XDecomposer: Learning Prior-Free Set Decomposition for Multiphase X-ray Diffraction

Hanyu Gao[1,2,*], **Bin Cao**[3,4,*,†], **Yunyue Su**[1], **Tong-Yi Zhang**[3], **Qiang Liu**[1,†]
[1]New Laboratory of Pattern Recognition (NLPR), Institute of Automation,
Chinese Academy of Sciences (CASIA)
[2]School of Advanced Interdisciplinary Sciences,
University of Chinese Academy of Sciences
[3]Guangzhou Municipal Key Laboratory of Materials Informatics,
The Hong Kong University of Science and Technology (Guangzhou)
[4]AI Lab, The Yangtze River Delta.

Abstract

Multiphase powder X-ray diffraction (PXRD) analysis remains a fundamental bottleneck in structure identification, as real-world synthesis often produces complex mixtures whose constituent phases (components) cannot be reliably disentangled. While recent advances in representation-based crystal retrieval and generation suggest the possibility of inferring structures directly from PXRD, existing approaches largely assume single-phase inputs and break down in multiphase settings. Here, we present XDecomposer, a prior-free framework for joint decomposition and identification of multiphase XRD patterns without requiring candidate phase lists, structural templates, or prior knowledge of phase number. We formulate multiphase diffraction analysis as a set prediction problem, where the model infers an unordered set of phase-resolved components, their mixture proportions, and corresponding structural representations within a unified architecture. A phase-query-driven decomposition mechanism, together with diffraction-consistent physical reconstruction, enables accurate source separation while preserving crystallographic fidelity. Extensive experiments on both simulated and experimental datasets show that XDecomposer substantially improves reconstruction accuracy and phase identification across diverse chemical systems, while maintaining strong generalization to unseen mixtures. These results provide a practical route toward data-driven, source-resolved multiphase XRD analysis and reduce long-standing dependence on prior-guided iteratively phase matching. The code is openly available at `https://github.com/Licht0812/XDecomposer`

1 Introduction

Powder X-ray diffraction (PXRD) is a foundational technique for determining crystal structure [1–4]. In a typical PXRD workflow, phase identification provides the basis for downstream structural refinement and quantitative analysis. Widely used refinement methods, such as Rietveld [5], Pawley [6], and Le Bail [7], generally assume that the constituent phases are known a priori; when this assumption is violated, subsequent analyses can become unreliable. Phase identification is relatively well established for single-phase samples [8, 9]. However, in practice, samples often contain multiple crystalline phases whose diffraction signals are superimposed in a single one-dimensional pattern. In such multiphase settings, the goal is no longer to identify a single phase, but to simultaneously

*Equal contribution.
†Corresponding authors: Bin Cao (bincao@hkust-gz.edu.cn) and Qiang Liu (qiang.liu@nlpr.ia.ac.cn).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

determine all constituent phases and estimate their relative abundances. Unknown compositions, severe peak overlap, and experimental distortions further complicate this task [10–13]. As a result, multiphase phase identification has become a central bottleneck in PXRD analysis.

From a machine learning perspective, single-phase identification can be cast as mapping a one-dimensional PXRD pattern to a single phase or structural representation. This setting has been widely studied using classification [9–11, 14–17], retrieval [18–21], and generative modeling [22–25]. By contrast, multiphase analysis is not simply a multi-label extension of single-phase identification. A mixed pattern is generated by multiple latent diffraction components whose identities, number, and proportions are generally unknown. This makes the task more naturally formulated as a single-channel blind source separation (BSS) problem: given a mixed PXRD pattern, a model must jointly identify the constituent phases, reconstruct the diffraction signal of each phase, and estimate their relative proportions. This formulation highlights the underdetermined nature of the problem, in which multiple unobserved components must be recovered from a single observed signal, often without component-level supervision (see Appendix A.3).

