Avaliação de Modelos de Recuperação de Informação Utilizando Biblioteca de Busca por Similaridade FAISS

José Carlos Ferreira Neto[1]

[1]Instituto de Informática – Universidade Federal de Lavras (UFLA)

Abstract. *This meta-paper describes the style to be used in articles and short papers for SBC conferences. For papers in English, you should add just an abstract while for the papers in Portuguese, we also ask for an abstract in Portuguese ("resumo"). In both cases, abstracts should not have more than 10 lines and must be in the first page of the paper.*

Resumo. *Este meta-artigo descreve o estilo a ser usado na confecção de artigos e resumos de artigos para publicação nos anais das conferências organizadas pela SBC. É solicitada a escrita de resumo e abstract apenas para os artigos escritos em português. Artigos em inglês deverão apresentar apenas abstract. Nos dois casos, o autor deve tomar cuidado para que o resumo (e o abstract) não ultrapassem 10 linhas cada, sendo que ambos devem estar na primeira página do artigo.*

1. Introdução

[Knuth 1984]...

2. Referencial Teórico

...

3. Estratégias Adotadas

Para proceder com a comparação dos modelos de busca, oito combinação de abordagens foram utilizadas, isto é, oito estratégias de modelos de busca foram implementados. A Figura 1 apresenta um fluxograma completo com todas as abordagens que foram utilizadas para compor as estratégias adotas.

Todas as estratégias passam pelas etapas elencadas na Figura 1 com diferença nas abordagens utilizadas. As estratégias são:

1. Não há pré-processamento, vetorização via BERT e indexação *Flat*;
2. Não há pré-processamento, vetorização via BERT e indexação *IVFFlat*;
3. Normalização das palavras, vetorização via TF-IDF e indexação *Flat*;
4. Normalização das palavras, vetorização via TF-IDF e indexação *IVFFlat*;
5. Remoção de *stopwords*, vetorização via BERT e indexação *Flat*;
6. Remoção de *stopwords*, vetorização via BERT e indexação *IVFFlat*;
7. Normalização das palavras, remoção de *stopwords*, vetorização via TF-IDF e indexação *Flat*;
8. Normalização das palavras, remoção de *stopwords*, vetorização via TF-IDF e indexação *IVFFlat*;

Na etapa de pré-processamento duas abordagens foram utilizadas, a normalização e a remoção das *stopwords*. A aplicação da primeira se dá por converter todos os caracteres em letras minúsculas. Já para a segunda, toda palavra presente no texto e no conjunto de *stopwords* foram removidas dos documentos. O conjunto de *stopwords* empregado está disponível na biblioteca NLTK.

