Fast, Approximate Vector Queries on Very Large Unstructured Datasets

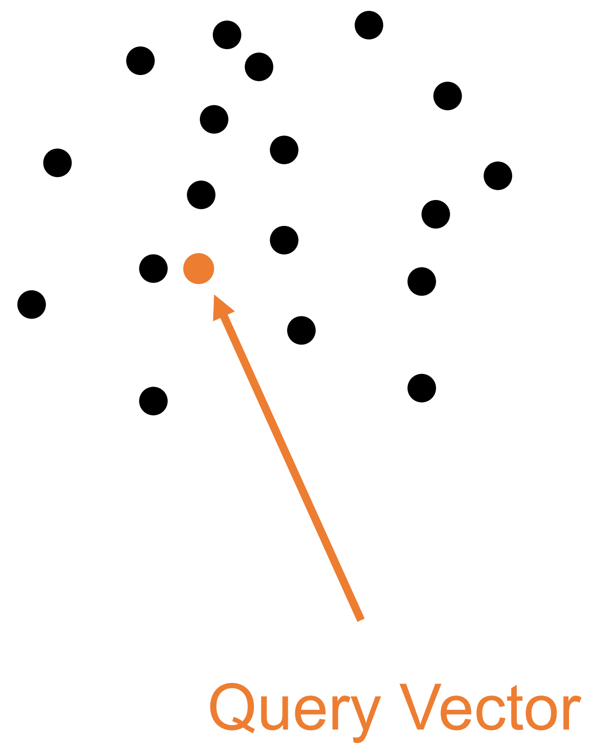
Zili Zhang, Chao Jin, Linpeng Tang, Xuanzhe Liu, Xin Jin

北京大學 PEKING UNIVERSITY

MOQI 墨奇科技

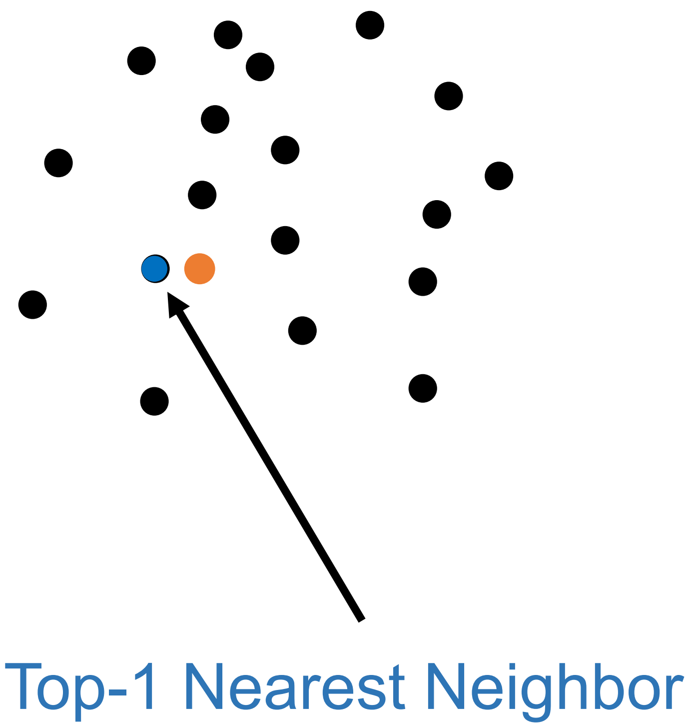
Vector Search

- What is Vector Search ?



