Efficient Sparse Retrieval with Lightweight Superblock Pruning

Parker Carlson, Wentai Xie, Rohil Shah, Tao Yang
University of California, Santa Barbara

Background & Motivation

- Recent retrieval algorithms BMP and SP [SIGIR ’24, ’25] use a very large number of clusters to accelerate retrieval

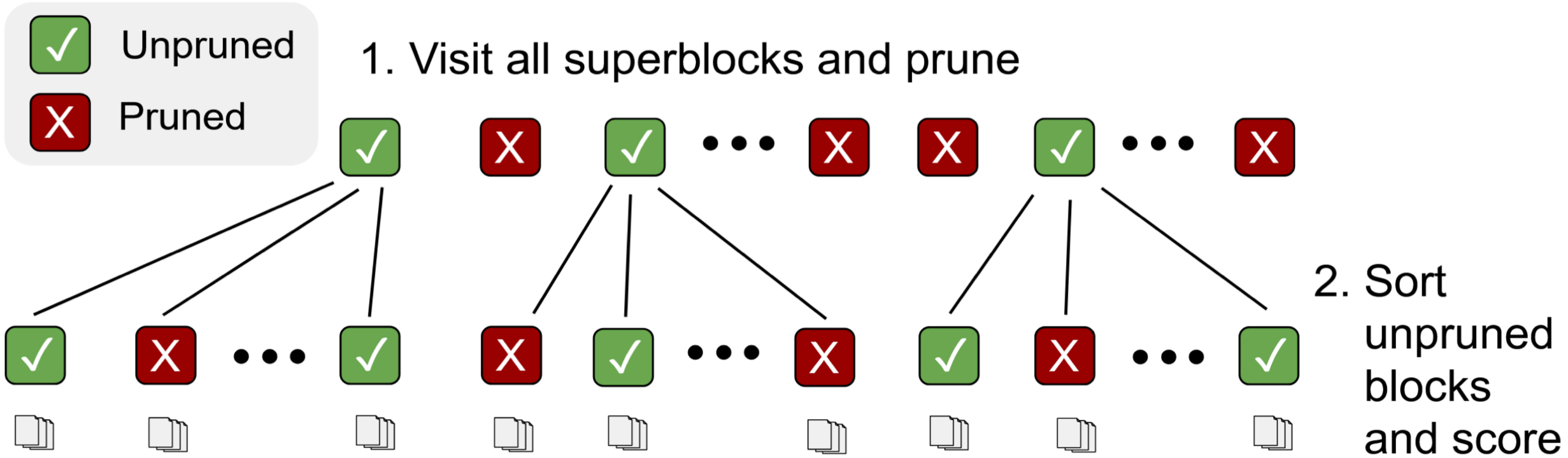
Background & Motivation

- Recent retrieval algorithms BMP and SP [SIGIR ’24, ’25] use a very large number of clusters to accelerate retrieval
- However, they both have large index sizes and struggle with cluster overpruning during approximate retrieval

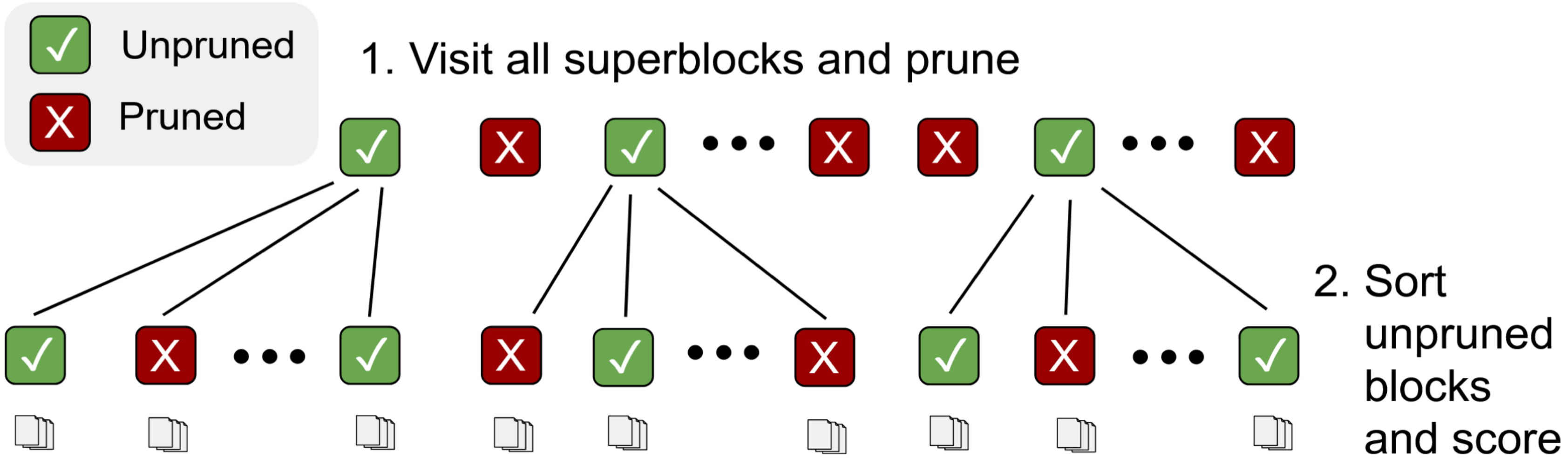
Background & Motivation

- Recent retrieval algorithms BMP and SP [SIGIR ’24, ’25] use a very large number of clusters to accelerate retrieval
- However, they both have large index sizes and struggle with cluster overpruning during approximate retrieval
- In this work, we investigate the design choices of BMP and SP and propose improvements to address their weaknesses

