Final Assignment Information Retreival

SHUBHAM BHATT S3287467

1 INTRODUCTION

Cross-encoder re-rankers employ Machine Learning (ML) models to retrieve and rank documents according to how relevant they are to a given query. This use-case, and re-rankers themselves are thus highly relevant in the domain of Information Retrieval. These methods of re-ranking have demonstrated their ability to increase the accuracy of results at the top quantum of rankings, hence enhancing the functionality of search engines. With regards to task-specific data, these models can be fine-tuned to further improve their performance.

The re-rankers encode the query and the document as a single input and compute the relevance between the two using transformer-based models like BERT, RoBERTa, among others. Their efficiency for relevance rating tasks is increased by this "cross-encoding" procedure. But optimising these cross-encoder re-rankers is still a complex and difficult research challenge.

The significance of comprehending the metrics and benchmarks used in model evaluation, as well as the relevance of data quality and quantity in model performance, will be highlighted through a thorough analysis of these re-rankers.

The Microsoft MARCO (MS MARCO) dataset [2], a sizable, real-world dataset for machine reading comprehension and question-answering, will be extensively used in this work. The MS MARCO dataset comprises millions of actual, anonymized Bing user searches, giving researchers the chance to create and improve algorithms based on real-world user experiences. It is a rich dataset that is useful for training cross-encoder re-rankers and offers a strong and varied set of queries and documents to assess the efficacy of our ensemble methodologies and fine-tuning techniques.

In this experiment, we used three distinct models:

- **cross-encoder/ms-marco-MiniLM-L-2-v2**: This condensed MiniLM model, which offers the benefits of reduced size and faster speed while keeping good language understanding skills, was optimised on the MS MARCO dataset.
- **cross-encoder/ms-marco-TinyBERT-L-2-v2**: This model, a variation of TinyBERT, is likewise adjusted on the MS MARCO dataset and offers a practical and portable substitute while maintaining the robust performance of the original BERT model.
- **distilroberta-base**: It is especially well-suited for deployment in contexts with limited resources since this condensed version of the RoBERTa model uses model compression techniques to give great performance in a more useful package.
- **MiniLM-L12-H384-uncased**: A small transformer-based language model called the "MiniLM-L12-H384-uncased" was created by Microsoft Research. With its case-insensitive training and 12-layer design with 384-dimensional hidden layers, it can process natural language tasks quickly and with little computational overhead.

Author's address: Shubham Bhatt s3287467, s.bhatt@umail.leidenuniv.nl.

2 TASK 1

Models	NDCG@10	Recall@100	**MAP@1000**	**Training Steps**
ms-marco-MiniLM-L-2-v2	67.70	49.55	43.42	44624/156250
ms-marco-TinyBERT-L-2-v2	**69.38**	**50.31**	**45.54**	**61999/156250**
distilroberta-base	60.88	47.99	41.07	8463/156250
MiniLM-L12-H384-uncased	68.06	50.44	45.10	11290/156250

2.1 Discussion

2.1.1 Results and Performance Variation. Variations in model performance can be attributed to their distinct architectures, training methods, and sizes of the models.

For instance, using knowledge distillation, MiniLM and TinyBERT were trained to resemble larger models, while DistilRoBERTa is a scaled-down version of RoBERTa. Performance discrepancies are a result of these various context interpretation mechanisms.

Performance is also impacted by the model's parameters and size. But even though it was smaller, TinyBERT performed the best, demonstrating the success of its distillation method. The total amount of training exercises may also have an effect. Greater performance is frequently the result of additional training steps because the model has more chances to learn from the input. This isn't always the case, though, as excessive training can result in overfitting.

2.1.2 Will TinyBERT perform the best for allnew unseen queries? No. If the distribution of the unseen data differs greatly from the distribution of the training and evaluation data, TinyBERT may not necessarily perform as well on unseen data. Each metric also evaluates a distinct component of the model's performance. Recall@100, for instance, gauges a model's capacity to locate pertinent documents within the top 100 results, while NDCG@10 gauges the calibre of the top 10 results. Depending on the strengths and flaws of a model, it may do well on one metric while failing miserably on another for the unseen data.

