A Pruning-Based Deep Learning Approach For Information Retrieval

Master of Science in Engineering in Computer Science

Deep Learning

Sapienza Università di Roma

Speakers

Luca Zanchetta	1848878
Pasquale Mocerino	1919964
Simone Scaccia	2045976

Task Description

Neural Inverted Index: a unified model replicating the behavior of a conventional index and performing enhanced retrieval by leveraging the power of neural networks.

→ Our **focus** is on **optimizing a DSI model**, denoted as f, which takes a query q as input and produces a ranked list of document IDs.

Dataset

- We used the **MS MARCO** dataset through the Pyserini toolkit:
  - Select the **most relevant K document IDs** for each query of the dataset, and use this data for training the model.
  - Select the **most relevant 1000 document IDs** only for the **Recall@1000** metric computation (see later).
- We used the **T5 tokenizer** for tokenizing both the queries and the document IDs.

Evaluation Metrics

We assessed our model using the following **metrics**:

- **MAP (Mean Average Precision):** the mean of the average precision scores from a set of queries.

- **Recall@1000:** the proportion of relevant document IDs found in the top-1000 results.

Baseline

- We used a **pre-trained T5** model from Hugging Face, embedded in a Pytorch Lightning module.
- We have **fine-tuned** all the layers of the pre-trained T5 model on our task, so that the model was able to predict a ranked list of document IDs, given an input query.
- We have considered **three versions** of the Hugging Face T5 model, having different sizes:
  - T5-large
  - T5-base
  - T5-small

Innovation

→ **Our approach:** employ the **Train-Prune-Recovery strategy** on the proposed baseline in order to let the model work in a resource-constrained environment.

- In particular, the pruning:
  - One-shot
  - Unstructured
  - Magnitude-based (L1 norm)

- We have conducted several experiments on the **pruning rate**, in order to find the model with the best metrics performance (see later).

Workflow (1/6)

Goal: comparing different versions of T5 model with their pruned counterparts.

- T5-large: Google Colab limitations (GPU RAM)

SAPIENZA
UNIVERSITÀ DI ROMA

Workflow (2/6)

➢ T5-base:

Table 1: Baseline *t5-base*

K	Batch size	Epochs	Learning rate	Patience	Test loss	MAP	Recall@1000
25	16	30	0.001	5	3.514	0.00100	0.0
20	16	30	0.001	5	4.628	0.00135	4.00E-05
5	8	20	0.001	∞	8.323	0.00352	5.00E-05
5	**8**	**23**	**0.001**	∞	**8.604**	**0.00563**	**0.00011**

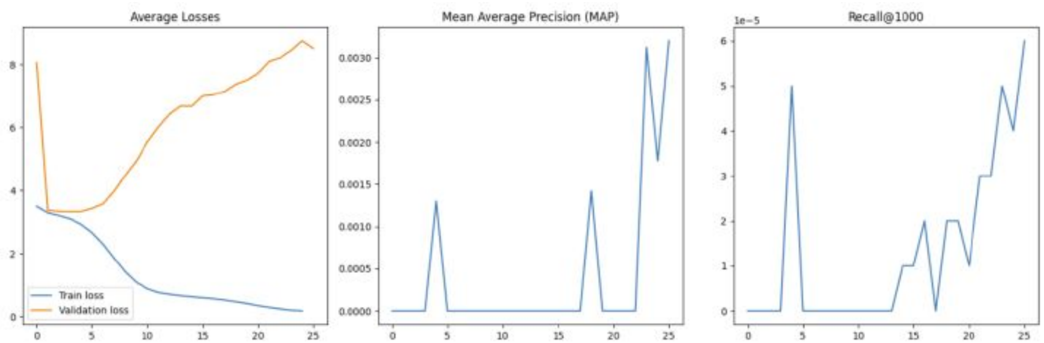


Figure 1: Plots of the best baseline *t5-base* result.

Workflow (3/6)

- T5-base observations:
  - We could train for **fewer epochs** with a **large K.**
  - **Performance** on metrics **increases** as the number of **training epochs increases**: **epoch-wise double descent?**
  - Due to Colab limitations, we **couldn't apply** the Train-Prune-Recovery strategy.

Workflow (4/6)

- T5-small:

Table 2: Baseline *t5-small*

K	Batch size	Epochs	Learning rate	Patience	Test loss	MAP	Recall@1000
25	8	20	0.001	∞	8.558	0.0	0.0
10	8	50	0.001	∞	11.197	0.00100	0.0
5	**8**	**25**	**0.001**	∞	**9.339**	**0.00119**	**2.00E-05**

