ConANN: Conformal Approximate Nearest Neighbor Search

Sonia Horchidan
KTH Royal Institute of Technology
Stockholm, Sweden
sfhor@kth.se

Fabian Zeiher
KTH Royal Institute of Technology
Stockholm, Sweden
zeiher@kth.se

Henrik Boström
KTH Royal Institute of Technology
Stockholm, Sweden
bostromh@kth.se

Paris Carbone
KTH Royal Institute of Technology
Stockholm, Sweden
parisc@kth.se

ABSTRACT

Approximate Nearest Neighbor (ANN) search is widely used in applications such as recommendation systems, search engines, and natural language processing. Indexing techniques like the Inverted File (IVF) offer efficiency at the cost of accuracy, yet lack formal mechanisms to quantify or control approximation error. Existing approaches that attempt to provide such guarantees typically rely on restrictive assumptions about underlying data distributions, which limits their generalizability. We introduce ConANN, the first framework to provide formal, distribution-free error guarantees for IVF-based ANN search by leveraging recent advances in Conformal Risk Control. Empirical evaluation across five standard benchmarks demonstrates that ConANN: (1) tightly controls approximation error, achieving a worst-case False Negative Rate deviation within 0.03 percentage points of the target; (2) provides formal guarantees without requiring expansion of the search space, and in some cases even reduces the number of probed clusters; (3) dynamically adapts the cluster probes required per query; and (4) incurs negligible overheads when compared to existing state-of-the-art baselines. ConANN is integrated into the FAISS vector search library, facilitating adoption in real-world ANN systems.

PVLDB Reference Format:
Sonia Horchidan, Fabian Zeiher, Henrik Boström, and Paris Carbone. ConANN: Conformal Approximate Nearest Neighbor Search. PVLDB, 19(1): 29 - 42, 2025.
doi:10.14778/3772181.3772184

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/soniahorchidan/conann.

1 INTRODUCTION

The rise of unstructured data, such as text, images, and audio, has transformed how we represent and process information. Today, much of this data is encoded into dense vector embeddings, enabling powerful semantic similarity tasks across domains such as search engines, recommendation systems, and natural language processing [12, 27]. The advent of large pre-trained large language models has further amplified the need to scale vector search within massively large, high-dimensional embedding spaces and meet the demands of modern applications [12, 41]. To address this need, *Approximate Nearest Neighbor* (ANN) methods have become a key component of data-intensive applications, striking a critical balance between computational efficiency and retrieval accuracy, and enabling scalable processing of billion-scale datasets [25, 26].

Modern ANN Methods, most notably, the *Inverted File* (IVF) *index* [6, 24, 44], are widely adopted in production vector-search systems. IVF partitions the space into clusters, allowing the system to narrow the search to a few centroid-neighboring partitions. This design reduces search latency by avoiding computation across the entire dataset. However, this efficiency comes at the cost of approximation error, as true nearest neighbors may lie outside the probed clusters. Despite its practical impact, this trade-off remains fundamentally heuristic: the number of clusters probed is tuned

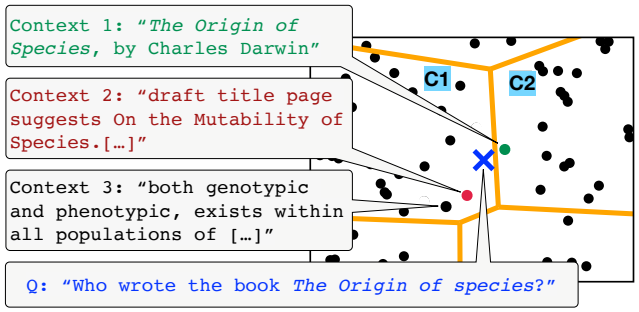

Figure 1: Example of approximate IVF search in a RAG-based LLMs using vector search for encoded passages. While an exact kNN search ($k = 1$) successfully retrieves Context 1 as relevant, the IVF-based approximate index mistakenly retrieves Context 2 if configured to probe only cluster ($C1$).

Table 1: Accuracy of RAG-based natural query answering using kNN and AkNN ($k = 1$) to retrieve the most relevant context based on its embedding proximity to the query.

Setup	Precision	Recall	F1	Search (ms)
LLM	0.051	0.209	0.082	0
kNN RAG	**0.118**	**0.558**	**0.195**	5.58
AkNN RAG	0.062	0.279	0.102	**0.146**

Proceedings of the VLDB Endowment, Vol. 19, No. 1 ISSN 2150-8097.
doi:10.14778/3772181.3772184

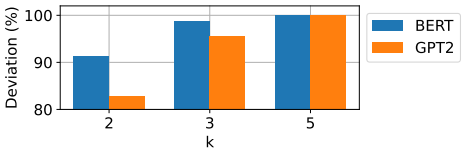

Figure 2: Local uniformity empirical experiment for the BERT and GPT-2 pre-trained embeddings.

manually, and there are no formal guarantees that the returned neighbors meet any accuracy thresholds.

RAG and the Case for Reliable ANN. The consequences of this trade-off are particularly apparent in Retrieval-Augmented Generation (RAG) pipelines, where approximate retrieval is used to condition large language models (LLMs) on relevant external knowledge [30]. In RAG systems, the semantic quality of the retrieved context directly influences the model's output. Accuracy losses in the retrieval accuracy can cascade into factually incorrect or irrelevant query answers. Figure 1 illustrates a scenario in which an IVF index, configured to search a single cluster, retrieves Context 2 as the most relevant context to a query Q encoded in Cluster C1, which the model then uses to answer the question. However, Context 1 is the more accurate response.

To assess the real-world impact, we implemented a simple RAG-based question answering system using the pre-trained LLaMA model[1]. As our query workload, we selected a subset of around 12k questions from the Natural Questions dataset [28], which consists of real user-issued search queries and associated short answers. We encoded both questions and candidate contexts using a dense embedding model and compared the accuracy of approximate IVF-based search against exact kNN retrieval. We configured the IVF index with 100 total clusters. Table 1 shows that, while AkNN achieved a 38× speedup, it exhibited a considerable drop in accuracy, measured here in Precision, Recall, and F1 score, showing that efficiency gains come at the cost of unpredictable accuracy loss. In high-stakes domains like healthcare or finance, even small drops in retrieval reliability can undermine user trust, legal compliance, or safety. This motivates a stronger requirement: *ANN systems must offer statistical accuracy guarantees that are enforceable at runtime.*

Limitations in Prior Work. While the interest in reliable ANN search is growing, existing methods fall into two categories: *best-effort* and *bounded-error* approaches. Best-effort methods, including those that can be facilitated by widely-used vector search libraries such as AnalyticDB-V [48] or Faiss [13, 25], as well as recent techniques that terminate early to minimize query latency [21, 31], rely heavily on sampling heuristics and empirically tuned parameters. These approaches optimize for performance but lack formal guarantees over standard metrics like recall or false negative rate, making their accuracy unpredictable. Further, the lack of guarantees can cause overly pessimistic results. In contrast, bounded-error methods aim to provide formal accuracy guarantees but do so under restrictive and often unrealistic assumptions. For instance, Auncel [50] assumes local uniformity in the distribution of data points to derive theoretical error bounds. These assumptions are rarely satisfied, for example, in real-world embeddings generated by deep language models like BERT or GPT, which are optimized for semantic similarity tasks and often exhibit varying density and non-linear structures across different regions [37]. To empirically assess the validity of the local uniformity assumption, we conducted a study on embeddings from BERT and GPT-2, analyzing local uniformity by comparing min-max normalized k nearest-neighbor distances to uniform distributions using a Kolmogorov-Smirnov (KS) test. Using a significance level of 0.05, we report the percentage of vectors for which the null hypothesis of uniformity was rejected. Figure 2 shows that for the majority of vectors, the local uniformity assumption is rejected, even for small values of k, and the rejection rate increases with larger neighborhoods. These findings question the applicability of methods that depend on such assumptions.

Contributions. We address this gap by introducing ConANN, a novel framework that leverages recent advances in uncertainty quantification for Machine Learning models [47] and Conformal Risk Control (CRC) [4] to provide formal, distribution-free bounded-error guarantees for IVF-based ANN search. ConANN treats ANN search as a black-box predictive task and dynamically calibrates the number of clusters probed per query to satisfy a user-specified recall tolerance. Unlike prior methods, ConANN makes no assumptions about the underlying vector data distribution and adapts automatically to varying query hardness and dataset structure. Furthermore, it requires no adjustments to the core IVF search algorithm. Our main contributions are the following:

- We adapt Conformal Risk Control to state-of-the-art Approximate Nearest Neighbor search methods, formulating the search problem as a black-box machine learning task.
- We introduce ConANN, the first framework offering formal recall guarantees for IVF-based ANN search, without requiring manual tuning or distribution assumptions.
- We evaluate ConANN on five standard ANN benchmarks, demonstrating tight recall control with a maximum deviation of 0.03 FNR from target error rates.
- We show that ConANN achieves these guarantees without increasing the search space relative to baselines, reducing cluster probes by up to 62.8%, with runtime overhead below 1.97 ms and calibration time under 4 minutes.
- We integrate ConANN into Faiss, showcasing its generality and ease of adoption in modern vector search systems.

2 PRELIMINARIES

This section reviews Approximate Nearest Neighbor (ANN) search, then introduces Conformal Prediction (CP) and Conformal Risk Control (CRC) as tools for principled error quantification. Table 2 summarizes the main notations used.

2.1 Approximate Nearest Neighbor Search

Exhaustive nearest-neighbor search is prohibitive on today's billion-vector datasets, so production systems accept approximate answers to cut latency. A standard choice for approximate search is the Inverted File Index (IVF) [6, 25]. During index construction, a vector quantizer (e.g., K-means) partitions the vector space into P cells, with the resulting centroids forming a coarse representation of the space. The search procedure in IVF balances precision and speed using a user-defined parameter, p, where $p \leq P$, which determines the number of clusters to probe. For a given query vector $\mathbf{x}$, the algorithm begins by examining the cluster with the closest centroid

[1] https://huggingface.co/ahxt/LiteLlama-460M-1T

Table 2: Summary of Notations

Notation	Description
k	Number of closest neighbors to be retrieved
$\mathbf{x}$	Query vector
P	Total number of clusters
p	Number of clusters searched
C_p	The p-th closest centroid to $\mathbf{x}$ (Voronoi cell)
S_p	ANN Search result after probing $C_1, ..., C_p$
S_{GT}	Exact kNN result set
$\mathcal{D}$	Dataset of vectors
N	$\lvert\mathcal{D}\rvert$
d	Vector dimensionality
M	Calibration dataset size
α	Error bound
α_p	Error after probing $C_1, ..., C_p$
$\pi(\mathbf{x}, y)$	The non-conformity score of input $\mathbf{x}$ with respect to label y
$\mathbf{q}^{(i)}$	The i-th element of the vector $\mathbf{q}$

and retrieves an intermediate list of k nearest neighbors. It then continues probing additional clusters, one at a time, until p clusters are searched or the desired accuracy is achieved.