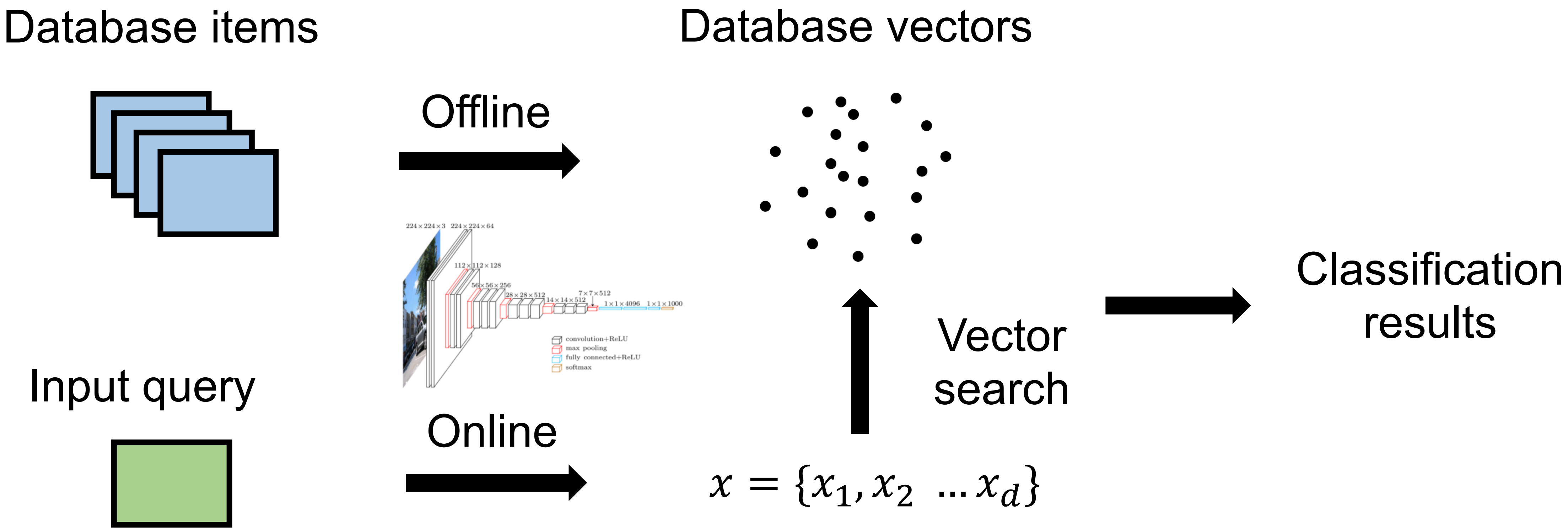
Vector Search

- What is Vector Search ?



Vector Search

➢ Vector Search in **Real-World Applications**



Vector Search

Exact K-NN search

- ➢ **high query latency**

Deep Learning model

- ➢ approximate result

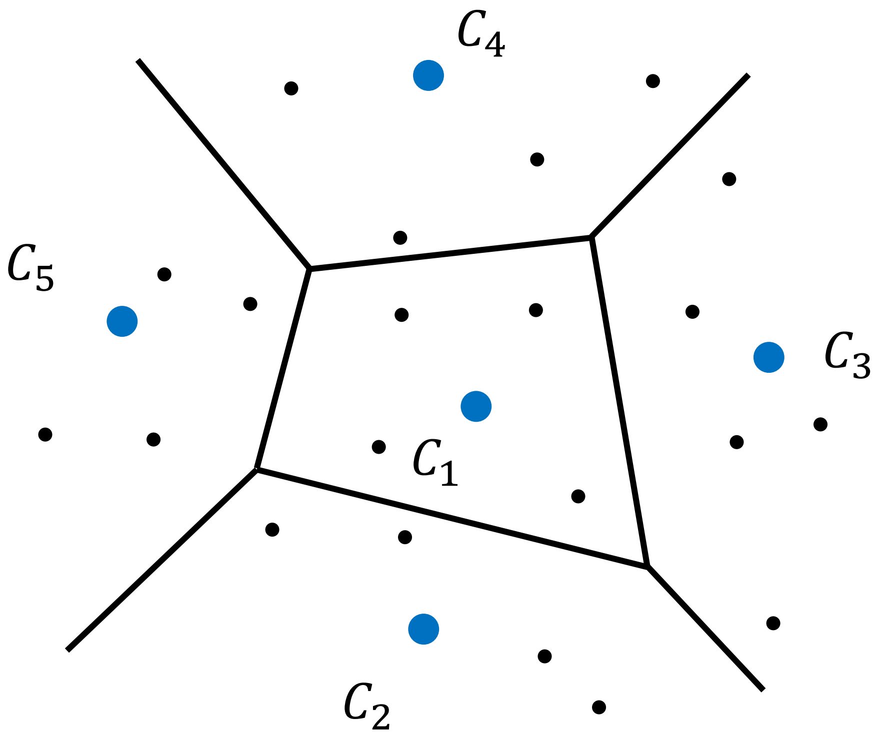
⬇

Approximate Vector Search

Approximate Vector Search

Inverted File Index (IVF)

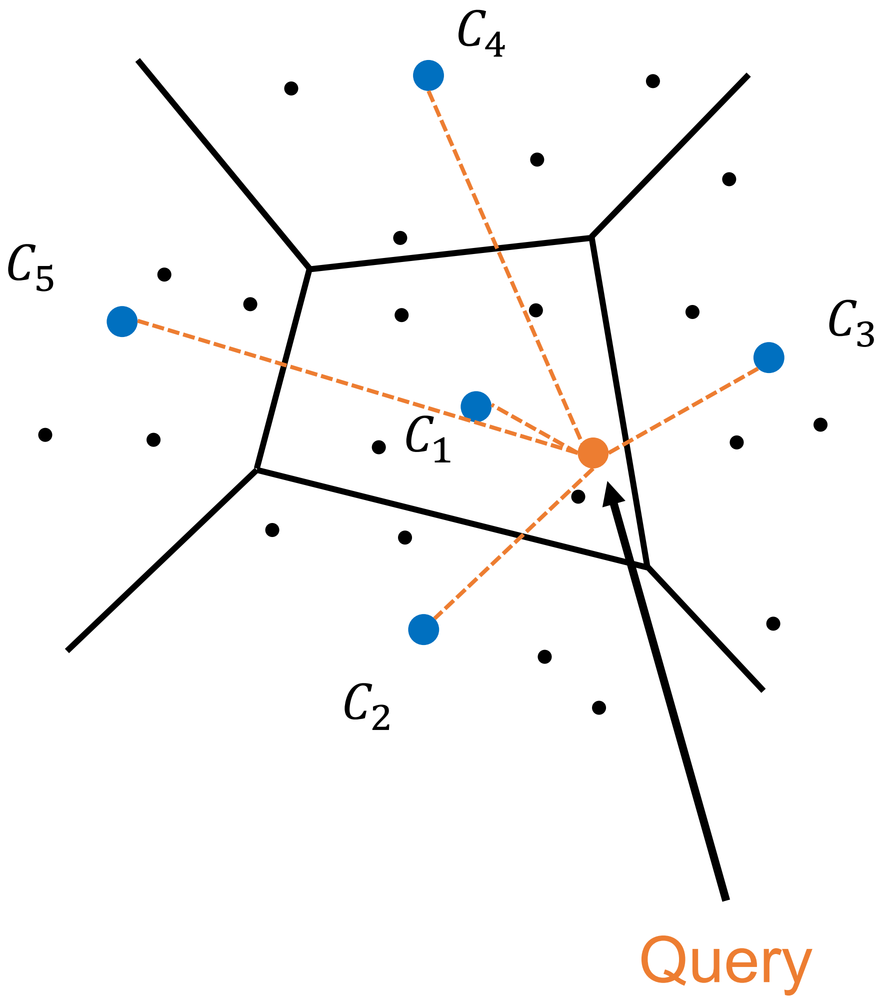
- partition the dataset into several clusters