Review: SP Architecture [SIGIR '25]



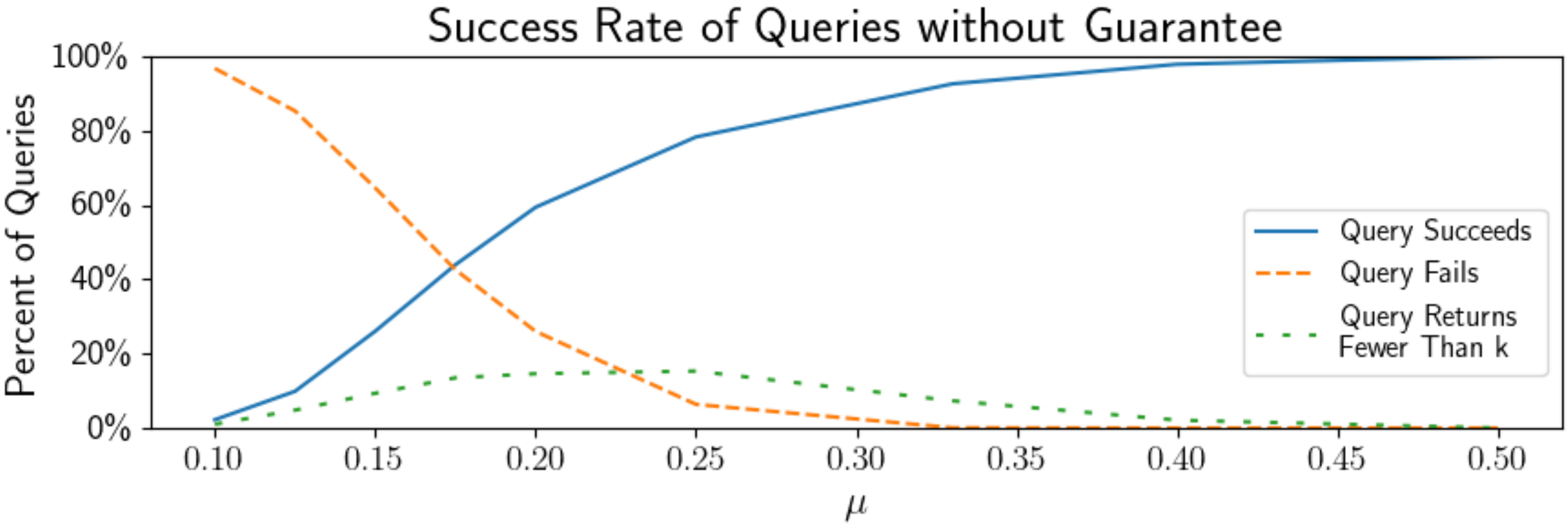
Review: SP Architecture [SIGIR '25]



- At the superblock level, use SB BoundSum and SB Avg to prune

$$SBBoundSum(C_i); = \sum_{t \in Q} q_t \cdot \max_{d \in C_i} w_{t,d}; \quad SBAvg(C_i) = \sum_{t \in Q} q_t \cdot (\frac{1}{|C_i|} \sum_{d \in C_i} w_{t,d})$$

Limitations of Superblock Pruning



Prune more (threshold overestimation rate) Prune less

Goals of Lightweight Superblock Pruning (LSP)

1. Address the superblock overpruning problem

Goals of Lightweight Superblock Pruning (LSP)

1. Address the superblock overpruning problem
2. What superblock heuristic is best?

Goals of Lightweight Superblock Pruning (LSP)

1. Address the superblock overpruning problem
2. What superblock heuristic is best?
3. Compress the index without affecting latency

Goals of Lightweight Superblock Pruning (LSP)

1. Address the superblock overpruning problem
2. What superblock heuristic is best?
3. Compress the index without affecting latency
4. Simplify parameter choices & enable zero-shot approximate retrieval

Goals of Lightweight Superblock Pruning (LSP)

1. Address the superblock overpruning problem
2. What superblock heuristic is best?
3. Compress the index without affecting latency
4. Simplify parameter choices & enable zero-shot approximate retrieval
5. Scale to larger datasets

Goals of Lightweight Superblock Pruning (LSP)

