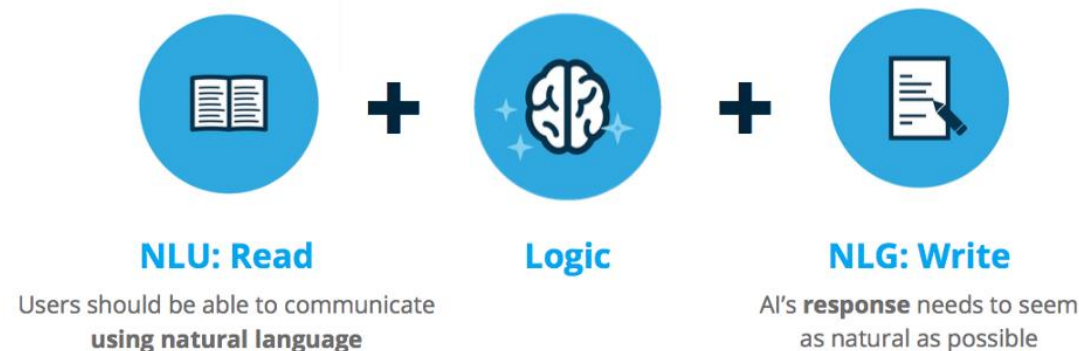
Linguagem Natural

- O conteúdo de um livro é uma fonte de linguagem natural.
- Quando você conversa com alguém, ouve ou escreve algo, você está usando linguagem natural.
- Diálogos em filmes, séries, também são exemplos de linguagem natural.
- Suas conversas em aplicativos de mensagem, são exemplos de linguagem natural.



Componentes do PLN

- **NLU (Natural Language Understanding):** É considerado o primeiro componente da PLN. É o processo de converter linguagem natural em uma representação útil usando ferramentas linguísticas. Exemplo: Análise de sentimento (positivo ou negativo) de expressões, análise de palavras-chave em um texto, dentre outros.
- **NLG (Natural Language Generation):** É considerado o segundo componente da PLN. É o processo de gerar linguagem natural a partir do output de uma máquina. Exemplo: Assistentes virtuais, como Alexa, Siri, Cortana, que respondem de acordo com a forma como foram desenvolvidas com o sistema de Inteligência Artificial. Outro exemplo também comum, é a geração de legendas a partir de um vídeo.



Exemplo - Python NLTK

Quando estamos iniciando uma etapa de análise num conjunto de dados, é essencial a etapa de análise estatística dos dados, quando estamos num processo de PLN, é essencial entendermos a frequência de distribuição de um determinado corpus

```
1  # Imports
2  import nltk
3  from nltk.corpus import webtext
4
5  # Fileids
6  print("\n")
7  print(webtext.fileids())
8
9  # Distribuição de frequência de um único arquivo
10 fileid = 'singles.txt'
11 wbt_words = webtext.words(fileid)
12 fdist = nltk.FreqDist(wbt_words)
13
14 # Report
15 print('\nContagem do número máximo de ocorrências do token "', fdist.max(), '" : ', fdist[fdist.m
16 print('\nNúmero total de tokens distintos : ', fdist.N())
17 print('\nA seguir estão os 10 tokens mais comuns')
18 print(fdist.most_common(10))
19 print("\n")
```