A natural approach is to exploit the additive structure of diffraction signals. Existing methods therefore adopt a sequential identification paradigm [12, 26]: they iteratively identify a dominant phase, subtract its contribution, and repeat on the residual. However, this strategy suffers from a fundamental limitation: phase proportions are unknown, making each subtraction inherently inaccurate. Moreover, peak overlap causes subtraction to distort signals from other phases, leading to error accumulation across iterations. As a result, even strong single-phase models often fail in multiphase settings, revealing the inadequacy of sequential identification and motivating the need for a new paradigm.

Here, we formulate prior-free multiphase XRD analysis as a blind source separation problem, in which a model directly decomposes a mixed diffraction pattern into an unordered set of single-phase components and their proportions. Existing approaches either rely on predefined phase sets or sequential subtraction, which limits their ability to jointly recover all components. To this end, we propose XDecomposer for direct whole-pattern decomposition. It is trained in two stages: self-supervised pre-training on large-scale single-phase data to learn general diffraction patterns, followed by supervised training on multiphase mixtures for component-level decomposition. XDecomposer integrates convolutional encoding for local peak features, Transformer-based global context modeling, a query-based mechanism to separate individual phases, and physically constrained reconstruction. Experiments on simulated and real-word experimental datasets show that XDecomposer accurately decomposes mixtures, reconstructs phase-specific signals, and estimates phase proportions, establishing a new perspective for multiphase PXRD analysis. The contributions are as follows:

- **Problem formulation.** We cast multiphase PXRD analysis as a permutation-invariant set prediction problem under unknown phase cardinality, providing an alternative to sequential identification and enabling direct, prior-free decomposition of mixture patterns.
- **Modeling approach.** We introduce XDecomposer, a query-based decomposition model that combines global context modeling with phase-conditioned latent modulation to separate phase-specific components from a single mixed diffraction pattern.
- **Physics-aware reconstruction.** We incorporate physically motivated constraints, including non-negativity and mixture consistency, through a mask-based reconstruction mechanism, improving stability and plausibility of the recovered components.
- **Empirical evaluation.** We conduct extensive experiments on both simulated and real-world datasets, showing that XDecomposer consistently outperforms strong baselines across multiple metrics and exhibits robust generalization to unseen mixtures.

2 Related Work

Query strategy For single-phase materials, identification typically matches measured diffraction patterns to reference entries, such as the Powder Diffraction File, using peak positions and relative intensities [26, 4] (see Appendix C.1). For multiphase systems, the observed pattern is treated as a superposition of phase signals and analyzed sequentially by identifying a dominant phase, subtracting its contribution, and repeating on the residual. Such methods therefore cannot jointly infer all mixture

components and, despite their effectiveness in controlled settings, depend strongly on candidate-phase priors and are sensitive to peak overlap, noise, and incomplete or biased reference databases.

Learning Strategies Recent advances leverage machine learning to automate phase identification from PXRD patterns. For single-phase materials, the problem is well-studied and typically formulated as classification, retrieval, or generation. Classification methods formulate phase identification as a supervised multi-class classification problem, directly assigning an input diffraction pattern to one of the predefined phase labels in the training database [9–11, 14–17]. Retrieval-based approaches, in contrast, learn feature representations that enable similarity search, ranking candidate phases by matching a query pattern against a database of known phases rather than directly predicting a class label [18–21]. Generation methods model the inverse diffraction process, reconstructing crystal structures from PXRD patterns to facilitate phase identification [22–25].

In contrast, multiphase analysis remains underexplored. Existing methods typically extend these formulations to multi-label classification [15, 16, 27, 28] or regression [29, 30], often with fixed candidate phase sets, or adopt sequential strategies that iteratively identify and subtract phase contributions from the mixture [12].

However, these methods remain fundamentally limited: predefined phase sets and labeled data restrict generalization to unseen mixtures; sequential subtraction accumulates errors under severe peak overlap; and generation-based methods may produce physically inconsistent structures. Consequently, current approaches struggle to reliably disentangle and recover individual phase components from multiphase PXRD patterns.

