NOVA SCHOOL OF
SCIENCE & TECHNOLOGY

Relatório do Projeto de RI

Fase 3: Word Embeddings

Autores:

Miguel França Nº55622

João Palmeiro Nº55926

Tiago Morais Nº56810

Nova School of Science & Technology, 10-01-2022

1. Introdução

No âmbito da cadeira de Recuperação de Informação, foi-nos pedida a realização de um trabalho para que pudéssemos complementar os conhecimentos obtidos nas aulas teóricas. Em termos gerais, o trabalho tem como base a análise de ensaios clínicos e casos de pacientes. A ideia principal seria usar diferentes métodos de representação do texto para os analisar e encontrar uma forma de classificar a relevância entre os ensaios clínicos e os casos de pacientes.

O projeto foi divido em três fases, algo que permitiu observar a evolução da tecnologia relacionada com esta área. Assim, começámos por ver modelos de representação de texto que permitem a atribuição de uma pontuação com base em métodos e algoritmos de análise específicos. Posteriormente, vimos formas de combinar esses modelos para gerar um outro mais robusto, baseado em aprendizagem automática, capaz de melhor classificar as relevâncias. Finalmente, o estudo teórico terminou em técnicas e algoritmos de análise de texto mais recentes. Os "Word Embeddings" trouxeram uma forma melhorada de representar palavras, baseando-se em técnicas como o "Word2Vec", capaz de estabelecer associações entre tokens com base na semântica. Já o "BERT" permitiu estabelecer relações entre tokens com base no contexto em que estes se encontram num determinado documento.

Desta feita, o foque da terceira fase do projeto foi o uso destes algoritmos melhorados para representar e visualizar relações entre tokens de um determinado documento, bem como programar um modelo capaz de aprender a classificar documentos relevantes e não relevantes para uma certa "query".

Seguidamente, apresentamos os seguintes tópicos por pontos: 2. Ambiente Experimental; 3. Métodos Implementados; 4. discussão dos resultados obtidos; 5. Conclusão.

2. Ambiente Experimental (Experimental Setup)

Como dito acima, o foque do projeto é a análise e processamento de texto. Neste caso em particular, queremos analisar ensaios clínicos, que correspondem aos documentos, e casos de pacientes, que correspondem às queries. Para esta fase do projeto, subdividimos bastante o corpus, escolhendo apenas uma query e dois documentos, sendo que apenas um deles fosse relevante para a query usada. Para cada um dos documentos, usámos apenas o campo "detailed_description", uma vez que era suficiente para a análise a que nos propúnhamos. Ficámos, portanto, com dois corpus, cada um deles com um token [CLS] no início e dois tokens [SEP], um que separa a query do documento e outro no final do corpus.

Concretamente para a divisão do texto em tokens e a extração dos seus embeddings, foi necessário usar o modelo BERT ("Bidirectional Encoder Representations from Transformers"). Além disso, também foi necessário visualizar os dados obtidos. Por tudo isto, o projeto foi desenvolvido em python, com bibliotecas já desenvolvidas e otimizadas para resolver estes problemas. Algumas das mais importantes são: transformers (modelos do tipo Transformer já treinados); bertviz (visualizar a attention em modelos do tipo Transformer, nomeadamente o BERT); sklearn (nomeadamente os algoritmos PCA e TSNE para reduzir as dimensões dos dados usados); numpy; torch (framework para aprendizagem automática, nomeadamente para trabalhar com tensors (matrizes de múltiplas dimensões)); matplotlib (para apresentar imagens dos dados).

3. Métodos implementados

Para conseguir processar o texto e extrair os embeddings iniciais e contextuais, usámos o modelo de Transformer BioBERT. Este é um modelo de representação de linguagem biomédica pré-treinado, que é mais indicado para os dados que estamos a usar neste projeto, ao invés de um modelo treinado com dados mais diversos e não tão relacionados com ensaios clínicos. Este modelo fornece formas de dividir o texto em tokens e posteriormente treinar e extrair os seus embeddings por camadas. Assim, é possível extrair os embeddings iniciais acedendo à camada zero e os embeddings finais acedendo à última camada (que neste modelo é a 13ª). É importante referir que a camada inicial contém os embeddings pré-treinados e independentes de contexto, o que não acontece com a última camada, onde já são consideradas as relações contextuais entre os tokens no corpus fornecido. Importa lembrar que para este trabalho escolhemos dois conjuntos limitados de tokens (um para cada corpus) de forma a produzir melhores visualizações. Estes

foram escolhidos de forma que tivessem uma relação significativa com o corpus em questão. Desta feita, a extração e treino dos embeddings ocorreu apenas para os tokens incluídos nos conjuntos mencionados.