Approximate Vector Search

Inverted File Index (IVF)

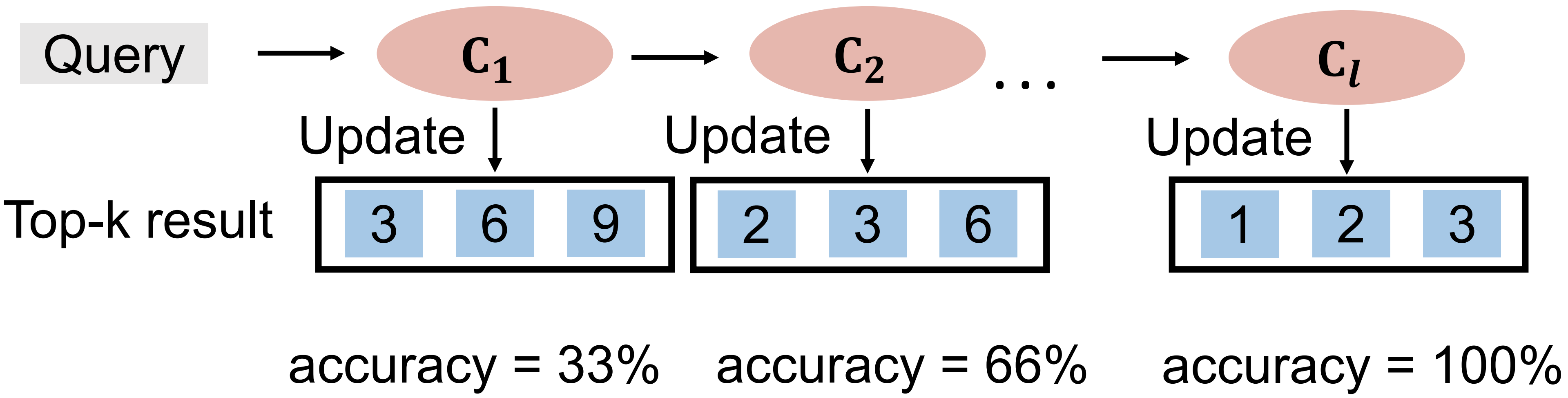
➢ search in the **top-n** clusters



Approximate Vector Search

Inverted File Index

- **top-n** decides the query latency and accuracy



Bounded Performance

Approximate algorithm

(Top-n)

Bounded Performance

- Time bound
- Error bound

Specify a Time/Error bound

Bounded Performance

- Time bound
- Error bound

Bounded Performance

Auncel

- The first distributed vector search engine that provides bounded performance

Bounded Performance : Error Bound

Faiss

- ➢ sample some queries to process vector search
- ➢ map the error bound to the corresponding top-n

Key: error bound		**Value**: Top-n
1%	→	256
5%	→	128
10%	→	64
15%	→	32
20%	→	16
30%	→	4

Bounded Performance : Error Bound

Faiss limitations

- ➢ different queries require different values of **Top-n**
- ➢ the worst case dominates the performance