We denote the AkNN search as S_p, aiming to approximate the ground truth S_{GT}. The quality of approximation is evaluated using recall, which quantifies the fraction of retrieved results present in S_{GT}, or equivalently, the False Negative Rate (FNR). As p approaches P, the search space is expanded and the FNR approaches 0.

$$FNR(S_p, S_{GT}) = 1 - \frac{|S_p \cap S_{GT}|}{k} \tag{1}$$

2.2 Conformal Prediction Methods

We build upon the Conformal Risk Control (CRC) framework [4], an extension of CP that supports arbitrary loss functions while preserving formal coverage guarantees. CP provides a model-agnostic approach for constructing prediction sets under a user-specified error rate. This section provides a brief introduction of both methods.

2.2.1 Conformal Prediction (CP). CP is a statistical framework that provides distribution-free, model-agnostic uncertainty quantification for predictive models [3]. The framework is applicable to both classification and regression problems, and allows for controlling the error rate, by turning point predictions into set predictions, i.e., sets of class labels for classification problems or prediction intervals for regression problems. The framework guarantees that an output set prediction will contain the true target (class label or regression value) with a user-defined probability (confidence level) $(1 - \alpha)$, where α, also known as *significance level*, represents the error bound. Originally, the framework was developed for an online, *transductive* setting, requiring retraining and calibration for each new observation. A recent computationally cheaper framework variant [47] known as *inductive* or *split* conformal prediction divides batches of the training set into proper training and calibration sets. The former is used to fit a model (using any algorithm), and the latter is used to compute so-called *non-conformity scores*, which encapsulate how unlikely a label y is for input x; when forming a prediction set $C(x_{M+1})$ for a test object, labels with a higher non-conformity score than the $(1 - \alpha)$ percentile (denoted $\hat{s}$) are rejected. The prediction sets are, therefore, defined as follows:

$$C(x_{M+1}) = \{y : \pi(x_{M+1}, y) \leq \hat{s}\}. \tag{2}$$

For a given calibration set $\{(x_i, y_i), \ldots, (x_M, y_M)\}$, and a test example (x_{M+1}, y_{M+1}), the framework guarantees that:

$$\mathbb{P}(y_{M+1} \in C(x_{M+1})) \geq 1 - \alpha, \tag{3}$$

This guarantee holds under the assumption of *exchangeability*, i.e., any permutation of $\{(x_i, y_i), \ldots, (x_{M+1}, y_{M+1})\}$ is equally probable. It should be noted that the framework makes no assumptions about the underlying algorithm and how the non-conformity scores are computed. In other words, the coverage rate in Eq. (3) is guaranteed no matter how we choose to implement the framework. However, for the informativeness (defined as how narrow the prediction sets are) of the prediction sets, the choice of learning algorithm and definition of non-conformity may have a large impact. These lenient requirements make CP broadly applicable across various Machine Learning models, with use-cases including classification, regression, anomaly detection, and active learning [43].

The key to CP's statistical guarantees lies in adjusting the prediction set size to fit the desired confidence level α. Higher accuracy requirements (lower α) result in larger prediction sets. The set size is tailored to the specific test instance, based on its non-conformity. Queries resembling the calibration data typically yield smaller, more informative sets, while outliers or less familiar inputs produce larger sets, reflecting greater model uncertainty. As the error probability is mathematically guaranteed, the efficiency (informativeness) of the predictions is critical for CP; a set containing all possible labels ensures full coverage but lacks practical utility.

2.2.2 Conformal Risk Control. Traditional CP methods offer coverage guarantees, as defined in Eq. (3). Conformal Risk Control [4] extends CP by generalizing the loss function to a broader class of non-increasing loss metrics (i.e., monotonic functions where the loss decreases as the prediction set size increases). This flexibility enables CRC to address application-specific performance criteria.

The goal of CRC is to construct prediction sets $C_\lambda(X_i)$ with user-defined guarantees on a specified loss function f. Given a calibration dataset $\{(x_i, y_i), \ldots, (x_M, y_M)\}$, a test example (x_{M+1}, y_{M+1}), and a prediction loss function $f(x, y)$, CRC ensures the following expectation bound:

$$\mathbb{E}\left[f(C_{\hat{\lambda}}(x_{M+1}), y_{M+1})\right] \leq \alpha, \tag{4}$$

where $\alpha \in (-\infty, B]$ is a user-specified risk upper bound. The new parameter $\hat{\lambda}$ is optimized during calibration and controls the conservativeness of the prediction set, balancing between tighter predictions and stricter adherence to the loss constraint. To obtain $\hat{\lambda}$, CRC evaluates the empirical risk over the calibration set for each candidate λ and identifies the smallest λ such that the expected calibration loss remains below α. Formally, this involves solving:

$$\hat{\lambda} = \inf\left\{\lambda \in [0, 1] : \frac{M}{M+1}\sum_{i=1}^{M} f(C_\lambda(x_i), y_i) + \frac{B}{M+1} \leq \alpha\right\} \tag{5}$$

The correction term $\frac{B}{M+1}$ accounts for sampling variability in the empirical risk estimate and ensures that the guarantee in Eq. (4) holds under finite-sample conditions. This adjustment prioritizes the risk constraint, potentially at the cost of larger prediction sets. Note that the correction term depends solely on the calibration set size M and does not scale with the total dataset size N. In practice, $\hat{\lambda}$ can be efficiently computed using binary search.

3 PROBLEM STATEMENT

We formulate the problem of providing statistical error guarantees for the IVF search. We define the optimization task and examine applying conformal methods to IVF, highlighting their benefits, prerequisites, and alignment with vector search requirements.

3.1 Objective

We reinterpret IVF search as a black-box approximation to the exact kNN retrieval, akin to how machine learning models approximate complex target functions. This analogy motivates the use of CP to quantify the approximation error.

We formalize the problem of providing statistical error guarantees for the AkNN error in IVF. Let $\mathbf{x}$ be the query vector, and S_p the AkNN result for x with parameter k, after probing clusters $C_1, C_2, ..., C_p$. Our goal is to find S_p corresponding to $\mathbf{x}$ such that the expected FNR, as defined in Eq. (1), is bounded by a user-defined threshold α. At the same time, we aim to minimize p, the number of clusters probed, to reduce computation time.

Importantly, achieving an FNR much lower than α is suboptimal in this setting. Since FNR can always be reduced by increasing p, overshooting the target leads to unnecessary computation without improving statistical guarantees. Thus, in addition to satisfying the constraint $\mathbb{E}[\text{FNR}(S_p, S_{GT})] \leq \alpha$, we aim to match α as closely as possible from below. In summary, we formulate this as the following constrained optimization problem:

$$\begin{aligned} \text{minimize} \quad & p \\ \text{subject to} \quad & \mathbb{E}\left[\text{FNR}(S_p, S_{GT})\right] \leq \alpha \\ & \left|S_p\right| = k; \quad p \leq P \end{aligned} \tag{6}$$

This formulation naturally gives rise to three key desiderata. First, the solution must guarantee *validity*: the expected False Negative Rate (FNR) across queries must remain below the user-specified threshold α, ensuring statistical reliability. Second, the solution should promote *efficiency* by minimizing p, the number of clusters probed during search. Third, the solution should exhibit *adaptivity*, adjusting p dynamically based on the difficulty of each query rather than relying on a fixed probing depth. We stress one final consideration: since the computational cost in IVF is primarily determined by cluster probes, reducing p leads directly to faster query processing. Therefore, the optimal solution must achieve tight calibration, minimizing unnecessary computation by keeping the actual FNR close to α without overshooting.

3.2 Applicability of Conformal Methods to IVF

First, we discuss the applicability of conformal methods to IVF. The design of IVF presents a unique opportunity for the application of conformal prediction techniques. While common statistical or learning-based methods for uncertainty quantification (UQ), such as ensembles [29] offer no formal guarantees or require complex assumptions about the underlying data distribution, conformal methods are distribution-agnostic, meaning they do not impose any assumptions about the distribution of the data. This makes them particularly well-suited to applications like IVF search, where the true distribution of the data is not known and is difficult to model due to high dimensionality and large data volumes.

The alignment of conformal methods with IVF systems goes beyond statistical benefits. Conformal methods require a calibration phase, but this step can be carried out offline, concurrently with the IVF index construction, and does not require constant updates during runtime. IVF indexes are typically built offline and retrained periodically, which aligns perfectly with the need for a fixed distribution in conformal prediction. As a result, once the index is built, the calibration step can be inherently embedded.

Regarding runtime overhead, conformal methods only require the computation of non-conformity scores, which can be directly derived from intermediary results already produced by the IVF search, followed by a simple threshold comparison to decide when to stop the probing. Conformal methods can treat the IVF search algorithms as black-box processes, making minimal modifications and introducing negligible performance penalties. In contrast, alternative UQ methods, such as Bayesian inference [17] or ensembles [29], typically require multiple forward passes or model replications, incurring substantial computational overheads that are often prohibitive in latency-sensitive ANN systems like IVF.

However, conformal methods do introduce specific requirements. Notably, conformal prediction assumes the data to be exchangeable. The calibration dataset must mirror the distribution of query vectors, which can be addressed at runtime by sampling from the query distribution.

Lastly, we highlight why CRC aligns particularly well with IVF. While standard inductive conformal prediction could, in principle, provide recall guarantees using class-conditional (Mondrian) conformal classifiers [47], doing so would require constructing a separate calibration set per class (i.e., per IVF cluster). This is infeasible in practice due to (1) scale, as IVF systems may contain thousands of clusters [25], and (2) sparsity, since many clusters might have too few samples for reliable calibration [23]. ConANN ultimately adopts Conformal Risk Control (CRC) as it is the only practical UQ framework that satisfies all three criteria essential for IVF integration: formal recall guarantees, distribution-free assumptions, and minimal runtime overhead.