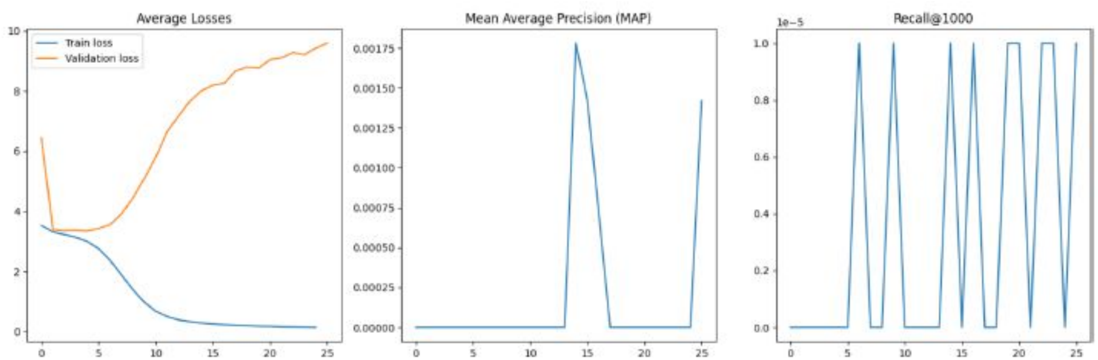


Figure 2: Plots of the best baseline *t5-small* result.

Workflow (5/6)

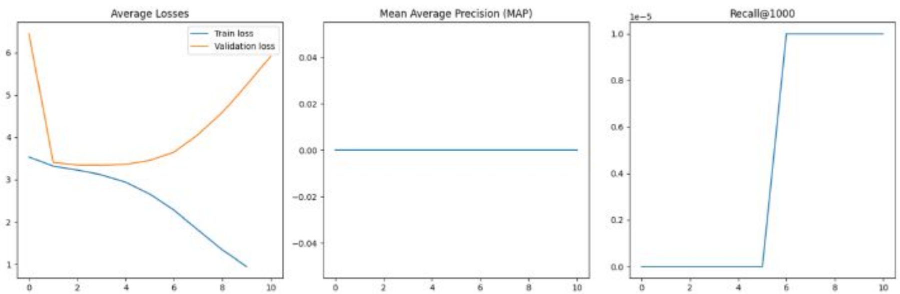
➢ T5-small pruned:

Table 3: Pruned *t5-small*

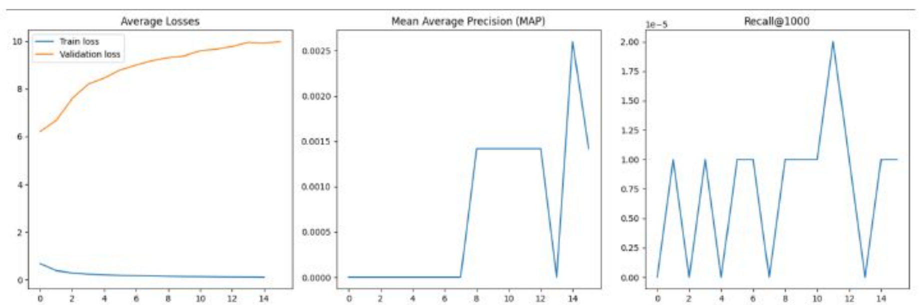
K	Batch size	Epochs (total)	Learning rate	Patience	Pruning rate	Test loss	MAP	Recall@1000
5	8	25	0.001	∞	0.1	9.596	0.00057	2.00E-05
5	8	25	0.001	∞	0.15	9.710	0.00071	1.00E-05
5	**8**	**25**	**0.001**	∞	**0.2**	**9.901**	**0.00214**	**2.00E-05**
5	8	25	0.001	∞	0.25	10.207	0.00062	1.00E-05
5	8	25	0.001	∞	0.3	10.352	0.00211	2.00E-05
5	8	25	0.001	∞	0.4	10.464	0.00071	1.00E-05
5	8	25	0.001	∞	0.5	10.768	0.00167	1.00E-05

Workflow (6/6)

- T5-small pruned:



Non-pruned baseline



Pruned baseline

Conclusions

- The **pruned** T5-small **outperforms on both metrics** the T5-small **baseline** when the pruning rate is 0.2 or 0.3:

Table 2: Baseline *t5-small*

K	Batch size	Epochs	Learning rate	Patience	Test loss	MAP	Recall@1000
25	8	20	0.001	∞	8.558	0.0	0.0
10	8	50	0.001	∞	11.197	0.00100	0.0
5	**8**	**25**	**0.001**	∞	**9.339**	**0.00119**	**2.00E-05**

Table 3: Pruned *t5-small*

K	Batch size	Epochs (total)	Learning rate	Patience	Pruning rate	Test loss	MAP	Recall@1000
5	8	25	0.001	∞	0.1	9.596	0.00057	2.00E-05
5	8	25	0.001	∞	0.15	9.710	0.00071	1.00E-05
5	**8**	**25**	**0.001**	∞	**0.2**	**9.901**	**0.00214**	**2.00E-05**
5	8	25	0.001	∞	0.25	10.207	0.00062	1.00E-05
5	8	25	0.001	∞	0.3	10.352	0.00211	2.00E-05
5	8	25	0.001	∞	0.4	10.464	0.00071	1.00E-05
5	8	25	0.001	∞	0.5	10.768	0.00167	1.00E-05

A Pruning-Based Deep Learning Approach For Information Retrieval

Thanks for your interest!

SAPIENZA
UNIVERSITÀ DI ROMA

Speakers

Luca Zanchetta	1848878
Pasquale Mocerino	1919964
Simone Scaccia	2045976