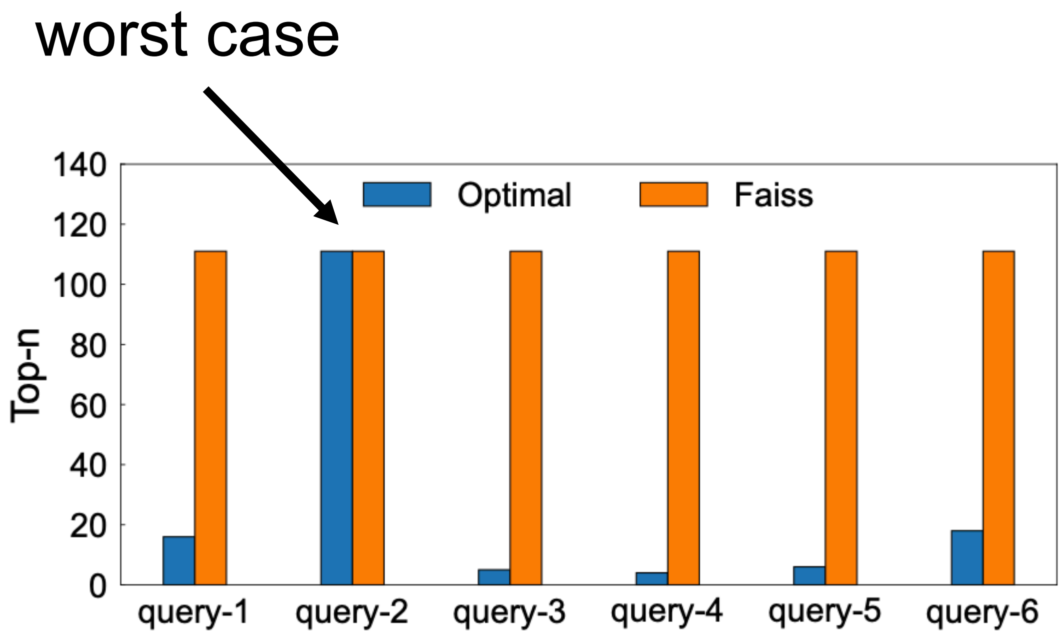


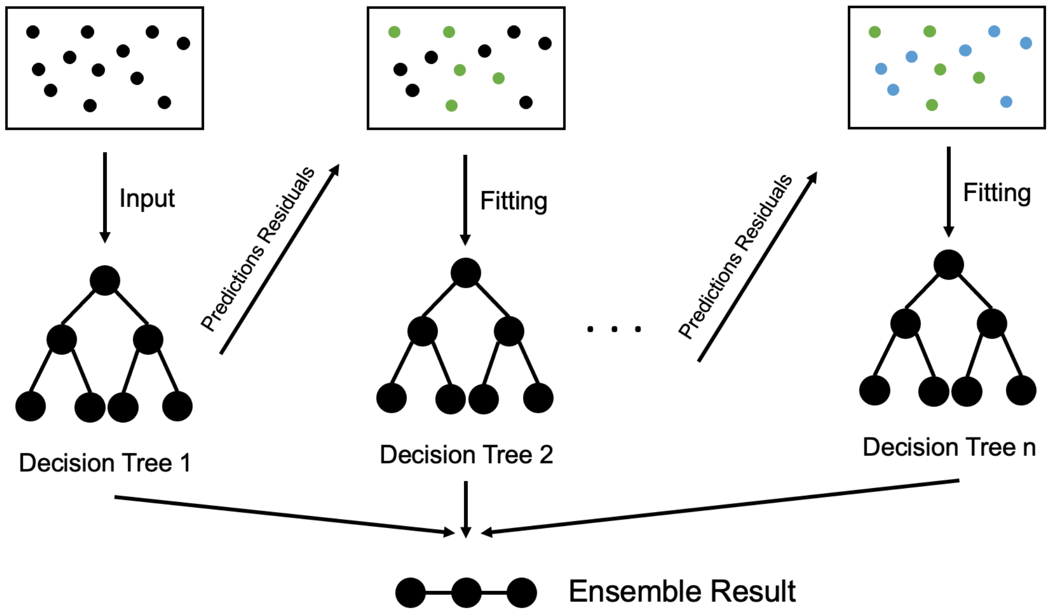
Figure 2: Redundant computation in Faiss.

query-agnostic

Bounded Performance : Error Bound

LAET

- ➢ use a gradient boosting decision tree to predict **top-n** for different queries
- ➢ the model is in-accurate and includes a multiplier to guarantee the bound



Bounded Performance : Error Bound

worst case multiplier

LAET Limitations

- the inaccurate model needs a very large multiplier
- including a complex model introduces large overhead

