Set-Encoder:

Permutation-Invariant Inter-Passage Attention for Listwise Passage Re-Ranking with Cross-Encoders

ECIR 2025, Lucca, Italy, 06.04.2025–10.04.2025

Ferdinand Schlatt   Maik Fröbe   Harrisen Scells
Shengyao Zhuang   Bevan Koopman   Guido Zuccon
Benno Stein   Martin Potthast   Matthias Hagen

Friedrich-Schiller University Jena   University of Kassel   CSIRO
University of Queensland   Bauhaus-Universität Weimar
hessian.AI   ScadDS.AI

Cross-Encoders

Comparing Point-, Pair-, and Listwise Models

Query | python course
Documents | Python is a great language to learn.
Pythons live in the rainforest.
Guido van Rossum invented Python.

Cross-Encoders

Comparing Point-, Pair-, and Listwise Models

Query: python course

Documents:

Python is a great language to learn.
Pythons live in the rainforest.
Guido van Rossum invented Python.

monoBERT (Pointwise) [Nogueira and Cho, arXiv'19]

[CLS] [SEP] [SEP] → BERT → 2.4
[CLS] [SEP] [SEP] → BERT → 0.1
[CLS] [SEP] [SEP] → BERT → 1.9

Cross-Encoders

Comparing Point-, Pair-, and Listwise Models

Query: python course

Documents:

Python is a great language to learn.
Pythons live in the rainforest.
Guido van Rossum invented Python.

monoBERT (Pointwise) [Nogueira and Cho, arXiv’19]

[CLS] [SEP] [SEP] → BERT → 2.4
[CLS] [SEP] [SEP] → BERT → 0.1
[CLS] [SEP] [SEP] → BERT → 1.9

Issue: The model scores each document independently.

Cross-Encoders

Comparing Point-, Pair-, and Listwise Models

Query: python course

Documents:
Python is a great language to learn.
Pythons live in the rainforest.
Guido van Rossum invented Python.

monoBERT (Pointwise) [Nogueira and Cho, arXiv'19]

[CLS] [SEP] [SEP] → BERT → 2.4
[CLS] [SEP] [SEP] → BERT → 0.1
[CLS] [SEP] [SEP] → BERT → 1.9

Issue: The model scores each document independently.

➜ Listwise (and pairwise) models enable interactions between documents.

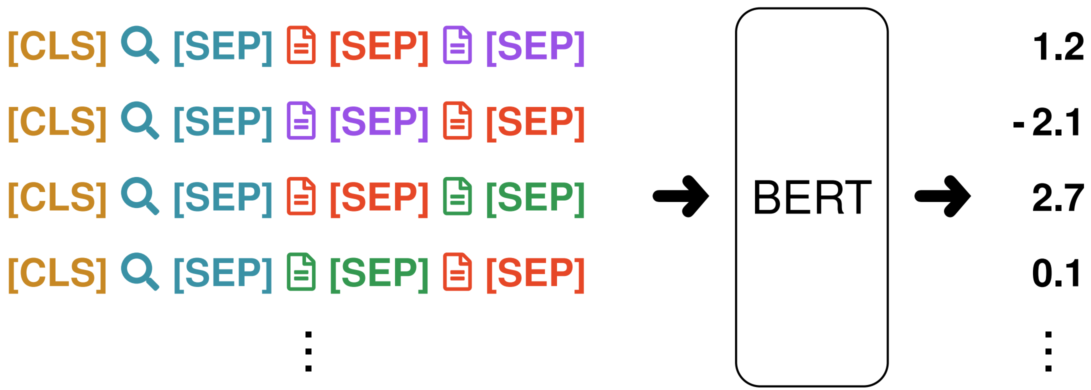
Cross-Encoders

Comparing Point-, Pair-, and Listwise Models

Query: python course

Documents:
Python is a great language to learn.
Pythons live in the rainforest.
Guido van Rossum invented Python.

duoBERT (Pairwise) [Nogueira et al., arXiv'20]



Cross-Encoders

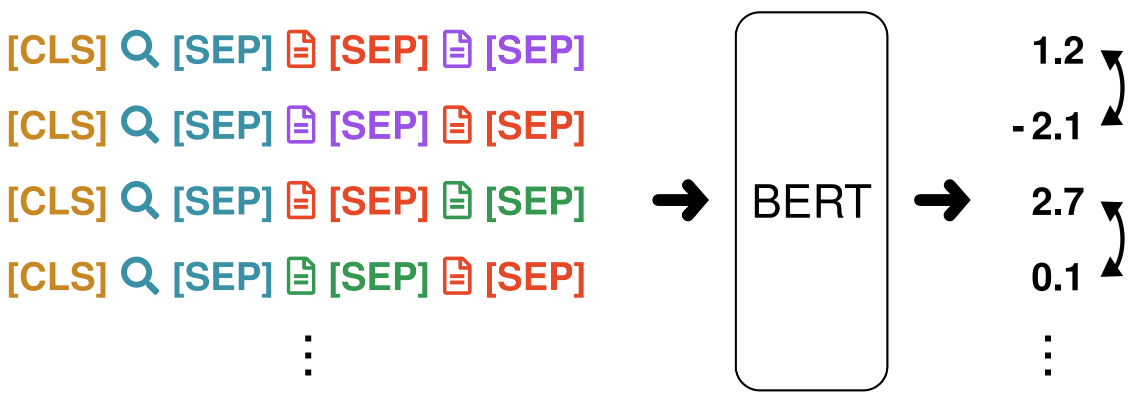
Comparing Point-, Pair-, and Listwise Models

Query python course

Documents

Python is a great language to learn.
Pythons live in the rainforest.
Guido van Rossum invented Python.

duoBERT (Pairwise) [Nogueira et al., arXiv'20]



Issue: Relevance scores are not symmetric.

Cross-Encoders

Comparing Point-, Pair-, and Listwise Models

Query: python course

Documents:

Python is a great language to learn.
Pythons live in the rainforest.
Guido van Rossum invented Python.