4 CONANN OVERVIEW

Incorporating Conformal Prediction with CRC for AkNN Search using Inverted File Indexing (IVF) presents two principal challenges. The first challenge is to select an appropriate non-conformity score, which must encapsulate the uncertainty inherent in the model with respect to a given query $\mathbf{x}$. The second stems from the structure of IVF, which incorporates an intermediate classification step that departs from the standard conformal prediction setup. Specifically, IVF first maps the query vector $\mathbf{x}$ to a subset of centroids, which are subsequently probed to identify the k-nearest neighbors. In contrast to conventional conformal prediction methods, which focus on minimizing the size of the prediction set (i.e., the set of nearest neighbors, in our case), IVF requires minimizing the number of clusters probed while ensuring that the result set has size k and satisfies statistical error guarantees.

To address these challenges, we frame the IVF-based AkNN problem as a multi-label classification task, treating each IVF cluster (Voronoi cell) as a distinct class. This enables the application of Conformal Ranking Classification with FNR as a function of search

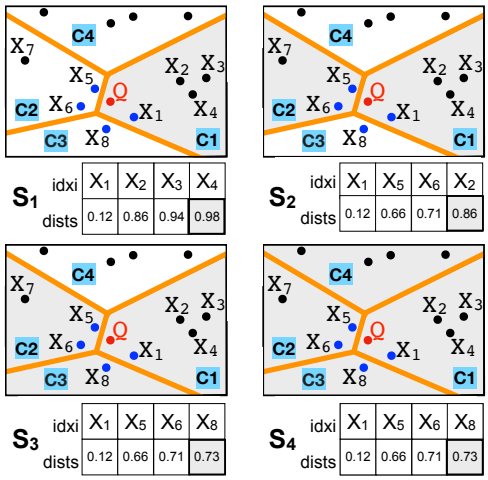

Figure 3: Example of non-conformity score computation before regularization.

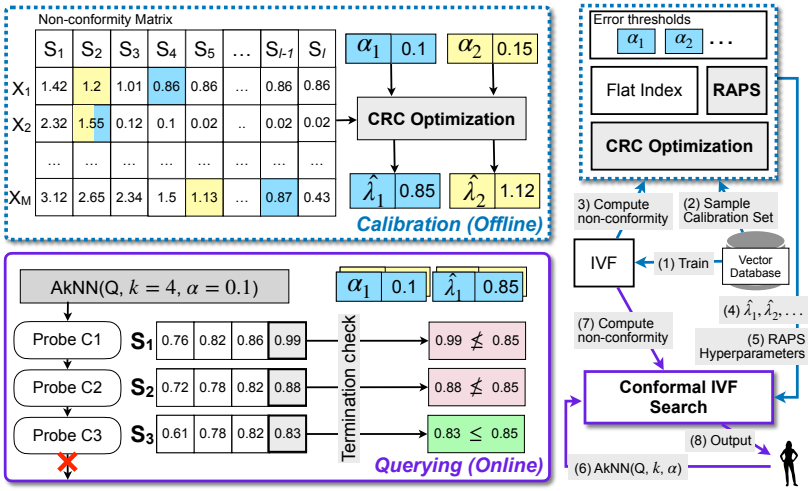

Figure 4: Overview of ConANN.

depth. In this framework, each search result S_p corresponds to a specific FNR value, which asymptotically approaches zero as the number of clusters probed p increases. We propose an effective calibration strategy and query algorithm to meet these challenges.

Figure 4 illustrates the full ConANN pipeline, containing an offline calibration stage and an online querying stage. We begin by training an IVF index on a sample dataset (Step 1). To compute the FNR required by the calibration process, we also require a flat index to obtain the ground-truth nearest neighbors for each query. We then sample the calibration query vectors (Step 2) and evaluate their approximate kNN results $S_1, \ldots, S_P$, where each S_p corresponds to probing p clusters. For each query vector, we compute the non-conformity score given the results S_p at each search depth (Step 3). CRC is applied to these scores to compute acceptance thresholds $\hat{\lambda}_j$ for different user-specified levels α_j (Step 4), as well as hyperparameters (Step 5), which will be described later in this section. At inference time (Steps 6, 7, 8), ConANN computes the non-conformity scores of the query vector , evaluates them against the optimized $\hat{\lambda}_j$, and returns the optimal set of kNN to meet the guarantee. We now detail the key components of the system.

4.1 Non-conformity Scores

We define non-conformity scores tailored to IVF-based AkNN search, where each score quantifies the quality of the result set S_p obtained after probing p clusters. Rather than computing the scores over all neighbors, we focus on a scalar summary: the distance to the k-th nearest neighbor in S_p. This definition aligns naturally with LLM retrieval scenarios. For example, as illustrated in Figure 1, the embedding corresponding to Context 1 lies closer to the query embedding, indicating higher semantic similarity and a greater likelihood of relevance. In contrast, Context 3 is embedded farther from the query, suggesting lower relevance. Further, this definition reduces computational and storage overhead, as the number of scores grows with the number of cluster probes P instead of the total database size N.

Formally, let S_p denote the set of database vectors retrieved after probing the top p clusters for a given query $\mathbf{x}$. We define the non-conformity score $\pi((\mathbf{x}, k), p)$ as the distance to the k-th nearest neighbor in S_p:

$$\pi((\mathbf{x}, k), p) = \min_{\mathbf{v} \in S_p} (d(\mathbf{x}, \mathbf{v}))^{(k)} \tag{7}$$

This score serves as a proxy for approximation quality: as p increases, S_p approaches the ground-truth set $S_{GT} = S_P$, and the k-th distance decreases accordingly. Larger scores indicate that relevant vectors remain unretrieved, implying a lower likelihood that the true k nearest neighbors are present in S_p. This behavior aligns with prior findings that query vectors located near Voronoi cell boundaries often require probing more clusters to retrieve their true neighbors [50]. To ensure robustness across vector spaces of varying scales and dimensions, we also apply min-max normalization to $\pi(q, k)$ across all calibration queries. These scores are then passed to the calibration procedure, but the same methodology is also followed at query time.

Figure 3 illustrates the computation of non-conformity scores in a 2-dimensional example using an IVF index with $P = 4$ clusters and a query vector $\mathbf{Q}$ seeking $k = 4$ neighbors. As clusters $C_1, C_2, \ldots, C_P$ are probed incrementally, we compute intermediate result sets S_p and record the distance to the k-th nearest neighbor in each set. In our example, after probing C_1 and C_2, the closest four neighbors found are X_1, X_5, X_6, and X_2, with corresponding distances of 0.12, 0.66, 0.71, and 0.86 to the query vector $\mathbf{Q}$. Our algorithm, therefore, records the distance to X_2 to build the nonconformity score of $\mathbf{Q}$ with respect to S_2. The search process continues until all P clusters are probed or an early stopping criterion is met. Once S_p contains all ground truth neighbors (in this example, S_3), the score stabilizes and remains unchanged, even if additional clusters are probed.

4.2 Conformal IVF Search

The goal of offline calibration is to determine the CRC acceptance threshold $\hat{\lambda}$ corresponding to a user-specified error tolerance α. The procedure begins by sampling a calibration set of size M and

computing non-conformity scores as described in Section 4.1. These scores are subsequently passed to the CRC optimization routine outlined in Figure 4. Prior to detailing the calibration algorithm, we highlight a key empirical observation: direct application of CRC often results in overly conservative prediction sets, a trend we revisit in Section 6. This behavior is consistent with prior work [5], which observes that conformal methods tend to be conservative when the number of classes is large (i.e., > 1000). In our setting, IVF clusters are treated as class labels; thus, the effective label space scales with dataset size. For example, with 1M vectors, P should be set to a value in the order of thousands[2]. Such settings pose challenges for classical conformal prediction techniques. To mitigate this, we adopt the Regularized Adaptive Prediction Sets (RAPS) framework [5], originally developed for multi-class classification, and adapt it to the ANN setting as detailed in Section 4.2.1. RAPS penalizes the inclusion of low-ranked predicted classes, promoting tighter, more focused prediction sets.

4.2.1 Regularization. We now describe the regularization step used to improve ConANN's efficiency. Our approach builds on the RAPS algorithm [5], originally developed for conformal classification, which adds lightweight regularization to conformity scores to produce more efficient (smaller) prediction sets without sacrificing coverage guarantees. Given a test example $(\mathbf{x}, y)$, the RAPS scores are computed by summing up the predicted probability of a class y (e.g., the softmax score returned by the model during inference) with the following regularization term:

$$\beta(\mathbf{x}, y) = \gamma \cdot (o_{\mathbf{x}}(y) - c_{reg})^{+} \tag{8}$$

The regularized term includes: (1) $o_{\mathbf{x}}(y)$, defined as the ranking of y among all the labels, given the scores π, and (2) two hyperparameters γ and c_{reg}, which are designed to promote small prediction set sizes. $(z)^{+}$ denotes the positive part of z. The hyperparameters are tuned during calibration on a small data sample using a grid search, where the conformal calibration procedure is run with different values to find the combination that minimizes prediction set size. As shown in [5], RAPS is generally robust to the choice of hyperparameters: the theoretical validity is maintained regardless of γ and c_{reg}, but good choices can substantially improve efficiency.

In our setting, we adapt RAPS regularization to the IVF-based AkNN search. The non-conformity score $\pi((\mathbf{x}, k), p)$ for a query vector $\mathbf{x}$ and candidate set S_p is defined as the (normalized) distance to the k-th nearest neighbor. To align with RAPS, which assumes that higher scores are better, we take the complement $1-\pi((\mathbf{x}, y), p)$. Our regularized non-conformity score $\hat{\pi}((\mathbf{x}, y), S_p)$ then becomes:

$$\hat{\pi}((\mathbf{x}, k), S_p) = (1 - \pi((\mathbf{x}, k), S_p)) + \gamma \cdot (p - c_{\text{reg}})^{+} \tag{9}$$

Here, p is the position (rank) of S_p among all candidate clusters or vectors. Crucially, in IVF search, the candidates are processed in order of increasing distance (i.e., better matches first), so the ranking $o_x(y)$ is known in advance and simplifies to p directly.

