UNICAMP

Article Presentation

IA368DD_2023S1: Deep Learning aplicado a Sistemas de Buscas

Student: Marcus Vinícius Borela de Castro

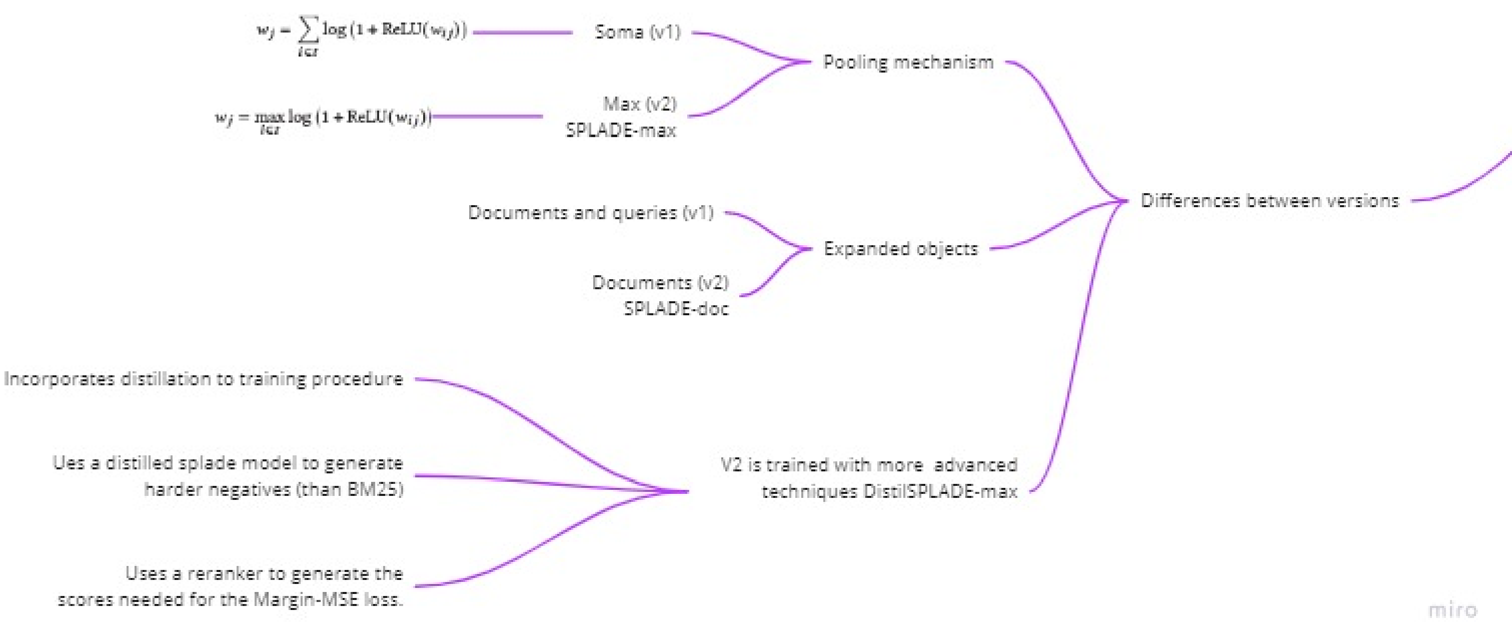
Splade

A model that both learns expansion and compression in an end-to-end manner.

The model generates a sparse BOW (the size of the vocabulary) for a text with expansion of important terms (synonyms and related, for example) and excludes unimportant terms (stop words, for example)

SPLADE
SParse **L**exical **A**n**D** **E**xpansion Model for Information Retrieval

Main concepts (history)