RankGPT (Listwise) [Sun et al., EMNLP'23]

Input		Model		Output
Prompt: ... Query: [1]: [2]: [3]:				1 > 3 > 2
Prompt: ... Query: [1]: [3]: [2]:				1 > 3 > 2
Prompt: ... Query: [2]: [1]: [3]:	→	GPT	→	1 > 3 > 2
Prompt: ... Query: [2]: [3]: [1]:				3 > 1 > 2
⋮				⋮

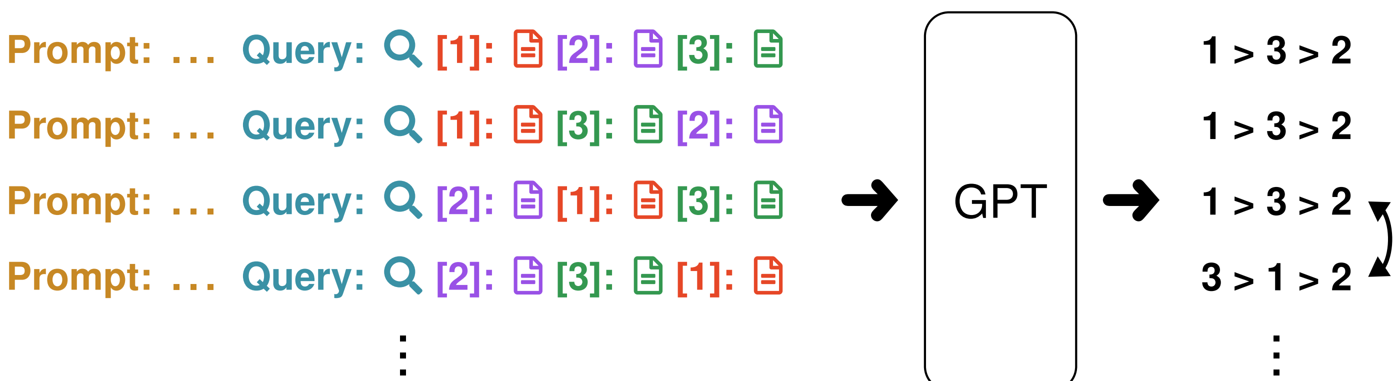
Cross-Encoders

Comparing Point-, Pair-, and Listwise Models

Query: python course

Documents:
Python is a great language to learn.
Pythons live in the rainforest.
Guido van Rossum invented Python.

RankGPT (Listwise) [Sun et al., EMNLP'23]



Issue: Relevance preference order is not consistent.

Set-Encoder

Attention Mechanism

[CLS] python course [SEP] Python is a great language to learn . [SEP]

[CLS] python course [SEP] Pythons live in the rainforest . [SEP]

[CLS] python course [SEP] Guido van Rossum invented Python . [SEP]

Set-Encoder

Attention Mechanism

[CLS] python course [SEP] Python is a great language to learn . [SEP]

[CLS] python course [SEP] Pythons live in the rainforest . [SEP]

[CLS] python course [SEP] Guido van Rossum invented Python . [SEP]

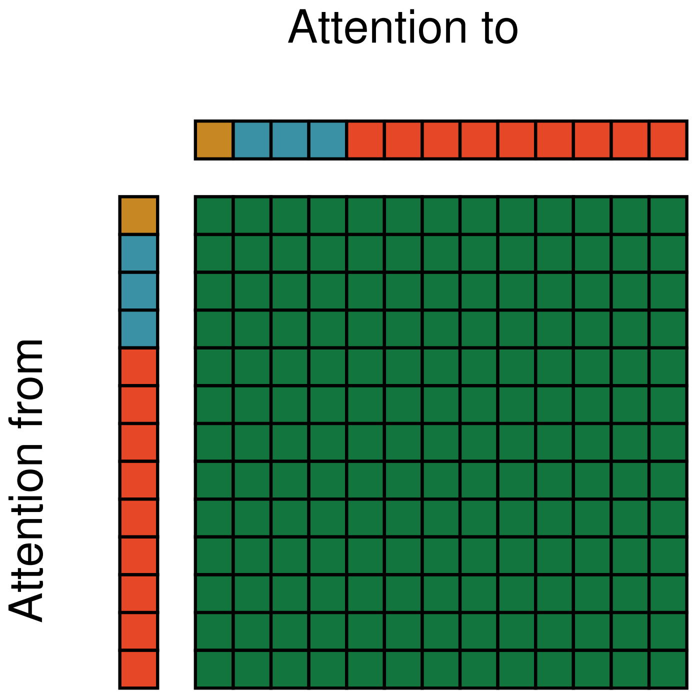
Set-Encoder

Attention Mechanism

[CLS] python course [SEP] Python is a great language to learn . [SEP]

[CLS] python course [SEP] Pythons live in the rainforest . [SEP]

[CLS] python course [SEP] Guido van Rossum invented Python . [SEP]



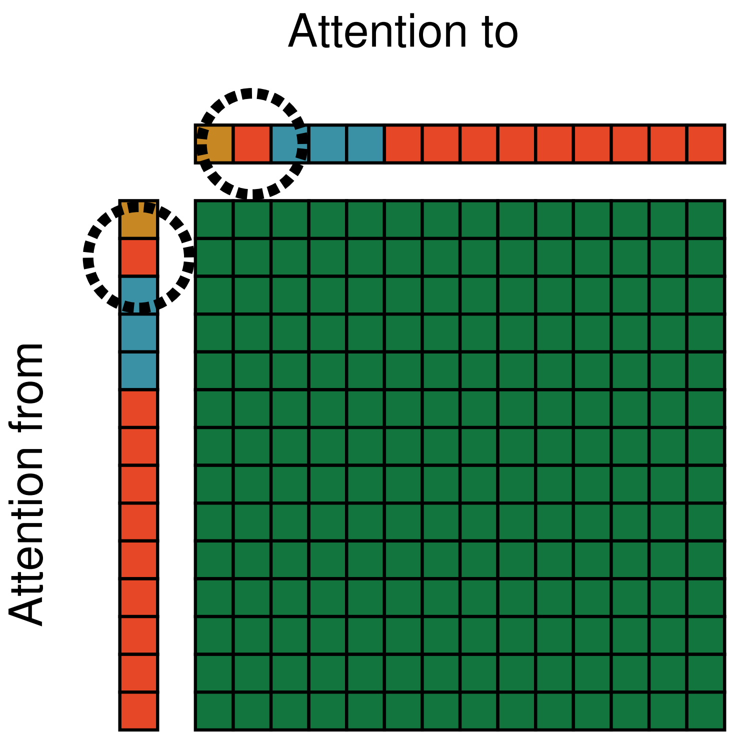
Set-Encoder

Attention Mechanism

[CLS] [INT] python course [SEP] Python is a great language to learn . [SEP]

