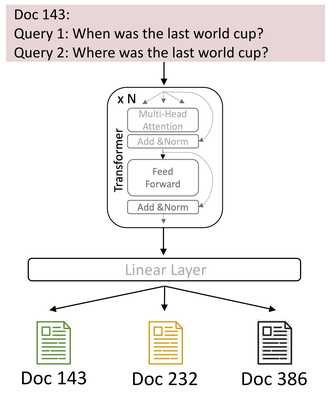
IncDSI: Incrementally Updatable Document Retrieval

Varsha Kishore, Chao Wan, Justin Lovelace, Yoav Artzi, Kilian Q. Weinberger

Paper Code

Background

- Differentiable search index [1] is a new paradigm for end-to-end document retrieval.
- Information about documents is stored within the parameters of a neural network.
- Model is trained to directly map queries and documents to corresponding IDs with cross-entropy loss.



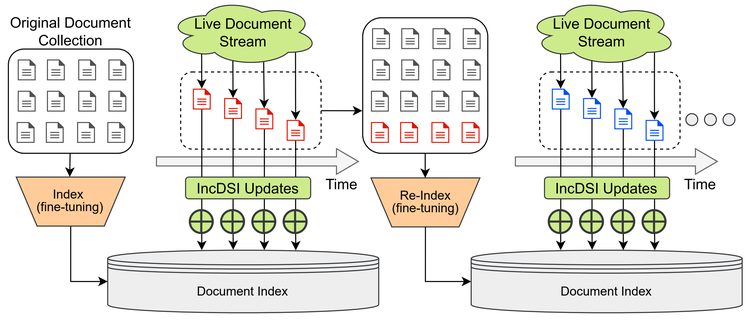
[1] Tay, Yi, et al. "Transformer memory as a differentiable search index." *Advances in Neural Information Processing Systems* 35 (2022): 21831-21843.

Pro: Good retrieval performance compared to baselines.

Con: New documents cannot be added to the model easily.

Online Updates

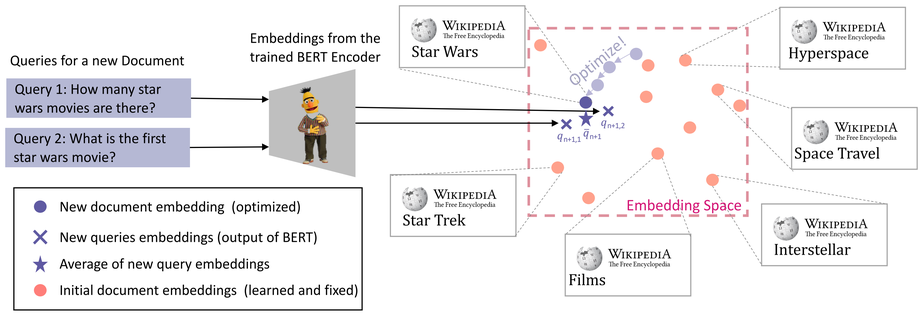
A document retrieval model is trained by indexing a given set of documents. New documents later become available and must be added to the index, with low computational overhead.



The system needs to satisfy the following two properties:

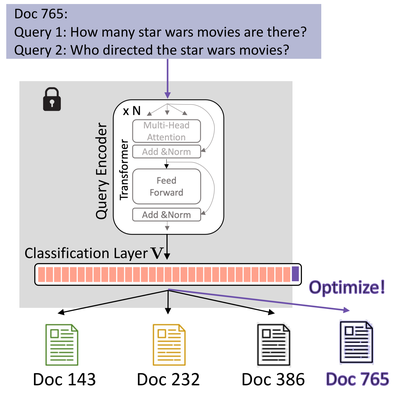
1. Information about new documents has to be incorporated into the already trained model, such that the documents can be retrieved for corresponding queries.
2. Information about previously indexed documents needs to be retained so queries corresponding to the old documents are still correctly mapped to the right document.

Adding Documents as Lightweight Optimization



Optimization

Insight: the DSI network can be viewed as a query encoder and a classification layer which has one document vector for every indexed document.



