Estudo sobre a variação temporal de Word Embedding

Clovis Henrique da Silva Chedid
DRE: 119004999
cchedid@cos.ufrj.br

Pedro Vitor Marques Nascimento
DRE: 116037448
pedromn@cos.ufrj.br

Thiago da Mota Souza
DRE: 119004826
thiagosz@cos.ufrj.br

***Resumo*—No contexto da disciplina Busca de Informações e Mineração de Textos da pós-graduação, este trabalho se propõe a descrever o processo de pesquisa e aprendizado neste tópico, além de documentar em detalhes a reprodução dos experimentos de um artigo da área.**

***Index Terms*—dynamic, word, embedding**

I. Introdução

Linguagens tem como uma de suas características a evolução e, portanto, a mudança. Uma destas mudanças acontece no significado nas palavras, de forma que o contexto em que são utilizadas pode mudar ao longo do tempo. Bloomfield and Hockett em seu livro *Language* [1] define isto como Mudança Semântica (*Semantic Change* no inglês). No contexto de NLP, Kutuzov et al. descreve um crescente interesse pela visualização e cálculo dessas mudanças usando *Word Embeddings*.

Este trabalho tem como objetivo descrever o processo de pesquisa e aprendizado por parte dos autores neste tópico, além de documentar em detalhes a reprodução do artigo de Yao et al..

Em termos de estrutura, este relatório começa descrevendo os principais conceitos relacionados ao campo na seção Fundamentação Teórica. Depois disso descrevemos o processo de estudo e reprodução do artigo escolhido. Por fim, na seção IV, documentamos os resultados.

II. Fundamentação Teórica

Este estudo foi feito utilizando como base o ferramental teórico da área de Processamento de Linguagem Natural (NLP). Nesta seção iremos introduzir brevemente os principais conceitos que embasam o trabalho.

A. Word Embedding

O processo de Word Embedding (WE) envolve mapear palavras para um espaço contínuo e multidimensional. Este mapeamento utiliza o contexto (as frases) que essas palavras se encontram, de forma que palavras de significado parecido fiquem próximas espacialmente [4].

Entende-se que em um espaço contínuo hipotético, palavras usadas em contextos semelhantes como ”rei” e ”rainha” ficariam próximas enquanto ”pneu” e ”pizza”, dois conceitos pouco similares, ficariam distantes.

Os WE podem servir de extratores de features para tarefas de classificação de texto. Muitos modelos de machine learning precisam receber os dados na forma de vetores e não são capazes de receber os dados brutos na sua forma de texto, portanto os WE são necessários como pré-pŕocessamento.

B. Mudança Semântica

Em seu livro *Language* [1], Bloomfield and Hockett definem *Semantic Change* (ou Mudança Semântica, em tradução livre) como as modifições no sentido das palavras, portanto nos contextos que são utilizadas.

Sabendo que se os contextos em que as palavras são usadas são passíveis de mudança, seus embeddings podem variar ao longo do tempo. Um exemplo dessa mudança seria o uso da palavra ”amazon”, que com o passar dos anos se aproximou mais de tecnologia e menos de natureza [3].

Como visto no trabalho de [3], esta característica da linguagem permite o uso de Word Embeddings para visualizar o significado de palavras separados em recortes temporais. Além disso, como comentado em [5], essas mudanças vão na via oposta à visão que em um corpus grande o sentido das palavras é estático.

1) Word Embedding Temporal: Considerando a Mudança Semântica, diversos autores propuseram métodos para computá-la [5]. Bamler and Mandt explica que pelo menos até 2016 as abordagens consistiam em dividir o corpus em diversas partes, cada uma relacionada a um intervalo de tempo, para então analisá-las separadamente. O autor enumera três problemas que podem surgir com esse método:

- Dividir o dataset em muitas partes pode deixar poucos dados para o treinamento de cada recorte. Isso introduz a possibilidade de *overfitting*.
- Pela propriedade não-convexa dos modelos de WE, treinar duas vezes o mesmo dado pode retornar valores de embedding diferentes.
- Por conta do tamanho finito dos datasets, é provável a existência de ruido (*noise*) nos resultados. Diferenciar ruído de uma mudança semântica vira uma questão, principalmente em recortes sucessivos.

2) Alinhamento de Embeddings Temporais: Em seu artigo [2], Kutuzov et al. salienta que a comparação entre embeddings de diferentes recortes temporais gerados conforme o procedimento descrito em II-B1 não pode ser feita de forma direta quando os embeddings são treinados de forma independentes em cada um dos recortes. Isto acontece porque os mapeamentos assim gerados não tem as dimensões

alinhadas, isto é, os vetores de WE não têm a mesma base canônica. Trabalhos anteriores Kutuzov et al. desenvolveram estratégias para resolver o problema do alinhamento após o treinamento independente dos embeddings. Zhang et al. propôs que esse alinhamento seja feito usando como base palavras cujo significado é aproximadamente fixo, as palavras âncoras [6]. Os autores definem os critérios para escolher boas palavras âncoras, um deles é a alta frequência de uso. Tendo as palavras âncora definidas, o alinhamento é feito a partir de um problema de otimização.