[CLS] [INT] python course [SEP] Pythons live in the rainforest . [SEP]

[CLS] [INT] python course [SEP] Guido van Rossum invented Python . [SEP]



1. Insert an extra [INT] token

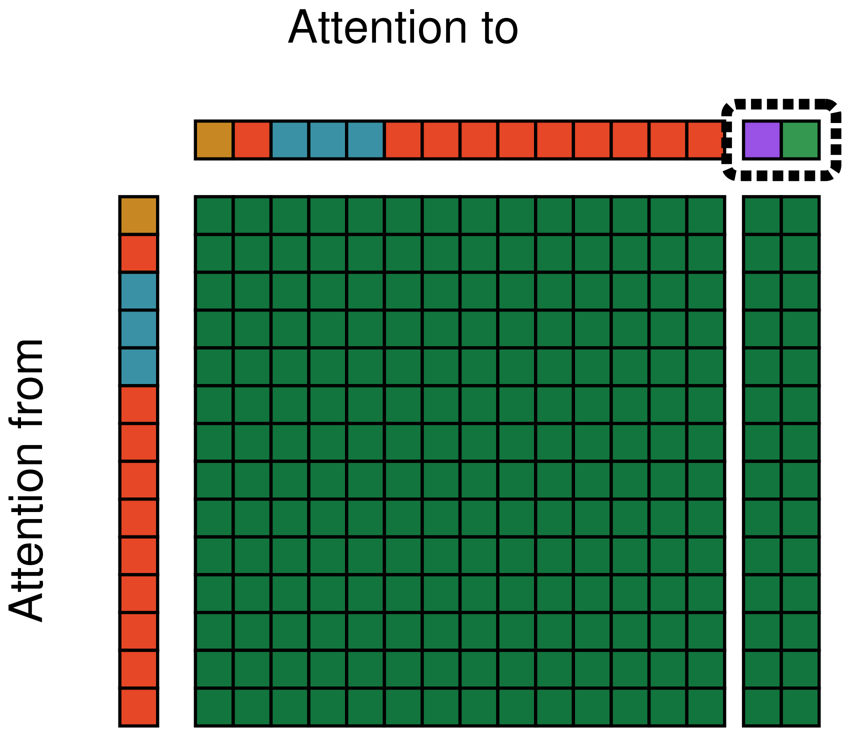
Set-Encoder

Attention Mechanism

[CLS] [INT] python course [SEP] Python is a great language to learn . [SEP]

[CLS] [INT] python course [SEP] Pythons live in the rainforest . [SEP]

[CLS] [INT] python course [SEP] Guido van Rossum invented Python . [SEP]



1. Insert an extra [INT] token
2. Allow a document to attend to all other documents' [INT] tokens

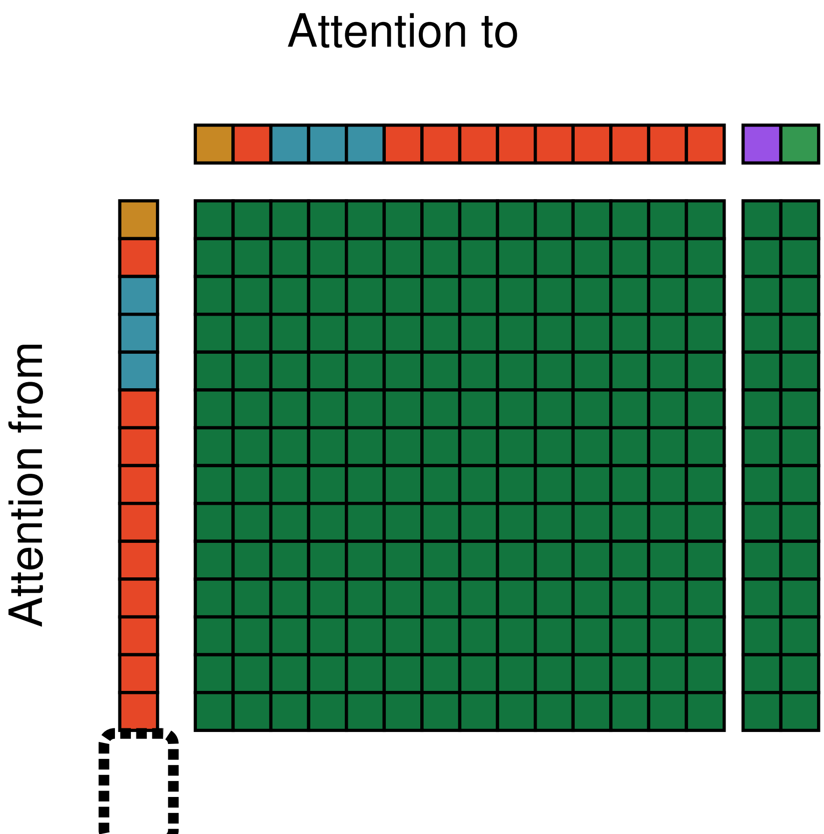
Set-Encoder

Attention Mechanism

[CLS] [INT] python course [SEP] Python is a great language to learn . [SEP]

[CLS] [INT] python course [SEP] Pythons live in the rainforest . [SEP]

[CLS] [INT] python course [SEP] Guido van Rossum invented Python . [SEP]