Blind Source Separation We relate multiphase PXRD decomposition to BSS, set prediction, and self-supervised representation learning. In BSS, permutation invariant training (PIT) addresses output permutation ambiguity [31, 32], while Conv-TasNet [33] and SepFormer [34] demonstrate the effectiveness of mask-based reconstruction and Transformer-based long-range modeling. In set prediction, Deep Sets [35], Set Transformer [36], and DETR [37] establish permutation-invariant and query-based prediction paradigms. In representation learning, Transformer [38], Masked Autoencoder [39], and FiLM [40] support global modeling, self-supervised pretraining, and conditional feature modulation. However, multiphase PXRD decomposition is not a generic source separation problem: diffraction patterns are governed by crystal-structure-dependent peak positions and intensities constrained by Bragg's law, while severe peak overlap makes phase-wise disentanglement highly ill-posed. Moreover, the target is an unordered set of constituent phases with variable cardinality rather than a fixed set of sources. Existing BSS and set prediction methods do not explicitly model these physics-driven constraints, limiting their effectiveness for physically consistent PXRD decomposition.

In summary, existing approaches, whether database-driven, learning-based, or adapted from generic BSS frameworks, fail to provide a prior-free, physically consistent, and unified solution for jointly decomposing and identifying multiphase PXRD patterns.

3 Preliminaries

3.1 Phase Identification

In X-ray diffraction analysis, diffraction patterns encode crystal structures through peak positions and intensities. For a single-phase material, the observed pattern can be approximated as a deterministic function of its crystal structure, including lattice parameters, atomic positions, and chemical elements. In practice, extrinsic factors such as sample quality, preferred orientation, and instrument conditions introduce distortions and reduce phase distinguishability (see Appendix A.3). A single-phase pattern can generally be represented as a function $x(\theta)$ over the diffraction angle θ.

For multiphase materials, the observed pattern is a superposition of single-phase signals governed by diffraction physics [41]. Given K constituent phases, the measured pattern $x(\theta)$ can be expressed as

$$x(\theta) = \sum_{k=1}^{K} w_k \, y_k(\theta) + \epsilon(\theta), \tag{1}$$

where $y_k(\theta)$ denotes the diffraction pattern of the k-th phase, w_k is its mixture coefficient, and $\epsilon(\theta)$ accounts for noise and trace impurities, whose contribution is negligible in multiphase decomposition. Recovering both $\{y_k\}_{k=1}^K$ and $\{w_k\}_{k=1}^K$ from a single observation $x(\theta)$ is the inherent mission of multiphase identification. Existing approaches typically restrict the solution space to predefined candidate phases [15, 16, 28] and rely on iterative fitting [12, 42]. In contrast, we address the realistic prior-free setting, where both the number of phases and their diffraction components are unknown and must be inferred directly from the observed mixture.

Novel crystal structure determination from multiphase diffraction data requires two sequential stages: mixture decomposition and crystal structure identification. The latter comprises both in-library and out-of-library settings (Appendix C.2). Combined with database matching, XDecomposer completes the full pipeline for in-library identification, whereas out-of-library identification requires integration with generative crystal structure prediction methods and remains an open research problem.

3.2 Problem Definition

Let the input pattern be sampled on a discrete 2θ grid of length L. After standard preprocessing, the mixed diffraction pattern is decomposed as

$$\mathcal{D}\big(x(\theta)\big) = \{\hat{y}_k\}_{k=1}^K, \qquad \text{subject to} \qquad x(\theta) \approx \sum_{k=1}^K \hat{y}_k, \quad 1 \le K \le K_{\max}, \tag{2}$$

where $\mathcal{D}(\cdot)$ maps the mixture to a set of decomposed single-phase contributions. Each $\hat{y}_k \approx w_k\, y_k(\theta) \in \mathbb{R}_+^L$ denotes the estimated contribution of the k-th phase, incorporating realistic experimental effects and phase proportions. Each component is assigned a confidence score p_k, and active phases are selected by thresholding with τ, yielding $\hat{\mathcal{Y}}^* = \{\hat{y}_k \mid p_k > \tau\}$.