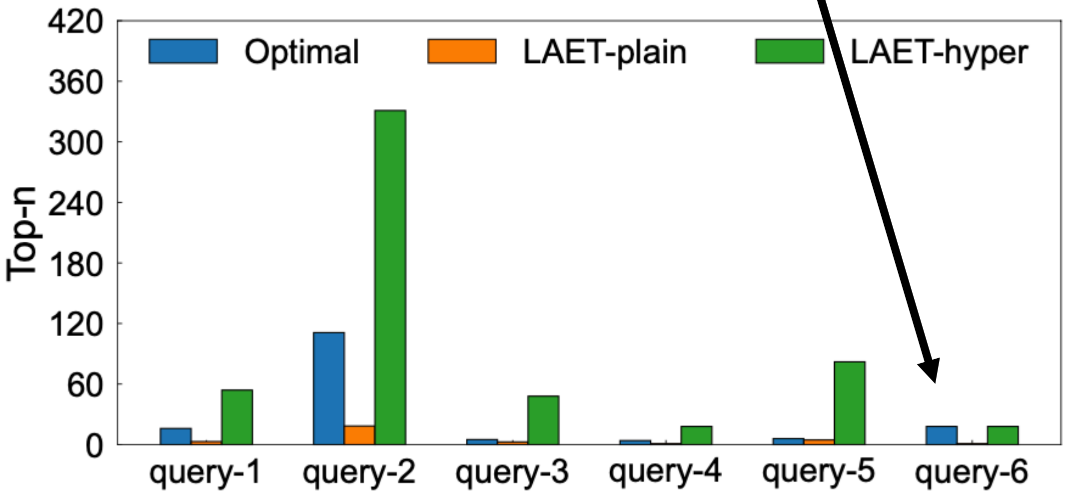


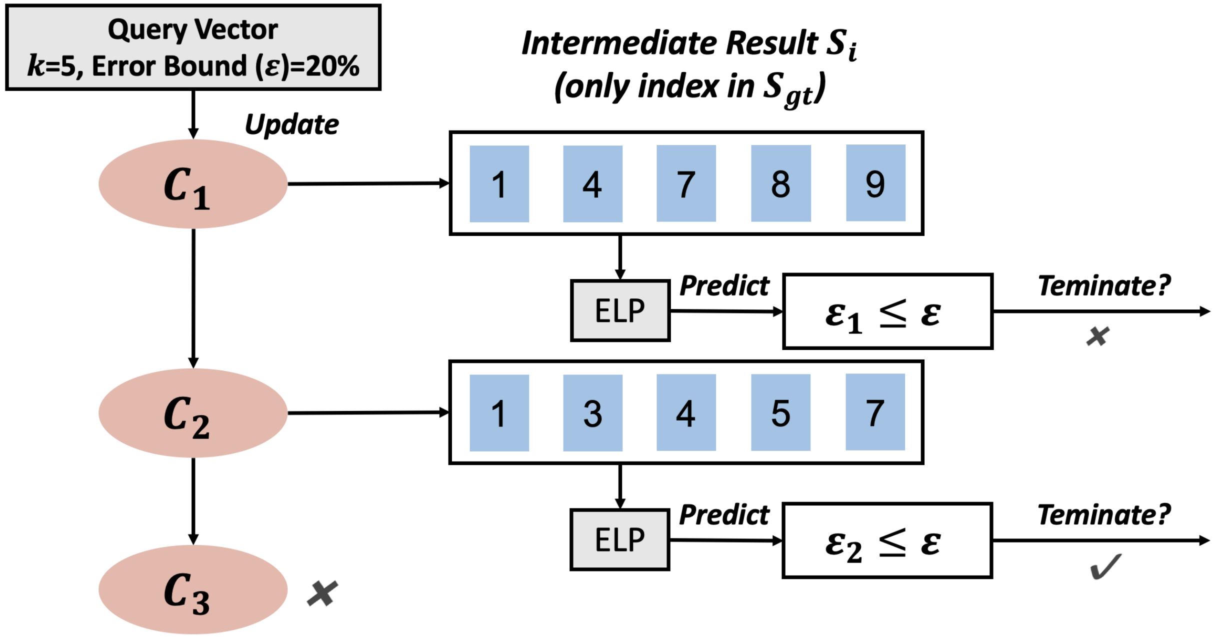
Figure 3: Redundant computation in LAET.

black-box fitting

Bounded Performance : Error Bound

Auncel

- ➢ terminates the query if the error bound is satisfied
- ➢ leverages high-dimensional geometry to profile the error