Para comparar os embeddings, foi necessário recorrer aos mecanismos de attention e self-attention. O primeiro olha para o contexto em que cada token ocorre no texto de forma a modificar o seu embedding, aproximando tokens com um contexto semelhante. O segundo permite a interação entre os tokens de input, calculando as attentions de cada um em relação com os outros.

O foque principal do trabalho é o uso do modelo para fazer vários tipos de visualização, enumerados e explicados de seguida:

- Visualização dos embeddings por camada:
  Este tipo de visualização é conseguido reduzindo a duas o número de dimensões dos embeddings através do algoritmo Principal Component Analisys (PCA). Este consegue diminuir para duas o número de dimensões dos embeddings representativos dos tokens escolhidos. Com este número, pode-se produzir um "scatter-plot" 2D com pontos representativos de cada token, sendo possível analisar a semelhança dos tokens com base nas distâncias entre os pontos que os representam (sendo menor distância equivalente a maior semelhança);
- Visualização da similaridade por camada:
  Esta visualização apresenta uma matriz que reproduz a similaridade de cada token a todos os outros, usando a distância do cosseno entre vetores de embeddings. As cores mais quentes (nomeadamente o amarelo) representam maior similaridade, enquanto cores mais frias (nomeadamente o roxo) apresentam menor similaridade. Assim, é expectável ver uma diagonal com cores próximas do amarelo, uma vez que esta representa a relação de um token a si mesmo (que não varia significativamente). Esta visualização foi usada para verificar como as similaridades eram alteradas entre o input e o output de cada camada;
- Visualização da self-attention por head:
  Esta visualização apresenta um gráfico interativo em que é possível selecionar certas camadas e heads. Em cada uma destas, é possível ver as ligações que são estabelecidas entre tokens num mecanismo de self-attention. O gráfico é especialmente importante pois mostra que ligações são estabelecidas em cada head, permitindo induzir que tipo de relação ou aprendizagem foi feita naquela fase.

4. Discussão de Resultados

O principal foque de análise e discussão nesta fase do projeto reside nas diferentes formas de visualização de dados explicadas no ponto anterior. Assim, apresentamos pontos separados para cada uma delas:

- Visualização dos embeddings por camada:

Para este modo de visualização, apresentamos dois plots, para os tokens do documento relevante e não relevante à query escolhida:

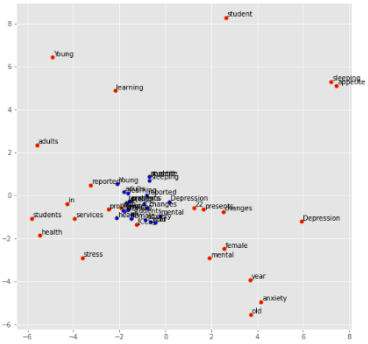

Figura 1 - scatter plot doc. relevante

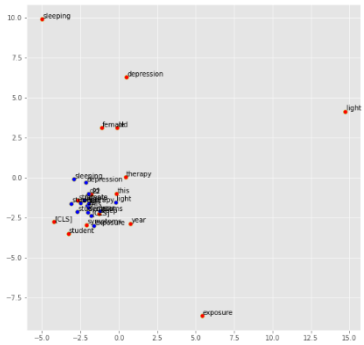

Figura 2 - scatter plot doc. não relevante

Estes plots mostram os pontos representativos dos tokens escolhidos. A vermelho estão os pontos da camada 0 (embeddings iniciais) e a azul estão os pontos da última camada (embeddings finais). Os embeddings iniciais apenas estão relacionados com a semântica pré-treinada da palavra, pelo que pontos com significados parecidos ficam mais próximos entre si.

Podemos verificar que, ao longo das camadas, houve uma adaptação dos embeddings, que resultou na maior proximidade de tokens que aparecem no mesmo contexto do corpus. Isto prova que o modelo se adapta, não só à semântica, como também à proximidade das palavras. Os gráficos não apresentam resultados demasiado diferentes, apesar da relevância para a query ser diferente. Isto acontece porque o modelo se adapta ao contexto independentemente deste fator.