A abordagem introduzida por Yao et al. consiste em abordar o problema do alinhamento concomitantemente ao do treinamento dos embeddings em difrentes recortes temporias pela modificação da função de custo utilizada. Os autores introduziram um termo de regularização que penaliza grandes mudanças entre embeddings de diferentes recortes. Esse termo de penaliza o distanciamento dos embeddings de uma palavra entre recortes temporais consecutivos. Junto com o termo de regularização, é introduzido mais um hiper-parâmetro a processo treinamento que funciona como um multiplicador ao novo regularizador criado. Este hiper-parâmetro tem uma faixa de valor que o projetista dos embeddings tem que ajustar, pois se muito próximo de zero, o efeito do novo regularizador é pequeno e o problema do alinhamento não será resolvido. Por outro lado, se for um valor muito grande, os Embeddings gerados serão pouco sensíveis as mudanças semânticas em diferentes recortes. No limite, um valor muito alto deste hiper-parâmetro reduz a estratégia ao cálculo de um embedding tradicional que não leva em consideração as diferenças no tempo.

C. Pointwise Mutual Information

Os autores desenvolveram os embeddings para que o produto interno entre dois embeddings de palavras distintas em um mesmo recorte de tempo seja igual ao PMI(Pointwise Mutual Information) entre as mesmas. Essa métrica captura a informação mutua na da co-ocorrência de pares de palavras em em uma vizinhança de tamanho fixo, escolhida como sendo 5 pelos autores. O problema de treinamento dos embeddings portanto é desenvolvido como sendo um problema de regressão.

III. Metodologia

Para reproduzir os resultados obtidos pelos autores, utilizou-se o mesmo dataset disponível no repositório que descrito em III-A para retreinar os embeddings III-B. Após concluído o treinamento, foram gerados gráficos com trajetórias conforme descrito em III-C. Ao que seguiu uma avaliação qualitativa dos resultados

A. Datasets

O corpus utilizado foi formado a partir de 99.872 artigos em inglês do jornal *The New York Times* [7] publicados entre Janeiro de 1990 e Julho de 2016 que foram divididos em recortes anuais de tempo, numerados de 0 a 26. No total Os autores removeram palavras deste corpus com frequência inferior a 200, mas não filtraram números ou stop-words. No total o vocabulário obtido tem 20.936 palavras. Conforme descrito em II-C, os autores extraíram desse corpus o PMI entre todos os pares de palavras para cada um dos períodos de tempo, gerando assim 27 arquivos no formato *csv* com tês colunas: o ínidice da palavra *A*, o índice da palavra *B*, o valor de PMI entre *A* e *B*. São esses os arquivos utilizados pelas rotinas de treinamento. Estes arquivos totalizaram 6,5GB de dados.

B. Treinamento dos Embeddings

O treinamento dos embeddings foi feito em uma máquina com processador Intel(R) Core(TM) i7-4790 de 3.60GHz, com 16GB de memória RAM, GPU NVIDIA GeForce GTX 1050 com 4GB de memória e com sistema Linux Ubuntu 18.04.4 LTS. As rotinas de treino foram desenvolvidas em Python 2.7 e utilizam a biblioteca Numpy. O treinamento demorou 10 min com esse sistema.

Foram mantidos os hiper-parâmetros estabelecidos pelos autores do estudo original: dimensão dos embeddings, 50; regularizador de Frobenius, 10; regularizador de posto, 100; regularizador de continuidade, 50. O problema de otimização foi resolvido utilizando a otimização por Gradiente Descendente com apenas 5 épocas, o que também foi mantido do trabalho original.

O resultado do treinamento foram 5 arquivos com o resultado parcial de cada iteração contendo os embeddings de todas as palavras do corpus para cada um dos 27 recortes temporais. Esses arquivos totalizaram 2,2GB de dados. As trajetórias geradas e apresentadas em IV foram geradas a partir do resultado da última iteração.

C. Gráfico de Trajetórias

A visualização de um hiper-espaço de dimensão 50 exige estratégias de projeção para um espaço de dimensão compreensível pelo ser humano. O objetivo é mostrar trajetórias dos embeddings de determinadas palavras ao longo do recortes temporais, explicitando a sua vizinhança para assim avaliar a mudanças semântica das mesmas. Para esse fim, os autores utilizaram a técnina de Projeção de Mainfold t-SNE(t-distributed Stochastic Neighbor Embedding), que minmização o Divergetne de Kullback-Leibler para mapear os espaço de alta dimensionalidade em um espaço 2D que posicione embeddings neste espaço segundo sua proximidade no espaço de mais alta dimensionalidade. Como uma técnica de machine learning, essa projeção deve ser aprendida dos próprios dados. Foi utilizada a implementação da biblioteca sklearn para esse fim.