Bounded Performance : Error Bound

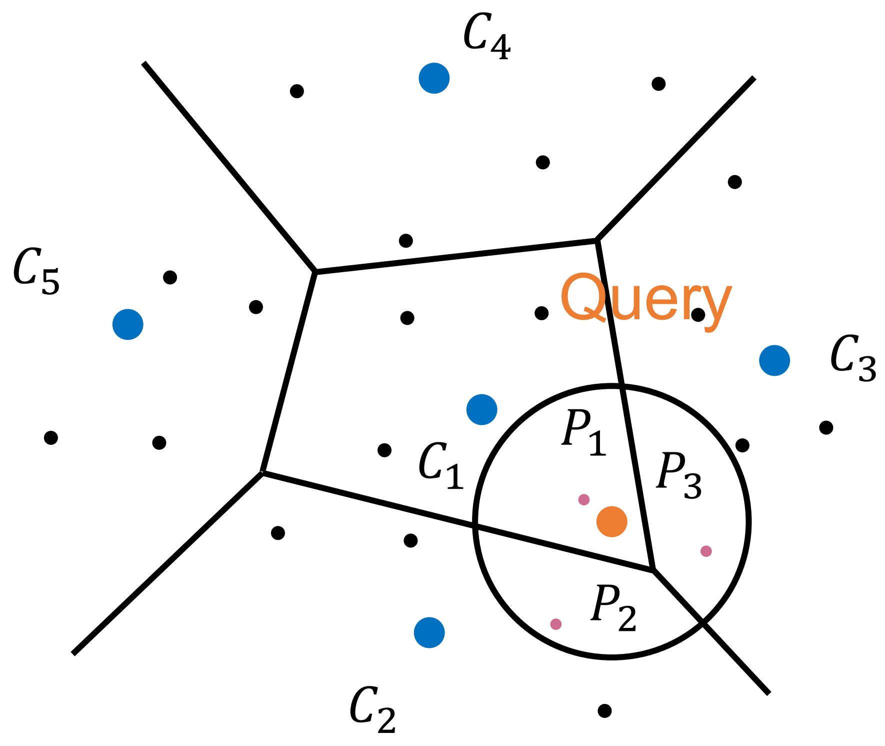
Auncel Profile

- the error is calculated as

$$1 - \frac{\mathrm{N}(P_1)}{top\text{–}k}$$

after processing cluster-1

- $\mathrm{N}(P_i) \approx \mathrm{V}(P_i)$



Bounded Performance : Time Bound

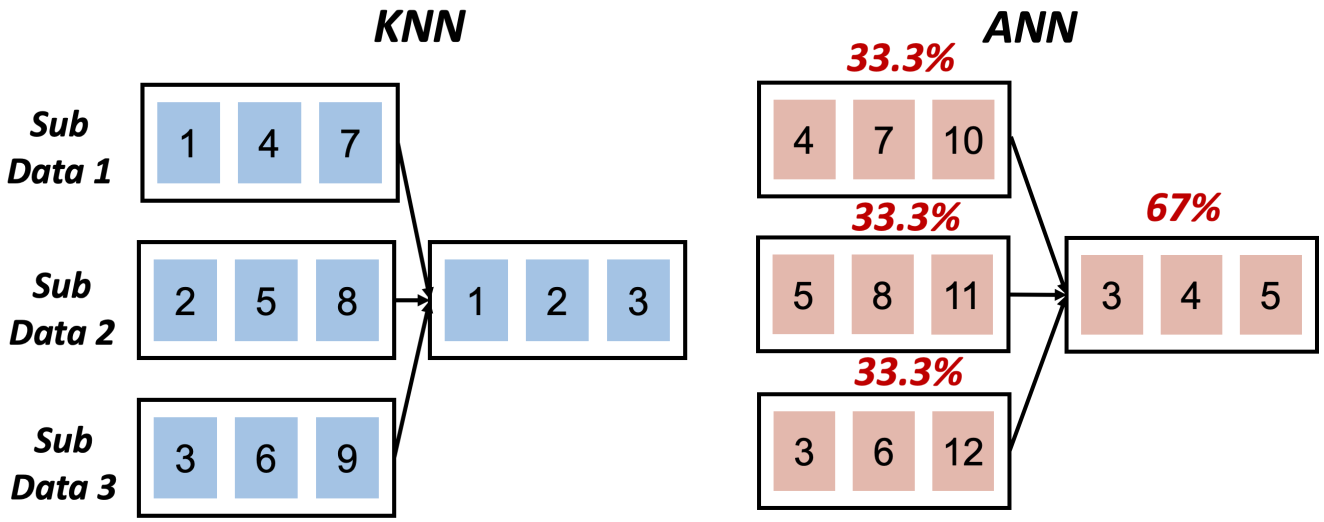
Auncel

- error decreases over time
- terminate the query when $t_{used} + t_{next\ cluster} \geq time\ budget$
- $t_{next\ cluster}$ is profiler from search history

Bounded Performance : Distributed Settings

Error Amplification

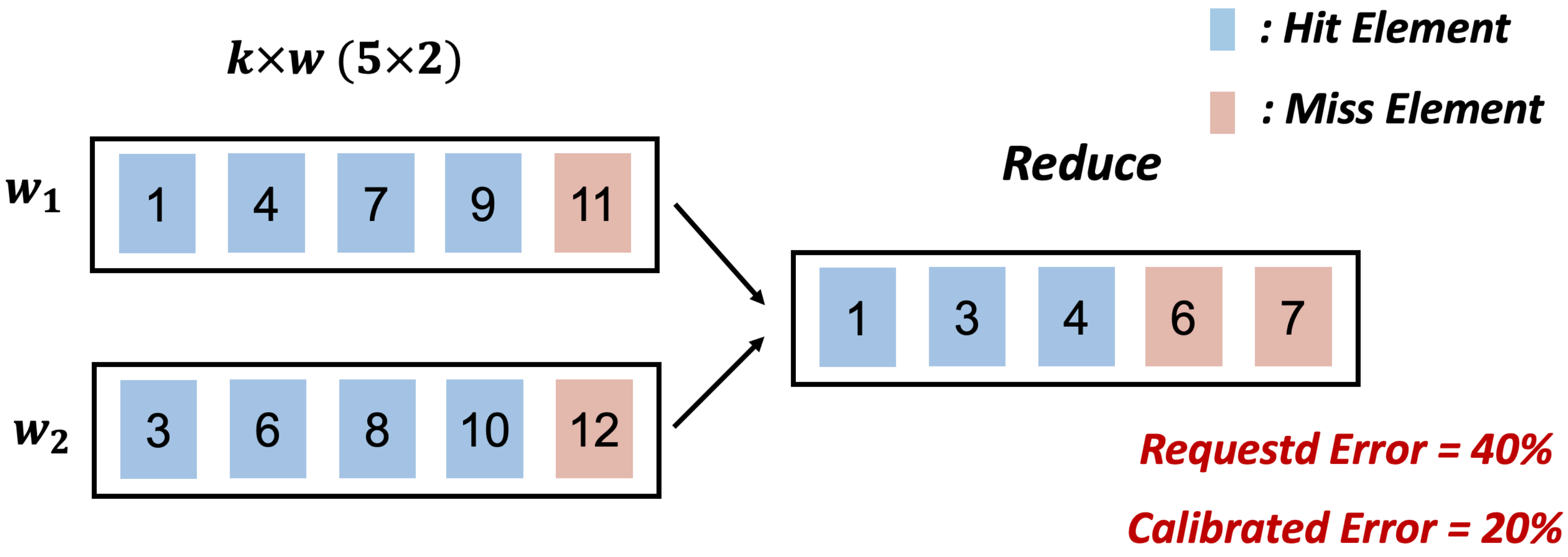
- sharding the dataset into a number of nodes
- the error is amplified and violates the given error bound



