**Conceitos Aprendidos**

**RESUMO 1**

**BERTopic** – método usado para encontrar tópicos: faz se clusters e consoante as palavras dos textos em cada um dos clusters é composto pelos tweets de um tópico.

**Embedding** - Cada palavra, frase ou documento é transformado num vetor de números.

Ex: “carro” e “automóvel” → vetores muito próximos.

**c-TF-IDF** – versão adaptada do TF-IDF que, em vez de olhar para cada documento separado, calcula a importância das palavras em grupos de documentos (ou seja, nos clusters).

**MiniLM** – modelo de linguagem “leve” baseado em transformers que gera representações de texto rápidas e eficientes, mantendo uma boa qualidade mesmo sem precisar de muito poder de computação.

**HDBSCAN** - algoritmo de clustering que não só agrupa pontos semelhantes, mas também atribui uma probabilidade de um ponto pertencer a um grupo. Apesar de cada tweet pertencer apenas a um tópico (cluster), ele pode falar de mais que um tema por isso, apesar de o BERTopic falhar nessa hipótese, este algoritmo ajuda a saber se havia algum outro tema no tweet. Ou seja, apesar de ajudar a ver isso, não resolve o problema de o BERTopic não considerar a hipótese e vários tópicos no mesmo tweet.

**RESUMO 2**

**CFDTM** - modelo que acompanha a evolução dos tópicos ao longo do tempo, evitando que eles fiquem repetitivos ou que apareçam palavras que não têm nada a ver com aquele momento.

**RESUMO 3**

**HDP (Hierarchical Dirichlet Process)** - modelo que permite descobrir automaticamente quantos tópicos existem num conjunto de documentos, sem precisar definir esse número antes. Cada grupo (ex: cada documento) tem o seu próprio DP local, mas todos partilham uma distribuição global de tópicos, pois um documento pode falar de vários tópicos, não apenas de um. Assim sendo, cada documento pode estar em vários tópicos ao mesmo tempo, com pesos diferentes. Ex: um discurso político pode ser 40% economia, 30% saúde, 30% educação.

**Gibbs Sampling** -método de inferência estatística que atualiza uma variável de cada vez; funciona, mas pode ser lento e preso a más soluções.

**MCMC** – Markov chain Monte Carlo - família de métodos que usam cadeias de Markov para explorar distribuições complexas.

**Metropolis-Hastings**: regra para decidir se uma mudança (como um split ou merge) deve ser aceite ou não.

**Burn-in** - fase inicial da cadeia MCMC em que o modelo ainda está a “aquecer” antes de estabilizar.

**RESUMO 4**

**Bootstrap resampling** - uma técnica estatística que “simula” várias versões dos dados originais (networks), para testar se os padrões que aparecem são mesmo sólidos ou só ruído

**Significance clustering** - em vez de confiar cegamente no algoritmo de clustering, eles verificam se as comunidades (clusters) se mantêm estáveis quando os dados são reamostrados

**Alluvial diagrams**: gráficos que mostram como grupos se dividem (splits) ou se juntam (merges) ao longo do tempo

**RESUMO 5**

Margarida Tese

**RESUMO 6**

**Latent Dirichlet Allocation (LDA)** - É um dos modelos mais famosos de tópicos. A ideia é que cada documento mistura vários temas em proporções diferentes. Por exemplo, um artigo pode ser 70% sobre medicina e 30% sobre estatística. (precisa que número de tópicos antes de treinar)

**Dynamic Topic Models (DTM) -** Uma evolução do LDA. Em vez de assumir que os tópicos ficam fixos no tempo, o DTM acompanha como os temas mudam ao longo dos anos.

**State Space Models**: São modelos matemáticos usados para lidar com coisas que mudam ao longo do tempo. Ajudam a representar como os tópicos vão evoluindo de um ano para o outro.

**RESUMO 7**

**TOT (Topics over Time)** - versão do LDA que inclui o tempo como variável contínua. Evita confundir eventos distantes, mas com vocabulário semelhante.

**Distribuição Beta por tópico** - cada tópico tem uma curva temporal que indica quando aparece e quando desaparece.

**Diferença para DTM** - TOT não usa janelas discretas de tempo nem depende de cadeia de Markov. Tempo é contínuo.

**RESUMO 8**

**Processo de Dirichlet (DP)** - criar grupos de dados sem decidir à partida quantos grupos existem.

**Processo de Dirichlet Hierárquico (HDP)** - Em vez de cada grupo inventar tópicos do zero, todos usam um “menu global” de tópicos, mas cada grupo escolhe as suas próprias combinações.

