Similaredade entre Documentos e suas Ramificações

Ronnald Rezende Machado
Instituto de Computação
Universidade Federal Fluminense - Niterói

Resumo

Automatizar a detecção de documentos similares é uma tarefa que trás consigo diferentes obstáculos. Fazendo uso de técnicas de aprendizagem de máquina, o trabalho em questão discute e propõe uma modelagem eficiente para a solução deste desafio.

Introdução

A Identificação de semelhança entre objetos é uma tarefa fundamental para o desenvolvimento do pensamento. Consagrada pelo aforismo "A é igual a A" o princípio da identidade se consolidou como uma das bases do pensamento lógico.

Levando em conta a importância da identificação de semelhanças entre documentos em ambientes educacionais, o trabalho em questão implementa técnicas de vetorização de documentos , sumarização de texto, discute técnicas de comparação de similaridade entre os vetores gerados e faz uso dos resultados obtidos para a construção de grafos não direcionados tendo como matéria prima, textos coletados em um curso online.

Preliminares

Antes do detalhamento do trabalho desenvolvido, faz-se necessário abordar os principais conceitos teóricos sobre os quais se baseiam as ferramentas e experimentos detalhados neste relatório.

Representação vetorial de palavras

Algumas técnicas que tem por objetivo a representação vetorial de palavras como o word2vec(Mikolov et al. 2013), glove(Pennington, Socher, and Manning 2014) e fasttext(Joulin et al. 2016) vêm recebendo atenção por alcançarem resultados significativos em problemas que envolvem linguagem natural.

O word2vec faz uso de redes neurais profundas para aprender o contexto das palavras alvo levando em consideração, as palavras que estão ao seu redor. Para tanto é possível usar dois algoritmos diferentes. O algoritmo chamado de CBOW não leva a ordem das palavras em consideração, sendo mais rápido de executar porém tende a ser menos preciso. Já o algoritmo skip-gram leva a ordem das palavras do texto em consideração.

Representação vetorial de documentos

Visando a representação vetorial de documentos , uma extensão do word2vec foi proposta por Le e Mikolov e denominada de doc2vec(Le and Mikolov 2014). O doc2vec modifica o algoritmo do word2vec com a intenção de considerar blocos maiores de texto, sendo possível relacionar rótulos com palavras ao invés de palavras com outras palavras. Assim como no word2vec, é possível usar dois algoritmos diferentes na fase de treinamento. Denominado como PV-DM (*Distributed Memory version of Paragraph Vector*) ele faz uso da ordem das palavras e também do rótulo do documento para o cálculo do vetor. Já no algoritmo PV-DBOW (*Distributed Bag of Words version of Paragraph Vector*) o rótulo do documento também é usado, porém a ordem das palavras não é levada em consideração.

Similaridade de documentos

Com o intuito de medir a similaridade entre os vetores gerados , são usados duas técnicas ao longo do trabalho:similaridade de cosseno e *word mover's distance*(Kusner et al. 2015).

Similaridade de cosseno

Sendo os vetores A e B, a similaridade de cosseno se dá por:

Figura 1: Similaridade de cosseno

$$\cos\theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \cdot \|\vec{b}\|}$$

Os valores retornados por essa função podem variar de -1 a 1 sendo o valor 1 correspondente a medida máxima de similaridade.

Word Mover's Distance

É um método que permite avaliar a similaridade entre dois documentos mesmo quando eles não tenham palavras em comum. No trabalho original os autores fizeram uso dos resultados obtidos através do word2vec para alimentar seu algoritmo.

Sumarização

Visando facilitar a avaliação da similaridade obtida, o presente trabalho implementa o algoritmo de sumarização TextRank(Mihalcea and Tarau 2004).

