

FIAP

NBA



Text Mining & Topic Modeling

Dheny R. Fernandes

1. Text Mining

1. Definição
2. Text Mining x Text Analytics
3. Técnicas
4. Aplicações

2. LDA

1. Introdução
2. Conceito
3. Formulação Matemática

Text Mining



Mineração de textos é o processo de **transformar textos** não estruturados num formato estruturado com o objetivo de **identificar padrões significativos e novos insights** de uma determinada linguagem natural.

Ao aplicar técnicas analíticas, como SVM, RNN, Ranking, etc., nos dados é possível explorar e descobrir relações ocultas que permeiam os dados não estruturados.

Dessa forma, a mineração de textos visa a aplicação de técnicas, muitas das quais já estudamos, para transformar textos brutos em valiosas fontes de informação.

Apesar de muitas vezes serem usados como sinônimos, ambos possuem suas nuances específicas, mas que se complementam para tornar a análise mais consistente.

Os dois identificam padrões textuais e tendências em dados não estruturados através do uso de machine learning, estatística e linguística. A grande diferença reside no fato que text analytics envolve o uso de ferramentas de visualização de dados, seja para ajudar na identificação dos padrões, seja para comunicar os resultados das análises para o público de interesse.

Muitas são as técnicas de Text Mining, e algumas delas estão presente em nossa disciplina:

- Recuperação da Informação:
 - **Retorna informação relevante ou documentos** baseado num conjunto de queries ou frases pré-definidas. É comumente usada em Search engines (Google, Bing).
- Natural Language Processing:
 - Envolve o uso de métodos para permitir que computadores **entendam linguagem humana**, escrita e verbal.
 - Exemplos:
 - Sumarização
 - POS-Tagging
 - Classificação de texto/Análise de sentimento

- Extração de Informação:
 - Foca no processo de extração de informação estruturada a partir de textos livres e armazena entidades, atributos e relações de informação numa base de dados.
 - Exemplos:
 - Feature Selection
 - Named-Entity Recognition (NER): encontra e categoriza entidades específicas em textos, tais como nomes, locais, produtos, etc.

Muitas são as aplicações de Text Mining, como temos visto ao decorrer da disciplina:

- Customer Service: chatbots, análise de review de produtos, análise de sentimento no atendimento.
- Gerenciamento de risco: provê informações ao monitorar mudanças de sentimento e extrair informações de relatórios de RI a bancos e casas de investimento. ([link](#))
- Spam Filtering: diferencia mensagens de e-mail/SMS/Whatsapp entre as que são autênticas e as que são Spam.

Latent Dirichlet Allocation (LDA)

Para qual grupo você atribuiria esse documento?

Você confiaria em uma decisão judicial tomada por um robô? A Estônia sim. O país está desenvolvendo um “**robô juiz**”, para analisar disputas legais simples envolvendo menos de € 7 mil (em torno de R\$ 30 mil). O governo espera que a tecnologia diminua a quantidade de processos para os juízes e funcionários do judiciário.

Funcionará assim: as duas partes enviam os documentos relevantes para o caso e a [inteligência artificial](#) toma a decisão — que pode ser revista por um juiz humano. O projeto ainda está no início, mas até o final do ano deve ser colocado em prática um piloto focado em disputas contratuais. Segundo o diretor do escritório de dados do governo, Ott Velsberg, os algoritmos serão ajustados de acordo com o retorno de advogados e juízes.



SAIBA MAIS

Robô alimenta quem não consegue comer por conta própria

Vai um gole? Robô barman

Aparentemente, trata-se de um documento que poderia ser dito membro da classe “Direito”.

Entretanto, ele possui algumas peculiaridades:

Entretanto, alguns termos podem por em xeque essa afirmação:

Você confiaria em uma **decisão judicial** tomada por um robô? A Estônia sim. O país está desenvolvendo um “**robô juiz**”, para analisar **disputas legais** simples envolvendo menos de € 7 mil (em torno de R\$ 30 mil). O **governo** espera que a **tecnologia** diminua a quantidade de **processos para os juízes e funcionários do judiciário**.