1. Insert an extra [INT] token
2. Allow a document to attend to all other documents' [INT] tokens

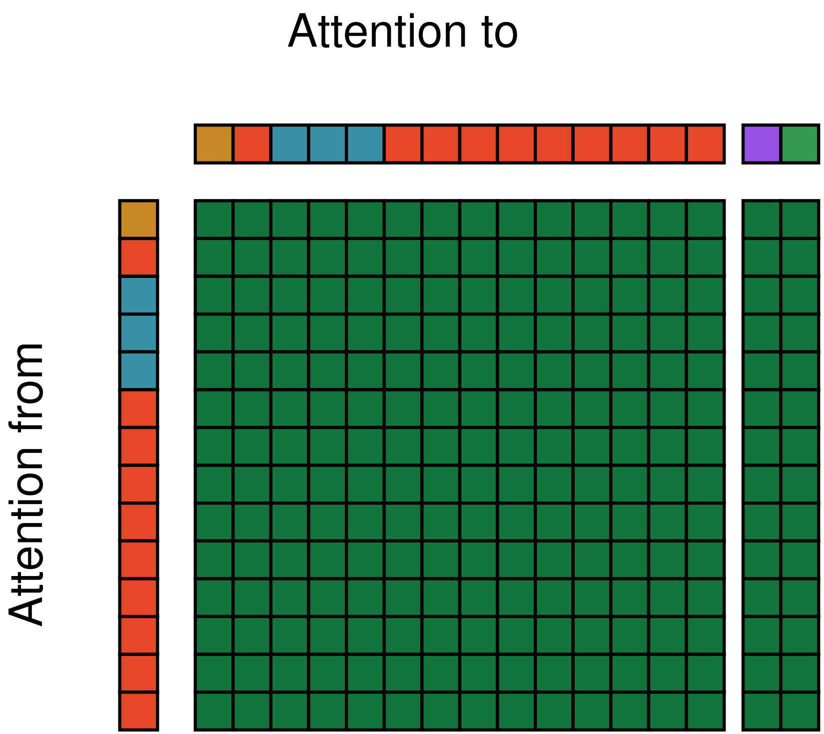
Set-Encoder

Attention Mechanism

[CLS] [INT] python course [SEP] Python is a great language to learn . [SEP]

[CLS] [INT] python course [SEP] Pythons live in the rainforest . [SEP]

[CLS] [INT] python course [SEP] Guido van Rossum invented Python . [SEP]



1. Insert an extra [INT] token
2. Allow a document to attend to all other documents' [INT] tokens

- [INT] tokens aggregate semantic information and shares information with other documents

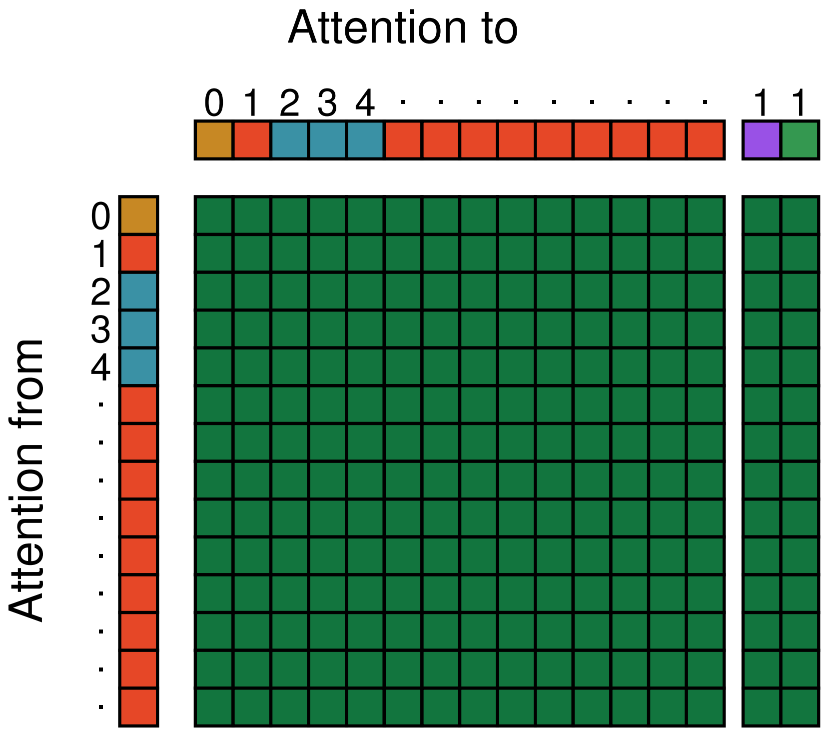
Set-Encoder

Attention Mechanism

[CLS] [INT] python course [SEP] Python is a great language to learn . [SEP]

[CLS] [INT] python course [SEP] Pythons live in the rainforest . [SEP]

[CLS] [INT] python course [SEP] Guido van Rossum invented Python . [SEP]



1. Insert an extra [INT] token
2. Allow a document to attend to all other documents' [INT] tokens

- [INT] tokens aggregate semantic information and shares information with other documents
- Permutation-invariant because all [INT] tokens share the same positional encoding

Set-Encoder

Effectiveness

nDCG@10 on TREC Deep Learning 2019 and 2020 passage and TIREx

Model	TREC DL 19	TREC DL 20	TIREx
BM25	0.480	0.494	0.286
monoT5 3B	0.705	0.715	0.313
RankT5 3B	0.710	0.711	**0.322**
RankGPT-4o	0.725	0.719	–
RankZephyr	0.719	0.720	0.320
Set-Encoder$_{\mathtt{BASE}}$			
Set-Encoder$_{\mathtt{LARGE}}$			

Set-Encoder

Effectiveness

nDCG@10 on TREC Deep Learning 2019 and 2020 passage and TIREx

Model	TREC DL 19	TREC DL 20	TIREx
BM25	0.480	0.494	0.286
monoT5 3B	0.705	0.715	0.313
RankT5 3B	0.710	0.711	**0.322**
RankGPT-4o	0.725	0.719	–
RankZephyr	0.719	0.720	0.320
Set-Encoder$_{\texttt{BASE}}$	0.724	0.710	0.311
Set-Encoder$_{\texttt{LARGE}}$			