This regularization promotes early termination by slightly penalizing candidates that appear later in the ranking, thereby reducing computational cost without violating the formal risk guarantees of CRC. As we will show in the following sections, this leads to faster termination in the IVF search.

[2]According to the Faiss guidelines, P should be set as $O(\sqrt{N})$.

Algorithm 1 Calibration Phase (Offline)

1: **Input:** Error bound α, total number of clusters P, calibration vectors $\mathcal{X} = \{x_i\}_{i=1}^{M}$, $\{k_i\}_{i=1}^{M}$, ground truths $\mathcal{S} = \{S_{GT_i}\}_{i=1}^{M}$.
2: **Output:** Optimal acceptance threshold $\hat{\lambda}$.
3: **for** $i \leftarrow 1$ **to** M **and for** $p \leftarrow 1$ **to** P **do**
4: $\quad S_{i,p} \leftarrow \texttt{IVF\_scan}(x_i, p)$
5: $\quad$ Compute nonconformity $\pi_{i,p} \leftarrow \texttt{score}(k_i, S_{i,p})$
6: Fit hyperparameters γ, c_{reg} using $\pi, \mathcal{S}$
7: **for** $i \leftarrow 1$ **to** M **and for** $p \leftarrow 1$ **to** P **do**
8: $\quad \hat{\pi}_{i,p} \leftarrow (1 - \pi_{i,p}) + \gamma \cdot \max(0, p - c_{\text{reg}})$
9: **for** each λ in a discretized grid over $[0, 1]$ **do**
10: $\quad$ **for** $i \leftarrow 1$ **to** M **do**
11: $\quad\quad$ Find $p' \leftarrow \min\{p : \hat{\pi}_{i,p} \leq \lambda\}$
12: $\quad\quad$ Select $\hat{S}_{i,\lambda} \leftarrow S_{i,p'}$
13: $\quad\quad$ $R_{\lambda,i} \leftarrow \texttt{FNR}(\hat{S}_{i,\lambda}, S_{GT_i}) - \alpha\frac{M+1}{M} - \frac{1}{M}$
14: $\hat{\lambda} \leftarrow \texttt{infimum}(R_\lambda, \lambda \in [0, 1])$
15: **return** $\hat{\lambda}$

4.2.2 Calibration. The offline calibration phase computes an acceptance threshold $\hat{\lambda}$ to provide distribution-free guarantees on the FNR during the AkNN search. Following the CRC framework, we frame the IVF search process as a structured prediction task and calibrate $\hat{\lambda}$ to satisfy a user-specified risk level α. Algorithm 1 summarizes the procedure.

Given a calibration set of vectors $\mathcal{X}$, corresponding k values, and their corresponding ground-truth nearest neighbors $\mathcal{S}$, we first simulate the IVF search behavior (Line 4), which scans only the p-th closest cluster to a given query vector. For each query $x_i \in \mathcal{X}$ and each possible number of probed clusters $p \in \{1, 2, ..., P\}$, we record the intermediate search result $S_{i,p}$ and compute a non-conformity score $\pi_{i,p}$ (Line 5), as described in Section 4.1, measuring the search quality relative to the ground truth. We employ a simplified version of the RAPS regularization to strengthen the relationship between the number of clusters probed and the search quality, as described by Eq. (9). γ and c_{reg} are hyperparameters fitted to a sample of 1000 vectors of additional unseen calibration data (Line 6). The regularization results in a matrix of scores, named non-conformity matrix in Figure 4, which is then fed to the CRC Optimization step, aiming to identify the optimal $\hat{\lambda}$ acceptance thresholds.

Next, we evaluate candidate thresholds sampled from a discretized grid over $[0, 1]$, with arbitrary precision. For each query and each candidate λ, we select the smallest number of clusters p' such that the regularized non-conformity score $\hat{\pi}_{i,p} \leq \lambda$. We then compute the corresponding empirical FNR across the calibration set. To account for the finite size of the calibration data, we adjust the empirical risk using the CRC correction, with $B = 1$ to denote the upper bound of the FNR. Finally, we select the smallest λ that satisfies the risk constraint, as defined in Eq. (5). Figure 4 shows the selection of λ values for two error bounds $\alpha_1 = 0.1$ and $\alpha_2 = 0.15$.

This calibration step is performed once, at the IVF index build time. It produces the acceptance threshold $\hat{\lambda}$ that governs the per-query behavior during online search, and the RAPS hyperparameters. For instance, the example provided in Figure 4 identifies $\hat{\lambda}_1 = 0.85$ for a requested FNR of 10%, and $\hat{\lambda}_2 = 1.12$ for 15%. We

Algorithm 2 IVF Search (Online)

1: **Input:** Query vector $\mathbf{x}$, k, optimal threshold $\hat{\lambda}$, calibrated hyperparameters γ, c_{reg}.
2: **Output:** Predicted output $\hat{S}_p$.
3: $p \leftarrow 1$
4: **for** $p \leftarrow 1$ **to** P **do**
5: $\quad \hat{S}_p \leftarrow \texttt{IVF\_scan}(\mathbf{x}, p)$
6: $\quad$ Compute nonconformity score $\pi_p \leftarrow \texttt{score}(k, \hat{S}_p)$
7: $\quad \hat{\pi}_p \leftarrow (1 - \pi_p) + \gamma \cdot \max(0, p - c_{\text{reg}})$
8: $\quad$ **if** $\hat{\pi}_p < \hat{\lambda}$ **then**
9: $\quad\quad$ **Break**
10: **return** $\hat{S}_p$

stress the following two aspects: (1) computing non-conformity scores does not add extra cost because they are based on intermediate results already produced by the standard IVF search, and (2) although different risk levels α require different thresholds $\hat{\lambda}$, all optimizations (Lines 11–18) can reuse the same set of precomputed non-conformity scores. As we will see later, the most time-consuming part of calibration is computing these scores.

4.2.3 Online IVF Search. The online search phase iteratively probes the IVF index to find the k-nearest neighbors of a query $\mathbf{x}$ while ensuring that the error guarantee, controlled by the calibrated threshold $\hat{\lambda}$, is satisfied. The procedure is summarized in Algorithm 2.

The IVF search is configured to scan all P clusters. As each cluster is probed, we compute the non-conformity score for the current set of nearest neighbors using Eq. (7), and apply the regularization described in Eq. (9). At every iteration, the regularized non-conformity score of the k-th farthest vector in the current result set is compared to $\hat{\lambda}$ (Line 8). According to the incremental nature of IVF search, these scores are computed without extra overhead, enabling early stopping: if the score drops below $\hat{\lambda}$, it indicates that the error guarantee has been met, and the search terminates early, returning the current set of k nearest neighbors S_p. This modification allows the CRC framework to minimize the number of clusters probed, rather than adjusting the prediction set size, aligning with the optimization objective described in Eq. (6). As clusters are probed sequentially, the search can terminate as soon as the guarantee is met, minimizing p. For example, in Figure 4, the search stops after probing the third cluster, where the non-conformity score (0.71) falls below $\hat{\lambda}_1 = 0.85$. If the stopping criterion is not met, the search continues probing one more cluster at a time until either the threshold is satisfied or all P clusters are exhausted.

This procedure ensures that the search probes only as many clusters as necessary to achieve the desired accuracy, reducing computation time and adapting to the difficulty of each query. Moreover, it naturally integrates with the standard IVF search process, since intermediate nearest neighbor results are already available as clusters are incrementally scanned.

5 EXPERIMENTS

In the empirical evaluation, we aim to answer the following research questions:

Table 3: Datasets summary.

Dataset	d	N	Query
BERT [12]	768	30522	10k
FASTTEXT [8]	300	1M	10k
GIST [24]	960	1M	1k
SIFT1M [33]	128	1M	10k
DEEP10M [42]	96	10M	10k

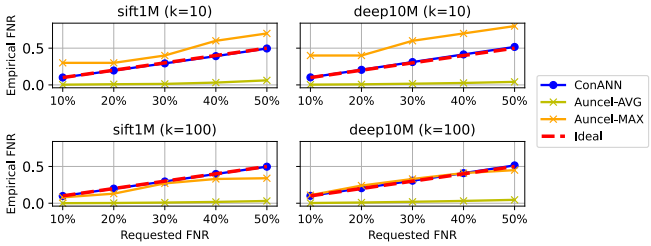

Figure 5: Validity results for Auncel.

(RQ1) Validity: Does ConANN accurately control the FNR across varying configurations? How does it compare against state-of-the-art Bounded-Error ANN methods and Best-Effort ANN methods?
(RQ2) Efficiency: How does ConANN compare to the baselines in terms of the average number of clusters searched? Does it expand the search space to provide formal error guarantees?
(RQ3) Adaptivity: Is ConANN capable of adapting the search space to the particularities of each query?
(RQ4) Overhead: What is the performance overhead of integrating conformal methods into the IVF search?

5.1 Setup

The following section outlines the experimental setup, including hardware and software configurations.

5.1.1 Hardware. The experiments were conducted on Google Cloud Platform (GCP) using a virtual machine with an `Intel(R) Xeon(R) CPU @ 2.80GHz`, 64 vCPUs, 256 GB RAM, and Ubuntu 20.04.6 LTS. The Intel-MKL SIMD instruction set was used throughout all the experiments to speed up the computation.

5.1.2 Datasets. We use high-dimensional datasets, ranging from 96 to 960 dimensions and 30k to 10M vectors, widely used in ANN research [19, 24, 40]. A summary is provided in Table 3. BERT [12] and FASTTEXT [8] are pre-trained word embeddings. The SIFT1M [33] and GIST [24] datasets are based on image data using their respective SIFT and GIST descriptors. The DEEP10M [42] dataset contains image embeddings generated by a convolutional neural network. Each dataset includes a uniform sample of 10k query vectors, with the exception of GIST, where we sample only 1k query vectors due to the prohibitively large dimensionality.