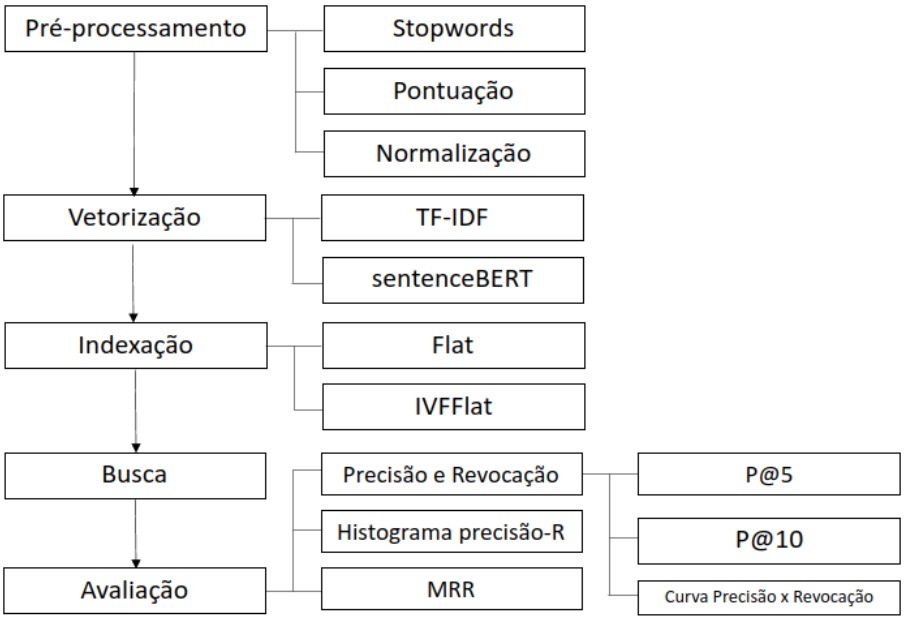

Figura 1. Abordagens Adotadas nas Estratégias Aplicadas

Duas abordagens foram adotadas para a vetorização dos documentos: TF-IDF e *sentence*-BERT. O TF-IDF foi aplicado utilizando a biblioteca *Scikit-Learn*. Para a geração dos *embeddings* via BERT, utilizou-se o modelo pré-treinado *multi-qa-mpnet-base-dot-v1*[1]. Este modelo foi ajustado para busca semântica utilizando 215 milhões de pares de pergunta e resposta de diversas fontes, este suporta sentenças de até 512 palavras, gerando um *embedding* de 768 dimensões.

Dois tipos de indexação foi aplicados, o *Flat* e o *IVFFlat*, ambas disponíveis na biblioteca FAISS.

A busca utilizando indexação *Flat* é realizada calculando a similaridade entre todos os documentos e a consulta. Já a indexação do tipo *IVFFlat* agrupa os documentos com base na distância euclidiana, 10 agrupamentos foram gerados neste trabalho. Na busca utilizando *IVFFlat* a consulta é comparada aos centroides de cada um destes agrupamentos, e o cálculo de similaridade é realizado apenas com os documentos presentes no agrupamento em que a consulta foi atribuído, reduzindo assim o espaço de busca. O resultado das buscas utilizando ambas abordagens ordena os documentos relevantes de acordo com a similaridade encontrada.

As estratégias adotadas são avaliadas através da precisão, revocação, precisão em 5 e 10 documentos recuperados (P@5 e P@10), curva de precisão vs. revocação, histograma da precisão-R e *Mean Reciprocal Rank*.

[1]Disponível em: https://huggingface.co/sentence-transformers/multi-qa-mpnet-base-dot-v1

4. Experimentos

Todos as bibliotecas mencionadas na seção 3 estão disponíveis para linguagem Python, e portanto, todos os experimentos foram executados utilizando esta linguagem. A biblioteca FAISS, que disponibiliza as opções de indexação e busca está disponível para ser executada via processamento de CPU ou GPU, neste trabalho utilizou-se o processamento via CPU. Assim sendo, a configuração da máquina utilizada é: processador (CPU) Ryzen 5 3600, 6-CORE, 12-THREADS, velocidade de 3.6GHz e duas memórias (RAM) DDR4 de 16GB e velocidade de 2666MHz.

Todos os códigos utilizados para executar os experimentos mencionados neste trabalho podem ser acessados no GitHub[2].

Os vetores produzidos pelo *sentence*BERT possuem 768 dimensões, enquanto que os vetores produzidos pelo TF-IDF possuem 13.171 dimensões para os documentos sem pré-processamento e 13.141 dimensões para os documentos com remoção das *stopwords*.

5. Resultados

A Figura 2 apresenta o *boxplot* dos resultados de precisão das estratégias adotadas. Com esse gráfico é possível visualizar o resumo estatístico da precisão para todas as consultas. A precisão média variou entre 20.1% para indexação do tipo *IVFFlat*, vetorização com TF-IDF e remoção de *stopwords* para 27.1% com indexação *Flat*, vetorização com BERT sem pré-processamento.

Comparativamente, utilizou-se teste-T bilateral para as médias de duas amostras independentes para identificar se há diferença estatística para os experimentos executados com e sem remoção de *stopwords*. A hipótese nula para este teste é de que as duas amostras independentes possuem valores médios idênticos. Para todos os casos, o valor-P observado foi maior do que 5% (intervalo de confiança de 95%), isto significa que, não podemos rejeitar nossa hipótese nula.