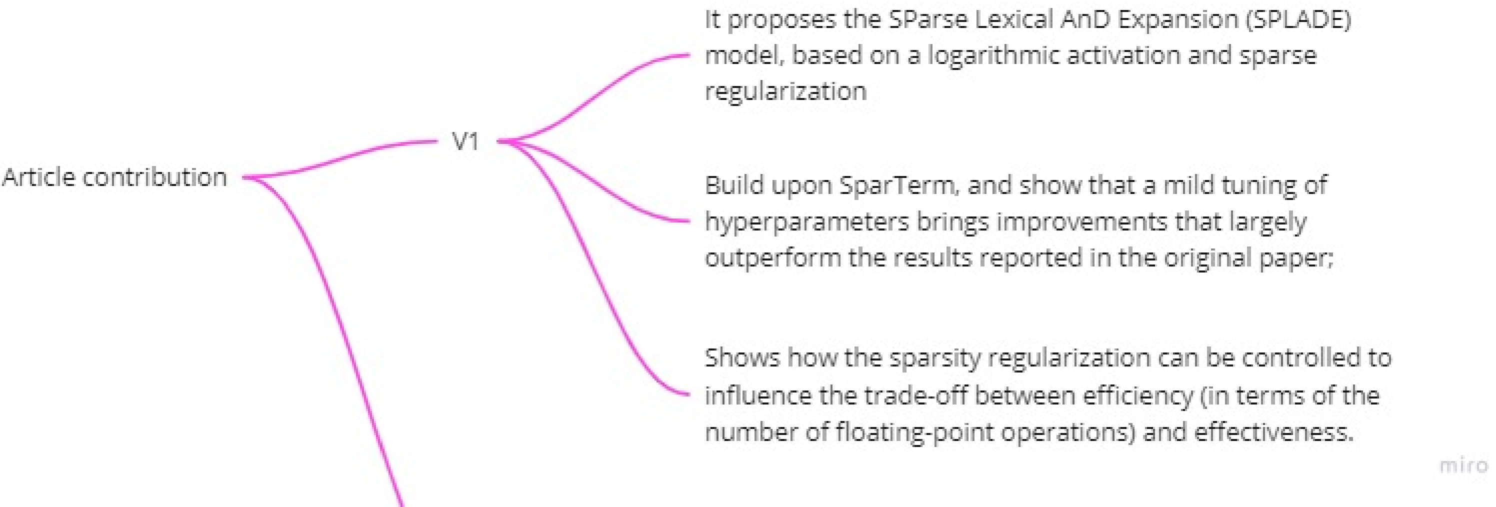


$w_j = \sum_{i \in t} \log(1 + \text{ReLU}(w_{ij}))$
Soma (v1)
$w_j = \max_{i \in t} \log(1 + \text{ReLU}(w_{ij}))$
Max (v2)
SPLADE-max
Pooling mechanism
Documents and queries (v1)
Documents (v2)
SPLADE-doc
Expanded objects
Differences between versions
Incorporates distillation to training procedure
Ues a distilled splade model to generate
harder negatives (than BM25)
Uses a reranker to generate the
scores needed for the Margin-MSE loss.
V2 is trained with more advanced
techniques DistilSPLADE-max
miro

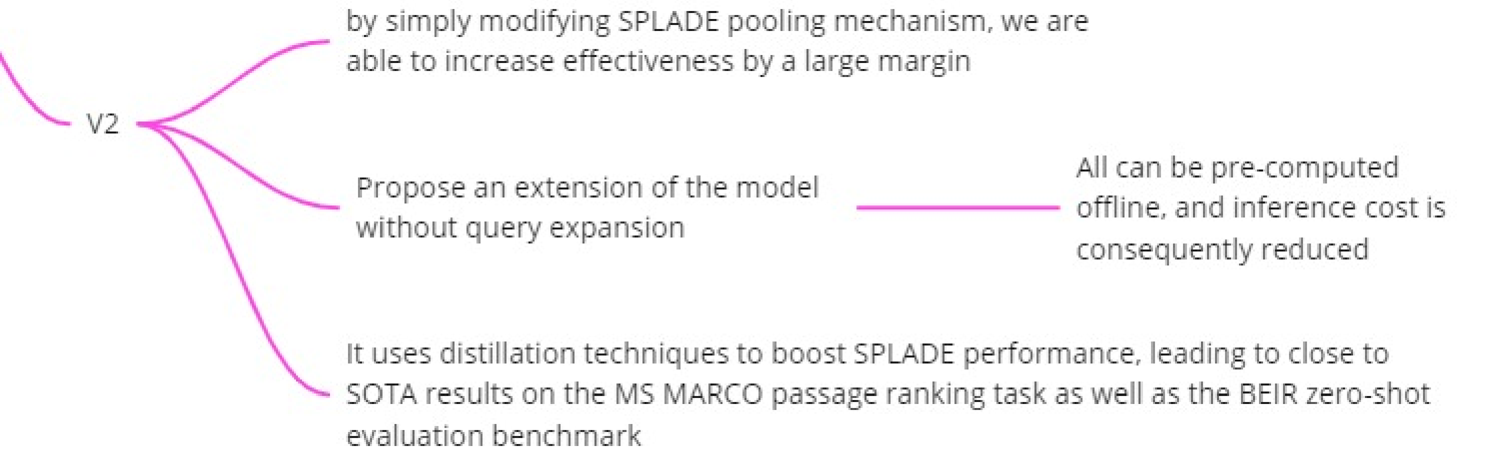
While BOW models remain strong baselines, they suffer from the long standing vocabulary mismatch problem, where relevant documents might not contain terms that appear in the query.

Thus, there have been approaches by learned (neural) rankers, with challenges regarding efficiency and scalability: therefore there is a need for methods where most of the computation can be done offline and online inference is fast.

So, there has been a growing interest in learning sparse representations for documents and queries.
By doing so, models can inherit from the desirable properties of BOW models like exact match of terms, efficiency of inverted indexes and **interpretability**.