5.1.3 Baselines. Auncel [50], a Bounded-Error-type method, provides a different type of error guarantee than our method. Specifically, it targets a bound on the *maximum* false negative rate (FNR), whereas we provide guarantees on the *average* FNR. Despite this difference, we include Auncel in our evaluation because it is the only existing method that attempts to control the approximation

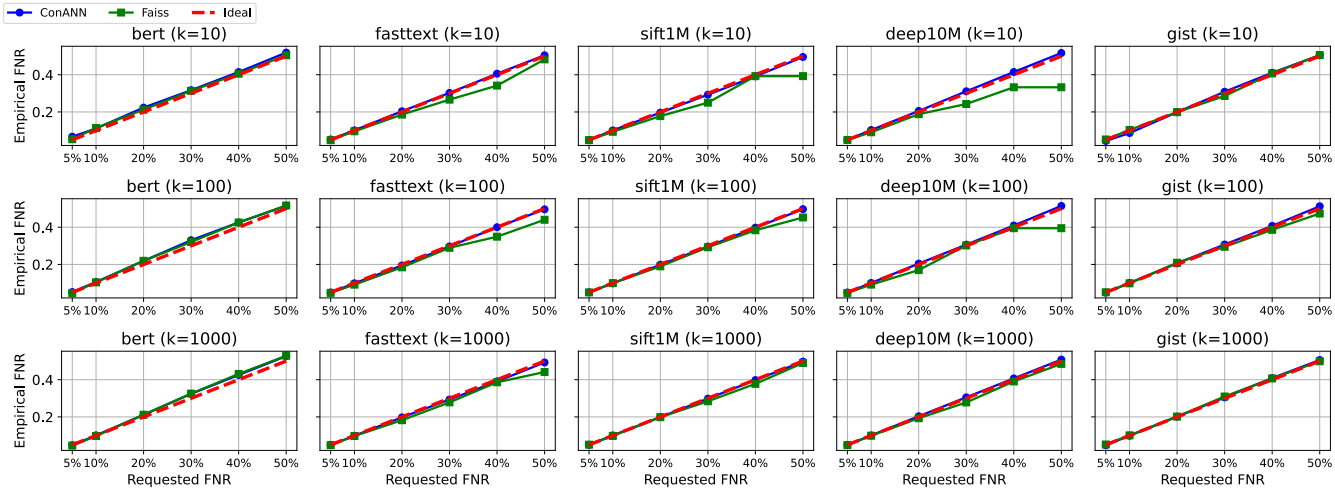

Figure 6: Validity results. ConANN offers effective FNR control over the tested datasets.

error. Our evaluation uses the hyperparameters reported in the original paper and tests the same datasets, as the tuning procedure is not disclosed. Since no other existing system offers formal false negative rate (FNR) guarantees under arbitrary data distributions, we compare ConANN against Faiss [25]. While Faiss does not natively support FNR-constrained configurations, it can be adapted into a best-effort baseline by tuning its probing strategy [50]. Specifically, we perform an offline calibration on a configuration query sample: starting from a small number of clusters, we incrementally increase the probe count until the empirical average FNR falls below the target threshold α. The smallest such probe count is then fixed and applied uniformly to all test-time queries.

5.1.4 Configuration. ConANN is integrated into Faiss 1.9.0 and uses Faiss' core features like distance computation and memory management. Parallel computing based on OpenMP is enabled with 60 threads for both ConANN and Faiss to ensure fairness. We used the L2-squared Euclidean distance as the distance metric in all the experiments. For the validity, efficiency, and overhead experiments the IVFFlat index was used, storing the vectors in their original state inside the index. Similar to related studies, the number of clusters P was fixed to 1024, except for the BERT dataset, where P was set to 128 clusters [50]. We trained the index on half of the dataset in the interest of time. For all datasets, we choose 50% of the query dataset for calibration and use the remaining for RAPS hyperparameter tuning (1000 query vectors) and testing. The min-max normalization of the non-conformity scores uses 0 as the minimum value. It approximates the maximum distance geometrically via the diagonal of the hyperrectangle containing all vectors, considering the dataset range and dimensionality. Exact distance computation is also feasible but costly, requiring quadratic time relative to dataset size. We use Brent's numerical optimization method [9] to find the optimal $\hat{\lambda}$. Without loss of generality, we calibrate and evaluate ConANN on three fixed values of k. However, the method's guarantees hold beyond this setting. Additional experiments calibrating over a range of k values, demonstrating runtime robustness in case of dynamic k changes from one query to the next, are available in our public GitHub repository.

5.1.5 Metrics. We measure empirical FNR using Eq. (1) to validate the adherence to the theoretical bound. For each query, we log the number of clusters searched to assess ConANN's adaptivity and compare its efficiency against Faiss. Two metrics are used: (1) the per-query cluster ratio, computed as ConANN's reported p divided by Faiss' static p, and (2) the average cluster ratio, defined as Faiss' static p over ConANN's average p. For example, a ratio of 1.2 implies Faiss searches 20% more clusters. We also measure ConANN's latency overhead by comparing end-to-end query times where p matches. Finally, we report ConANN's calibration time.

5.2 Experimental Results

We evaluate ConANN on five standard ANN datasets under varying k and FNR settings, and observe that it:

(1) Consistently meets user-defined FNR bounds across diverse datasets and calibration sizes (Figure 6), whereas Auncel and Faiss frequently deviate from the target FNR.
(2) Provides formal error guarantees proven empirically, without increasing the search demand and even decreasing the number of clusters needed to be probed up to 62.8% when compared to Faiss (Figure 7),
(3) Dynamically adapts the search space per query (Figure 8), regardless of dataset characteristics, and
(4) Introduces minimal latency (less than 1.97 ms in our experiments, Figure 9), with calibration completing in under 4 minutes regardless of dataset size or complexity.

5.2.1 Validity. We begin by evaluating Auncel, which aims to guarantee a bound on the *maximum* false negative rate (FNR). Using the hyperparameters reported in the original paper, we assess Auncel on the SIFT1M and DEEP10M datasets, as no tuning method is provided to test other datasets. As shown in Figure 5, Auncel frequently violates its maximum FNR targets, particularly for $k = 10$, and offers no control over the average FNR. In fact, the average FNR often approaches that of exhaustive search, indicating highly conservative behavior. Due to these shortcomings, we omit Auncel from subsequent comparisons.

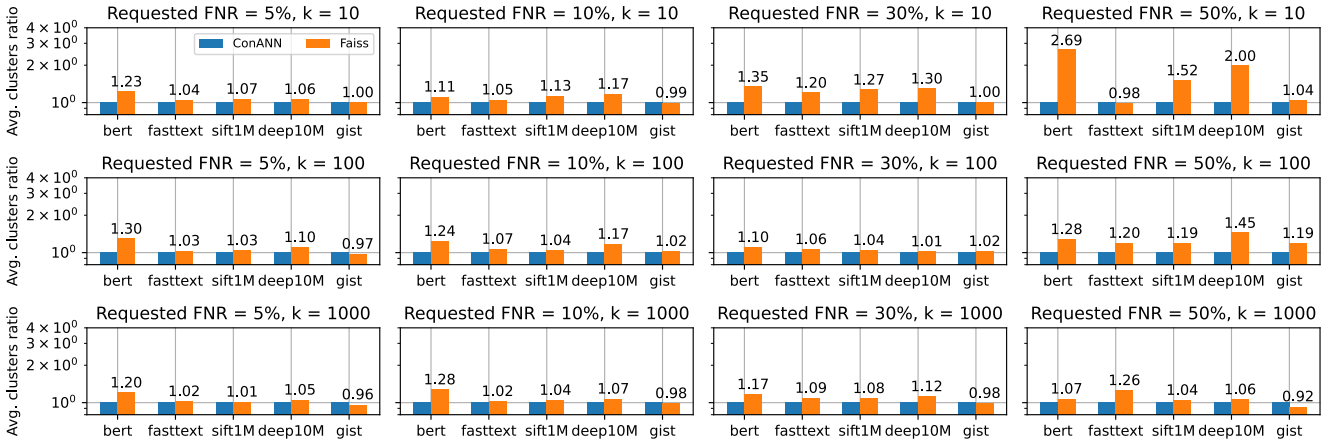

Figure 7: Efficiency results. ConANN matches or even reduces the probing depth when compared to standard methods.

We assess the validity of ConANN by measuring how well the empirical FNR tracks the requested FNR. Figure 6 presents results for $k \in \{10, 100, 1000\}$ and requested FNRs ranging from 5% to 50%. The red dashed line indicates the ideal behavior, where empirical FNR exactly matches the user-specified threshold. ConANN closely follows this ideal FNR across all conditions, indicating that it reliably satisfies the used-defined constraint. Notably, ConANN maintains this alignment even under challenging settings, and deviates at most by only 0.03 percentage points. For instance, on the GIST dataset with $k = 10$, where calibration data is limited to 500 vectors, ConANN still tracks the target line well.

In comparison, Faiss tends to undershoot the target FNR across several settings, which is particularly noticeable on FASTTEXT, SIFT1M, and DEEP10M for $k = 10$ and $k = 100$. This conservative tendency leads to lower empirical FNRs than requested, indicating unnecessarily large search spaces. While this preserves a pessimistically low FNR, it undermines efficiency, as we will note in the next section. The trend attenuates for larger values of k, where both methods converge more closely to the ideal.

On the BERT dataset, ConANN exhibits marginal deviations of the requested FNR across targets. This behavior likely arises from the nature of the non-conformity scores. Although the calibration sample is large and uniformly drawn, the scores on BERT are less discriminative, potentially due to dense, semantically clustered embeddings and small distances that cause precision losses during the min-max normalization. As a result, the correspondence between quantile thresholds and true query difficulty can be slightly weakened, leading to negligible perturbations from the target FNR, bounded in expectation. This behavior aligns with the theoretical expectations of CRC [4]. However, further investigations could help refine the quality of the non-conformity scores.

Overall, these findings affirm ConANN's capacity to maintain reliable FNR control, with only slight deviations that remain within the expected probabilistic bounds of the CRC framework.

5.2.2 Efficiency. We measure the efficiency of the methods using the ratio of average clusters searched by each method to achieve the desired FNR. First, we vary the requested FNR and the k value. The results are showcased in Figure 7.

Table 4: Average clusters ratio for ConANN against Faiss for variable total number of clusters P ($k = 100$, requested FNR = 10%). A ratio of 1.1 indicates that Faiss searched 10% more clusters on average than ConANN. Our results across datasets show that ConANN is not sensitive to varying P.