- Visualização da similaridade por camada:

Neste caso, mostramos todas as matrizes de similaridade entre camadas consecutivas. Cada uma delas apresenta a comparação entre os embeddings de input e output de cada camada:

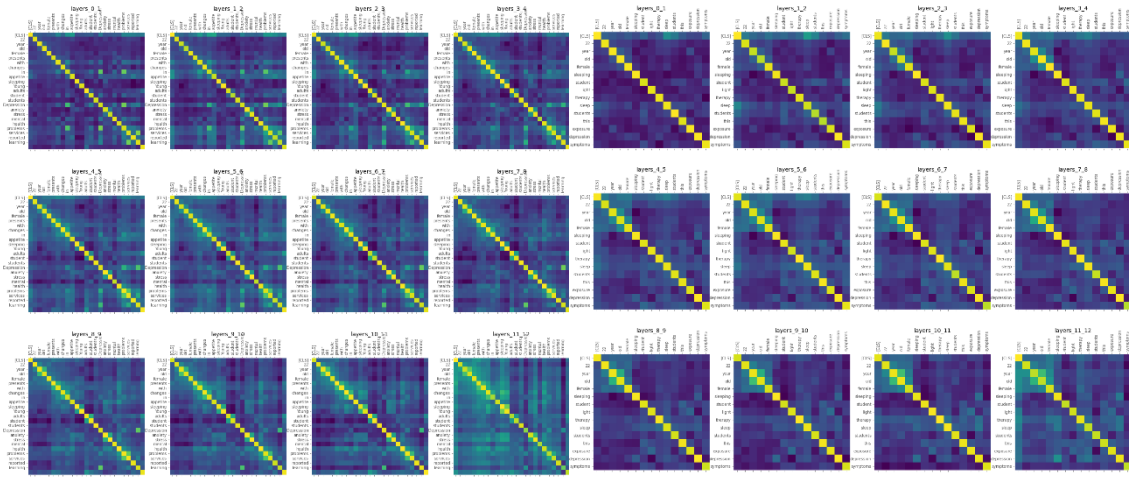

Figura 3 – similarity matrix doc. relevante

Figura 4 – similarity matrix doc. não relevante

Os gráficos mostram que os embeddings vão mudando entre camadas, produzindo cores mais quentes caso haja menos mudanças. Isto permite deduzir que, no caso do documento relevante, há uma maior convergência, visto haver cada vez menos alterações entre camadas consecutivas. Já para o documento não relevante isto verifica-se em menor escala, não havendo tanta convergência.

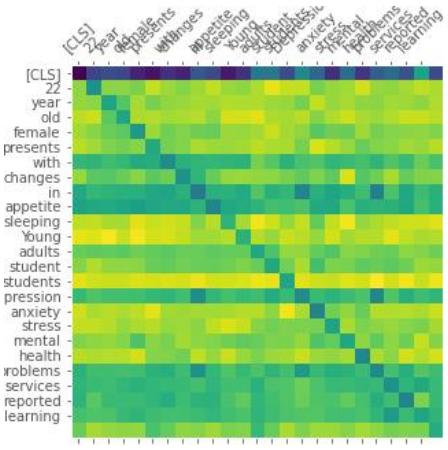

Figura 5 – similarity matrix entre camada 0 e 12 doc. relevante

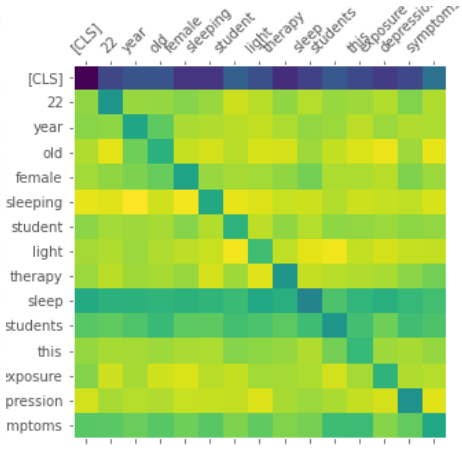

Figura 6 – similarity matrix entre camada 0 e 12 doc. não relevante

Já estes plots apresentam as diferenças entre a primeira e a última camada, mostrando o quanto os embeddings foram alterados entre estas. As cores estão invertidas para melhor visualização, portanto cores mais quentes correspondem a maiores mudanças. Isto permite verificar que houve mudanças apreciáveis entre estas. É ainda possível verificar que, para o documento relevante, os tokens mais correlacionados com o tema do ensaio e da query são aqueles que mais mudam os embeddings, tal como acontece com os termos "students", "young", "health" (o documento e query estão relacionados com stress nos mais jovens). Em relação ao documento não relevante, também há uma adaptação ao corpus escolhido, mas que não está relacionada com a query (o embedding de "sleep", por exemplo, não muda demasiado).