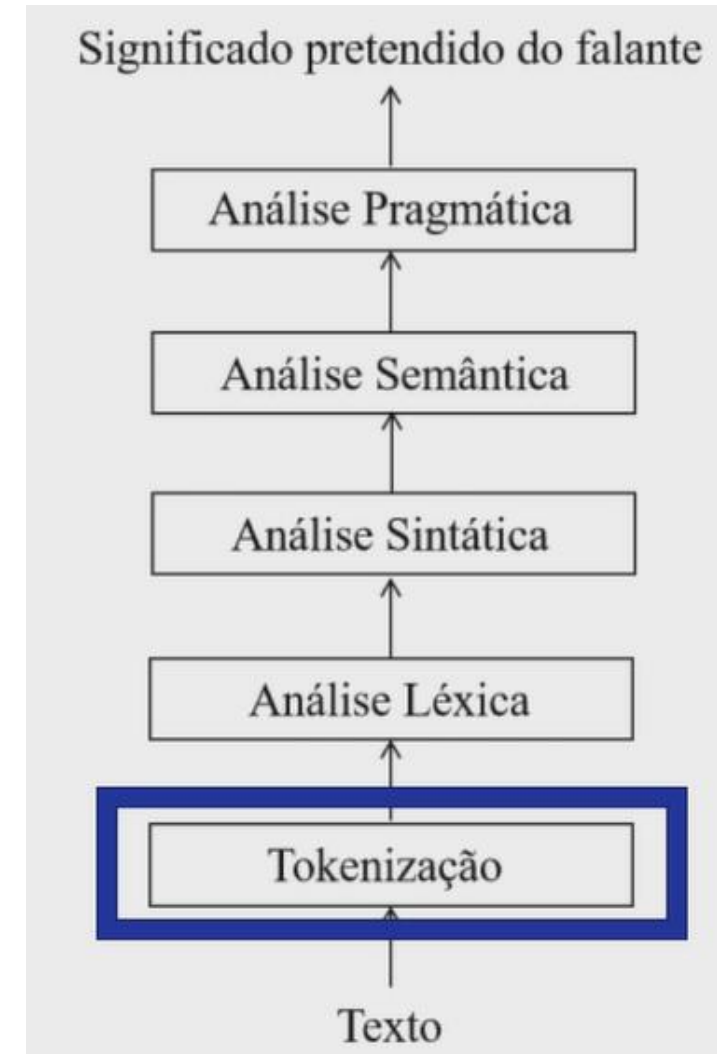
Exemplo - Python NLTK

Quando estamos iniciando uma etapa de análise num conjunto de dados, é essencial a etapa de análise estatística dos dados, quando estamos num processo de PLN, é essencial entendermos a frequência de distribuição de um determinado corpus

```
['firefox.txt', 'grail.txt', 'overheard.txt', 'pirates.txt', 'singles.txt', 'wine.txt']  
Contagem do número máximo de ocorrências do token " , " : 539  
Número total de tokens distintos : 4867  
A seguir estão os 10 tokens mais comuns  
[(',', 539), ('.', 353), ('/', 110), ('for', 99), ('and', 74), ('to', 74), ('lady', 68), ('-', 66), ('seeks', 60), ('a', 52)]
```

Etapas de Análises

- Podemos decompor as etapas de análise em PLN em cinco etapas:



Tokenização

- Também conhecida como segmentação de palavras, quebra a sequência de caracteres em um texto localizando o limite de cada palavra, ou seja, os pontos onde uma palavra termina e outra começa.
- Para fins de linguística computacional, as palavras assim identificadas são frequentemente chamada de **tokens**.