**Chinese Restaurant Process (CRP)** - Uma metáfora: tens um restaurante chinês com mesas infinitas. Cada novo cliente (documento) pode sentar-se numa mesa já ocupada (reforçando a popularidade desse prato) ou abrir uma mesa nova (novo cluster). – permite criar tópicos sem ter um número definido de clusters

**Chinese Restaurant Franchise (CRF)** (Versão do HDP) - É como ter uma cadeia de restaurantes chineses (um documento). Cada restaurante (documento) tem várias mesas (clusters locais (dentro de um documento)) e os clientes agrupam se por essas mesas. Cada mesa depois pede um prato do menu global (tópico global). – pode haver vários pratos por documento e não se precisa de definir o número de clusters.

**Diferença para o RESUMO 3** – este é mais explicativo do modelo em si (fundamentos teóricos e utilidade). O outro é mais inferência prática de como implementar o HDP de forma mais eficaz através de um algoritmo de MCMC (split-merge).

**RESUMO 9**

**Evolutionary clustering** – Clustering de dados que mudam com o tempo, garantindo que os clusters de hoje façam sentido, mas também se pareçam com os de ontem.

**Snapshot quality** - Mede quão bem o agrupamento de agora representa os dados deste momento

**History cost** - Mede quanto os agrupamentos mudaram em relação ao passo anterior

**Change parameter (cp)** - Botão de ajuste que decide se preferes mais estabilidade (valor alto) ou mais precisão instantânea (valor baixo)

**RESUMO 10**

**Optimal Transport (OT)** - É uma maneira matemática de medir a diferença entre duas distribuições (por exemplo, dois tópicos representados por palavras). Mede o “custo” para mover a probabilidade de uma distribuição para outra.

**Entropic Regularization** - Adiciona uma penalização suave que “espalha” a massa transportada, tornando o cálculo mais rápido e estável. É como permitir pequenas imprecisões para ganhar muita velocidade.

**Sinkhorn Algorithm** - Um método iterativo muito eficiente que ajusta uma matriz para que as somas das linhas e colunas coincidam com as distribuições originais — e com isso calcula a distância.

**Sinkhorn Distance** - A distância OT regularizada com entropia — mais rápida e mais suave, ideal para grandes datasets.

**Matrix Scaling** - Processo de multiplicar linhas e colunas por fatores até as margens ficarem corretas — é o coração do algoritmo de Sinkhorn.

**Distribution Similarity** - A ideia geral de comparar duas distribuições — aqui, aplicada a tópicos representados por pesos de palavras.

**RESUMO 11**

**Two-sample test** - É um teste estatístico para ver se dois conjuntos de dados vêm da mesma distribuição. Exemplo: tweets democratas vs. republicanos falam do mesmo tema?  
**Kernel** - Função matemática que mede semelhança entre pontos num espaço possivelmente de alta dimensão — ajuda a captar relações não lineares.

**Reproducing Kernel Hilbert Space (RKHS)** - Um espaço matemático onde cada distribuição é representada como um vetor (embedding). Lá, é fácil medir distâncias.

**Maximum Mean Discrepancy (MMD)** - Mede a diferença média entre duas distribuições no espaço do kernel. Se for 0, as distribuições são iguais; se for grande, são diferentes.

**Non-parametric** - Não assume nenhuma forma específica (como normal ou gaussiana) — ideal para dados complexos como texto.

**Bootstrap** - Técnica para estimar a incerteza de um teste repetindo a amostragem várias vezes.

“To statistically validate whether two topic distributions correspond to distinct narratives, we employ the Maximum Mean Discrepancy (MMD) test (Gretton et al., 2012), a non-parametric kernel-based two-sample test. The MMD quantifies the divergence between the mean embeddings of two distributions in a reproducing kernel Hilbert space. Applying this test to topic representations across consecutive time windows provides quantitative evidence for true merges or splits, beyond simple similarity metrics.”

**RESUMO 12**

**DTQ (Dynamic Topic Quality)** - Esse mede como a qualidade dos tópicos muda ao longo do tempo.

**TTQ (Temporal Topic Quality)** – consistência temporal de cada tópico ao longo do tempo

**TQ (Topic Quality)** – coherence + diversity

**Coherence (Coerência)** – A coerência continua sendo o quanto as palavras dentro de um tópico realmente se relacionam.

**Diversity (Diversidade)** - E a diversidade mede o quanto os tópicos são diferentes uns dos outros.