3.3 Data Synthesis

Under standard diffraction conditions, a mixed PXRD pattern can be approximated as a linear superposition of its constituent phase signals, as shown in Eq. 1. We first construct simulated and experimental single-phase PXRD libraries, and then generate mixed PXRD data using a multiphase mixing strategy. Details of single-phase data generation and processing are given in Appendices B.2 and B.3, and the mixing strategy is described in Appendix B.4.

Simulated data: We parse 100,315 crystal structures from the Materials Project database [43] (MP500) and generate 20 single-phase PXRD patterns per structure using PySimXRD [10], yielding $100{,}315 \times 20$ simulated patterns. These patterns are then mixed with phase fractions $\{w_k\}$ and normalized to $\max_\theta x(\theta) = 1$. Simulations are performed on a high-performance computing cluster with perturbations in peak position, peak width, background, and instrumental geometry.

Experimental data: Real single-phase patterns are first corrected by WPEM [44] to remove baseline and background, and are then mixed using controllable phase fractions. The mixtures are generated on the fly during data loading to increase sample diversity.

4 Methodology

4.1 Overall Framework

We formulate prior-free multiphase XRD decomposition as a set-prediction-based one-dimensional BSS problem. XDecomposer uses a pretrained global bottleneck to capture transferable peak structure and is trained end-to-end with permutation-invariant separation loss, slot-activity supervision, and mixture-consistency regularization (Fig. 1). It enables one-shot inference directly from mixed patterns without candidate phase lists, structural templates, or database retrieval. Given a preprocessed mixed diffraction pattern $x \in \mathbb{R}_+^L$, the model predicts a fixed-size set of $K_{\max} = 4$ output slots,

$$\hat{\mathcal{Y}} = \{(\hat{y}_k, p_k)\}_{k=1}^{K_{\max}},$$

where $\hat{y}_k \in \mathbb{R}_+^L$ denotes the k-th candidate contribution and $p_k \in [0, 1]$ represents its activity probability. During training, the input is a multiphase mixture x, and the supervision target is the set