1. Set-Encoder is on-par with SOTA re-rankers in-domain

Set-Encoder

Effectiveness

nDCG@10 on TREC Deep Learning 2019 and 2020 passage and TIREx

Model	**TREC DL 19**	**TREC DL 20**	**TIREx**
BM25	0.480	0.494	0.286
monoT5 3B	0.705	0.715	0.313
RankT5 3B	0.710	0.711	**0.322**
RankGPT-4o	0.725	0.719	–
RankZephyr	0.719	0.720	0.320
Set-Encoder$_{\mathtt{BASE}}$	0.724	0.710	0.311
Set-Encoder$_{\mathtt{LARGE}}$	**0.727**	**0.735**	<u>0.321</u>

1. Set-Encoder is on-par with SOTA re-rankers in-domain and out-of-domain.

Set-Encoder

Effectiveness

nDCG@10 on TREC Deep Learning 2019 and 2020 passage and TIREx

Model	**TREC DL 19**	**TREC DL 20**	**TIREx**
BM25	0.480	0.494	0.286
monoT5 3B	0.705	0.715	0.313
RankT5 3B	0.710	0.711	**0.322**
RankGPT-4o	0.725	0.719	–
RankZephyr	0.719	0.720	0.320
Set-Encoder$_{\texttt{BASE}}$	0.724	0.710	0.311
Set-Encoder$_{\texttt{LARGE}}$	**0.727**	**0.735**	<u>0.321</u>

1. Set-Encoder is on-par with SOTA re-rankers in-domain and out-of-domain.
2. Despite being distilled from RankZephyr, the Set-Encoder is more effective.

Set-Encoder

Effectiveness

nDCG@10 on TREC Deep Learning 2019 and 2020 passage and TIREx

Model	TREC DL 19	TREC DL 20	TIREx
BM25	0.480	0.494	0.286
monoT5 3B	0.705	0.715	0.313
RankT5 3B	0.710	0.711	**0.322**
RankGPT-4o	0.725	0.719	–
RankZephyr	0.719	0.720	0.320
Set-Encoder$_{\texttt{BASE}}$	0.724	0.710	0.311
Set-Encoder$_{\texttt{LARGE}}$	**0.727**	**0.735**	<u>0.321</u>

1. Set-Encoder is on-par with SOTA re-rankers in-domain and out-of-domain.
2. Despite being distilled from RankZephyr, the Set-Encoder is more effective.

➜ LLM-rankers are not permutation-invariant and affected by the first-stage.

Set-Encoder

Permutation Invariance

Re-ordering input documents affects previous listwise model’s ranking preferences.

Set-Encoder

Permutation Invariance

Re-ordering input documents affects previous listwise model's ranking preferences.

We create corrupted BM25 rankings to test a model's robustness to permutations.

1. Inverse ideal ranking
2. Randomly shuffled ranking
3. Original BM25 ranking
4. Ideal ranking

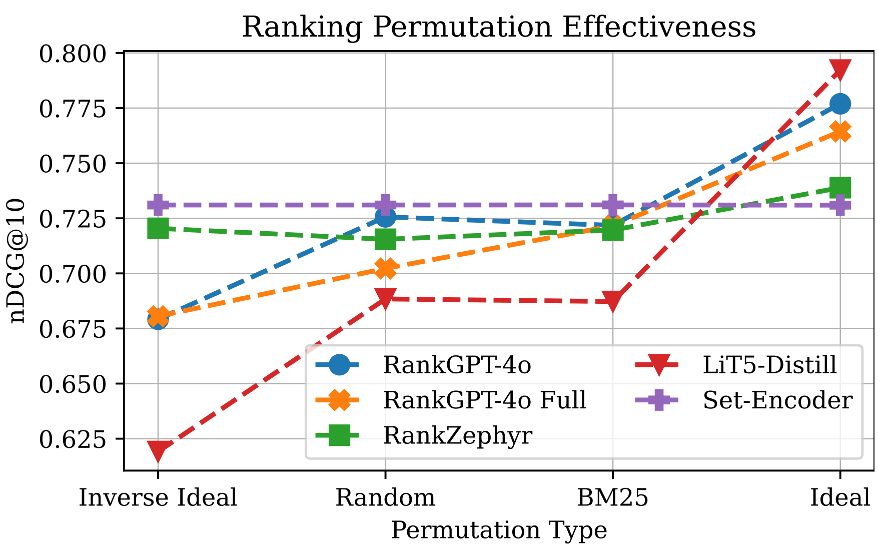
Set-Encoder

Permutation Invariance

Re-ordering input documents affects previous listwise model's ranking preferences.

We create corrupted BM25 rankings to test a model's robustness to permutations.

1. Inverse ideal ranking
2. Randomly shuffled ranking
3. Original BM25 ranking
4. Ideal ranking



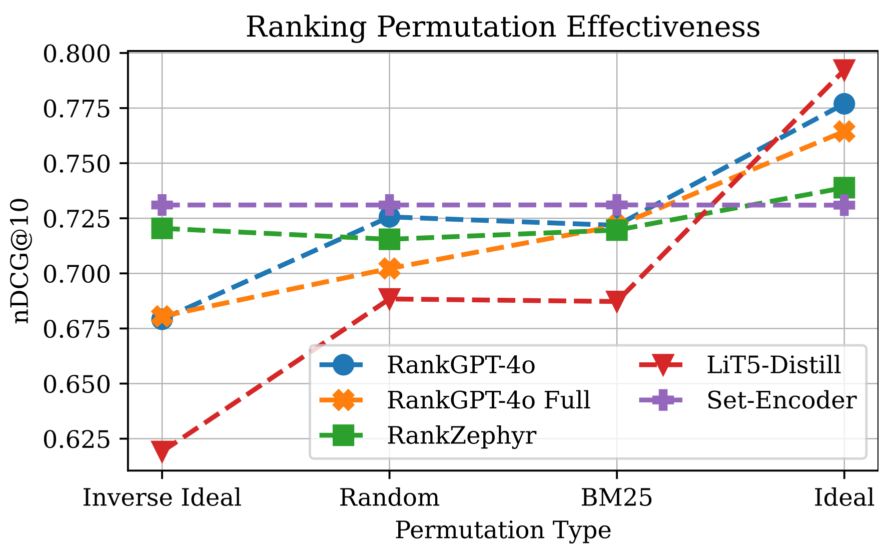
Set-Encoder

Permutation Invariance

Re-ordering input documents affects previous listwise model's ranking preferences.