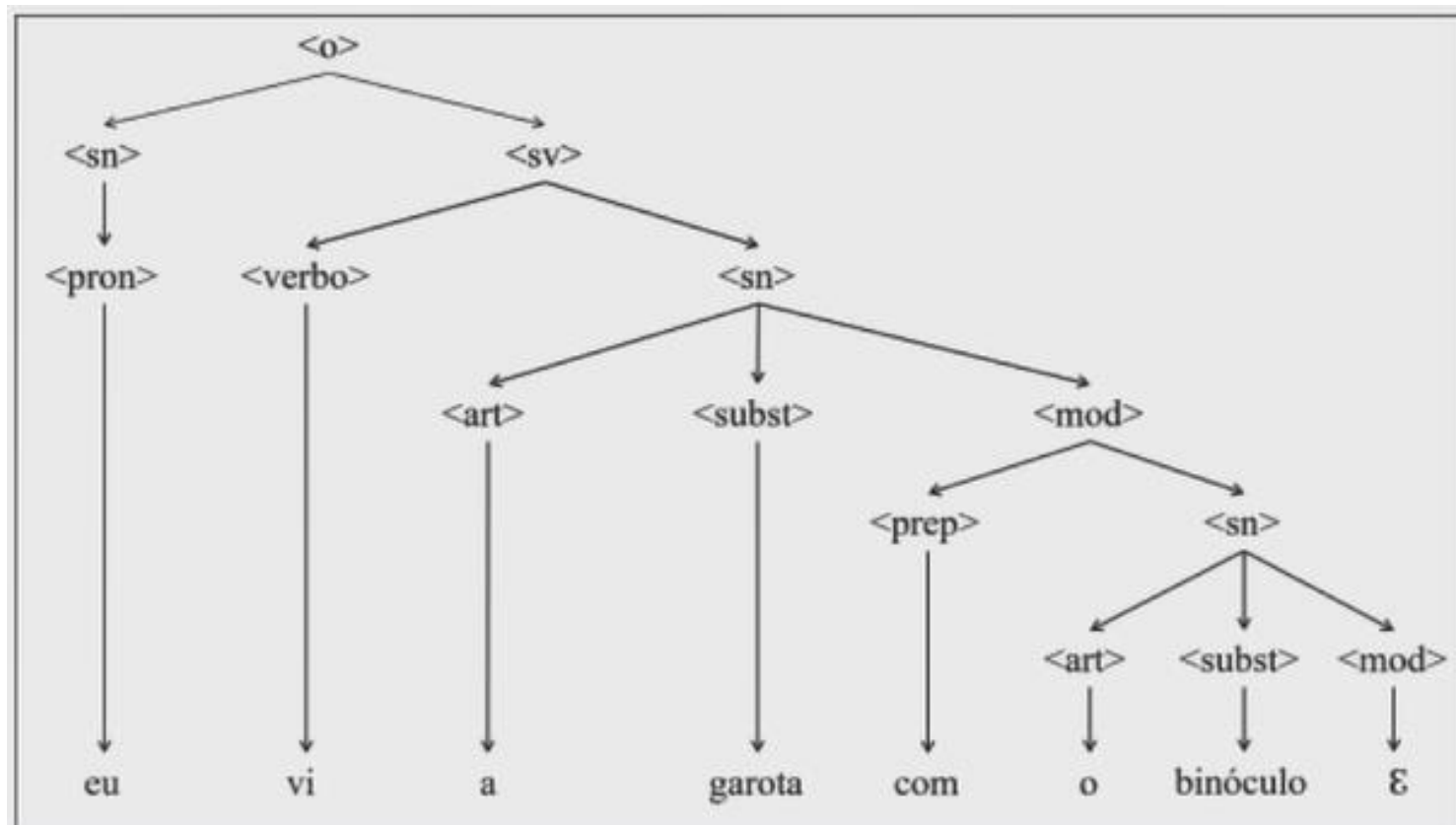


Análise Léxica

- Uma palavra pode ser pensada de duas maneiras: como uma sequência de caracteres no texto em execução, como por exemplo, o verbo ENTREGAR, ou como um objeto mais abstrato que é o termo principal para um conjunto de sequência de caracteres.
- Podemos aplicar a técnica do **stemming**.
- O **stem**, portanto é o chamado radical da palavra. Para a palavra ENTREGAR, o seu stem seria ENTREG, pois a partir do stem, podem ser criadas outras palavras.

Análise Sintática

- Uma das formas de se representar a análise sintática é por meio de gramáticas e árvores sintáticas.



Análise Semântica

- Refere-se à análise do significado das palavras, expressões fixadas, sentenças inteiras e enunciados no contexto.
- Um requerimento frequentemente identificado nessa etapa, é a resolução da ambiguidade.



Análise Pragmática

- A análise pragmática foge à estrutura de apenas uma frase, uma vez que ela busca nas outras frases a compreensão do texto que falta àquela frase em análise.
- As estruturas mais utilizadas nessa etapa são as gramáticas baseadas em casos, que são gramáticas semânticas não-terminais para formar padrões, onde caso uma dada frase encaixe na construção padrão, ela poderá ser reconhecida dentro de um contexto.