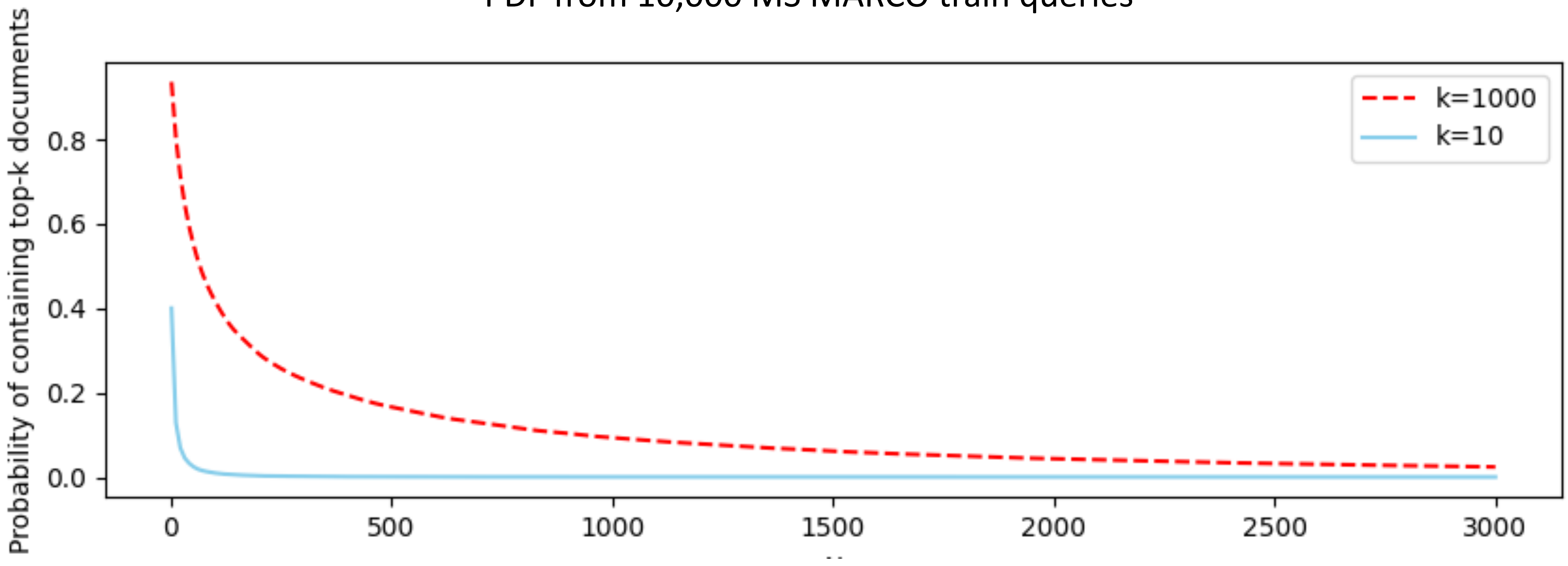
1. Address the superblock overpruning problem
2. What superblock heuristic is best?
3. Compress the index without affecting latency
4. Simplify parameter choices & enable zero-shot approximate retrieval
5. Scale to larger datasets

Key Idea:
Guarantee Searching a Minimum Number of Superblocks

Address Superblock Overpruning

- If we only search the top-gamma SBs, how good are the results?

PDF from 10,000 MS MARCO train queries



MS MARCO has about 1.1M clusters total

UC SANTA BARBARA

Address Superblock Overpruning

Table 1: Confidence $P_\gamma(I)$ based on MS MARCO training data

Guaranteed number of clusters

γ	100	200	300	1000	2000	3000
$k = 10, b \times c = 64$	98.9%	99.6%	99.8%	$\sim$ 100%	$\sim$100%	$\sim$100%
$k = 10, b \times c = 128$	99.0%	99.6%	99.8%	$\sim$100%	$\sim$100%	$\sim$100%
$k = 10, b \times c = 256$	99.0%	99.6%	99.8%	$\sim$100%	$\sim$100%	$\sim$100%
$k = 1,000, b \times c = 64$	56.4%	69.4%	74.6%	89.4%	95.0%	97.1%
$k = 1,000, b \times c = 128$	59.6%	71.4%	77.3%	90.8%	95.8%	97.6%
$k = 1,000, b \times c = 256$	62.2%	73.7%	79.4%	91.9%	96.5%	98.1%

Retrieval Depth (k) &
Cluster size (b*c)

What Superblock Heuristic is Best?

- **LSP/0** — Only search gamma superblocks

What Superblock Heuristic is Best?

- **LSP/0** — Only search gamma superblocks
- **LSP/1** — Use threshold overestimation, guaranteeing at least gamma superblocks are scored

What Superblock Heuristic is Best?