Funcionará assim: as duas partes enviam os documentos relevantes para o caso e a **inteligência artificial** toma a decisão — que pode ser revista por um juiz humano. O projeto ainda está no início, mas até o final do ano deve ser colocado em prática um piloto focado em **disputas contratuais**. Segundo o diretor do escritório de dados do **governo**, Ott Velsberg, os **algoritmos** serão ajustados de acordo com o retorno de advogados e juízes.



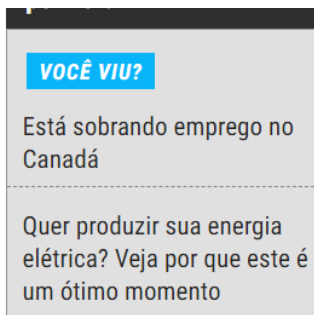
SAIBA MAIS

Robô alimenta quem não consegue comer por conta própria

Vai um gole? Robô barman

E esse documento, para qual grupo deveria ser atribuído?

Essa não é a primeira iniciativa do governo da Estônia em inteligência artificial. Desde o ano passado, Velsberg tem trabalhado para aplicar *machine learning* em serviços públicos. Funcionários públicos já foram substituídos por algoritmos na realização de 13 funções. No país que se tornou uma verdadeira sociedade digital, apenas três serviços exigem a presença física de um cidadão em uma instituição do governo: casamento, divórcio e transferência de imóvel.



Por exemplo, já não é mais necessário enviar inspetores para checar a propriedade de cada fazendeiro que recebe um subsídio do governo para cortar os seus campos de feno. Por meio de imagens de satélite feitas a cada semana, um algoritmo é capaz de avaliar a condição dos campos.

Com o novo sistema, o governo economizou € 665 mil (em torno de R\$ 2,9 milhões) no primeiro ano, já que os servidores não precisam mais se dirigir a todas as propriedades para fazer a verificação.

Ambos os textos fazem parte do mesmo documento, evidenciando que, num mesmo documento, podem existir vários tópicos distintos.

Isto se torna um problema para os paradigmas de classificação e agrupamento, já que seus objetivos é definir uma única classe/grupo para uma determinada amostra

Levando isso em consideração, *Topic Modelling* foi criado como tentativa de solucionar esse problema, atribuindo a um determinado documento probabilidades de pertencimento a um número k de tópicos.

Ainda assim, é considerado um algoritmo não-supervisionado, já que não possui informação prévia a respeito dos documentos.

Sua aplicação estende-se além de texto, e pode ser usado para imagens, DNA, etc.



SKY WATER TREE
MOUNTAIN PEOPLE



SCOTLAND WATER
FLOWER HILLS TREE



SKY WATER BUILDING
PEOPLE WATER



FISH WATER OCEAN
TREE CORAL



PEOPLE MARKET PATTERN
TEXTILE DISPLAY



BIRDS NEST TREE
BRANCH LEAVES

Aqui, vamos entender como usar LDA para fazer modelagem de tópicos e encontrar padrões de tópicos em textos

Tópic Modelling refere-se à tarefa de identificar **tópicos que melhor descrevem um conjunto de documentos.**

Como é uma tarefa não-supervisionada, não há conhecimento dos tópicos de antemão. Assim, eles **emergirão** durante o processo de modelagem.

Para realizar modelagem de tópicos, vamos utilizar LDA, cuja principal ideia é especificada da seguinte maneira:

“Cada documento pode ser descrito por uma distribuição de tópicos e cada tópico pode ser descrito por uma distribuição de palavras”.

Assim, podemos dizer que um documento foi gerado a partir de um conjunto de tópicos e cada tópico a partir de um conjunto de palavras

Ok, mas como eu faço para descobrir os tópicos?

De maneira simples, **engenharia reversa**.