P	fasttext	sift1M	deep10M	gist
512	1.09	1.06	1.22	1.01
768	1.06	1.11	1.15	1.00
1024	1.07	1.04	1.17	1.02
1536	1.05	1.05	1.15	1.00
2048	1.01	1.02	1.13	1.00

Across all experiments, ConANN either matches or reduces the search space compared to Faiss, achieving the same empirical FNR. This demonstrates ConANN's ability to adapt the search space size based on the characteristics of each query vector. In contrast, Faiss often searches more clusters than necessary. For instance, a Faiss-to-ConANN ratio of 1.35 means Faiss searches 35% more clusters, incurring unnecessary computational overhead without any gain in accuracy. This inefficiency stems from Faiss's conservative strategy for approximating the FNR. As illustrated in Figure 6 for the SIFT1M dataset at $k = 10$, Faiss' deviation from the target FNR leads directly to excess cluster searches. Specifically, while Faiss initially searches only 7% more clusters than ConANN at a 5% requested FNR, this gap widens substantially, to 52%, as the FNR target decreases. A similar trend is observed for the DEEP10M dataset at $k = 10$, where Faiss searches twice as many clusters as ConANN for 50% requested FNR because of the empirical FNR deviation.

ConANN's efficiency advantage is even more pronounced on BERT, where the larger calibration data sample (about one-third of the entire dataset) allows ConANN to better capture the data distribution. For smaller calibration data sizes, the results remain valid but are more conservative. For example, on GIST, where the calibration sample consists of 500 vectors only, the CRC framework is more conservative, leading to less efficient search behavior. While the validity property is still met, the efficiency gains observed for other datasets are not present on GIST. This is expected and consistent with the design of the CRC framework, which prioritizes valid coverage over aggressive optimization. However, the number of clusters searched by ConANN does not deviate significantly from Faiss in our experiments, showing the efficiency of RAPS.

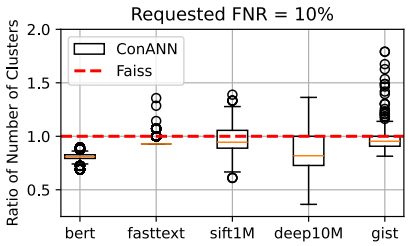

Figure 8: Adaptivity results ($k = 100$, requested FNR = 10%). ConANN adjusts the search effort by query difficulty.

We also assess the robustness of our method across varying P values, as shown in Table 4. We observe minimal changes in the average clusters searched ratio compared to Faiss, demonstrating that ConANN is not sensitive to P. On the DEEP10M dataset, the improvement becomes more pronounced as P decreases, peaking at a 22% ratio for $P = 512$, suggesting that an optimal P choice can yield further efficiency gains. However, since the data distribution is unknown, selecting the ideal P can be challenging. Nonetheless, our results show that ConANN remains both efficient and valid, regardless of P selection. Across diverse datasets and varying total number of clusters, ConANN's empirical FNR deviates by no more than 0.002 percentage points from the target, demonstrating robustness across all conditions.

Our results demonstrate ConANN's ability to guarantee the user-defined error threshold without searching unnecessary clusters, and, in many cases, even reducing the search space. This holds consistently across different values of k, requested FNRs, and datasets.

5.2.3 Adaptivity. Figure 8 highlights ConANN's ability to adapt the size of the search space to meet a fixed FNR target across diverse datasets. Specifically, we evaluate the ratio of the number of clusters searched by ConANN to that of Faiss, under a requested FNR of 10% with $k = 100$. This ratio, centered around the red dashed line (baseline Faiss performance), quantifies ConANN's adaptivity: values below 1 indicate more efficient search due to adaptive behavior, while values above 1 imply overcompensation. We observe that ConANN effectively tailors the number of clusters searched across datasets. For example, on the BERT dataset, the method consistently selects fewer clusters than Faiss, leveraging the availability of high-quality calibration data, as discussed above. In contrast, datasets like DEEP10M and SIFT1M exhibit a wider spread in the number of clusters searched, reflecting the intrinsic variability in query difficulty and data distribution. Despite this, ConANN maintains an overall balanced ratio below 1, indicating robust adaptivity without excessive conservativeness. Interestingly, while FASTTEXT and GIST show some outlier queries requiring substantially more clusters, the median remains close to or below 1.0, reinforcing ConANN's ability to adapt to both easy and hard queries. The wide spread on GIST is likely caused by the small calibration sample used, as discussed above; the queries corresponding to the outliers are likely not well represented in the calibration distribution, leading the method to behave conservatively in order to maintain validity. These results confirm that ConANN dynamically adjusts the retrieval process in response to query-level and dataset-level characteristics, rather than relying on static thresholds.

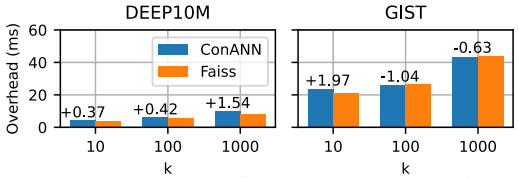

Figure 9: Latency overhead (requested FNR = 10%). ConANN introduces minimal latency overheads.

Table 5: Calibration breakdown per dataset, measured in seconds ($k = 100$, requested FNR = 10%). ConANN completes the calibration in under 4 minutes for all the tested datasets.

Step	bert	fasttext	sift1m	deep10m	gist
Index Building	0.727	25.610	10.043	20.229	52.013
Compute Scores	2.107	68.268	33.874	212.666	20.807
Pick Hyperparameters	2.131	5.186	4.617	5.570	0.556
Regularize Scores	0.011	0.155	0.154	0.150	0.018
Optimize (sec)	1.015	5.021	3.981	6.270	0.821
Total	**5.264**	**78.630**	**42.626**	**224.656**	**22.202**

5.2.4 Runtime Overhead. As described in Algorithm 2, ConANN extends the IVF search by adding a lightweight termination check at runtime with a single if statement. We stress that the non-conformity scores computation is already performed by Faiss, which, by design, probes the clusters iteratively, starting from 1, and computes intermediary steps. Therefore, the integration with ConANN is straightforward. We first measure the runtime cost of ConANN by measuring the per-query latency for a requested FNR of 10% across different values of k and averaging the results. We chose two datasets for this experiment, DEEP10M and GIST, to perform the measurements on a very high-dimensional dataset (GIST) and a large dataset with low dimensionality (DEEP10M). We compare only the queries where both methods searched the same number of clusters to quantify the overhead added by ConANN on top of the index search. For the DEEP10M dataset, 727, 585, and 629 queries were used for $k = 10$, 100, and 1000, respectively. For GIST, 308, 38, and 55 queries were found for $k = 10$, 100, and 1000, respectively. Figure 9 shows the total latency incurred by ConANN and Faiss, with the annotated values indicating the additional overhead (in milliseconds) introduced by ConANN relative to Faiss. On both datasets, ConANN introduces minimal overheads, of a maximum of 1.97ms on GIST for $k = 10$. The measured overhead is always negligible and can also be attributed to system-level factors such as CPU scheduling, which is likely the case for the GIST dataset at $k = 100$ and $k = 1000$, where we measure a negative overhead. Overall, ConANN imposes negligible runtime latency overhead across datasets and k values, while providing formal error guarantees. This reinforces its practicality as a plug-in extension for conventional ANN methods in latency-sensitive applications.

5.2.5 Calibration Overhead. We evaluate the one-time calibration cost of ConANN by profiling the end-to-end time across its four main stages: non-conformity score computation, RAPS hyperparameter selection, score regularization, and $\hat{\lambda}$ optimization. All measurements are performed for a fixed configuration with a requested FNR of 10% and $k = 100$. Table 5 shows a detailed breakdown for five datasets. The total calibration time ranges from 5.26s (BERT) to 224.66s (DEEP10M), with the compute scores step being the

Table 6: RAPS improvement in terms of average number of clusters searched. ConANN achieves notable reductions in average number of clusters searched when employing RAPS.

Method	bert	gist1m	deep10m
w/o RAPS	123.56	1008.03	991.5
ConANN	33.79	66.33	9.37

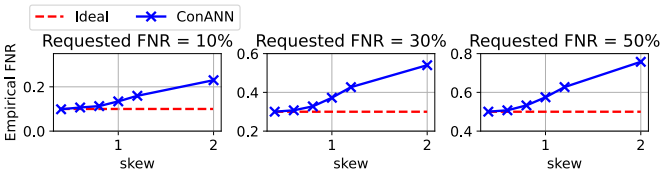

Figure 10: Validity under power-law skew on SIFT1M (k = 100). Skew controls the bias towards popular queries.

dominant contributor. Importantly, the score computation is independent of downstream optimization and can be reused across multiple calibration runs, amortizing its cost. While calibration for large datasets like DEEP10M incurs higher cost, it remains a one-time offline process. For smaller or more structured datasets such as GIST and BERT, the overhead is minimal.

6 DISCUSSION

The effect of regularization. The regularization step (RAPS) refines the non-conformity scores, enhancing their discriminative power. As IVF search increases the number of cluster probes, relying solely on the k-th nearest neighbor distance often results in many equal non-conformity scores. RAPS mitigates this by distinguishing between these probes. In our experiments, the regularization step significantly improves search efficiency. Without RAPS, ConANN is more conservative in the number of clusters searched, leading to valid but less efficient retrieval. As shown in Table 6, RAPS results in notable reductions: from 123.56 to 33.79 on BERT (-72.7%), from 1008.03 to 66.33 on GIST1M (-93.4%), and from 991.5 to 9.37 on DEEP10M (-99.1%). These findings align with previous work [5], which reported up to 10x reductions in prediction set size, and confirm that RAPS enhances efficiency across various datasets while maintaining ConANN's adherence to FNR constraints.

Query Distribution Shift. As noted in Section 3.2, ConANN assumes the calibration and test queries come from the same distribution. To test robustness, we simulate a power-law skew in the test queries (0.0 for uniform, up to a factor of 2) while keeping the calibration queries uniformly sampled. Figure 10 shows that ConANN can tolerate small drifts, but its empirical FNR increases with skew, deviating from the target. At 10% target FNR, the observed FNR rises to 22% under strong skew, and the gap grows at higher targets. These results highlight the need for drift adaptation techniques. A naive solution is to recalibrate periodically, ensuring the calibration sample matches the current query distribution. More principled alternatives could target drift detection, where a violation of the exchangeability assumption is detected using conformal test martingales [47]. Others could leverage recent methods like Weighted Conformal Prediction [7, 46] and Adaptive Conformal Inference [20], which adjust nonconformity scores to account for distribution shifts, for instance by penalizing the impact of old queries. Integrating such techniques for ConANN is an important direction for future work.