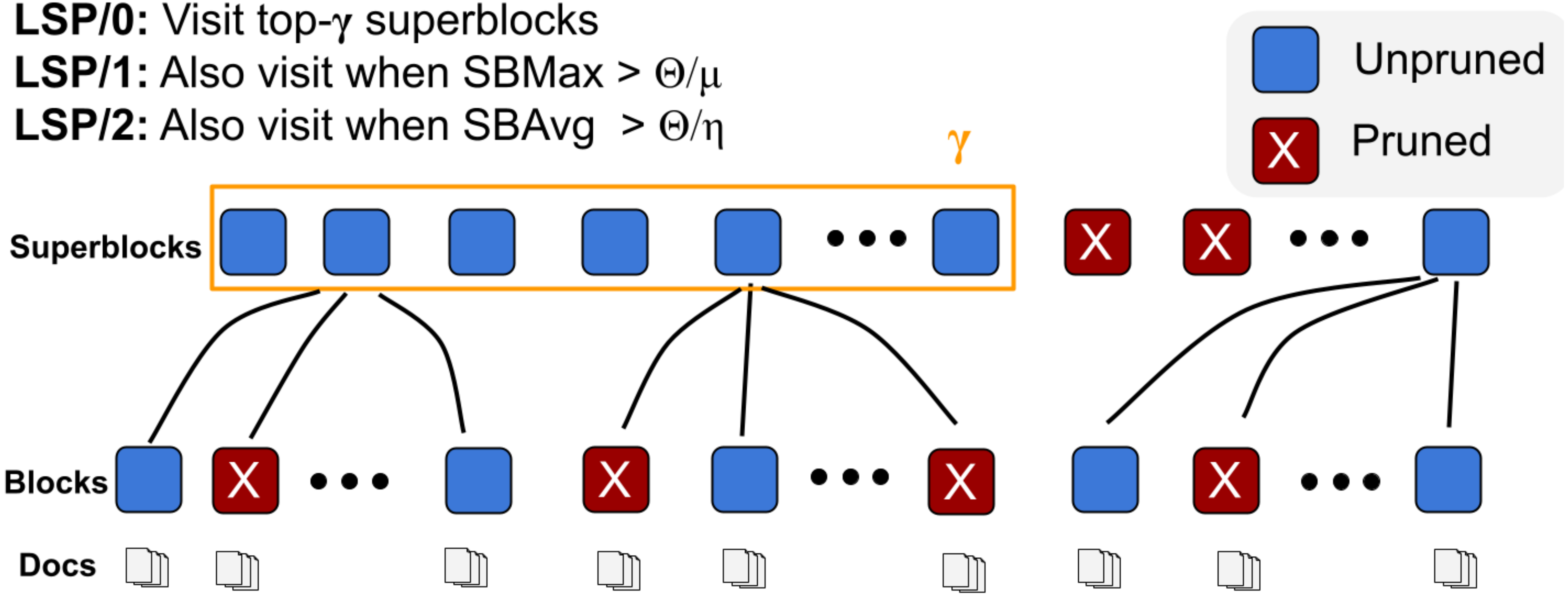
- **LSP/0** — Only search gamma superblocks
- **LSP/1** — Use threshold overestimation, guaranteeing at least gamma superblocks are scored
- **LSP/2** — Same as SP (Max & Avg), but guarantee at least gamma superblocks are scored

What Superblock Heuristic is Best?

- **LSP/0** — Only search gamma superblocks
- **LSP/1** — Use threshold overestimation, guaranteeing at least gamma superblocks are scored
- **LSP/2** — Same as SP (Max & Avg), but guarantee at least gamma superblocks are scored

LSP Architecture

LSP/0: Visit top-γ superblocks
LSP/1: Also visit when SBMax > Θ/μ
LSP/2: Also visit when SBAvg > Θ/η



Evaluation & Results

Evaluation Methodology

- In-Domain: MS MARCO passages (8.8M), Dev set, DL19, DL 20
- OOD: BEIR-13 Datasets
- Scalability: MS MARCO passages v2 (138M), Dev1 set
- Baselines:
  - MaxScore (PISA) [IPM ’95; OSIRRC@SIGIR ’19]
  - BMP [SIGIR ’24]
  - SP [SIGIR ’25]
  - SeismicWave [SIGIR ’24; CIKM ’24]

What Superblock Heuristic is Best?

- In-domain grid search on MS MARCO (SPLADE++)

Recall Preserved	**93%** MRT	**95%** MRT	**97%** MRT	**98%** MRT	**99%** MRT
			k=10		
LSP/0	195	214	275	281	320
LSP/1	191	215	248	285	315
LSP/2	474	477	527	527	716
Seismic-W.	159	179	221	323	402
+*knn*	**79**	**90**	**99**	**115**	**132**
			k=1000		
LSP/0	571	571	656	**765**	1030
LSP/1	**516**	**562**	**647**	780	**961**
LSP/2	846	937	1173	1280	1710
Seismic-W.	713	876	955	1080	1800
+*knn*	2330	2330	2330	2360	2440

MRT in ***microseconds***

What Superblock Heuristic is Best?

- In-domain grid search on MS MARCO (SPLADE++)

overfit to dev queries => best-case performance

Recall Preserved	93% MRT	95% MRT	97% MRT	98% MRT	99% MRT
			k=10		
LSP/0	195	214	275	281	320
LSP/1	191	215	248	285	315
LSP/2	474	477	527	527	716
Seismic-W.	159	179	221	323	402
+*knn*	**79**	**90**	**99**	**115**	**132**
			k=1000		
LSP/0	571	571	656	**765**	1030
LSP/1	**516**	**562**	**647**	780	**961**
LSP/2	846	937	1173	1280	1710
Seismic-W.	713	876	955	1080	1800
+*knn*	2330	2330	2330	2360	2440

MRT in ***microseconds***

What Superblock Heuristic is Best?