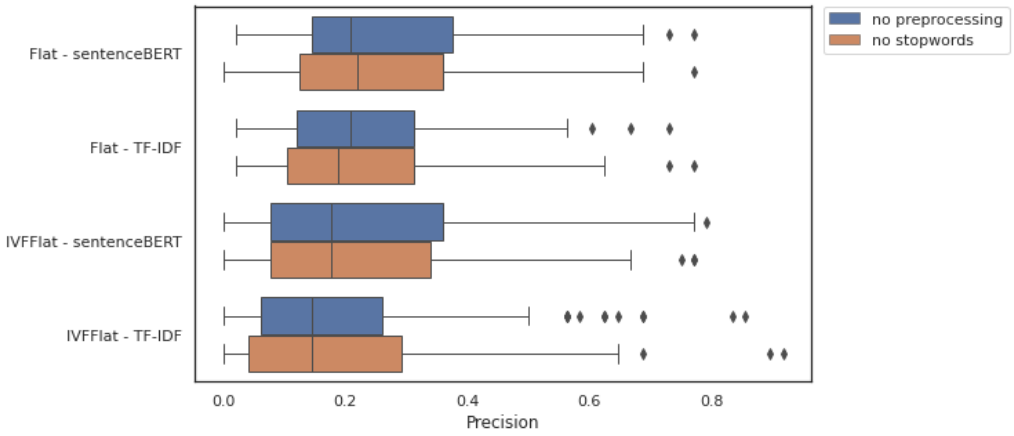

Figura 2. Precisão

A Figura 3 apresenta os resultados de revocação. O teste-T foi implementado com os resultados obtidos, e semelhante ao observado com a precisão, não foi possível

[2]Códigos disponíveis em: https://github.com/jcfneto/ir

perceber diferença estatística (p-Valor $>$ 5%) entre as estratégias utilizando ou não a remoção de *stopwords*.

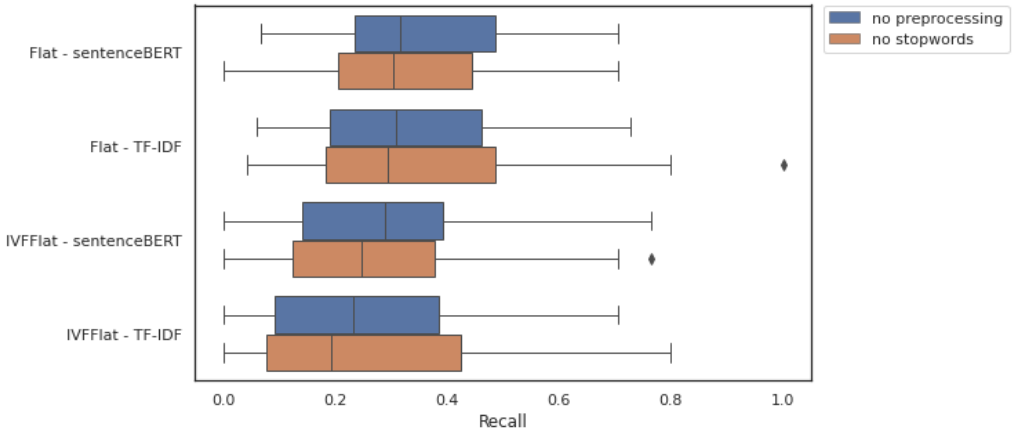

Figura 3. Revocação

As curvas precisão vs revocação são apresentada na Figura 4. Este gráfico é gerado a partir da precisão média obtida para os 11 níveis de revocação (de 0% a 100%). O gráfico da esquerda apresenta os resultados para as estratégias com a geração dos vetores com os documentos sem pré-processamento, enquanto o gráfico da direita apresenta os resultados para as estratégias que utilizaram os textos com remoção de *stopwords*. É possível observar que, os valores de precisão das estratégias nos níveis finais de revocação ($>$ 60%) invertem nos dois gráficos. Enquanto a estratégia representada pela linha preta finaliza com 100% de revocação e precisão maior que 10% no gráfico a esquerda, no gráfico a direita a mesma estratégia finaliza com precisão inferior a 10%. Esse mesmo padrão pode ser observado nas outras estratégias.