- Visualização da self-attention por head:

Esta visualização mostra relações entre tokens por head e por layer. Em quase todas escolhemos os tokens em que o corpus continha um documento relevante, visto serem os exemplos que apresentavam padrões mais interessantes. Além disto, quando se tratou do caso do documento não relevante, houve menos ligações (e também foram menos acentuadas) entre as palavras, em particular ao token [CLS]. Em baixo mostramos alguns exemplos de aprendizagem:

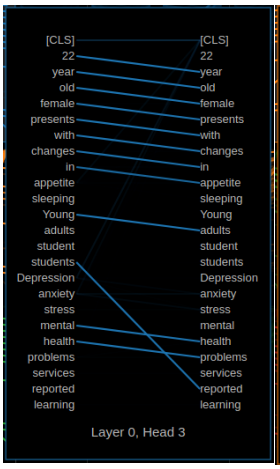

Figura 7 – Padrão 1

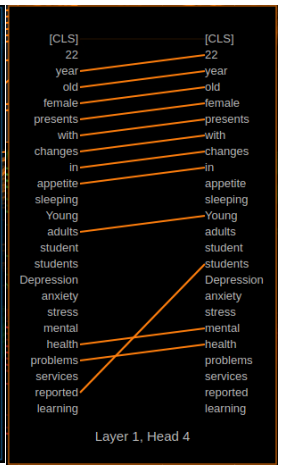

Figura 8 – Padrão 2

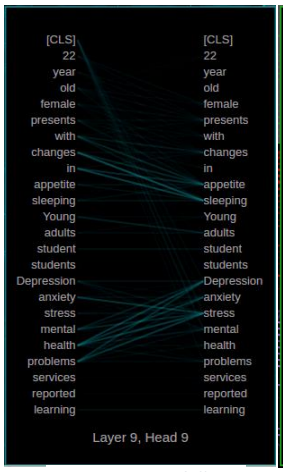

Figura 9 – Padrão 3

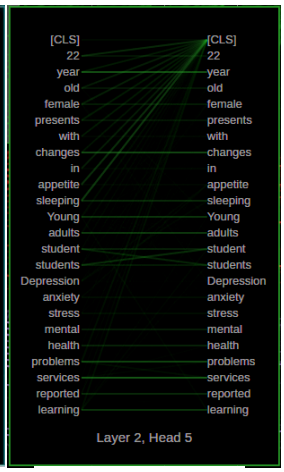

Figura 10 – Padrão 4

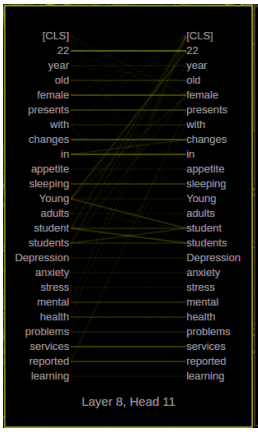

Figura 11 – Padrão 4

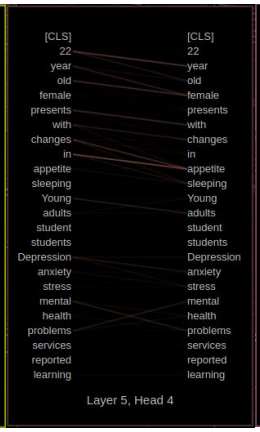

Figura 12 – Padrão 5

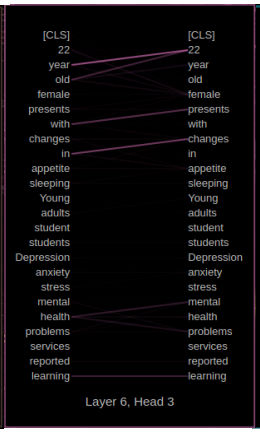

Figura 13 – Padrão 1 e 2

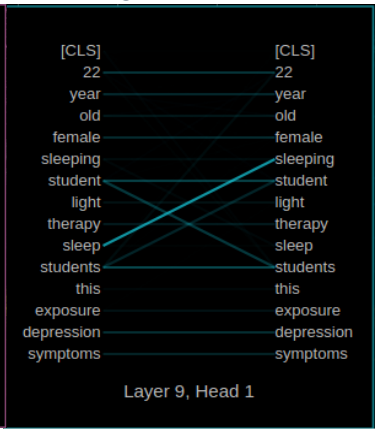

Figura 14 – Padrão 4 não rel.

Figura 7 – Esta head em específico correlaciona as palavras com a palavra imediatamente a seguir (as primeiras palavras escolhidas deste conjunto formam parte de uma frase que também está na query).

Figura 8 – Este exemplo é parecido com o anterior, mas demonstra a atenção para com a palavra anterior, ligando-as.

Figura 9 – Esta head identifica palavras e expressões que estão relacionadas entre si (na mesma frase), ainda que não estejam completamente próximas no texto. A frase é: “Depression, anxiety and stress are among the primary causes of disease rates worldwide and are the most prevalent mental health problems in the U.S Como podemos ver, as palavras “mental”, “health” e “problems” referem “Depression”, “anxiety” e “stress”, sendo esse tipo de relação que a head captura.

Figura 10 – Este exemplo captura relações entre palavras de frases diferentes. Neste caso, é apresentada uma relação morfológica, como ocorre entre as palavras “student” e “students”.

Figura 11 – Este também apresenta o mesmo padrão referido na figura anterior, mas captura também a semântica, ligando palavras como “Young” e “22”, “Young” e “student”.

Figura 12 – Esta head captura a possível previsão de palavras, aqui evidenciado pela previsão da palavra “problems” por parte da palavra “mental”, estando separadas entre si pela palavra “health”.