- In-domain grid search on MS MARCO (SPLADE++)

Recall Preserved	**93%** MRT	**95%** MRT	**97%** MRT	**98%** MRT	**99%** MRT
			k=10		
LSP/0	195	214	275	281	320
LSP/1	191	215	248	285	315
LSP/2	474	477	527	527	716
Seismic-W.	159	179	221	323	402
+*knn*	**79**	**90**	**99**	**115**	**132**
			k=1000		
LSP/0	571	571	656	**765**	1030
LSP/1	**516**	**562**	**647**	780	**961**
LSP/2	846	937	1173	1280	1710
Seismic-W.	713	876	955	1080	1800
+*knn*	2330	2330	2330	2360	2440

LSP/1 is *slightly* better than LSP/0

What Superblock Heuristic is Best?

- In-domain grid search on MS MARCO (SPLADE++)

LSP/0 is ~2x slower than SeismicWave for k=10

Recall Preserved	93% MRT	95% MRT	97% MRT	98% MRT	99% MRT
			k=10		
LSP/0	195	214	275	281	320
LSP/1	191	215	248	285	315
LSP/2	474	477	527	527	716
Seismic-W.	159	179	221	323	402
+*knn*	**79**	**90**	**99**	**115**	**132**
			k=1000		
LSP/0	571	571	656	**765**	1030
LSP/1	**516**	**562**	**647**	780	**961**
LSP/2	846	937	1173	1280	1710
Seismic-W.	713	876	955	1080	1800
+*knn*	2330	2330	2330	2360	2440

What Superblock Heuristic is Best?

- In-domain grid search on MS MARCO (SPLADE++)

LSP/0 is ~2x slower than SeismicWave for k=10

Recall Preserved	**93%** MRT	**95%** MRT	**97%** MRT	**98%** MRT	**99%** MRT
			k=10		
LSP/0	195	214	275	281	320
LSP/1	191	215	248	285	315
LSP/2	474	477	527	527	716
Seismic-W.	159	179	221	323	402
+*knn*	**79**	**90**	**99**	**115**	**132**
			k=1000		
LSP/0	571	571	656	**765**	1030
LSP/1	**516**	**562**	**647**	780	**961**
LSP/2	846	937	1173	1280	1710
Seismic-W.	713	876	955	1080	1800
+*knn*	2330	2330	2330	2360	2440

but is ~30%-2x faster for k=1000

Zero-Shot Approximate Retrieval

- Goal: test *realistic* performance

Zero-Shot Approximate Retrieval

- Goal: test *realistic* performance
- Can we fix approximate retrieval parameters using in-domain data and still achieve good results OOD?

Zero-Shot Approximate Retrieval

- Goal: test *realistic* performance
- Can we fix approximate retrieval parameters using in-domain data and still achieve good results OOD?
- How to test?
  - Fix configuration based on train data / in-domain grid search

Zero-Shot Approximate Retrieval

- Goal: test *realistic* performance
- Can we fix approximate retrieval parameters using in-domain data and still achieve good results OOD?
- How to test?
  - Fix configuration based on train data / in-domain grid search
- LSP/0 for k=10:
  - gamma=250, query pruning rate=0.33, ~500K clusters

    (0.05% of clusters)

