1. É correto afirmar que uma das principais formas de se fazer com que um modelo computacional aprenda linguagem é a partir de uma forma eficiente de se representar as palavras de forma numérica? Explique.

2. É possível fazer operações de similaridade entre palavras, usando como representação o *One-Hot Encoding*? Por quê?

3. Explique de que forma as representações de palavras foram evoluindo entre *One-Hot Encoding* → TF-IDF → LSA.

4. O uso de engenharia de *features* em problemas de NLP tem uma principal dificuldade. Qual é? Qual a vantagem da engenharia de features em relação ao *Bag of Words*?

5. Qual foi a grande vantagem obtida a partir da representação por *word embeddings* estáticos, com o Word2Vec? Como essa vantagem se compara em relação ao uso de engenharia de *features*?

6. É correto afirmar que as dimensões de *word embeddings* representam características concretas de palavras? Explique.

7. Por que se diz que os vetores de *word embeddings* são representações **densas** de palavras? Como elas se opõem a representações esparsas por *One-Hot Encoding*?

8. Discorra sobre a diferença entre os algoritmos de treino *Continuous Bag of Words* e *Skip-Gram*.

9. Quais as diferenças entre os algoritmos dos modelos Word2Vec, FastText, Wang2Vec e GloVe?

10. Ao se usar um determinado modelo de *word embeddings*, explique porque é necessário fazer o mesmo pré-processamento que foi feito para este modelo no *corpus* de treino do seu modelo de NLP.

11. Qual o principal ganho que se tem com o *Transfer Learning* promovido pelo uso de *word embeddings*? Explique.

12. Qual a principal limitação dos *word embeddings* **estáticos**? Explique.

13. Faça o download do dump mais recente do Wikipedia da língua portuguesa.
    Dicas:
    Nome do arquivo: ptwiki-latest-pages-articles.xml.bz2
    Local: https://dumps.wikimedia.org/ptwiki/latest/

a. Qual o formato do arquivo? Que informações contém nele?
b. Execute o pré-processamento do *WikiExtractor* para extrair o conteúdo textual dos artigos do *dump* baixado. Explique o que é feito por este script.
c. Descreva o resultado da execução do script. Quais arquivos foram gerados?

14. O 1 Billion Word Language Model Benchmark é um benchmark utilizado para treino e avaliações de modelos de linguagem. Em seu repositório existe um script para pré-processamento do corpus que executa várias etapas necessárias para o treino dos modelos. Adapte o script para fazer o mesmo pré-processamento no dump do Wikipedia ***resultante do pré-processamento realizado no exercício anterior***.
    Dica: Simplifique o processo fazendo com que o WikiExtractor produza somente um arquivo de saída.
    a. Descreva o que é feito pelo script.
    b. Descreva o resultado da execução do script. Quais arquivos foram gerados em cada etapa?

15. Utilize o repositório de pré-processamento do NILC no arquivo de texto resultante do exercício anterior. Faça o pré-processamento do Wikipedia de acordo com o script de pré-processamento deste repositório.

Para os 4 próximos exercícios a seguir, utilize o corpus resultante deste pré-processamento do exercício 16.

16. Faça o treino de um modelo Word2Vec utilizando o gensim. Consulte a documentação do mesmo para saber como é.

17. Faça o treino de um modelo FastText utilizando o gensim. Consulte a documentação do mesmo para saber como é.

18. Faça o treino de um modelo Wang2Vec utilizando o repositório do modelo.

19. Faça o treino de um modelo GloVe utilizando o repositório do modelo.

20. Faça as avaliações contidas no repositório do NILC, para cada um dos modelos que você treinou:
    a. https://github.com/nathanshartmann/portuguese_word_embeddings#semantic-similarity-evaluation
    b. https://github.com/nathanshartmann/portuguese_word_embeddings#syntactic-and-semantic-analogies-evaluation

Como os seus modelos se comparam aos resultados reportados pelo NILC em https://arxiv.org/abs/1708.06025? Analise as diferenças.