O algoritmo TextRank transforma cada sentença do texto em um grafo onde cada nó representa uma sentença e as arestas representam a medida de similaridade entre todas as sentenças em um determinado texto como demonstrados na figura 2.Tendo Si e Sj como sentenças e w como palavras, a similaridade é calculada por:

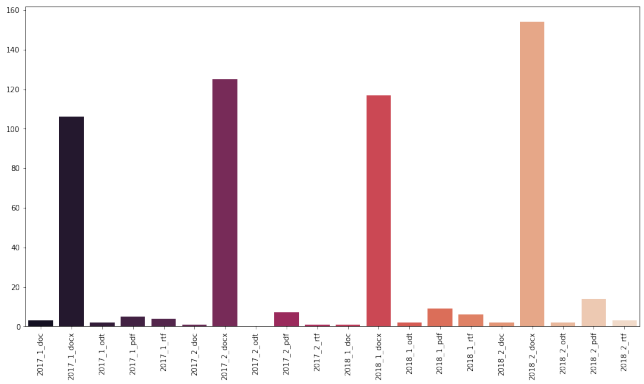
Figura 2: TextRank

$$Sim(S_i, S_j) = \frac{|\{w_k | w_k \in S_i \& w_k \in S_j\}|}{log(|S_i|) + log(|S_j|)}$$

Processamento dos documentos

O primeiro passo necessário para a construção do sistema se resume na análise quantitativa dos arquivos que serão avaliados.

Figura 3: Tipos de documentos disponíveis



Como observado na figura 3 temos predominantemente arquivos em formato docx, totalizando 502 arquivos únicos. Como o número de arquivos no formato docx é suficiente para o desenvolvimento da análise, os documentos em outros formatos foram desconsiderados.

O segundo passo consiste na normalização dos textos, ou seja, a retirada de caracteres especiais e pontos finais para os textos que servirão de entrada para o doc2vec e apenas a normalização, sem a retirada dos pontos finais, para os textos que servirão de entrada para o algoritmo textrank, que por definição precisa de um delimitador de sentença.[1]

Doc2vec

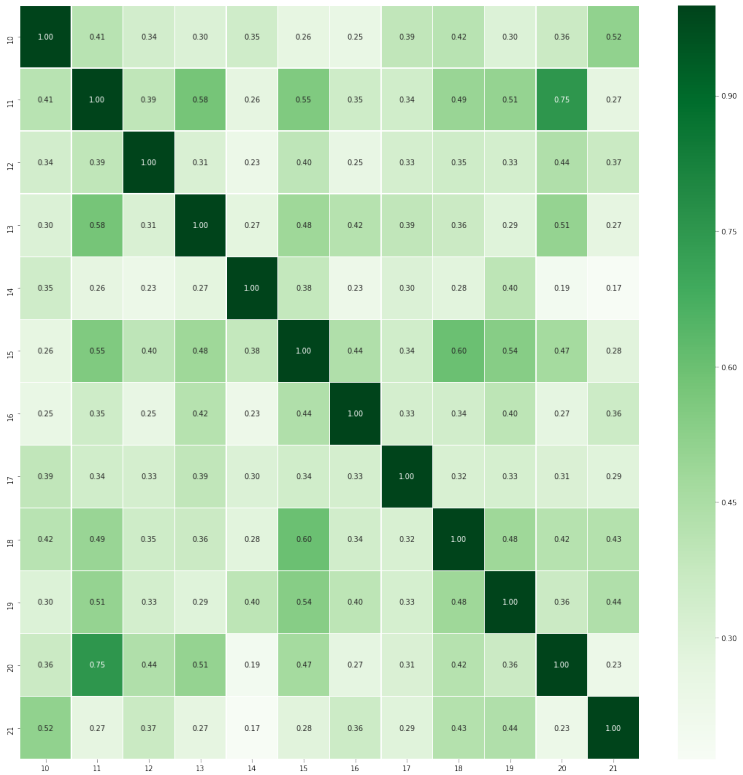
O modelo usado ao longo do trabalho é o PV-DM, configurado com os seguintes hiperparâmetros:

- Dimensionalidade do vetor de características : 100
- Épocas :10
- Taxa de aprendizagem Inicial: 0.025
- Distância máxima entre a palavra atual e a prevista em uma frase: 3
- Decaimento da taxa de aprendizagem a cada época: -0.0002

Medidas de Similaridade entre documentos

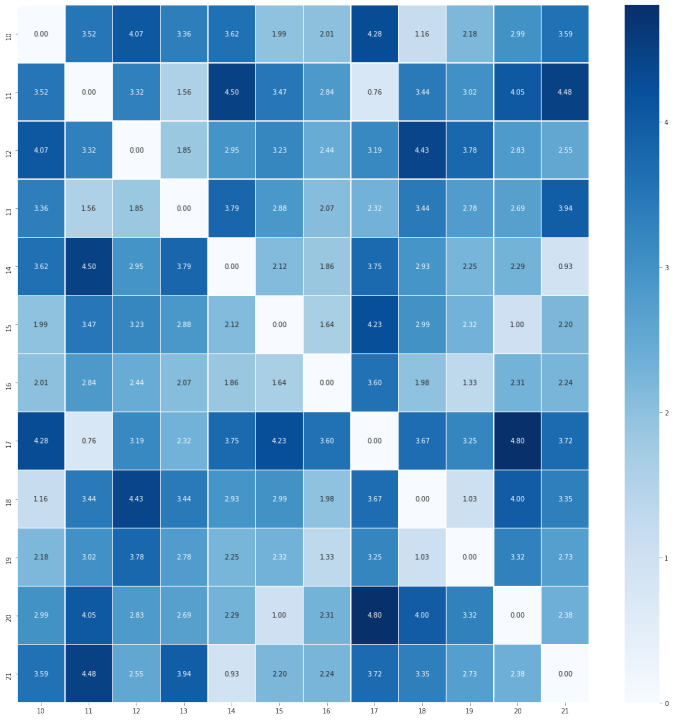
Após obter os vetores através do doc2vec duas medidas de similaridade são usadas. A intenção desse tópico é comparar o resultado obtido entre as duas e justificar a escolha do melhor resultado para a construção do grafo de relações entre os documentos. Para exemplificar a comparação e facilitar o entendimento, matrizes de correlação envolvendo o documento 10 até o documento 21 foram construídas e são representadas pelas figuras X e Y.

Figura 4: Similaridade de cosseno



[1] Por questões de privacidade, o nome dos autores são removidos dos textos.

Figura 5: Word mover's distance



Tendo em conta a quantidade de comparações possíveis, selecionei o exemplo da comparação entre os documentos '2017-2-257756' e '2017-2-257741'. O valor obtido na similaridade do cosseno é 0.75, ou seja, assumindo que o valor esperado para documentos idênticos é 1 em uma escala entre 1..-1. Já usando o word mover's distance o valor obtido é de 3.84, sugerindo assim, pouca similaridade entre o conteúdo dos documentos comparados.

Por definição, documentos com alto grau de semelhança, devem possuir resultados similares também no algoritmo textrank. Para o documento '2017-2-257756' , a sumarização resultante foi:

- *"**Contudo, em sentido oposto, ao mesmo tempo em que observamos a crescente diversidade e pluralismo religioso no mundo, continuamos a verificar a manifestacao da intolerancia religiosa.**Por mais que, por muitas vezes, tentemos negar sua existencia, talvez ate como forma de vergonha, fato e que a intolerancia, nao apenas religiosa, vem ganhando forca em nossa sociedade, como podemos concluir com base nos textos, na analise das explicacoes e definicoes desse flagrante movimento de intolerancia e fundamentalismo."*

Já para o documento '2017-2-257741' o resultado foi:

- *"O cenario de conflitos religiosos e de intolerancia faz ainda menos sentido quando nos deparamos com informacoes, dados e estatisticas acerca da diversidade religiosa cada vez maior no Brasil e no mundo.**Contudo, inversamente, ao mesmo tempo em que observamos a crescente diversidade e pluralismo religioso no mundo, nos deparamos com o fenomeno de intolerancia religiosa.**"*

Para validar usamos mais um exemplo, agora com os do documento '2017-1-182232' e '2017-2-257741' que possuem um índice de 0.98 na similaridade de cosseno e 2.11 no word mover's distance. O resultado da sumarização do documento '2017-1-182232' foi:

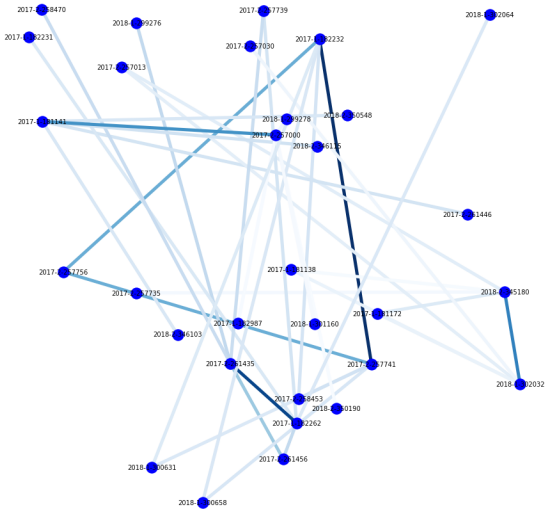
- "*O cenario de conflitos religiosos e de intolerancia faz ainda menos sentido quando nos deparamos com informacoes, dados e estatisticas acerca da diversidade religiosa cada vez maior no Brasil e no mundo.Contudo, inversamente, ao mesmo tempo em que observamos a crescente diversidade e pluralismo religioso no mundo, nos deparamos com o fenomeno de intolerancia religiosa.*"

Tendo em vista a similaridade latente resultante do algoritmo de sumarização podemos concluir que a similaridade de cosseno obtém melhores resultados com os dados e o modelo desenvolvido neste estudo. Além disso, vale destacar o tempo de execução dos algoritmos. Ao executar a similaridade de cosseno, o tempo gasto na comparação entre todos os 502 documentos foi de 15 segundos contra 426 segundos no word mover's distance.

Visualização em Grafos

Considerando a alta dimensionalidade dos dados e a consequente dificuldade de navegação entre os documentos e suas similaridades, este trabalho propõe a construção de grafos não direcionados interconectando os documentos. Além de proporcionar uma navegação aprimorada entre os documentos, a estrutura de grafos possibilita a implementação de algoritmos de mineração específicos. É possível construir grafos que exibam todos os documentos ou apenas documentos que obedeçam a uma determinada regra. Nos exemplos a seguir, o grafo G é dado por : Para todo documento X que possua similaridade com algum outro em uma taxa de 0.65, obtenha os 5 documentos mais similares a X. Considerando essa regra, o grafo G obtido possui 31 nós.

Figura 6: Grafo com arestas valoradas



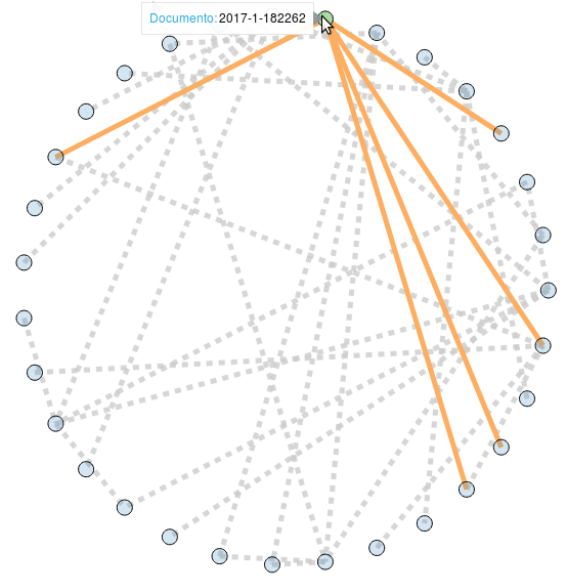
Na figura 6 os nós representam os documentos e as arestas representam o valor da similaridade entre os documentos, sendo que, as arestas mais escuras simbolizam uma maior similaridade.