OOD Results on BEIR-13

Fastest on 10/13 datasets

Dataset	Corpus Size	Safe nDCG	BMP nDCG	BMP MRT	BMP GB	SP nDCG	SP MRT	SP GB	SeismicWave nDCG	SeismicWave MRT	SeismicWave GB	LSP/0 nDCG	LSP/0 MRT	LSP/0 GB
Arguana	8.7K	**52.0**	51.1	0.414	0.070	48.7	0.704	0.096	49.5	**0.089**	0.188	50.3	0.222	**0.023**
C-FEVER	5.4M	23.0	24.3	9.45	15.3	**24.5**	10.5	58.3	24.4	**0.453**	7.47	23.6	0.623	**6.14**
DBPedia	4.6M	**43.7**	43.3	6.34	19.8	43.6	2.02	48.3	43.2	0.381	5.63	42.9	**0.330**	**4.79**
FEVER	5.4M	78.8	79.9	7.38	15.2	79.5	4.36	58.3	**80.0**	0.501	7.47	79.2	**0.461**	**6.13**
FiQA	57K	34.7	35.5	0.639	0.405	35.5	0.598	0.666	**35.6**	0.841	0.713	35.1	**0.186**	**0.093**
Hotpot	5.2M	**68.7**	68.6	9.62	21.4	68.5	3.73	54.0	68.3	0.442	5.73	67.1	**0.422**	**5.04**
NFCorpus	3.6K	34.7	35.3	0.104	0.030	34.9	0.123	0.035	**35.6**	**0.045**	0.080	35.2	0.084	**0.011**
NQ	2.7M	53.8	53.8	4.61	14.8	**53.9**	2.11	30.3	53.8	0.363	7.04	53.4	**0.294**	**3.57**
Quora	520K	83.4	**83.5**	0.945	1.27	83.2	0.521	4.51	83.4	0.305	0.693	**83.5**	**0.193**	**0.294**
SciDocs	25K	15.9	15.8	0.444	0.213	15.9	0.499	0.306	15.8	0.239	0.413	**16.2**	**0.175**	**0.052**
SciFact	5K	70.4	70.7	0.350	0.046	70.5	0.469	0.055	70.8	**0.090**	0.115	70.8	0.129	**0.014**
T-COVID	171K	72.7	72.8	1.57	1.10	72.1	1.18	1.92	72.8	1.94	1.98	**72.9**	**0.241**	**0.227**
Touche	380K	24.7	**27.3**	1.77	2.61	**27.3**	0.793	4.56	27.2	0.861	4.32	**27.3**	**0.215**	**0.575**
Average	1.9M	50.5	**50.9**	3.22	7.10	50.6	2.12	20.1	50.8	0.504	3.22	50.6	**0.275**	**2.07**
vs. LSP/0	–	-0.7%	+0.6%	9.3x	3.7x	+0.2%	5.7x	8.0x	+0.4%	2.0x	5.0x	–	–	–
k = 100	1.9M	50.6	**51.0**	6.13	7.10	50.7	3.47	20.1	50.9	1.06	3.22	50.9	**0.490**	**2.07**
k = 1000	1.9M	50.6	50.9	13.9	7.10	50.7	8.75	20.1	50.9	2.52	3.22	**51.0**	**1.31**	**2.07**

OOD Results on BEIR-13

Dataset	Corpus Size	Safe	BMP			SP			SeismicWave			LSP/0		
		nDCG	nDCG	MRT	GB	nDCG	MRT	GB	nDCG	MRT	GB	nDCG	MRT	GB
Arguana	8.7K	**52.0**	51.1	0.414	0.070	48.7	0.704	0.096	49.5	**0.089**	0.188	50.3	0.222	**0.023**
C-FEVER	5.4M	23.0	24.3	9.45	15.3	**24.5**	10.5	58.3	24.4	**0.453**	7.47	23.6	0.623	**6.14**
DBPedia	4.6M	**43.7**	43.3	6.34	19.8	43.6	2.02	48.3	43.2	0.381	5.63	42.9	**0.330**	**4.79**
FEVER	5.4M	78.8	79.9	7.38	15.2	79.5	4.36	58.3	**80.0**	0.501	7.47	79.2	**0.461**	**6.13**
FiQA	57K	34.7	35.5	0.639	0.405	35.5	0.598	0.666	**35.6**	0.841	0.713	35.1	**0.186**	**0.093**
Hotpot	5.2M	**68.7**	68.6	9.62	21.4	68.5	3.73	54.0	68.3	0.442	5.73	67.1	**0.422**	**5.04**
NFCorpus	3.6K	34.7	35.3	0.104	0.030	34.9	0.123	0.035	**35.6**	**0.045**	0.080	35.2	0.084	**0.011**
NQ	2.7M	53.8	53.8	4.61	14.8	**53.9**	2.11	30.3	53.8	0.363	7.04	53.4	**0.294**	**3.57**
Quora	520K	83.4	**83.5**	0.945	1.27	83.2	0.521	4.51	83.4	0.305	0.693	**83.5**	**0.193**	**0.294**
SciDocs	25K	15.9	15.8	0.444	0.213	15.9	0.499	0.306	15.8	0.239	0.413	**16.2**	**0.175**	**0.052**
SciFact	5K	70.4	70.7	0.350	0.046	70.5	0.469	0.055	70.8	**0.090**	0.115	70.8	0.129	**0.014**
T-COVID	171K	72.7	72.8	1.57	1.10	72.1	1.18	1.92	72.8	1.94	1.98	**72.9**	**0.241**	**0.227**
Touche	380K	24.7	**27.3**	1.77	2.61	**27.3**	0.793	4.56	27.2	0.861	4.32	**27.3**	**0.215**	**0.575**
Average	1.9M	50.5	**50.9**	3.22	7.10	50.6	2.12	20.1	50.8	0.504	3.22	50.6	**0.275**	**2.07**
vs. LSP/0	–	-0.7%	+0.6%	9.3x	3.7x	+0.2%	5.7x	8.0x	+0.4%	2.0x	5.0x	–	–	–
$k = 100$	1.9M	50.6	**51.0**	6.13	7.10	50.7	3.47	20.1	50.9	1.06	3.22	50.9	**0.490**	**2.07**
$k = 1000$	1.9M	50.6	50.9	13.9	7.10	50.7	8.75	20.1	50.9	2.52	3.22	**51.0**	**1.31**	**2.07**