Em passos:

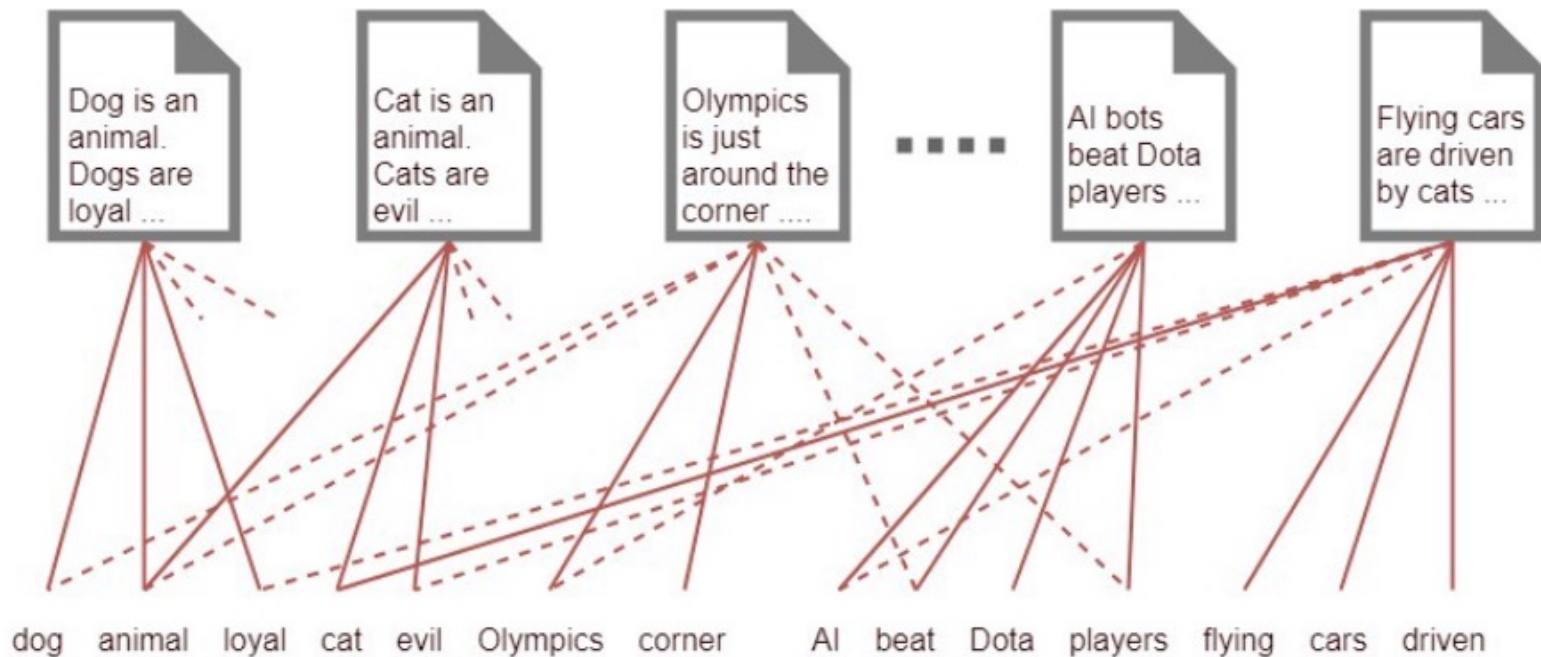
- Assuma que existam k tópicos no conjunto de documentos
- Distribua esses k tópicos sobre um documento m atribuindo à cada palavra um tópico
- Para cada palavra w no documento m , assumo que seu tópico está incorreto, mas todas as demais palavras estão atribuídas ao tópico correto
- Probabilisticamente, atribua cada palavra w a um tópico baseado em duas coisas:
 - Que tópicos existem no documento m
 - Quantas vezes a palavra w foi atribuída a um tópico em particular sobre todos os documentos
- Repita para cada documento

Mas antes de entender à fundo esse passo-a-passo, vamos analisar o problema de modelagem de tópicos mais de perto.