Personalized Querying. State-of-the-art methods for error bounding in ANN often rely on sampling techniques that require the sample distribution to match the data distribution. In contrast, ConANN operates by ensuring that the query distribution aligns with the calibration data, opening the door for personalized predictions based on specific workloads. This distinction could enable ConANN to adapt more effectively to varying query characteristics. Further experiments are needed to explore this potential fully, as our current setup follows the ANN literature to sample queries uniformly at random. That said, this insight suggests opportunities for optimization in real-world applications. For example, if an application frequently queries vectors close to cluster centroids, ConANN could leverage this characteristic to reduce the number of clusters searched, ultimately decreasing query latency and improving efficiency.

Marginal Coverage. ConANN guarantees marginal coverage, similar to other conformal methods: the false negative rate (FNR) constraint holds on average over the query distribution, not for individual queries. It is well-established that achieving conditional coverage (i.e., for each query) without assumptions on the data distribution, or with finite calibration data, is provably impossible for conformal methods [16]. While this limits fine-grained control, it is a necessary trade-off for distribution-free guarantees. Crucially, ConANN still provides practical, per-query adaptivity while ensuring global reliability, addressing a gap in the literature where state-of-the-art methods either provide formal guarantees or adapt to data distributions, but not both. Therefore, ConANN represents a step forward in error bounding for ANN systems.

7 RELATED WORK

Error Guarantees for ANN. Auncel [50] is the only known method offering bounded-error guarantees for IVF. It assumes a uniform distribution of result vectors within a hypersphere around the query, computing error by comparing the hypersphere's volume to that of searched clusters. As shown in Section 1, this assumption breaks down on complex, non-uniform datasets (e.g., word embeddings), limiting practical applicability. To date, no other existing method offers explicit recall guarantees for ANN search [14]. Prior work instead focuses on bounding distance errors, targeting primarily latency improvements. Locality-Sensitive Hashing methods [11, 22, 35] follow this trend but suffer from high memory overhead and poor recall in practice [14, 45]. Conceptually related to LSH, ADSampling [18] leverages distance-based random projections and adaptive dimension sampling via hypothesis testing to refine distance estimates, but lacks auto-tuning, hindering real-world usability. RaBitQ [19] proposes a drop-in replacement for PQ [24] with tighter distance bounds, but still does not address recall guarantees directly, as even minor distance approximation errors can lead to poor recall [14]. LEAT [31] predicts p per query to optimize efficiency, introducing iterative probing later used in Auncel and ConANN, though without error bounds. A recent development, the Subspace Collision Framework [49], proposes an ANN search method using random subspace sampling. It scores vectors by their frequency as close neighbors across sampled subspaces, selecting top candidates for re-ranking via exact distances.

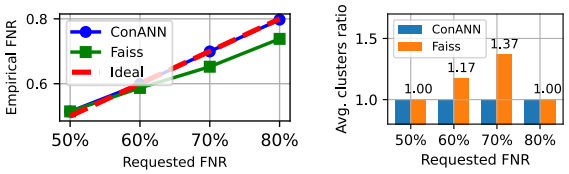

Figure 11: Validity (left) and efficiency (right) results for IVF-PQ on SIFT1M ($k = 100$). ConANN effectively controls the recall of a quantized IVF index.

While offering theoretical guarantees without query sampling, it assumes uniformly distributed distances across dimensions and is incompatible with existing index structures.

Applications of Conformal Methods. Conformal prediction methods have been widely used in applications such as medical diagnosis [34], image classification [5], and natural language processing [15], where reliable uncertainty quantification is critical. Next, a related line of work applies conformal prediction with kNN for regression tasks, as explored in prior research [38]. kNN-based regression compares data points to their neighbors to quantify their dissimilarity. In contrast, our work directly quantifies the approximation error introduced by the approximate kNN search. Lastly, applications of conformal prediction to data-management problems are still rare. dbET introduces a strategy to extend existing DBMS query plan selection methods with conformal prediction [32]. Their work produces execution time interval estimations guaranteed to cover the true execution time with probability $(1 - \alpha)$, which can be later fed into query optimization techniques.

Approximate Query Processing. Approximate query processing (APQ) with error bounds has been an active research area in relational DBMS systems [1, 10, 39]. BlinkDB [2] allows users to trade accuracy for speed on a per-query basis. It uses a sampling strategy that ensures sufficient representation of underrepresented data. The sample size is dynamically adjusted based on user time constraints or error bounds. This concept was extracted into a middleware approach for VerdictDB [39], which uses query rewriting to apply APQ on existing database systems such as SparkSQL. However, these methods are not tailored for vector data.

8 FUTURE WORK

Improved Retrieval with Confidence. A promising direction for future work is the incorporation of confidence-aware retrieval into ConANN. Rather than returning nearest neighbors based solely on vector similarity, ConANN could be extended to output calibrated confidence scores that quantify the likelihood of a result being relevant to a given query. For instance, in IVF-based document retrieval, this approach would allow ConANN to return not only top-ranked candidates but also an associated measure of confidence grounded in prior knowledge. This departs from conventional distance-based methods by enabling the system to differentiate between semantically ambiguous and highly relevant candidates, even if they are similarly distant in the latent embedding space. Such confidence estimates could further support adaptive ranking or filtering strategies, down-weighting or excluding low-confidence results to enhance both the precision and interpretability of retrieval outcomes.

Applicability beyond IVF. ConANN's core components, including the non-conformity score and ranked cluster probing, naturally extend to IVF-based indexing variations. This enables straightforward adaptation within the broader family of IVF methods. Our preliminary experiments on the SIFT1M dataset show that ConANN can effectively control the error rate even when IVF is used in combination with a vector compression method like Product Quantization [24, 26] (IVF-PQ)[3], without incurring additional search overheads, thereby demonstrating its plug-and-play nature. The experiment, depicted in Figure 11, reveals several key observations. First, due to the coarse quantization inherent to IVF-PQ, the minimum achievable FNR is approximately 50%. Additionally, when the requested FNR exceeds 80%, only one cluster needs to be searched. At the 50% FNR level, both ConANN and Faiss search all clusters, as this is the only way to ensure the target recall. As the allowed error increases, ConANN begins to search fewer clusters, mirroring trends observed in our main experiments. At the 80% requested FNR, both ConANN and Faiss search only a single cluster per query. However, ConANN exhibits a distinct behavior in this regime: to satisfy the validity constraint, it abstains from searching on a subset of queries (roughly 800), prioritizing a reliable expected guarantee. The average cluster ratio in Figure 11 excludes these abstentions. These preliminary results suggest that ConANN can generalize to compressed and quantized search settings, which may enable scalability to real-world billion-scale datasets. Nonetheless, the CRC framework is a distribution-free wrapper for any black-box model and is, therefore, not limited to IVF. This opens the possibility of extending the approach to fundamentally different ANN indexes, such as Hierarchical Navigable Small World (HNSW) graphs [36].

Metric-agnostic Design. Moreover, ConANN is distance metric-agnostic, meaning it can be used with diverse distance measures beyond Euclidean distance, such as cosine similarity, Manhattan distance, or even domain-specific metrics. This broad applicability makes ConANN a versatile tool for improving reliability in nearest neighbor search tasks, paving the way for adaptable, efficient, and robust retrieval systems across a wide range of domains.

9 CONCLUSIONS

We presented ConANN, the first distribution-free framework that delivers formal error guarantees for IVF-based Approximate Nearest Neighbor search. Leveraging Conformal Risk Control, ConANN dynamically adapts search effort per query, ensuring rigorous error control without assumptions on data distribution. Empirical results demonstrate that ConANN reliably satisfies target error rates, reduces search costs, and incurs minimal computational overhead. Integrated into the FAISS library, ConANN not only enhances the robustness of ANN systems but also highlights the broader potential of conformal methods in data management tasks beyond traditional predictive models.

ACKNOWLEDGMENTS

This work was supported by the Wallenberg Foundation (WASP-NEST Data-Bound), the Google Cloud Research Credits program, Vinnova (2023-01406), the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research Council through grant agreement no. 2022-06725, and Digital Futures.

[3]The IVF-PQ index was configured with 256 clusters per 8 subvectors.

REFERENCES

[1] Sameer Agarwal, Henry Milner, Ariel Kleiner, Ameet Talwalkar, Michael I. Jordan, Samuel Madden, Barzan Mozafari, and Ion Stoica. 2014. Knowing when you're wrong: building fast and reliable approximate query processing systems. In *International Conference on Management of Data, SIGMOD 2014, Snowbird, UT, USA, June 22-27, 2014*, Curtis E. Dyreson, Feifei Li, and M. Tamer Özsu (Eds.). ACM, 481–492. https://doi.org/10.1145/2588555.2593667

[2] Sameer Agarwal, Barzan Mozafari, Aurojit Panda, Henry Milner, Samuel Madden, and Ion Stoica. 2013. BlinkDB: queries with bounded errors and bounded response times on very large data. In *EuroSys*. ACM, 29–42.

[3] Anastasios N. Angelopoulos and Stephen Bates. 2021. A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification. *CoRR* abs/2107.07511 (2021). arXiv:2107.07511 https://arxiv.org/abs/2107.07511

[4] Anastasios Nikolas Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. 2024. Conformal Risk Control. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net. https://openreview.net/forum?id=33XGfHLtZg

[5] Anastasios Nikolas Angelopoulos, Stephen Bates, Michael I. Jordan, and Jitendra Malik. 2021. Uncertainty Sets for Image Classifiers using Conformal Prediction. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net. https://openreview.net/forum?id=eNdiU_DbM9

[6] Artem Babenko and Victor S. Lempitsky. 2015. The Inverted Multi-Index. *IEEE Trans. Pattern Anal. Mach. Intell.* 37, 6 (2015), 1247–1260. https://doi.org/10.1109/TPAMI.2014.2361319

[7] Rina Foygel Barber, Emmanuel J. Candès, Aaditya Ramdas, and Ryan J. Tibshirani. 2023. Conformal prediction beyond exchangeability. *The Annals of Statistics* 51, 2 (April 2023). https://doi.org/10.1214/23-AOS2276

[8] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomás Mikolov. 2017. Enriching Word Vectors with Subword Information. *Trans. Assoc. Comput. Linguistics* 5 (2017), 135–146. https://doi.org/10.1162/TACL_A_00051

[9] Richard P Brent. 2013. *Algorithms for minimization without derivatives*. Courier Corporation.