We create corrupted BM25 rankings to test a model's robustness to permutations.

1. Inverse ideal ranking
2. Randomly shuffled ranking
3. Original BM25 ranking
4. Ideal ranking



1. Previous listwise re-rankers are biased by the order of the input documents.

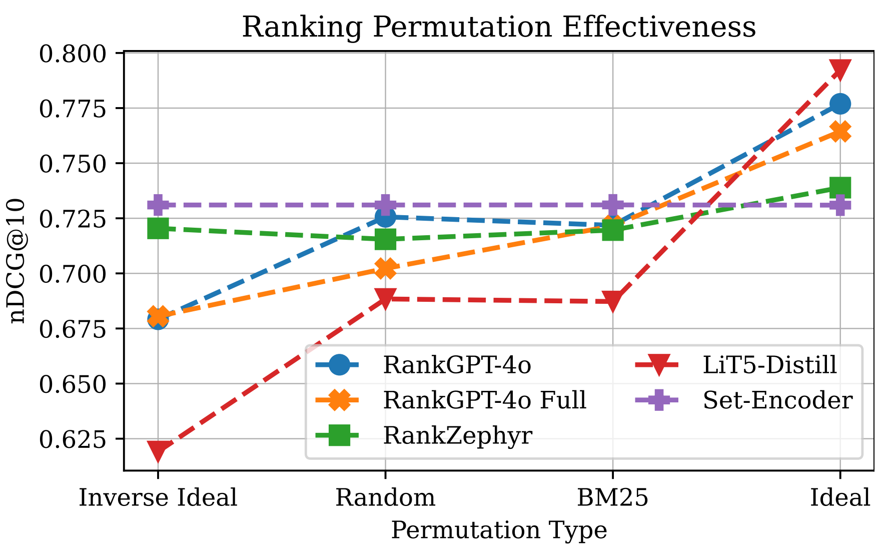
Set-Encoder

Permutation Invariance

Re-ordering input documents affects previous listwise model's ranking preferences.

We create corrupted BM25 rankings to test a model's robustness to permutations.

1. Inverse ideal ranking
2. Randomly shuffled ranking
3. Original BM25 ranking
4. Ideal ranking



1. Previous listwise re-rankers are biased by the order of the input documents.
2. Set-Encoder is invariant to the order of the input documents.

Set-Encoder

Conclusion

We introduced the Set-Encoder architecture that enables inter-document interactions in a permutation-invariant way.

Set-Encoder

Conclusion

We introduced the Set-Encoder architecture that enables inter-document interactions in a permutation-invariant way.

- The Set-Encoder is on-par with SOTA LLM re-rankers.

Set-Encoder

Conclusion

We introduced the Set-Encoder architecture that enables inter-document interactions in a permutation-invariant way.

- The Set-Encoder is on-par with SOTA LLM re-rankers.
- Permutation invariance is crucial for robustness and efficiency.

Set-Encoder

Conclusion

We introduced the Set-Encoder architecture that enables inter-document interactions in a permutation-invariant way.

- The Set-Encoder is on-par with SOTA LLM re-rankers.
- Permutation invariance is crucial for robustness and efficiency.
- **BUT:** Document interactions may not be necessary for Cranfield-style tasks.

Set-Encoder
Conclusion

We introduced the Set-Encoder architecture that enables inter-document interactions in a permutation-invariant way.

- The Set-Encoder is on-par with SOTA LLM re-rankers.
- Permutation invariance is crucial for robustness and efficiency.
- **BUT:** Document interactions may not be necessary for Cranfield-style tasks.

Checkpoints are released on HF and can be used with our Lightning IR framework.

Lightning IR

Set-Encoder

Conclusion

We introduced the Set-Encoder architecture that enables inter-document interactions in a permutation-invariant way.

- The Set-Encoder is on-par with SOTA LLM re-rankers.
- Permutation invariance is crucial for robustness and efficiency.
- **BUT:** Document interactions may not be necessary for Cranfield-style tasks.

Checkpoints are released on HF and can be used with our Lightning IR framework.

Lightning IR

Code and paper @
webis-de/set-encoder

Set-Encoder

Conclusion

We introduced the Set-Encoder architecture that enables inter-document interactions in a permutation-invariant way.

- The Set-Encoder is on-par with SOTA LLM re-rankers.
- Permutation invariance is crucial for robustness and efficiency.
- **BUT:** Document interactions may not be necessary for Cranfield-style tasks.

Checkpoints are released on HF and can be used with our Lightning IR framework.

Lightning IR

Thanks!

Code and paper @
webis-de/set-encoder

Set-Encoder

Listwise Re-Ranking

We build a synthetic task which requires document interactions.

Set-Encoder

Listwise Re-Ranking

We build a synthetic task which requires document interactions.

MS MARCO contains many lexical near-duplicates.

Set-Encoder

Listwise Re-Ranking

We build a synthetic task which requires document interactions.

MS MARCO contains many lexical near-duplicates. (Not actual MS MARCO data.)

Python is a great language to learn.

Python is a great language to learn now.

Pythons live in the rainforest.

Guido van Rossum invented Python.

Set-Encoder

Listwise Re-Ranking

We build a synthetic task which requires document interactions.

MS MARCO contains many lexical near-duplicates. (Not actual MS MARCO data.)

Python is a great language to learn.