Imagine um conjunto de 1000 palavras e 1000 documentos. Assuma que cada documento possua, em média, 500 dessas palavras. Como descobrir a categoria que cada documento pertence?

- Uma maneira é conectar cada documento a cada palavra baseado em sua aparição no documento

Teríamos algo semelhante à isso:



É possível ver que alguns documentos são conectados pelo mesmo conjunto de palavras, o que permite inferir **algum tipo de relação entre eles**.

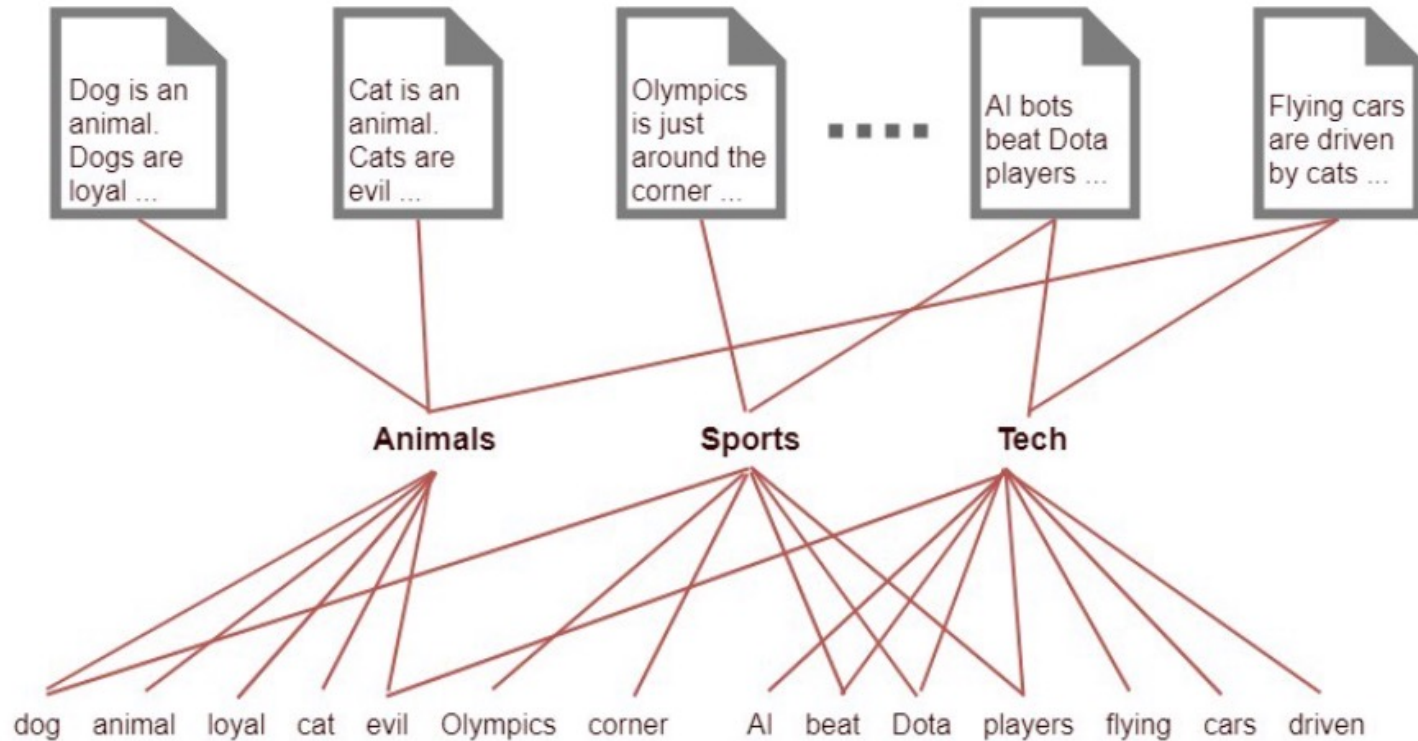
Entretanto, essa abordagem **não é escalável**, pois ficaria impossível, visualmente falando, conseguir identificar todas as relações.