Figura 13 – Esta imagem apresenta um exemplo em que a head identifica dois padrões distintos (já evidenciados nos padrões das figuras 7 e 8), capaz de associar uma palavra quer com a palavra que a antecede, quer com a que a precede.

Figura 14 – Esta imagem mostra apenas uma relação curiosa entre um verbo no infinitivo e o seu gerúndio, lingando “sleep” a “sleeping”. Este caso é de um documento não relevante, embora isto não tenha influência na forma como a relação é estabelecida entre as palavras do documento.

5. Conclusões:

Este projeto trouxe a possibilidade de aplicar e pôr em prática os conhecimentos teóricos estudados nas aulas. Enfrentar as dificuldades reais de programar, usar ou analisar os resultados de algoritmos de recuperação de informação fez-nos aprender mais sobre os mesmos e perceber a sua verdadeira utilidade em questões reais, como é o caso da atribuição de casos de pacientes a ensaios clínicos. Além disso, ver a evolução da tecnologia foi muito interessante, atravessando modelos simples baseados na frequência das palavras/tokens no corpus e outros capazes de melhor perceber as semelhanças semânticas e a correlação entre tokens olhando para o contexto em que estão inseridos.

Esta fase diz respeito aos últimos modelos referidos, que acabam por ser mais robustos e fazer uma melhor análise do texto. Os modelos word2vec e BERT são pré-treinados com um grande conjunto de documentos, o que os torna muito bons a perceber a semelhança semântica entre palavras. Além disso, o BERT permite um treino mais personalizado, usando várias layers e heads para aprender várias correlações entre palavras, baseando-se no contexto em que estas ocorrem no corpus dos dados fornecidos.

Desta feita, e olhando para a evolução das técnicas usadas ao longo do projeto, seria de esperar que nesta fase conseguíssemos classificar ainda melhor os ensaios clínicos face aos casos de pacientes. Infelizmente, não conseguimos pôr em prática a fase de treino da regressão logística, mas podemos estimar que os resultados da classificação seriam ainda melhores que nas fases anteriores, visto estarmos a considerar um modelo de word embeddings capaz de melhor captar as correlações entre tokens.

Dito isto, acreditamos que analisar os dados nos ajudou a perceber como os algoritmos funcionam, o que eles são capazes de fazer e a forma como os poderíamos usar a nível pessoal ou profissional para melhorar uma atividade de recuperação de informação.