Uma navegação mais aprimorada pode ser contemplada na figura 8 e na figura 9. O grafo em questão foi desenvolvido visando a experiência de um eventual usuário final.

Figura 7: Grafo circular

Ao clicar em cada nó é possível visualizar seu rótulo bem como um subgrafo que demarca apenas as suas conexões.

Figura 8: Seleção de subgrafo



Detecção de Comunidades

Para a detecção de comunidades o algoritmo Girvan-Newman(Girvan and Newman 2002) foi usado. Em linhas gerais o algoritmo de Girvan-Newman detecta comunidades removendo progressivamente as arestas do grafo original. O algoritmo remove a aresta com a maior centralidade a cada passo da interação. A medida que o gráfico se divide em pedaços, a estrutura das comunidade é obtida.

Para o grafo G obtivemos 6 comunidades, com os seguintes documentos:

1. '2017-1-181138','2017-1-181172','2017-2-257013','2017-2-257030','2017-2-257735','2018-1-302032','2018-2-345180'
2. '2017-1-181141','2017-2-261446', '2018-2-346103','2018-2-346115','2018-2-350548'
3. '2017-1-182231','2017-1-182262', '2017-2-261456', '2018-1-302064'
4. '2017-1-182232','2017-2-257741', '2017-2-257756', '2017-2-258453', '2018-1-300631', '2018-1-300658'
5. '2017-1-182987', '2017-2-257000', '2018-1-299278', '2018-1-301160', '2018-2-350190'
6. '2017-2-257739', '2017-2-258470', '2017-2-261435', '2018-1-299276'

Conclusão

Tendo em vista os resultados obtidos é possível afirmar que o uso do doc2vec aliado ao cosseno de similaridade proporciona bons resultados na busca pela similaridade entre documentos. Outra conclusão importante foi a constatação de que a similaridade de cosseno obteve um melhor desempenho se comparado ao word mover's distance. Quanto ao algoritmo de sumarização textrank, ficou claro que ele trabalha muito bem em conjunto com o doc2vec, fornecendo um resumo do documento alvo de forma eficiente. Para a visualização dos resultados obtidos a implementação de grafos se fez necessária e trouxe como bônus uma visão facilitada do universo de documentos bem como a separação dos documentos em comunidades.

Como extensão deste trabalho sugiro que cada nó no grafo receba como atributo o texto obtido através do algoritmo de sumarização além do rótulo.

Fica claro que a abordagem usada neste trabalho é efetiva na identificação de plágios, criação de caminhos entre documentos e construções de subgrupos de documentos com forte correlação.

A modelagem usada pode ser aplicada em contexto educacionais como também pode ser estendida para outros domínios.

References

Girvan, M., and Newman, M. E. J. 2002. Community structure in social and biological networks. *PNAS* 99(12):7821–7826.

Joulin, A.; Grave, E.; Bojanowski, P.; and Mikolov, T. 2016. Bag of tricks for efficient text classification.

Kusner, M.; Sun, Y.; Kolkin, N.; and Weinberger, K. 2015. From word embeddings to document distances. In Bach, F., and Blei, D., eds., *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, 957–966. Lille, France: PMLR.

Le, Q. V., and Mikolov, T. 2014. Distributed representations of sentences and documents.

Mihalcea, R., and Tarau, P. 2004. TextRank: Bringing order into texts. In *Proceedings of EMNLP-04and the 2004 Conference on Empirical Methods in Natural Language Processing*.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space.

Pennington, J.; Socher, R.; and Manning, C. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 1532–1543.