Isto é, precisamos diminuir o número de conexões.

Podemos resolver esse problema usando uma camada oculta (*latent*), supondo que existam k tópicos que apareçam em todos os documentos.

Assim, posso usar essa informação conectando palavras a tópicos, dependendo quão bem essa palavra se ajuste a esse tópico, e então conectar os tópicos aos documentos com base nos tópicos abordados em cada documento.

Isso simplificaria as coisas, fornecendo essa representação:

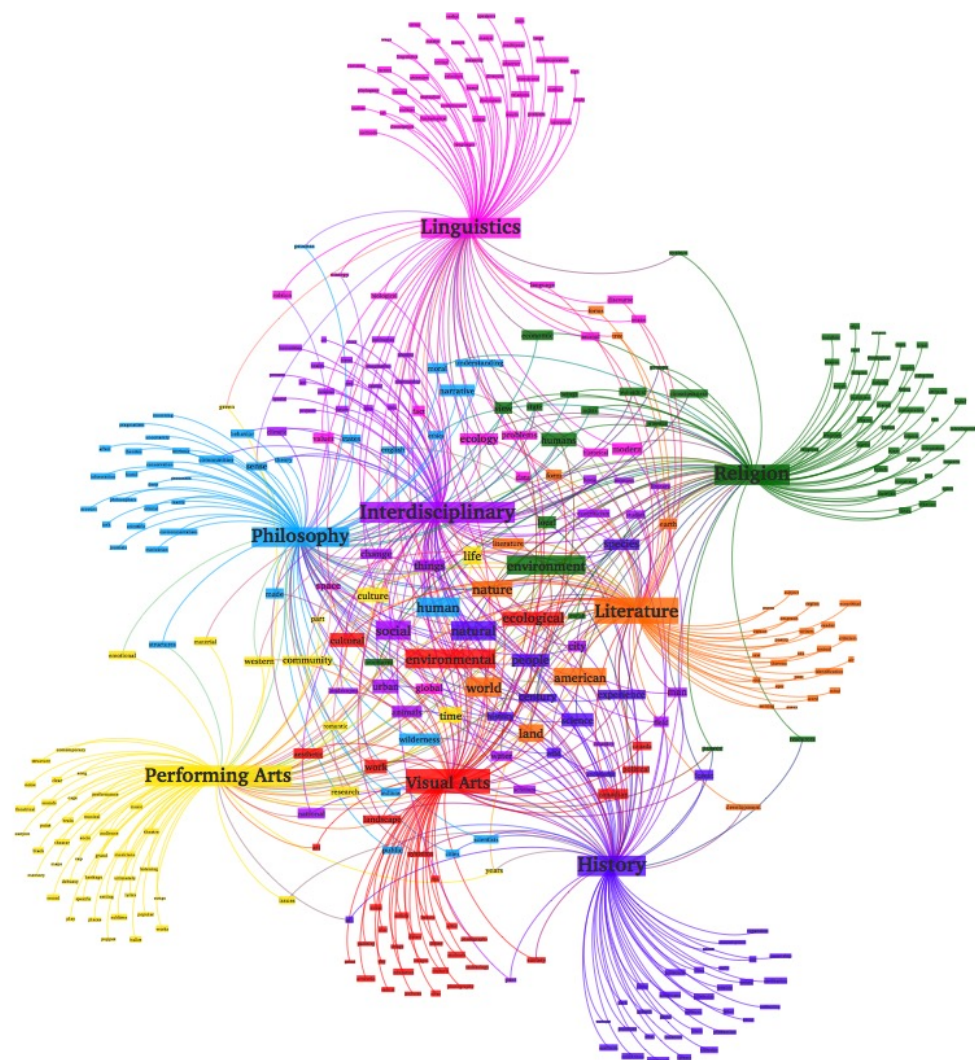


Importante notar que os tópicos “animals”, “tech” e “sports” são apenas para efeito de ilustração.