Bounded Performance : Distributed Settings

Auncel solution: error calibration

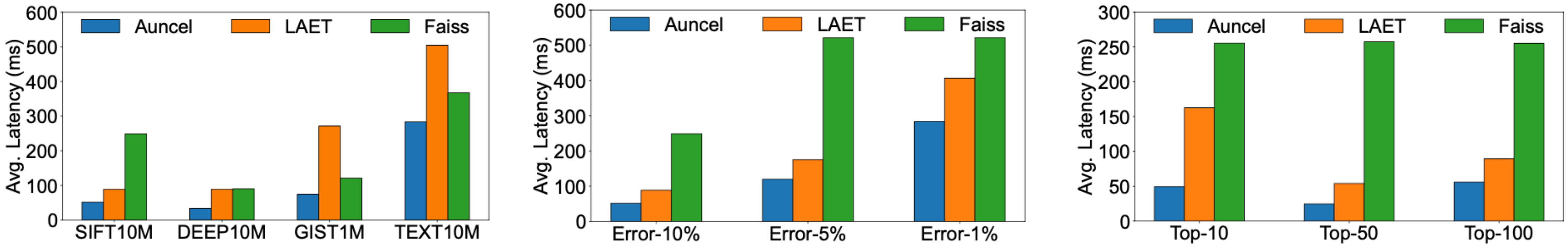
- reduce the error by $\frac{1}{top\text{-}k}$ each time until the global error bound is guaranteed



Evaluation: Set up

- Implementation
  - ~3000 LoC C++
  - Faiss
- Testbed
  - AWS c5.4xlarge (single node)
  - Four c5.metal (distributed settings)
- Datasets
  - Images: Sift, Gist, Deep (with one billion of items)
  - Text: Text-to-Images

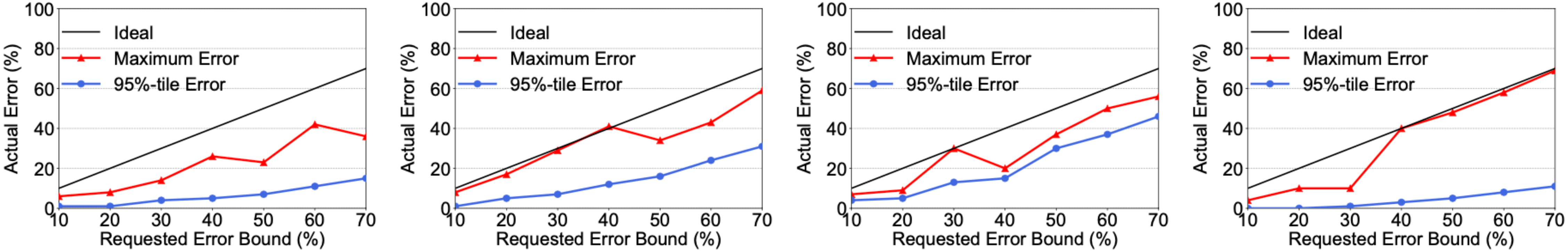
Evaluation: End-to-end Latency



Lower Latency

- ~3× lower query latency on average than baseline systems
- Outperform baselines under different datasets, error bounds and top-k values

Evaluation: Effectiveness

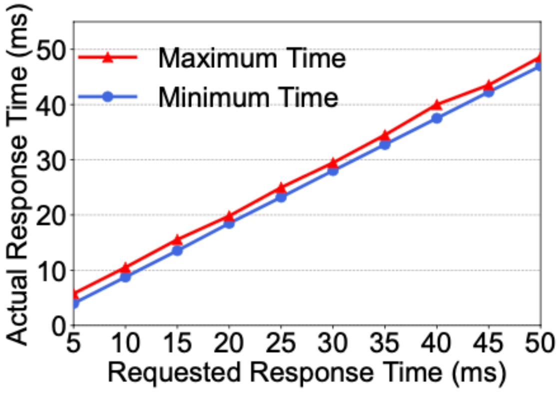


(a) SIFT10M. (b) DEEP10M. (c) GIST1M. (d) TEXT10M.