As palavras que aparecem na vizinhanças das trajetórias são escolhidas a partir dos embeddings em seu espaço 50-dimensional a seguinte forma. Dada a palavra cuja trajetória se deseja visualizar, para cada corte de tempo do corpus, adicione as palavras que tem maior PMI com a palavra selecionada.

Com tais trajetórias, é possível ver a mudança semântica de uma palavra a partir da proximidade com palavras em sua trajetória.

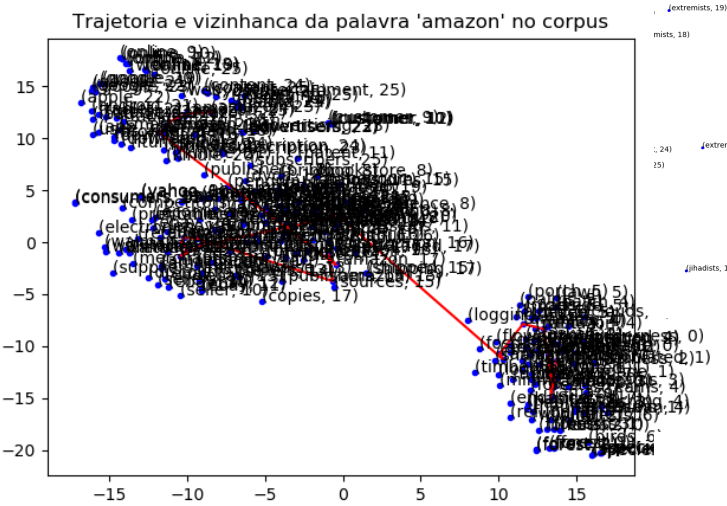

Figura 1. Vizinhança dos Embeddings da Palavra Amazon

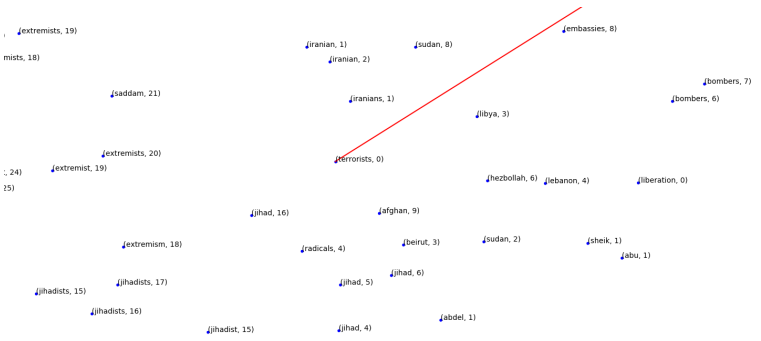

Figura 2. Vizinhança do Embedding de microsoft

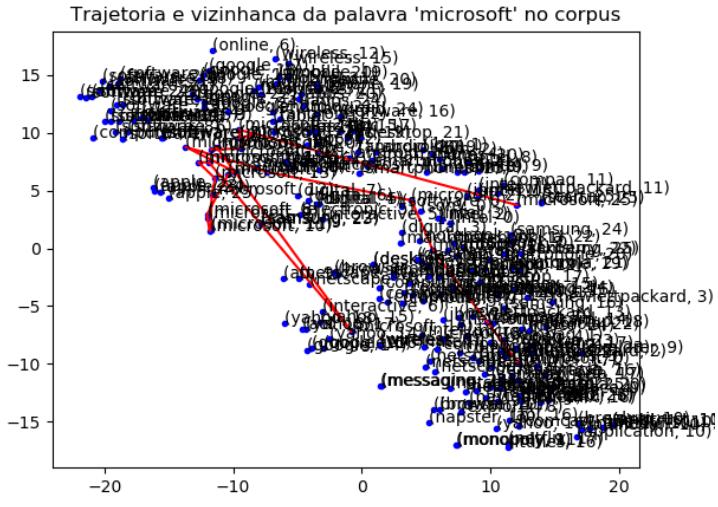

Figura 3. Vizinhança do Embedding de microsoft

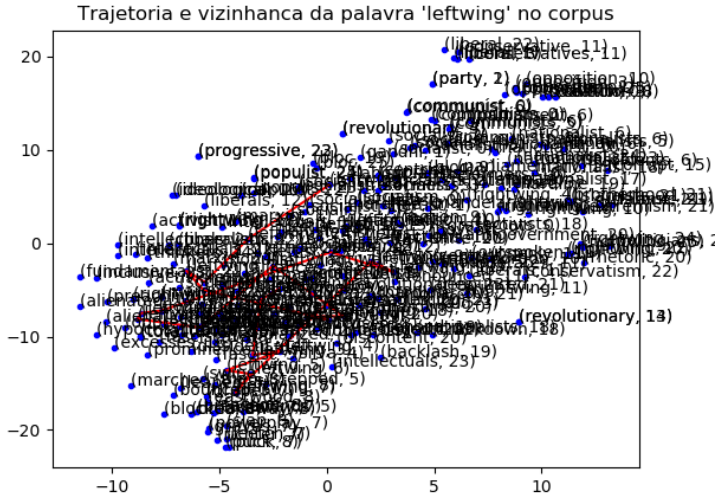

Figura 4. Vizinhança do Embedding de leftwing

IV. Resultados

Nesta sessão apresentamos algumas das trajetória obtidas utilizando o procedimento e os dados descritos em III. Seguem alguns exemplos de tais trajetórias e algumas interpretações que pudemos tirar das mesmas.

Na Figura 1 temos a trajetória da palavra *Amazon* ao longo do tempo do estudo para uma vizinhança de tamanho 10. Nela é possível perceber dois clusters de pontos em torno dos embeddings de Amazon para o recorte de tempo 0 e o recorte de tempo 26. No primeiro cluster há palavras relacionadas a ecologia na vizinhança do embedding, equanto que no segundo essas palavras da vizinhança são relaciandas a tecnologia. Isso acontece porque a a palavra Amazon no corpus passsou a aprecer muito mais frquentemente em textos ligados a tecnologia nos últimos anos do que quando no contexto de ecologia.

a Figura 2 mostra um a vizinhança de perto da palavra terrorists. Neste figura destacamos que no Corte de tempo 0 a palavra aparece próxima a palavras do Corte de Tempo 26, como Drone por exemplo. Isso talvez transpareça mais sobre a mudança semântica da própria palavra Drone do que da palavra terrorist.

A trajetória da palavra Microsoft, 3, apresenta um aspecto de zig zag. Na maior parte, a vizinhança é composta de nomes de empresas do ramo de tecnologia e percebe-se que algumas como Neptser aparece na vizinhança no início da trajetória, enquanto outras como google aprecem nas viznhanças no fim da trajetória. A palavra leftwing, Figura 4, é outra cuja trajetória tem evolução errática entre termos como: revolução, ditadura, progressista e partido.