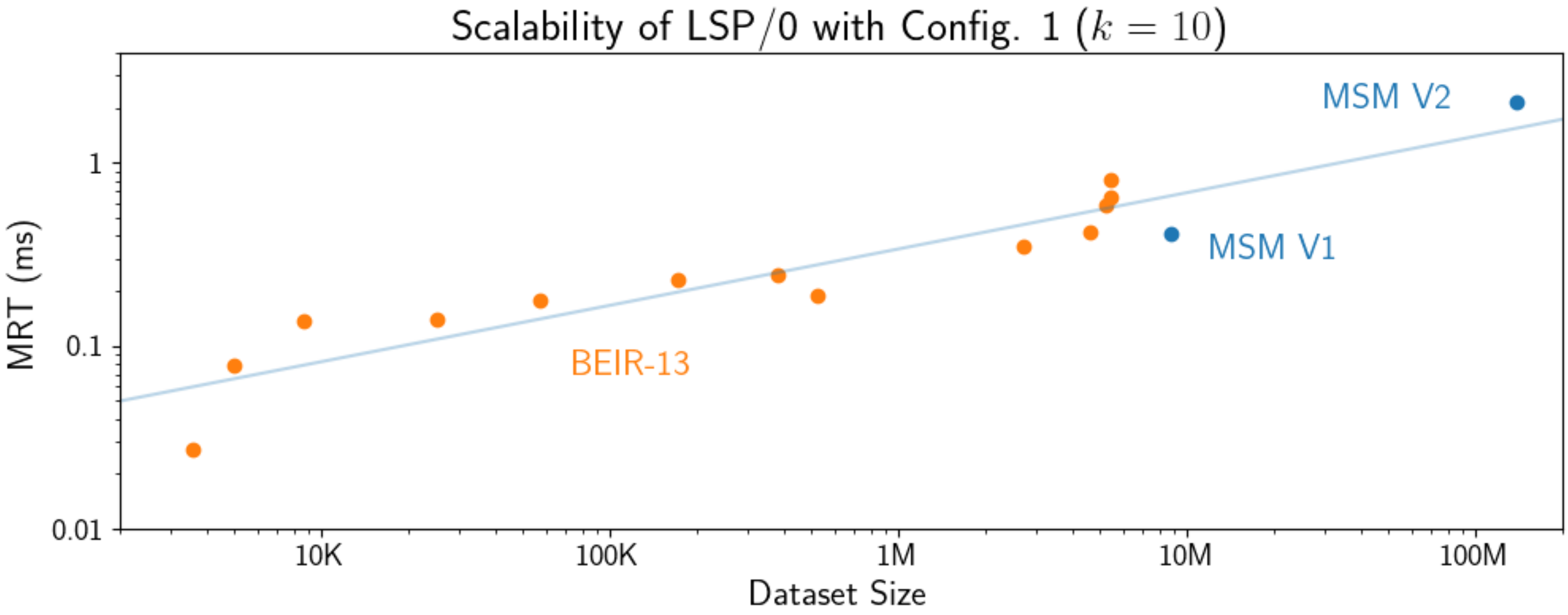
5.7x faster than SP,
8x less storage

OOD Results on BEIR-13

Dataset	Corpus Size	Safe	BMP			SP			SeismicWave			LSP/0		
		nDCG	nDCG	MRT	GB	nDCG	MRT	GB	nDCG	MRT	GB	nDCG	MRT	GB
Arguana	8.7K	**52.0**	51.1	0.414	0.070	48.7	0.704	0.096	49.5	**0.089**	0.188	50.3	0.222	**0.023**
C-FEVER	5.4M	23.0	24.3	9.45	15.3	**24.5**	10.5	58.3	24.4	**0.453**	7.47	23.6	0.623	**6.14**
DBPedia	4.6M	**43.7**	43.3	6.34	19.8	43.6	2.02	48.3	43.2	0.381	5.63	42.9	**0.330**	**4.79**
FEVER	5.4M	78.8	79.9	7.38	15.2	79.5	4.36	58.3	**80.0**	0.501	7.47	79.2	**0.461**	**6.13**
FiQA	57K	34.7	35.5	0.639	0.405	35.5	0.598	0.666	**35.6**	0.841	0.713	35.1	**0.186**	**0.093**
Hotpot	5.2M	**68.7**	68.6	9.62	21.4	68.5	3.73	54.0	68.3	0.442	5.73	67.1	**0.422**	**5.04**
NFCorpus	3.6K	34.7	35.3	0.104	0.030	34.9	0.123	0.035	**35.6**	**0.045**	0.080	35.2	0.084	**0.011**
NQ	2.7M	53.8	53.8	4.61	14.8	**53.9**	2.11	30.3	53.8	0.363	7.04	53.4	**0.294**	**3.57**
Quora	520K	83.4	**83.5**	0.945	1.27	83.2	0.521	4.51	83.4	0.305	0.693	**83.5**	**0.193**	**0.294**
SciDocs	25K	15.9	15.8	0.444	0.213	15.9	0.499	0.306	15.8	0.239	0.413	**16.2**	**0.175**	**0.052**
SciFact	5K	70.4	70.7	0.350	0.046	70.5	0.469	0.055	70.8	**0.090**	0.115	70.8	0.129	**0.014**
T-COVID	171K	72.7	72.8	1.57	1.10	72.1	1.18	1.92	72.8	1.94	1.98	**72.9**	**0.241**	**0.227**
Touche	380K	24.7	**27.3**	1.77	2.61	**27.3**	0.793	4.56	27.2	0.861	4.32	**27.3**	**0.215**	**0.575**
Average	1.9M	50.5	**50.9**	3.22	7.10	50.6	2.12	20.1	50.8	0.504	3.22	50.6	**0.275**	**2.07**
vs. LSP/0	–	-0.7%	+0.6%	9.3x	3.7x	+0.2%	5.7x	8.0x	+0.4%	2.0x	5.0x	–	–	–
k = 100	1.9M	50.6	**51.0**	6.13	7.10	50.7	3.47	20.1	50.9	1.06	3.22	50.9	**0.490**	**2.07**
k = 1000	1.9M	50.6	50.9	13.9	7.10	50.7	8.75	20.1	50.9	2.52	3.22	**51.0**	**1.31**	**2.07**