Python is a great language to learn now.

Pythons live in the rainforest.

Guido van Rossum invented Python.

Fine-tune models to rank according to relevance and put duplicates at the end.

Set-Encoder

Listwise Re-Ranking

We build a synthetic task which requires document interactions.

α-nDCG@10 ($\alpha = 0.99$) on the synthetic task

Model	**TREC DL 19**	**TREC DL 20**
First Stage	0.700	0.722
RankGPT-4o	0.741	0.773
RankZephyr	0.700	0.760
monoELECTRA	0.785	0.753
Set-Encoder	**0.821**	**0.803**

Set-Encoder

Listwise Re-Ranking

We build a synthetic task which requires document interactions.

α-nDCG@10 ($\alpha = 0.99$) on the synthetic task

Model	TREC DL 19	TREC DL 20
First Stage	0.700	0.722
RankGPT-4o	0.741	0.773
RankZephyr	0.700	0.760
monoELECTRA	0.785	0.753
Set-Encoder	**0.821**	**0.803**

➜ Set-Encoder improves over baselines in novelty-aware re-ranking.

Set-Encoder

Listwise Re-Ranking

Previous listwise re-rankers are biased by the order of the input documents.

Set-Encoder

Listwise Re-Ranking

Previous listwise re-rankers are biased by the order of the input documents.

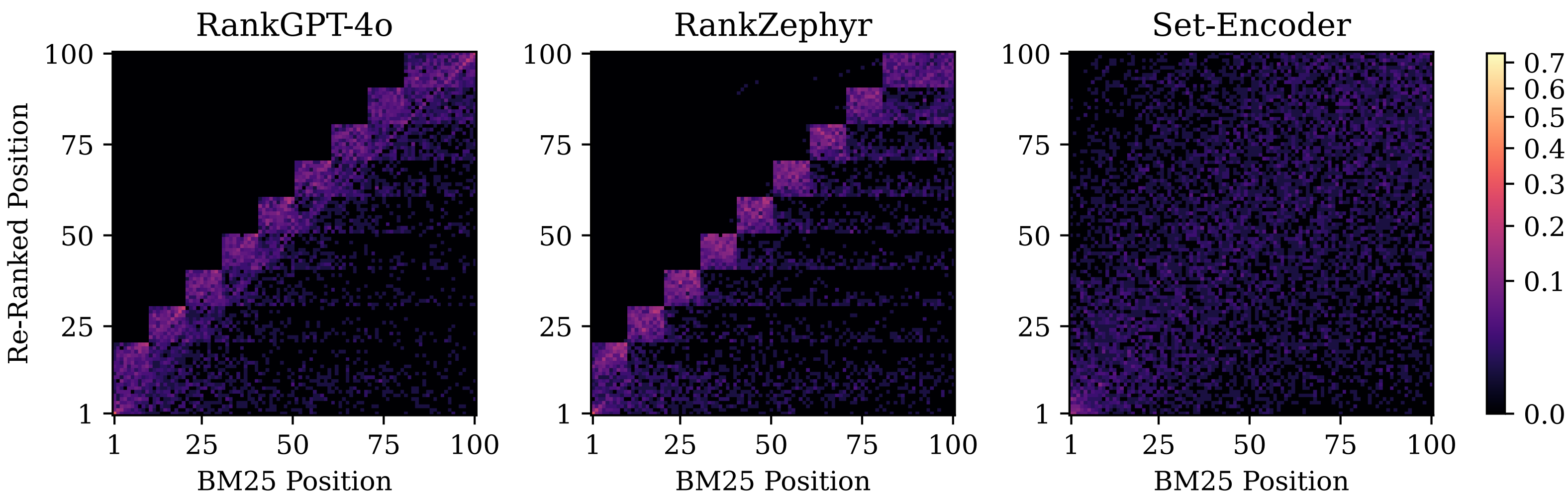
What do these biases look like? ➜ Plot the original vs re-ranked position.

Set-Encoder

Listwise Re-Ranking

Previous listwise re-rankers are biased by the order of the input documents.

What do these biases look like? ➜ Plot the original vs re-ranked position.

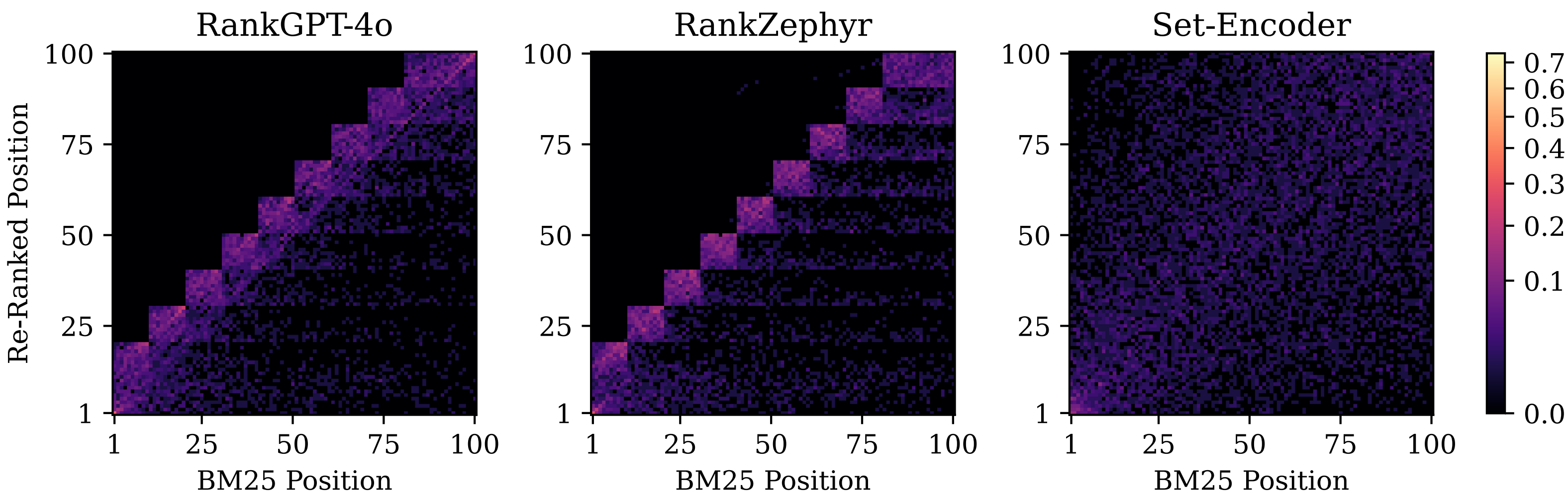


Set-Encoder

Listwise Re-Ranking

Previous listwise re-rankers are biased by the order of the input documents.