Os tópicos são representados, de fato, como uma lista ponderada de palavras. Exemplo:

- $(0.3 * \text{Cats}, 0.4 * \text{Dogs}, 0.2 * \text{Loyal}, 0.1 * \text{Evil})$ representando “animals”

Visualmente, o que temos é algo do tipo:



Os tópicos são representados por listas de palavras

human	evolution	disease	computer
genome	evolutionary	host	models
dna	species	bacteria	information
genetic	organisms	diseases	data
genes	life	resistance	computers
sequence	origin	bacterial	system
gene	biology	new	network
molecular	groups	strains	systems
sequencing	phylogenetic	control	model
map	living	infectious	parallel
information	diversity	malaria	methods
genetics	group	parasite	networks
mapping	new	parasites	software
project	two	united	new
sequences	common	tuberculosis	simulations

Podemos representar o workflow de topic modelling da seguinte maneira:

Documents

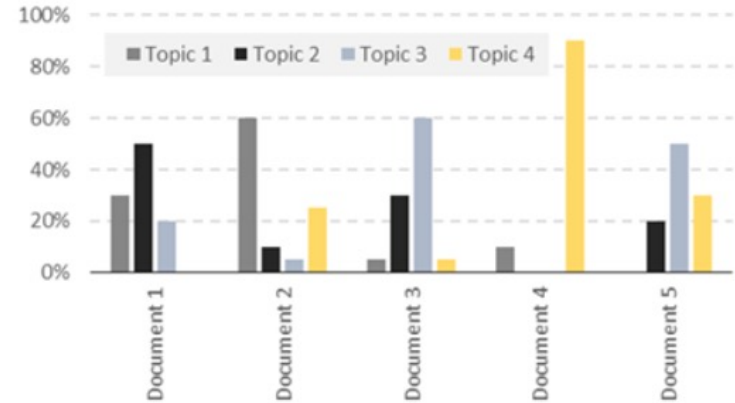


LDA

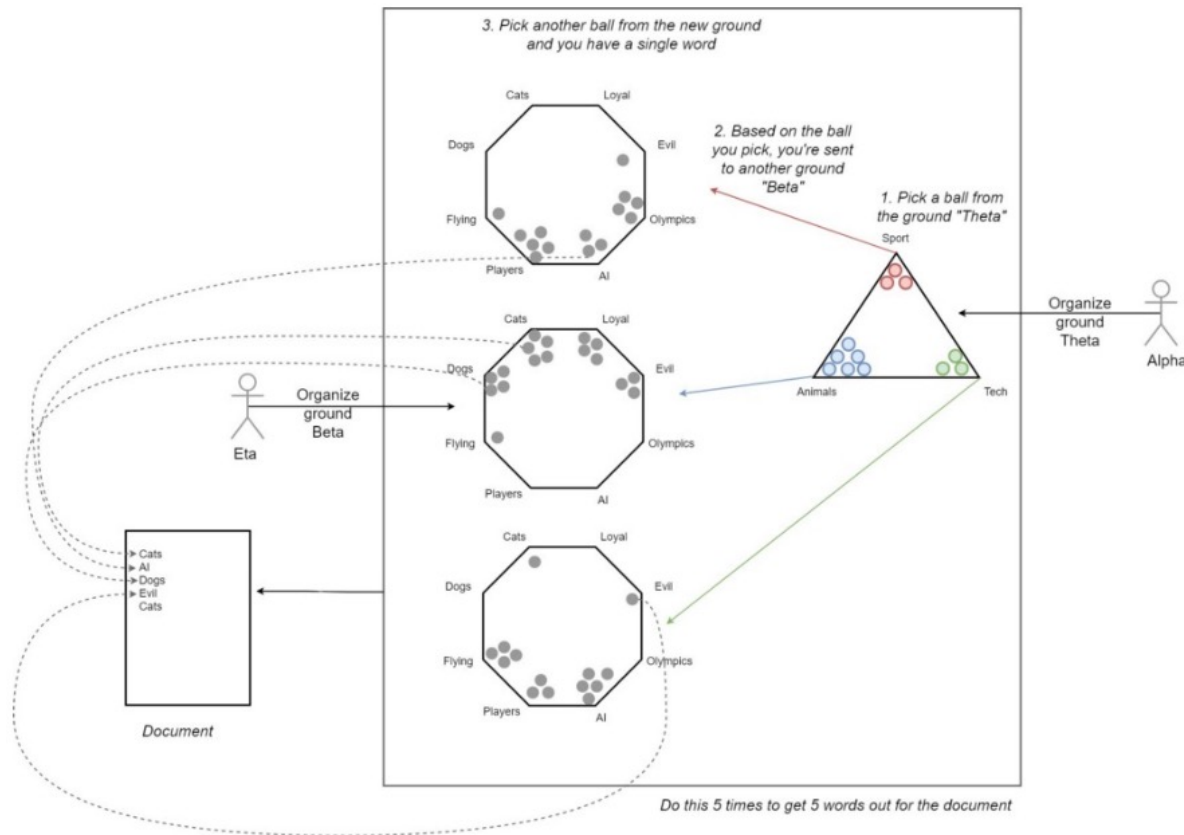
Creation of topics

	weight	words
Topic 1	3%	flower
	2%	rose
	1%	plant
...		
Topic 2	2%	company
	1%	wage
	1%	employee