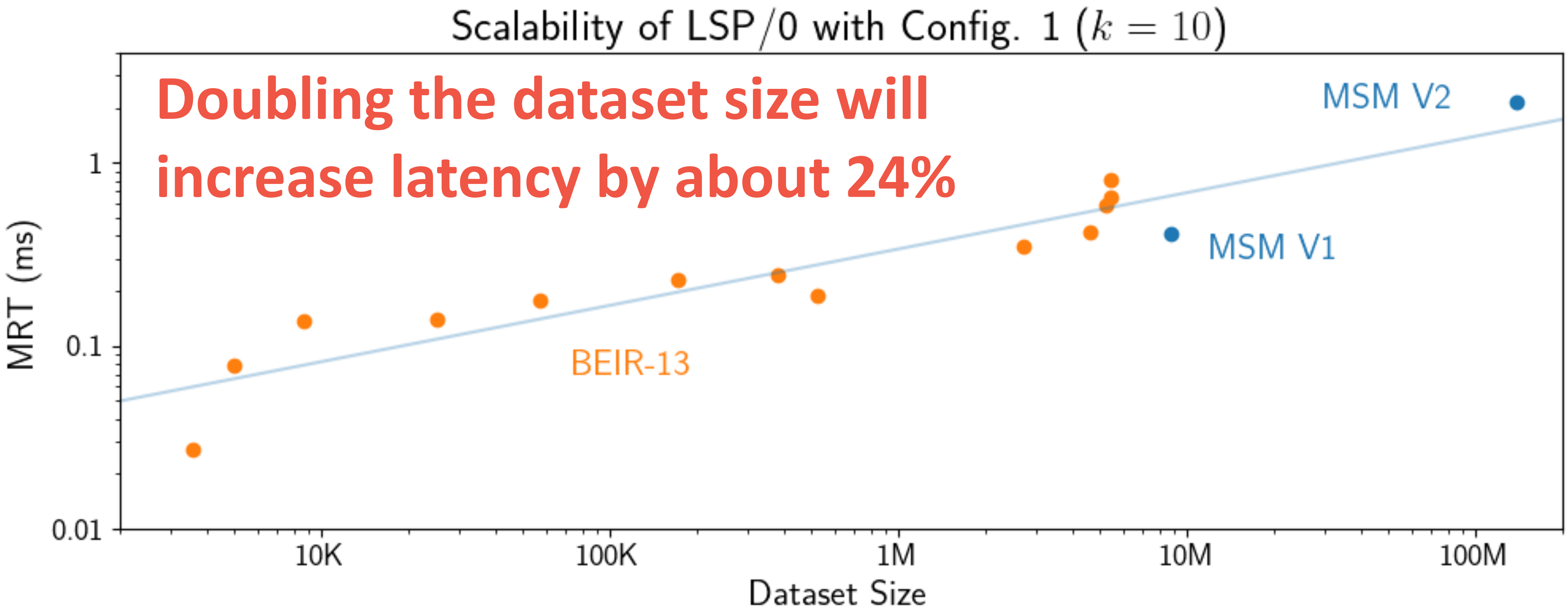
Trend holds for
k=100 and k=1000

Scalability of LSP with Fixed Configuration



UC SANTA BARBARA

Scalability of LSP with Fixed Configuration



UC SANTA BARBARA

Conclusions

- Simple parameter configurations with strong performance across OOD datasets and models

Conclusions

- Simple parameter configurations with strong performance across OOD datasets and models
- When a grid search is possible, fastest on k=1000, but slower at k=10 than Seismic

Conclusions

- Simple parameter configurations with strong performance across OOD datasets and models
- When a grid search is possible, fastest on k=1000, but slower at k=10 than Seismic
- A simple guarantee of top clusters outperforms more complex pruning heuristics (LSP/1 & LSP/2); backed by a probabilistic relevance model

This work is supported by the
SIGIR Student Travel Grant

UC SANTA BARBARA