Resumo

Análise	Descrição
Léxica	Estuda a construção das palavras, com seus radicais e afixos, que correspondem a partes estáticas e variantes das palavras, como as inflexões verbais
Sintática	Envolve a análise das palavras em uma sentença de acordo com a gramática e arranjo das palavras de uma maneira que mostre a relação entre as palavras. A sentença: "A escola vai para o menino" seria rejeitada por um analisador sintático em português.
Semântica	Processo de mapeamento de sentenças de uma linguagem visando a representação de seu significado, baseado nas construções obtidas na análise sintática
Pragmática	Processamento da forma que a linguagem é utilizada para comunicar e como os significados na análise semântica agem sobre as pessoas e seu contexto

Tokenização

- Tokenization é o processo de dividir uma string em listas de pedaços ou tokens.
- Exemplo: Uma palavra é um token em uma sentença, e por sua vez, uma sentença é um token em um parágrafo.
- Existem várias técnicas variadas para aplicar o Tokenization, vamos demonstrar aqui no artigo, a Tokenization em sentenças e em palavras.



Tokenização

```
1 from nltk.tokenize import sent_tokenize, word_tokenize
2
3 # Texto
4 frase = "Aprendendo processamento linguagem natural. Python e NLTK facilitam nossa vida!"
5
6 # Tokenization em sentenças
7 sent_tokens = sent_tokenize(frase)
8 print(sent_tokens)
```

```
['Aprendendo processamento linguagem natural.', 'Python e NLTK facilitam nossa vida!']
```

Como estamos tokenizando apenas as sentenças, nesse caso, o NLTK vai entender que na nossa frase, o que estiver antes do ponto é uma sentença, e o que estiver depois do ponto, é outra frase.

Tokenização

```
1  from nltk.tokenize import sent_tokenize, word_tokenize
2
3  # Texto
4  frase = "Aprendendo processamento linguagem natural. Python e NLTK facilitam nossa vida!"
5
6  # Tokenization em sentenças
7  word_tokens = word_tokenize(frase)
8  print(word_tokens)
```

```
['Aprendendo', 'processamento', 'linguagem', 'natural', '.', 'Python', 'e', 'NLTK', 'facilitam', 'nossa', 'vida', '!']
```

Note que agora, o algoritmo fez a tokenization para separar as palavras, e não mais as sentenças. Uma dúvida que poderia ficar é a seguinte: Como descartar alguns caracteres de um corpus?

Stopwords

- Stopwords são palavras comuns que normalmente não contribuem para o significado de uma frase, pelo menos com relação ao propósito da informação e do processamento da linguagem natural.
- Antes de aplicar stopword, precisamos consultar as palavras que por padrão, a biblioteca do NLTK, cadastrou como stopword. Para cada idioma, haverão stopwords diferentes.



Stopwords

```
1 from nltk.corpus import stopwords
2
3 english_stops = set(stopwords.words('english'))
4
5 print(english_stops)
```

```
{'such', 'needn't', 'only', 'their', 'all', 'through', 'now', 'below', 'hadn', 'who', 'for', 'where', 'don't', 'won', 'ours', 'no', 'themselves', 'weren', 'too', 'more', 'by', 'o', 'isn', 'his', 'being', 'can', 'didn't', 'them', 'shouldn', 'wouldn', 'if', 'after', 'she', 'until', 'those', 'you', 'me', 'between', 'was', 'but', 'does', 'that'll', 'd', 'very', 'didn', 'our', 'couldn't', 'it's', 'ain', 'll', 'myself', 'what', 'few', 'during', 'have', 'whom', 'same', 'how', 'before', 'into', 'which', 'once', 'haven't', 'its', 'other', 'because', 'we', 'don', 'wasn', 'has', 'you're', 'nor', 'were', 'down', 'herself', 's', 'hasn't', 'will', 'on', 'is', 'you'd', 'and', 'isn't', 'been', 'any', 'should', 'yourself', 'mustn't', 'won't', 'while', 't', 'be', 'when', 'than', 'ourselves', 'these', 'they', 'then', 'further', 'at', 'doesn', 'it', 'y', 've', 'do', 'weren't', 'couldn', 'hadn't', 'above', 'having', 'yourselves', 'her', 'hasn', 'over', 'she's', 'about', 'doing', 'both', 'not', 'again', 'a', 'to', 'your', 'i', 'am', 'this', 'out', 'my', 'against', 'own', 'up', 'mustn', 'yours', 'in', 'doesn't', 'just', 're', 'you've', 'wouldn't', 'him', 'from', 'aren', 'you'll', 'mightn', 'wasn't', 'mightn't', 'off', 'shan', 'as', 'of', 'there', 'each', 'theirs', 'under', 'aren't', 'shan't', 'are', 'the', 'here', 'haven', 'himself', 'hers', 'with', 'shouldn't', 'needn', 'he', 'an', 'most', 'should've', 'or', 'm', 'that', 'some', 'did', 'had', 'why', 'itself', 'so', 'ma'}
```