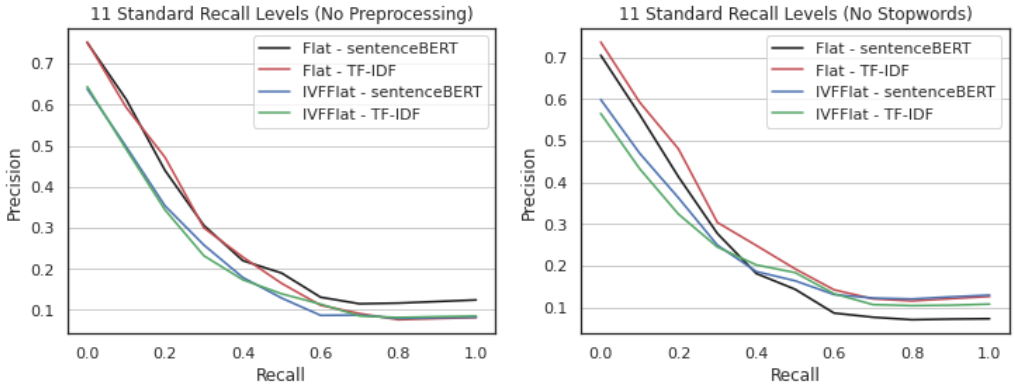

Figura 4. Precisão vs Revocação

As Figuras 5 e 6 apresentam o gráfico de violino para os resultados da precisão em 5 e 10 documentos, respectivamente. Este gráfico exibe um resumo estatístico e a distribuição dos resultados. Na maioria dos casos é possível ver que, não há diferença dos quartis para as distribuições em azul (sem processamento) e cinza (sem *stopwords*), o que é comprovado pelos resultados do teste-T, onde em todos os casos p-Valor $>$ 5%.

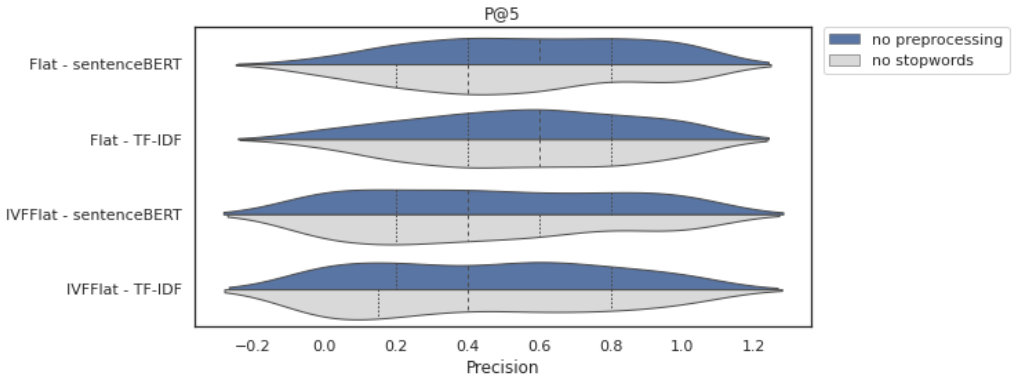

Figura 5. Precisão em 5 Documentos

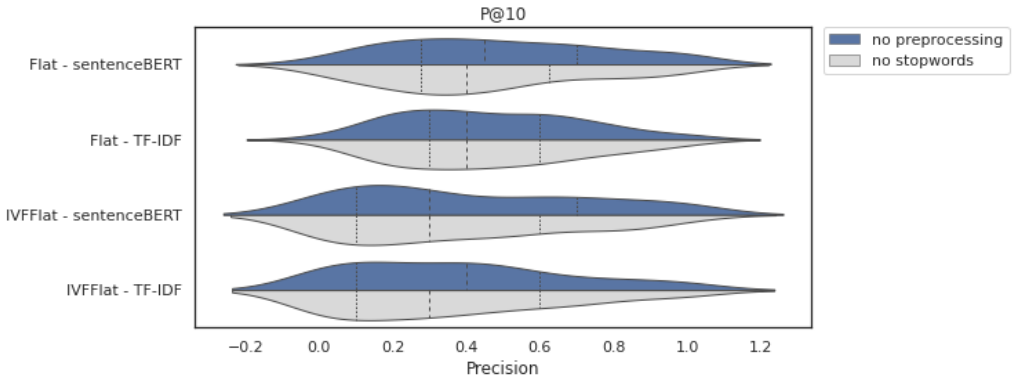

Figura 6. Precisão em 10 Documentos

O MRR é apresentado na Figura 7. É possível notar que, com a utilização dos documentos com remoção de *stopwords* menos documentos relevantes foram recuperados nas primeiras posições e mais consultas ficaram sem documentos relevantes dentro das cinco primeiras posição. Isto foi observado em todas as estratégias, exceto para a estratégia que utiliza indexação *Flat* e vetorização TF-IDF.