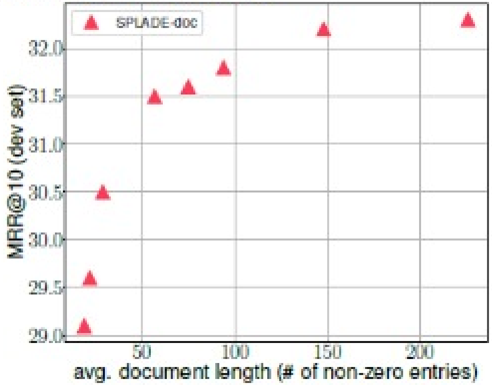
Article contribution
V1
It proposes the SParse Lexical AnD Expansion (SPLADE) model, based on a logarithmic activation and sparse regularization
Build upon SparTerm, and show that a mild tuning of hyperparameters brings improvements that largely outperform the results reported in the original paper;
Shows how the sparsity regularization can be controlled to influence the trade-off between efficiency (in terms of the number of floating-point operations) and effectiveness.
miro


V2
by simply modifying SPLADE pooling mechanism, we are able to increase effectiveness by a large margin
Propose an extension of the model without query expansion
All can be pre-computed offline, and inference cost is consequently reduced
It uses distillation techniques to boost SPLADE performance, leading to close to SOTA results on the MS MARCO passage ranking task as well as the BEIR zero-shot evaluation benchmark
miro

- Interesting/unexpected results

Different regularization strength λd
"little is a lot"

Figure 2: Performance vs average document length (number of non-zero dimensions in document representations) for SPLADE-doc models trained with different regularization strength λ_d on MS MARCO.



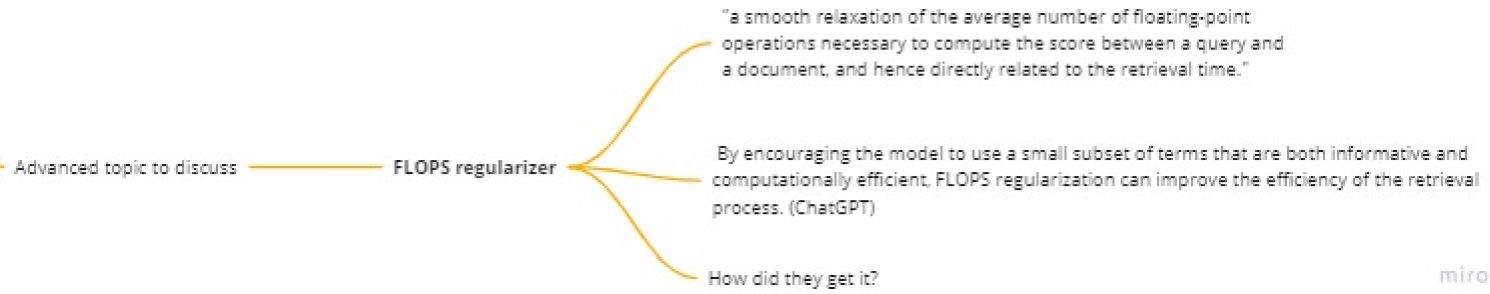
The results are competitive with SOTA dense retrieval methods (MS-Marco and TREC DL 2019)

Furthermore, DistilSPLADE-max is able to outperform all other methods in most datasets of the BEIR benchmark

Table 1: Evaluation on MS MARCO passage retrieval (dev set) and TREC DL 2019.

model	MS MARCO dev		TREC DL 2019	
	MRR@10	R@1000	NDCG@10	R@1000
Dense retrieval				
Siamese (ours)	0.312	0.941	0.637	0.711
ANCE [29]	0.330	0.959	0.648	-
TCT-ColBERT [16]	0.359	0.970	0.719	0.760
TAS-B [11]	0.347	0.978	0.717	0.843
RocketQA [24]	0.370	0.979	-	-
Sparse retrieval				
BM25	0.184	0.853	0.506	0.745
DeepCT [4]	0.243	0.913	0.551	0.756
doc2query-T5 [20]	0.277	0.947	0.642	0.827
SparTerm [1]	0.279	0.925	-	-
COIL-tok [9]	0.341	0.949	0.660	-
DeepImpact [18]	0.326	0.948	0.695	-
SPLADE [8]	0.322	0.955	0.665	0.813
Our methods				
SPLADE-max	0.340	0.965	0.684	0.851
SPLADE-doc	0.322	0.946	0.667	0.747
DistilSPLADE-max	0.368	0.979	0.729	0.865

- Basic doubts that may arise
  - DistilSPLADE-max only expands documents? (like SPLADE-doc)
  - **Sparsity regularization**
    - In SPLADE, the sparsity regularization term is based on the L1-norm of the weight vector, which encourages the model to set many of the weights to zero. This helps to reduce the impact of noisy or irrelevant terms in the retrieval process, and can improve the efficiency and effectiveness of the model. (ChatGPT)
    - Is it correct?


Advanced topic to discuss
FLOPS regularizer
"a smooth relaxation of the average number of floating-point operations necessary to compute the score between a query and a document, and hence directly related to the retrieval time."
By encouraging the model to use a small subset of terms that are both informative and computationally efficient, FLOPS regularization can improve the efficiency of the retrieval process. (ChatGPT)
How did they get it?
miro