Stopwords

```
1 from nltk.corpus import stopwords
2
3 portuguese_stops = set(stopwords.words('portuguese'))
4
5 print(portuguese_stops)
```

```
{'tenha', 'esteja', 'ne', 'houveríamos', 'esses', 'teria', 'na', 'o', 'haja', 'havemos', 'tenhamos', 'e', 'houverá', 'es  
tava', 'um', 'pela', 'houver', 'éramos', 'para', 'aquilo', 'eram', 'tiver', 'estiveram', 'vos', 'tivesse', 'no', 'minha',  
, 'mas', 'houveriam', 'eu', 'suas', 'tu', 'estivéssemos', 'as', 'estiverem', 'às', 'estivermos', 'estas', 'muito', 'tive  
rmos', 'lhes', 'como', 'quem', 'estive', 'houve', 'aos', 'tinham', 'teríamos', 'formos', 'tereí', 'uma', 'seu', 'não',  
esta', 'em', 'isto', 'numa', 'fôramos', 'hei', 'este', 'tenham', 'houverem', 'estivessem', 'aquelas', 'vocês', 'do', 'se  
ria', 'nos', 'minhas', 'fomos', 'estiver', 'dela', 'fosse', 'estão', 'ã', 'qual', 'nossa', 'tive', 'nós', 'estivera', 'q  
ue', 'estamos', 'quando', 'tivêramos', 'fui', 'tivera', 'era', 'houvéssemos', 'tenho', 'ele', 'houvéramos', 'houvermos',  
, 'teus', 'mesmo', 'lhe', 'tiverem', 'num', 'meus', 'aquele', 'os', 'depois', 'tinha', 'tua', 'estavam', 'se', 'hajam',  
houvessem', 'houveremos', 'nosso', 'seja', 'pelo', 'essas', 'sem', 'tivéssemos', 'fôssemos', 'foi', 'você', 'tuas', 'del  
as', 'sou', 'houvesse', 'sejam', 'só', 'estejam', 'te', 'houverão', 'terá', 'houvenos', 'dele', 'estivemos', 'tém', 'est  
ã', 'ao', 'tiveram', 'essa', 'meu', 'ela', 'eles', 'dos', 'seus', 'estávamos', 'terão', 'serei', 'esse', 'houveram', 'se  
ríamos', 'são', 'esteve', 'será', 'há', 'também', 'forem', 'foram', 'de', 'sonos', 'houveria', 'aqueles', 'tivemos', 'es  
tes', 'já', 'nossas', 'aquela', 'houverei', 'sua', 'teriam', 'estivêramos', 'sejamos', 'seremos', 'pelos', 'teu', 'fosse  
m', 'for', 'serão', 'pelas', 'estejamos', 'hajamos', 'nossos', 'seriam', 'teve', 'elas', 'temos', 'até', 'tínhamos', 'te  
remos', 'estivesse', 'por', 'tivessem', 'mais', 'das', 'da', 'hão', 'nas', 'deles', 'isso', 'houvera', 'nem', 'estou',  
com', 'fora', 'tem', 'ou', 'a', 'entre'}
```

Stopwords

```
1 from nltk.corpus import stopwords
2 portuguese_stops = set(stopwords.words('portuguese'))
3 palavras = ["Estou", 'estudando', 'sobre', 'um', 'tema', 'interessante', 'em', 'PLN']
4 print([palavra for palavra in palavras if palavra not in portuguese_stops])
```

```
['Estou', 'estudando', 'sobre', 'tema', 'interessante', 'PLN']
```

Como vimos acima, artigos como: um, em, foram descartados pelo stopword, conforme o que aplicamos no exemplo.

Stemming

- Stemming está relacionado com a técnica de remoção de sufixos e prefixos numa palavra, pois a título de PLN, eles não tem importância nenhuma.
- Com o NLTK, podemos usar algumas variações de Stemming para atender a objetivos diversos.