[10] Surajit Chaudhuri, Bolin Ding, and Srikanth Kandula. 2017. Approximate Query Processing: No Silver Bullet. In *Proceedings of the 2017 ACM International Conference on Management of Data, SIGMOD Conference 2017, Chicago, IL, USA, May 14-19, 2017*, Semih Salihoglu, Wenchao Zhou, Rada Chirkova, Jun Yang, and Dan Suciu (Eds.). ACM, 511–519. https://doi.org/10.1145/3035918.3056097

[11] Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab S. Mirrokni. 2004. Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the twentieth annual symposium on Computational geometry (SCG '04)*. Association for Computing Machinery, New York, NY, USA, 253–262. https://doi.org/10.1145/997817.997857

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, Jill Burstein, Christy Doran, and Thamar Solorio (Eds.). Association for Computational Linguistics, 4171–4186. https://doi.org/10.18653/V1/N19-1423

[13] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The Faiss library. (2024). arXiv:2401.08281 [cs.LG]

[14] Karima Echihabi, Themis Palpanas, and Kostas Zoumpatianos. 2021. New Trends in High-D Vector Similarity Search: AI-driven, Progressive, and Distributed. *Proc. VLDB Endow.* 14, 12 (2021), 3198–3201. https://doi.org/10.14778/3476311.3476407

[15] Adam Fisch, Tal Schuster, Tommi S. Jaakkola, and Regina Barzilay. 2021. Efficient Conformal Prediction via Cascaded Inference with Expanded Admission. In *International Conference on Learning Representations*. https://openreview.net/forum?id=tnSo6VRLmT

[16] Rina Foygel Barber, Emmanuel J Candes, Aaditya Ramdas, and Ryan J Tibshirani. 2021. The limits of distribution-free conditional predictive inference. *Information and Inference: A Journal of the IMA* 10, 2 (2021), 455–482.

[17] Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016 (JMLR Workshop and Conference Proceedings)*, Maria-Florina Balcan and Kilian Q. Weinberger (Eds.), Vol. 48. JMLR.org, 1050–1059. http://proceedings.mlr.press/v48/gal16.html

[18] Jianyang Gao and Cheng Long. 2023. High-Dimensional Approximate Nearest Neighbor Search: with Reliable and Efficient Distance Comparison Operations. *Proceedings of the ACM on Management of Data* 1, 2 (June 2023), 1–27. https://doi.org/10.1145/3589282

[19] Jianyang Gao and Cheng Long. 2024. RaBitQ: Quantizing High-Dimensional Vectors with a Theoretical Error Bound for Approximate Nearest Neighbor Search. *Proceedings of the ACM on Management of Data* 2, 3 (May 2024), 1–27. https://doi.org/10.1145/3654970

[20] Isaac Gibbs and Emmanuel Candes. 2021. Adaptive Conformal Inference Under Distribution Shift. In *Advances in Neural Information Processing Systems*, Vol. 34. Curran Associates, Inc., 1660–1672. https://proceedings.neurips.cc/paper/2021/hash/0d441de75945e5acbc865406fc9a2559-Abstract.html

[21] Anna Gogolou, Theophanis Tsandilas, Karima Echihabi, Anastasia Bezerianos, and Themis Palpanas. 2020. Data Series Progressive Similarity Search with Probabilistic Quality Guarantees. In *SIGMOD Conference*. ACM, 1857–1873.

[22] Piotr Indyk and Rajeev Motwani. 1998. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing (STOC '98)*. Association for Computing Machinery, New York, NY, USA, 604–613. https://doi.org/10.1145/276698.276876

[23] Hervé Jégou, Matthijs Douze, and Cordelia Schmid. 2010. Improving Bag-of-Features for Large Scale Image Search. *Int. J. Comput. Vis.* 87, 3 (2010), 316–336. https://doi.org/10.1007/S11263-009-0285-2

[24] Hervé Jégou, Matthijs Douze, and Cordelia Schmid. 2011. Product Quantization for Nearest Neighbor Search. *IEEE Trans. Pattern Anal. Mach. Intell.* 33, 1 (2011), 117–128. https://doi.org/10.1109/TPAMI.2010.57

[25] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data* 7, 3 (2019), 535–547.

[26] Hervé Jégou, Romain Tavenard, Matthijs Douze, and Laurent Amsaleg. 2011. Searching in one billion vectors: Re-rank with source coding. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 861–864. https://doi.org/10.1109/ICASSP.2011.5946540 ISSN: 2379-190X.

[27] Yehuda Koren, Robert M. Bell, and Chris Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *Computer* 42, 8 (2009), 30–37. https://doi.org/10.1109/MC.2009.263

[28] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural Questions: a Benchmark for Question Answering Research. *Trans. Assoc. Comput. Linguistics* 7 (2019), 452–466.

[29] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.). 6402–6413. https://proceedings.neurips.cc/paper/2017/hash/9ef2ed4b7fd2c810847ffa5fa85bce38-Abstract.html

[30] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *NeurIPS*.

[31] Conglong Li, Minjia Zhang, David G. Andersen, and Yuxiong He. 2020. Improving Approximate Nearest Neighbor Search through Learned Adaptive Early Termination. In *SIGMOD Conference*. ACM, 2539–2554.

[32] Yifan Li, Xiaohui Yu, Nick Koudas, Shu Lin, Calvin Sun, and Chong Chen. 2023. dbET: Execution Time Distribution-based Plan Selection. *Proc. ACM Manag. Data* 1, 1 (2023), 31:1–31:26. https://doi.org/10.1145/3588711

[33] David G. Lowe. 2004. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* 60, 2 (2004), 91–110. https://doi.org/10.1023/B:VISI.0000029664.99615.94

[34] Charles Lu, Andréanne Lemay, Ken Chang, Katharina Höbel, and Jayashree Kalpathy-Cramer. 2022. Fair Conformal Predictors for Applications in Medical Imaging. In *AAAI*. AAAI Press, 12008–12016.

[35] Qin Lv, William Josephson, Zhe Wang, Moses Charikar, and Kai Li. 2007. Multi-probe LSH: efficient indexing for high-dimensional similarity search. In *Proceedings of the 33rd international conference on Very large data bases (VLDB '07)*. VLDB Endowment, Vienna, Austria, 950–961.

[36] Yury Malkov, Alexander Ponomarenko, Andrey Logvinov, and Vladimir Krylov. 2014. Approximate nearest neighbor algorithm based on navigable small world graphs. *Information Systems* 45 (Sept. 2014), 61–68. https://doi.org/10.1016/j.is.2013.10.006

[37] Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *ICLR (Workshop Poster)*.

[38] H. Papadopoulos, V. Vovk, and A. Gammerman. 2011. Regression Conformal Prediction with Nearest Neighbours. *Journal of Artificial Intelligence Research* 40 (April 2011), 815–840. https://doi.org/10.1613/jair.3198

[39] Yongjoo Park, Barzan Mozafari, Joseph Sorenson, and Junhao Wang. 2018. VerdictDB: Universalizing Approximate Query Processing. In *SIGMOD Conference*. ACM, 1461–1476.

[40] Yun Peng, Byron Choi, Tsz Nam Chan, Jianye Yang, and Jianliang Xu. 2023. Efficient Approximate Nearest Neighbor Search in Multi-dimensional Databases. *Proc. ACM Manag. Data* 1, 1 (2023), 54:1–54:27. https://doi.org/10.1145/3588908

[41] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1, 8 (2019), 9.

[42] Yandex Research. 2020. Benchmarks for Billion-Scale Similarity Search. https://research.yandex.com/blog/benchmarks-for-billion-scale-similarity-search. Accessed: 2024-12-19.

[43] Glenn Shafer and Vladimir Vovk. 2008. A Tutorial on Conformal Prediction. *J. Mach. Learn. Res.* 9 (2008), 371–421. https://doi.org/10.5555/1390681.1390693

[44] Sivic and Zisserman. 2003. Video Google: a text retrieval approach to object matching in videos. In *Proceedings Ninth IEEE International Conference on Computer Vision*. 1470–1477 vol.2. https://doi.org/10.1109/ICCV.2003.1238663

[45] Yufei Tao, Ke Yi, Cheng Sheng, and Panos Kalnis. 2009. Quality and efficiency in high dimensional nearest neighbor search. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data (SIGMOD '09)*. Association for Computing Machinery, New York, NY, USA, 563–576. https://doi.org/10.1145/1559845.1559905

[46] Ryan J Tibshirani, Rina Foygel Barber, Emmanuel Candes, and Aaditya Ramdas. 2019. Conformal Prediction Under Covariate Shift. In *Advances in Neural Information Processing Systems*, Vol. 32. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2019/hash/8fb21ee7a2207526da55a679f0332de2-Abstract.html

[47] Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. 2022. *Algorithmic Learning in a Random World*. Springer Nature.

[48] Chuangxian Wei, Bin Wu, Sheng Wang, Renjie Lou, Chaoqun Zhan, Feifei Li, and Yuanzhe Cai. 2020. AnalyticDB-V: A Hybrid Analytical Engine Towards Query Fusion for Structured and Unstructured Data. *Proc. VLDB Endow.* 13, 12 (2020), 3152–3165. https://doi.org/10.14778/3415478.3415541

[49] Jiuqi Wei, Xiaodong Lee, Zhenyu Liao, Themis Palpanas, and Botao Peng. 2025. Subspace Collision: An Efficient and Accurate Framework for High-dimensional Approximate Nearest Neighbor Search. *Proceedings of the ACM on Management of Data* 3, 1 (2025), 1–29.

[50] Zili Zhang, Chao Jin, Linpeng Tang, Xuanzhe Liu, and Xin Jin. 2023. Fast, Approximate Vector Queries on Very Large Unstructured Datasets. In *NSDI*. USENIX Association, 995–1011.