2.1.3 Limitations.

- Computational Efficiency: Smaller models like TinyBERT and MiniLM are more computationally efficient than larger models, making them more suitable for deployment in real-world applications. However, they might not capture as much information as larger models.
- Overfitting: If the models are trained for too many steps, they might start to overfit to the training data, reducing their generalization capability on unseen data.
- Interpretability: Like most deep learning models, these models lack interpretability. It's hard to understand why they make certain predictions, which can be a problem in scenarios where interpretability is important.

3 TASK 2

The Ranx library is used for fast-ranking evaluation metrics [1]. To this end we use several fusion algorithms, the results of which are depicted in the table below:

Algorithm	NDCG@10	Recall@100	MAP@1000
CombMIN	0.25	0.67	0.23
CombMAX	0.38	0.83	0.34
CombMED	0.31	0.81	0.29
CombSUM	0.40	0.83	0.35
CombANZ	0.30	0.81	0.29
CombMNZ	0.41	0.83	0.36

3.1 Discussion

3.1.1 Results. The CombMNZ algorithm consistently outperforms the competition on all three criteria (NDCG@10, Recall@100, and MAP@1000). Performance-wise, the CombSUM algorithm comes in close second. Across all criteria, the CombMIN algorithm performs the poorest. The performance of the remaining algorithms (CombMAX, CombMED, and CombANZ) is in the middle.

3.1.2 Reason for CombMNZ's Effectivity. The effectiveness of CombMNZ may be due to its methodology, which takes into account both the scores actually given to a document as well as the number of systems that contribute to it (which can suggest its significance). CombMNZ is able to balance the degree of agreement among systems and the acuity of individual scores because to this.

3.1.3 Usefulness of Ensemble Methods. Information retrieval ensemble methods integrate the findings of various retrieval systems to produce a single ranking that is, ideally, superior to all independent rankings. They mitigate the weaknesses of various systems while utilizing their strengths. The findings emphasise the potential benefits of adopting ensemble approaches by showing that some fusion techniques can perform better than others. However, depending on the precise data, the retrieval systems involved, and the fusion mechanism used, the effectiveness of these strategies may change. As a result, even if ensemble approaches can have a lot of advantages, their selection and use must be adapted to the particulars of the current issue.

4 TASK 3

We present the performance of various combinations of models using the best fusion algorithm obtained from the previous task, i.e CombMNZ.

Combination	NDCG@10	Recall@100	MAP@1000
distilroberta-base + ms-marco-TinyBERT-L-2-v2	0.36	0.77	0.31
distilroberta-base + ms-marco-MiniLM-L-2	0.34	0.84	0.29
ms-marco-TinyBERT-L-2-v2 + ms-marco-MiniLM-L-2	0.32	0.69	0.28

4.1 Discussion

4.1.1 Results. The most effective combination was "distilroberta-base + ms-marco-TinyBERT-L-2-v2," whereas the poorest combination was "ms-marco-TinyBERT-L-2-v2 + ms-marco-MiniLM-L-2." This shows that "ms-marco-TinyBERT-L-2-v2" and "distilroberta-base" have complementary strengths that enhance the performance of the ensemble. The worst-performing duo, on the other hand, might not have this synergy or do poorly alone.

4.1.2 Performance Variations. Yes, the performance of the ensemble approach varies among various model assemblages. This is most likely caused by the distinct advantages and disadvantages of each model. Depending on the particular model combination, combining several models can maximize their individual strengths and reduce their shortcomings, resulting in a range of performance levels.

4.1.3 Unintuitive Results. Unexpectedly, the combination of the two models, "ms-marco-TinyBERT-L-2-v2 + ms-marco-MiniLM-L-2," which were both optimized using the same dataset, fared the poorest. This implies that these models might have similar flaws, and combining them wouldn't be as beneficial as combining models from more diverse backgrounds. These findings highlight the value of model diversity in ensemble techniques.

REFERENCES

[1] Elias Bassani and Luca Romelli. Ranx.fuse: A python library for metasearch. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, CIKM '22, page 4808–4812, New York, NY, USA, 2022. Association for Computing Machinery.

[2] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. Ms marco: A human-generated machine reading comprehension dataset. 2016.