Levando em consideração os resultados médios apresentados até o momento, duas estratégias se mostram mais robusta dado as condições do experimento. A primeira é a indexação *Flat* com vetorização BERT sem pré-processamento e a segunda é o conjunto de indexação *Flat* com vetorização TF-IDF com remoção de stopwords. A Figura 8 apresenta o histograma da precisão-R para as 20 primeiras consultas. Observa-se que, a primeira estratégia possui precisão superior a segunda em 50% das consultas, a segunda em 30% das consultas e em 20% existe um empate.

O último fator avaliado foi o tempo de execução das 1.239 consultas para cada uma das estratégias. A Tabela 1 apresenta os tempos em segundos. Nota-se que, as estratégias utilizando indexação *IVFFlat* foram em média 6.5 vezes mais rápidas em relação a indexação *Flat*. Em relação ao tipo de vetorização, o ganho em velocidade produzido pela indexação *IVFFlat* em relação a *Flat* foi maior na vetorização TF-IDF, com um ganho de 7.1x, enquanto que na vetorização via BERT o ganho foi de 5.8x.

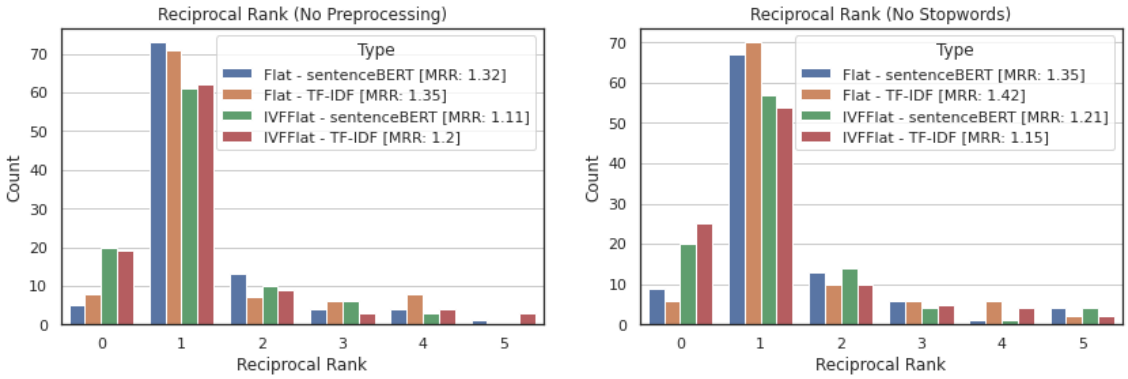

Figura 7. *Mean Reciprocal Rank* (MRR)

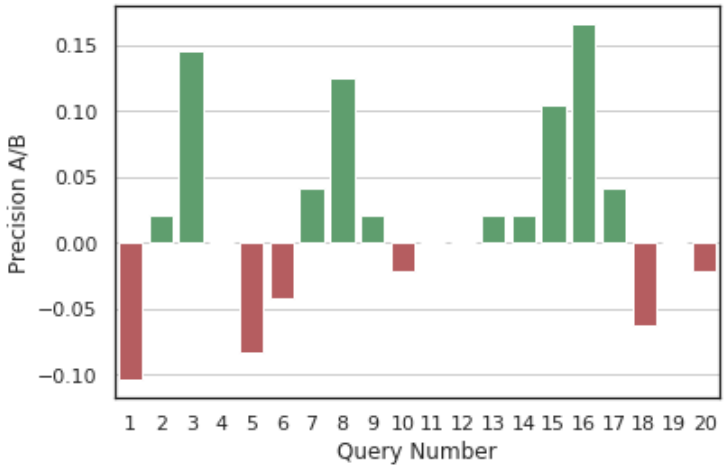

Figura 8. Tempo Total de Execução das Consultas

Referências

Knuth, D. E. (1984). *The TEX Book*. Addison-Wesley, 15th edition.

Tabela 1. Tempo Total de Execução das Consultas

Preprocessing	Flat - BERT	Flat - TF-IDF	IVFFlat - BERT	IVFFlat - TF-IDF
no preprocessing	0.072673	1.272436	0.013532	0.198257
no stopwords	0.073614	1.295599	0.012794	0.166142