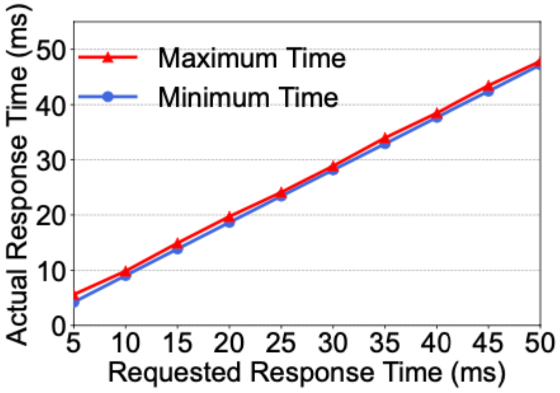
Effectiveness

- ➢ Adapts to different error bounds

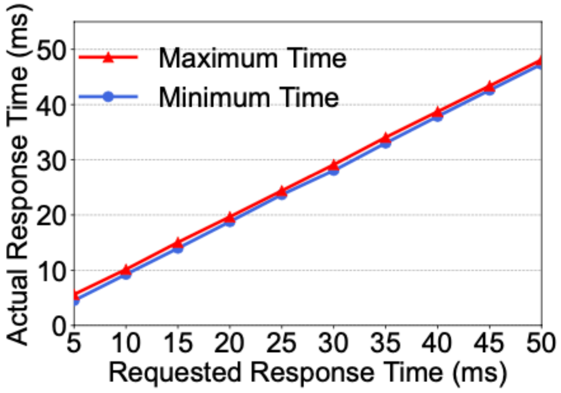
Evaluation: Effectiveness



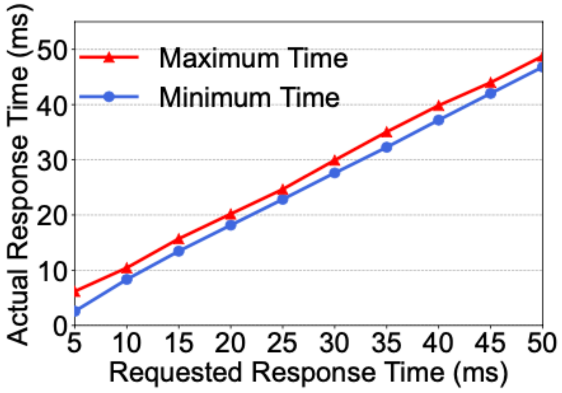
(a) SIFT10M.



(b) DEEP10M.



(c) GIST1M.

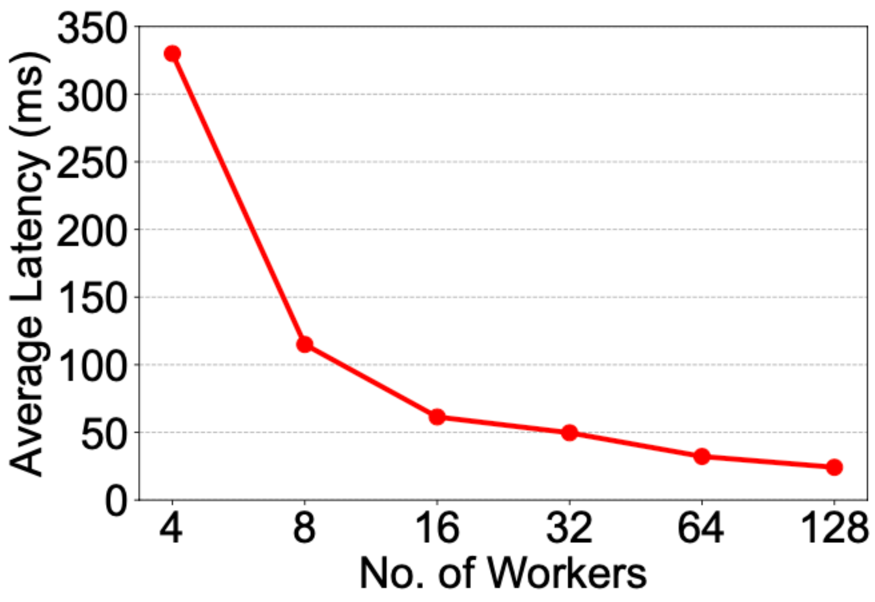


(d) TEXT10M.

Effectiveness

- Adapts to different time bounds

Evaluation: Scalability



Scale Ideally

- latency is reduced by half when the number of worker is doubled

Evaluation: More experiments

- Validation of the mathematical formulation
- Validation of local unformal distribution
- Runtime profile overhead
- System building time
- …

Conclusion

- Auncel: a fast, approximate vector query engine on very large unstructured datasets
  - propose white box and query aware error-latency-profile to guarantee bounded performance
  - apply probability theory to calibrate error bounds and scale to multiple workers ideally

zzlcs@pku.edu.cn

Thanks!