What do these biases look like? ➜ Plot the original vs re-ranked position.



A substantial number of previous works attempt to mitigate these positional biases.
[Zhuang et al., SIGIR'24; Parry et al., arXiv'24]

➜ Making the model permutation-invariant is a more principled approach.

Set-Encoder
TREC DL

Model	Size	TREC DL 19		TREC DL 20	
		BM25	CBv2	BM25	CBv2
First Stage	–	0.480†	0.732†	0.494†	0.724†
RankGPT-4o	N/A	0.725	0.784	0.719	0.793
RankGPT-4o Full	N/A	<u>0.732</u>	0.781	0.711	0.796
RankZephyr	7B	0.719	0.749	0.720	<u>0.798</u>
LiT5-Distill	220M	0.696	0.753	0.679†	0.744†
monoT5 3B	3B	0.705	0.745	0.715	0.757
RankT5 3B	3B	0.710	0.752	0.711	0.772
monoELECTRA	110M	0.720	0.768	0.711†	0.770
	330M	**0.733**	0.765	<u>0.727</u>	**0.799**
Set-Encoder	110M	0.724	<u>0.788</u>	0.710†	0.777
	330M	0.727	**0.789**	**0.735**	0.790

Set-Encoder
TREC DL Novelty

	Model	Prompt / Loss		nDCG		α-nDCG	
				2019	2020	2019	2020
(1)	First Stage	–		0.732	0.724	0.700†	0.722†
(2)	RankGPT-4o	Relevance		0.784†	0.793†	0.750	0.759
(3)		Novelty		0.778†	**0.806**†	0.741	0.773
(4)	RankGPT-4o Full	Relevance		0.781†	0.796†	0.738	0.763
(5)		Novelty		0.785†	0.803†	0.750	0.771
(6)	RankZephyr	Relevance		0.749	0.798†	0.699†	0.765
(7)		Novelty		0.753	0.800†	0.700†	0.760
(8)	**Model**	**1st $\mathcal{L}$**	**2nd $\mathcal{L}$**				
(9)	monoELECTRA	$\mathcal{L}_{\text{InfoNCE}}$	$\mathcal{L}_{\text{RankNet}}$	0.768†	0.770†	0.718†	0.745†
(10)			$\mathcal{L}_{\text{NA-RankNet}}$	0.704	0.675	0.785	0.753
(11)	Set-Encoder	$\mathcal{L}_{\text{InfoNCE}}$	$\mathcal{L}_{\text{RankNet}}$	0.780†	0.757†	0.733†	0.747†
(12)			$\mathcal{L}_{\text{NA-RankNet}}$	0.714	0.651	0.779	0.743†
(13)		$\mathcal{L}_{\text{DA-InfoNCE}}$	$\mathcal{L}_{\text{RankNet}}$	**0.788**†	0.777†	0.740†	0.752†
(14)			$\mathcal{L}_{\text{NA-RankNet}}$	0.710	0.690	**0.821**	**0.803**
(15)	Set-Enc. ~~[INT]~~	$\mathcal{L}_{\text{DA-InfoNCE}}$	$\mathcal{L}_{\text{NA-RankNet}}$	0.707	0.670	0.773	0.748†

Set-Encoder

Out-of-Domain Re-Ranking

Model	Size	Antique	Args.me	ClueWeb09	ClueWeb12	CORD-19	Cranfield	Disks4+5	GOV	GOV2	MEDLINE	NFCorpus	Vaswani	WaPo	G. Mean
First Stage	–	.516†	.405	.177†	**.364**†	.586†	**.012**	.424†	.259†	.467†	.385	.281†	.447†	.364†	.286
RankZephyr	7B	.534†	.364†	.213	.303	**.767**†	.009	.542	.349	.560	**.460**†	.314	.512	**.508**	.320
LiT5-Distill	220M	.576†	.395	.214	.275†	.686	.011	.495†	.304†	.534†	.354†	.293†	.429†	.470	.302
monoT5 3B	3B	.590	.415	.188†	.323	.649†	.011	.526	.345	.529†	.395	.319	.474†	.469	.313
RankT5 3B	3B	.598	**.421**	**.227**	.336	.713	.010	.538	**.353**	.528†	.406	**.323**	.459†	.468†	**.322**
m.ELECTRA	110M	.593	.375†	.209	.295	.692	.010	.507†	.305†	.541†	.399	.306	.522	.458†	.309
	330M	.575†	.369†	.221	.313	.716	.008	**.546**	.344	.572	.419	.316	.526	.504	.318
Set-Encoder	110M	.594	.375†	.216	.299	.683	.010	.513†	.306†	.543†	.396	.306	.523	.461†	.311
	330M	**.606**	.409	.226	.310	.702	.009	.534	.334	**.573**	.405	.313	**.530**	**.508**	.321

Set-Encoder

Efficiency

Model	Size	Time	Memory
RankGPT-4o	N/A	18.773	N/A
RankGPT-4o Full	N/A	7.362	N/A
RankZephyr	7B	24.047	15.48
LiT5-Distill	220M	2.054	2.69
monoT5$_{3B}$	3B	0.998	29.36
RankT5$_{3B}$	3B	0.942	29.04
monoELECTRA	110M	0.139	1.18
	330M	0.215	2.69
Set-Encoder	110M	0.147	1.25
	330M	0.219	2.60