Stemming

```
1  # Import
2  from nltk.stem import PorterStemmer
3  from nltk.stem import LancasterStemmer
4
5
6  # Cria o Stemmer
7  stemmer = PorterStemmer()
8
9  # Aplica o Stemmer
10 print("\nPorterStemmer")
11 print(stemmer.stem('studying'))
12 print(stemmer.stem('studied'))
13
14 # Cria o Stemmer
15 stemmer2 = LancasterStemmer()
16
17 # Aplica o Stemmer
18 print("\nLancasterStemmer")
19 print(stemmer2.stem('studying'))
20 print(stemmer2.stem('studied'))
```

```
PorterStemmer
stude
studi
```

```
LancasterStemmer
stud
study
```

Q1) [FAURGS UFRGS 2018] Uma nuvem de palavras é um recurso gráfico (usado principalmente na internet) para descrever os termos mais frequentes de um determinado texto. O tamanho da fonte em que a palavra é apresentada é uma função da frequência da palavra no texto: palavras mais frequentes são desenhadas em fontes de tamanho maior, palavras menos frequentes são desenhadas em fontes de tamanho menor. Qual é a técnica de análise de dados descrita pelo texto acima?

- a) Processamento de Linguagem Natural.
- b) Agrupamento.
- c) Classificação.
- d) Redes Neurais.
- e) Regressão Linear.

Q1) [FAURGS UFRGS 2018] Uma nuvem de palavras é um recurso gráfico (usado principalmente na internet) para descrever os termos mais frequentes de um determinado texto. O tamanho da fonte em que a palavra é apresentada é uma função da frequência da palavra no texto: palavras mais frequentes são desenhadas em fontes de tamanho maior, palavras menos frequentes são desenhadas em fontes de tamanho menor. Qual é a técnica de análise de dados descrita pelo texto acima?

a) **Processamento de Linguagem Natural.**

b) Agrupamento.

c) Classificação.

d) Redes Neurais.

e) Regressão Linear.

Q2) [FCC TRF4 2019] Um Analista necessita desenvolver uma aplicação chatbot que simula um ser humano na conversação com as pessoas. Para isso o Analista deve usar pesquisa em Processamento de Linguagem Natural – PLN que envolve três aspectos da comunicação, quais sejam,

- a) Som, ligado à fonologia, Estrutura que consiste em análises morfológica e sintática e Significado que consiste em análises semântica e pragmática.
- b) Áudio, ligado à fonologia, Estrutura que consiste em análises de línguas estrangeiras e Significado que consiste em análises semântica e pragmática.
- c) Conversação, ligado à tecnologia de chatbot, Semântica que consiste em análises de línguas estrangeiras e Arquitetura Spelling que realiza as análises sintática e pragmática.
- d) Business Intelligence, ligado à tecnologia OLAP, Mining que consiste em análises de línguas em geral e Spelling que realiza as funções de chatbot.
- e) Áudio, ligado à fonologia, Estrutura que consiste em análises semântica e pragmática e Significado que consiste em análise das línguas em geral.

Q2) [FCC TRF4 2019] Um Analista necessita desenvolver uma aplicação chatbot que simula um ser humano na conversação com as pessoas. Para isso o Analista deve usar pesquisa em Processamento de Linguagem Natural – PLN que envolve três aspectos da comunicação, quais sejam,

a) Som, ligado à fonologia, Estrutura que consiste em análises morfológica e sintática e Significado que consiste em análises semântica e pragmática.

b) Áudio, ligado à fonologia, Estrutura que consiste em análises de línguas estrangeiras e Significado que consiste em análises semântica e pragmática.

c) Conversação, ligado à tecnologia de chatbot, Semântica que consiste em análises de línguas estrangeiras e Arquitetura Spelling que realiza as análises sintática e pragmática.

d) Business Intelligence, ligado à tecnologia OLAP, Mining que consiste em análises de línguas em geral e Spelling que realiza as funções de chatbot.

e) Áudio, ligado à fonologia, Estrutura que consiste em análises semântica e pragmática e Significado que consiste em análise das línguas em geral.