V. Conclusão

Neste trabalho foi reproduzido o trabalho de [5] Zijun Yao, et al. Ficou demonstrada a versatilidade do uso de embeddings

para o estudo da variação de sentido semântico das palavras em um corpus ao longo do tempo, bem como os cuidados que devem ser tomados na comparação de tais embeddings. A avaliação da qualidade foi feita por meio de una exploração visual de trajetórias dos embeddings em um espaço projetado e das palavras na vizinhança de tais trajetórias. Diversos fatores e suas influências sobre o resultado final não puderam ser explorados por limitações de tempo e escopo, mas listamos algumas que acreditamos poderia ser alvo de estudos futuros: a influência dos hiper-parâmetros sobre o resultado final, a influência do tamanho da vizinhança do cálculo do PMI e ainda a possibilidade de usar modelos de machine learning/deep learning para obter análises semelhantes. A técninca de Embeddings dinâmicos aqui explorada teve, além das trajetórias geradas, o mérito de ser um modelo simples e de facil treinamento que pode ser utilizado em trabalhos futuros.

REFERÊNCIAS

[1] L. Bloomfield and C. Hockett, *Language*. University of Chicago Press, 1984. [Online]. Available: https://books.google.com.br/books?id=87BCDVsmFE4C

[2] A. Kutuzov, L. Øvrelid, T. Szymanski, and E. Velldal, "Diachronic word embeddings and semantic shifts: a survey," *arXiv preprint arXiv:1806.03537*, 2018.

[3] Z. Yao, Y. Sun, W. Ding, N. Rao, and H. Xiong, "Dynamic word embeddings for evolving semantic discovery," *WSDM 2018 - Proceedings of the 11th ACM International Conference on Web Search and Data Mining*, vol. 2018-Febua, pp. 673–681, 2018.

[4] D. Jurafsky, *Speech & language processing*. Pearson Education India, 2000.

[5] R. Bamler and S. Mandt, "Dynamic word embeddings," *arXiv preprint arXiv:1702.08359*, 2017.

[6] Y. Zhang, A. Jatowt, S. S. Bhowmick, and K. Tanaka, "The past is not a foreign country: Detecting semantically similar terms across time," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 10, pp. 2793–2807, 2016.

[7] "The New York Times," https://www.nytimes.com/, note = Accessed: 2020-10-01.