We add a new document by optimizing a single new document vector. Given document vectors $\mathbf{V} \in \mathbb{R}^{n \times d}$, average new query embedding $\bar{\mathbf{q}}_{n+1}$, and average old query embeddings $\bar{\mathbf{z}}_j$, we find a new vector $\mathbf{v}_{n+1}$ for a new document by solving the following optimization problem:

$$\min \quad \|\mathbf{v}_{n+1}\|_2^2$$

$$\text{s.t.} \quad \bar{\mathbf{q}}_{n+1}^T \mathbf{v}_{n+1} > \max_{1 \le j \le n} \bar{\mathbf{q}}_{n+1}^T \mathbf{v}_j$$

New document must be scored higher than existing documents for new queries

$$\forall_j \bar{\mathbf{z}}_j^T \mathbf{v}_{n+1} < \bar{\mathbf{z}}_j^T \mathbf{v}_j$$

Queries for the original documents should not retrieve the new document

The optimization problem can be solved by re-writing the constraints as loss violations (with margins) and minimizing the following loss function:

$$\mathcal{L}(\mathbf{v}_{n+1}) = \lambda_2 \|\mathbf{v}_{n+1}\|_2^2 + \lambda_1 \max(0, (\max_j(\bar{\mathbf{q}}_{n+1}^T \mathbf{v}_j) - \bar{\mathbf{q}}_{n+1}^T \mathbf{v}_{n+1})) + \gamma_1)^2 + (1-\lambda_1) \sum_j \max(0, \bar{\mathbf{z}}_j^T \mathbf{v}_{n+1} - \bar{\mathbf{z}}_j^T \mathbf{v}_j + \gamma_2)^2$$

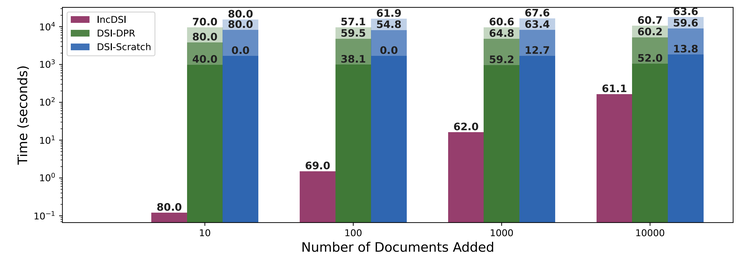
Loss is minimized with L-BFGS optimization!

Baselines

- DPR: A standard dual-encoder model that is trained with contrastive loss. The frozen model can be used to encode new documents and queries.
- Continual training with frozen DPR (DSI-DPR): A model with a frozen DPR encoder and a trainable classification layer is continually fine-tuned with new documents and queries.
- Continual training (DSI-Scratch): A DSI model is continually fine-tuned with new documents and queries.

Results

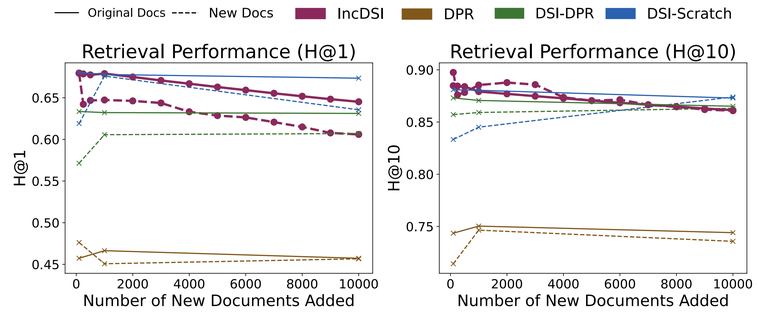
Time required to add new documents:



Numbers on the bars are hit@1 for new documents. Lighter shades in the stacked bars indicate later checkpoints from training the model (epochs 1,5,10).

Each new document can be added in 20-50ms per document. IncDSI is magnitudes faster!

Retrieval performance as documents are added:



Retrieval performance is close to training from scratch for queries from old and new documents!