Q3) [IADES APEX BRASIL 2018] A escolha de qual modelo deve-se usar para se analisar um conjunto de dados depende do domínio do problema analisado. Acerca dessa escolha de modelos, na análise de dados no domínio de textos em linguagem natural, é correto afirmar que

- a) n-grams são modelos muito utilizados por serem simples e, em geral, produzirem bons resultados.
- b) bag-of-words é considerado um modelo complexo quando comparado com outros de análise de texto, sendo de difícil implementação.
- c) textos em linguagem natural não podem ser analisados, pois são compostos de letras e não números.
- d) o modelo TFIDF produz bons resultados, mas não pode ser usado para classificação.
- e) redes neurais não podem ser utilizadas no domínio de texto.

Q3) [IADES APEX BRASIL 2018] A escolha de qual modelo deve-se usar para se analisar um conjunto de dados depende do domínio do problema analisado. Acerca dessa escolha de modelos, na análise de dados no domínio de textos em linguagem natural, é correto afirmar que

- a) n-grams são modelos muito utilizados por serem simples e, em geral, produzirem bons resultados.
- b) bag-of-words é considerado um modelo complexo quando comparado com outros de análise de texto, sendo de difícil implementação.
- c) textos em linguagem natural não podem ser analisados, pois são compostos de letras e não números.
- d) o modelo TFIDF produz bons resultados, mas não pode ser usado para classificação.
- e) redes neurais não podem ser utilizadas no domínio de texto.

Q4) [FEPESE CIASC 2017] Qual técnica de mineração de texto permite agrupar termos ou padrões similares a partir de vários documentos, e pode ser executada de modo top-down ou bottom-up através da aplicação de métodos de hierarquização, distribuição, densidade, entre outros?

- a) Clustering
- b) Summarização
- c) Contagem de Frequência
- d) Recuperação de Informação
- e) Extração de Informação

Q4) [FEPESE CIASC 2017] Qual técnica de mineração de texto permite agrupar termos ou padrões similares a partir de vários documentos, e pode ser executada de modo top-down ou bottom-up através da aplicação de métodos de hierarquização, distribuição, densidade, entre outros?

a) Clustering

b) Summarização

c) Contagem de Frequência

d) Recuperação de Informação

e) Extração de Informação

Q5) [IADES METRÔ-DF 2014] A mineração de texto consiste basicamente na extração de informação de qualidade a partir de textos em linguagem natural. Esse processo possui normalmente cinco fases principais. Com relação à fase que permite a recuperação da informação minerada, assinale a alternativa correta.

- a) Coleta.
- b) Pré-processamento.
- c) Indexação.
- d) Algoritmo.
- e) Análise de resultados.

Q5) [IADES METRÔ-DF 2014] A mineração de texto consiste basicamente na extração de informação de qualidade a partir de textos em linguagem natural. Esse processo possui normalmente cinco fases principais. Com relação à fase que permite a recuperação da informação minerada, assinale a alternativa correta.

a) Coleta.

b) Pré-processamento.

c) Indexação.

d) Algoritmo.

e) Análise de resultados.

Q6) [CS-UFG IF Goiano 2019] É um meio de encontrar padrões interessantes ou úteis em um contexto de informações textuais não estruturadas, combinado com alguma tecnologia de extração e de recuperação da informação, processo de linguagem natural e de sumarização/indexação de documentos. (Dixson, 1997, apud TRYBULA, 1999).

O conceito apresentado pelo autor se refere ao processo de

- a) mineração de dados.
- b) ontologia.
- c) redes semânticas.
- d) mineração de texto.

Q6) [CS-UFG IF Goiano 2019] É um meio de encontrar padrões interessantes ou úteis em um contexto de informações textuais não estruturadas, combinado com alguma tecnologia de extração e de recuperação da informação, processo de linguagem natural e de sumarização/indexação de documentos. (Dixson, 1997, apud TRYBULA, 1999).

O conceito apresentado pelo autor se refere ao processo de

- a) mineração de dados.
- b) ontologia.
- c) redes semânticas.
- d) mineração de texto.

GABARITO

Q1 – LETRA A.

Q2 - LETRA A.

Q3 - LETRA A.

Q4 - LETRA A.

Q5 - LETRA C.

Q6 - LETRA D.