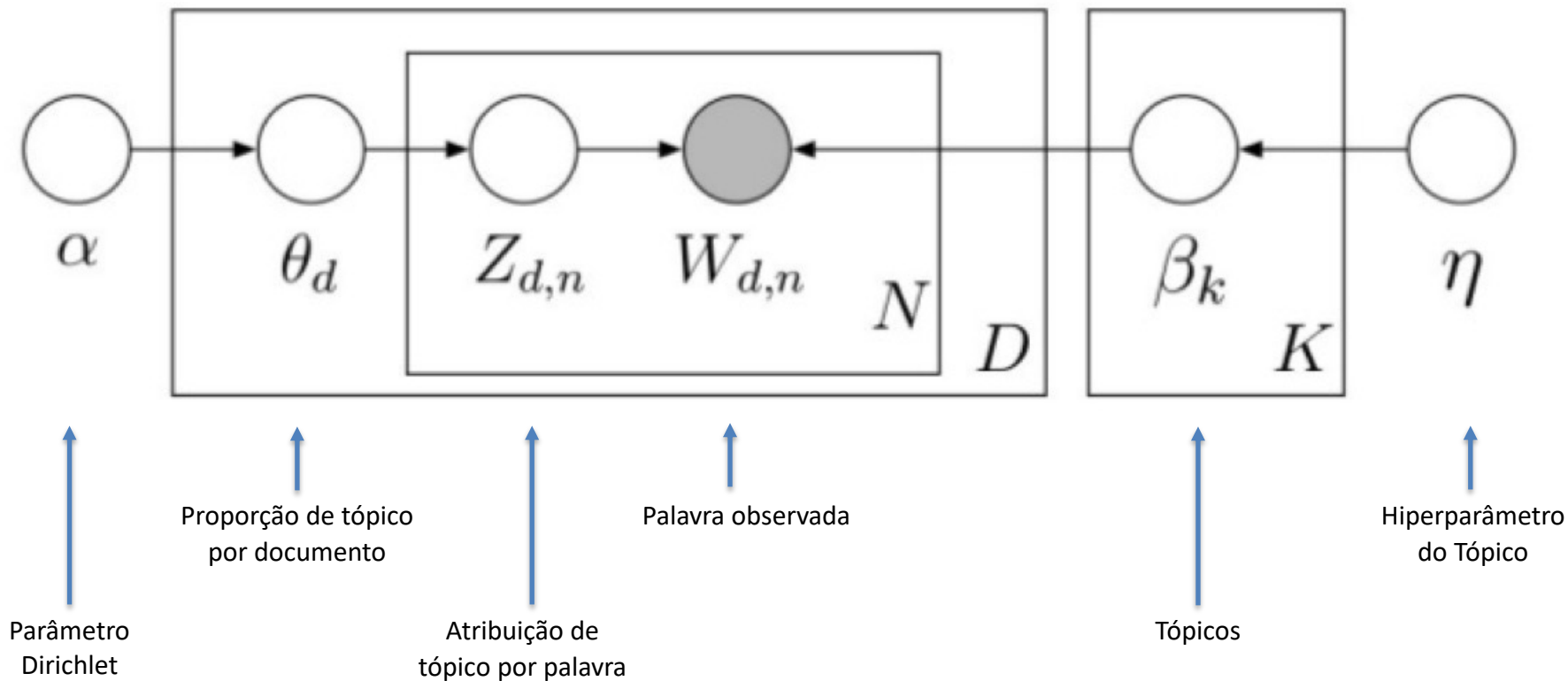
Topics allocation to documents

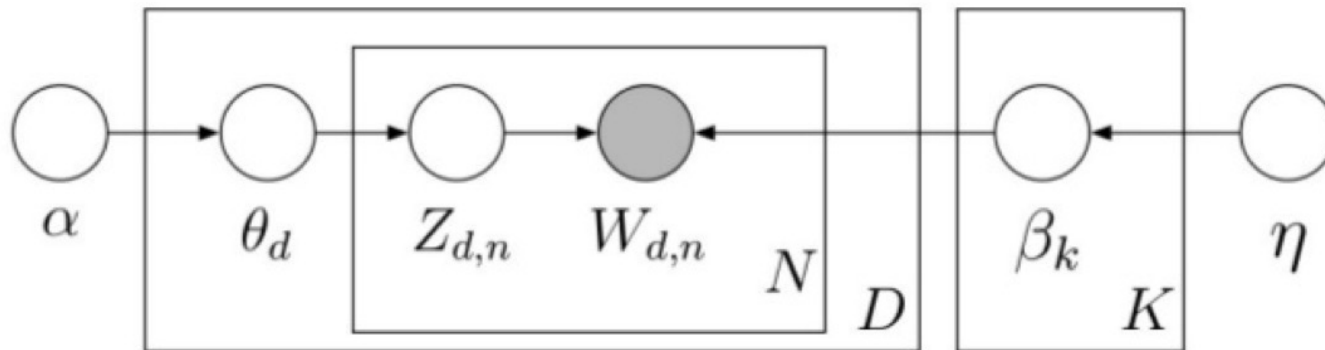


Mas precisamos, ainda, entender como LDA imagina que os documentos são gerados.



Basicamente, toda formulação matemática é baseada neste diagrama:





K é o número total de tópicos

β_k um tópico; uma distribuição sobre o vocabulário

D número total de documentos

θ_d proporção de tópicos por documento

N número total de palavras num documento

$Z_{d,n}$ atribuição de tópico por palavra

$W_{d,n}$ palavra observada

α parâmetro da distribuição de Dirichlet


η hiperparâmetro de tópico

Algumas coisas importantes de notar:

- $W_{d,n}$ é a probabilidade de uma palavra condicionada por $Z_{d,n}$ e β_k
- θ_d é a matriz de tópicos que representa a probabilidade do documento i conter palavras que pertençam ao tópico j
- $\beta_{i,j}$ representa a probabilidade do tópico i conter a palavra j
- Tanto θ_d quanto $\beta_{i,j}$ são uma Distribuição de Dirichlet

Obrigado!

profdheny.fernandes@fiap.com.br

 /dhenyfernandes

FIAP MBA⁺

Copyright © 2022 | Professor Dheny R. Fernandes

Todos os direitos reservados. Reprodução ou divulgação total ou parcial deste documento, é expressamente proibido sem consentimento formal, por escrito, do professor/autor.

FIAP