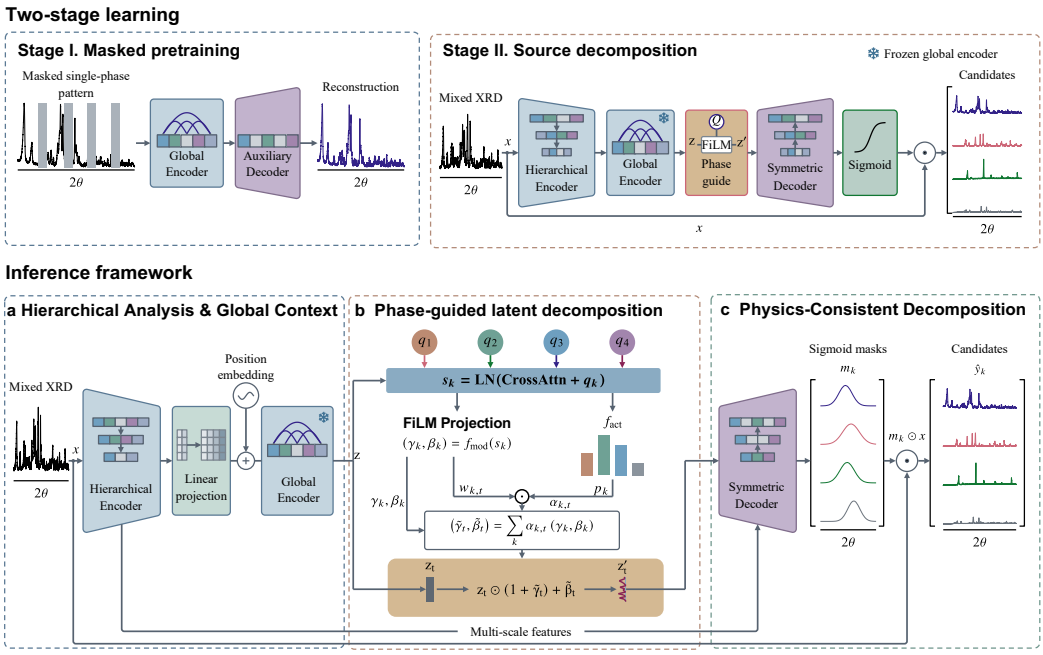


Figure 1: Overall framework of XDecomposer. **Part I** shows the two-stage training strategy: masked-reconstruction pretraining of the global-context encoder $G(\cdot)$ on single-phase patterns, followed by mixture training with $G(\cdot)$ frozen. **Part II** presents the inference framework including **a**, hierarchical analysis and global context modeling; **b**, phase-guided latent decomposition; and **c**, physics-consistent reconstruction.

of constituent single-phase components, zero-padded to four slots when fewer than four phases are present. During inference, inactive slots are removed according to their activity scores, yielding a variable-size set of active components.

4.2 Module

Hierarchical analysis and global context modeling. As shown in Fig. 1**a**, the input mixed diffraction pattern is first processed by a hierarchical one-dimensional convolutional analyzer $E(\cdot)$ to extract multi-scale local spectral features, yielding a latent representation $h = E(x) \in \mathbb{R}^{T\times D}$, where T is the latent sequence length and D is the feature dimension. The shallow convolutional layers capture local peak shapes, peak-width variations, and intensity perturbations. The deeper layers enlarge the receptive field and encode mid-range contextual relations among neighboring peak groups. Intermediate-scale features are further retained and fused during reconstruction to preserve the resolution of peak positions and line shapes.

The latent features h are then projected to the target dimension, combined with positional encodings, and passed to a global context encoder $G(\cdot)$, which produces a shared latent representation $z = G(h) \in \mathbb{R}^{T\times D}$ This global context modeling is motivated by the observation that multiple Bragg reflections from the same phase exhibit long-range dependencies in both peak positions and relative intensities. These dependencies are jointly governed by structure factors and symmetry [45, 46]. Consequently, purely local convolutions are insufficient to recover phase-consistent structural patterns.

Phase-guided latent decomposition. Fig. 1**b** illustrates the phase-guided latent decomposition module, which decomposes the shared latent space into phase-conditioned responses. We introduce a set of learnable phase queries $Q = \{q_k\}_{k=1}^{K_{\max}}$. For each slot k, q_k attends to the shared latent sequence z through cross-attention:

$$s_k = \mathrm{CrossAttn}(q_k, z, z), \qquad k = 1, \ldots, K_{\max}. \tag{3}$$

The slot summary s_k is projected to an activity logit $a_k = f_{\mathrm{act}}(s_k)$ and FiLM modulation parameters $(\gamma_k, \beta_k) = f_{\mathrm{mod}}(s_k)$, with activity probability $p_k = \sigma(a_k)$.

To obtain stable and complementary slot assignments, we combine position-wise spatial competition with sample-level activity gating. At latent position t, the spatial competition weight is computed from the similarity between the slot modulation vector γ_k and the local representation z_t:

$$w_{k,t} = \frac{\exp\big(\langle \gamma_k, z_t \rangle / \sqrt{D}\big)}{\sum_{j=1}^{K_{\max}} \exp\big(\langle \gamma_j, z_t \rangle / \sqrt{D}\big)}, \tag{4}$$

where $\sum_k w_{k,t} = 1$. The effective slot weight is then defined as $\alpha_{k,t} = w_{k,t} p_k$, where $w_{k,t}$ enforces local competition and p_k controls whether slot k is active for the current sample.

The effective weights are used to aggregate slot-specific FiLM parameters [40] into spatially adaptive modulation terms:

$$\tilde{\gamma}_t = \sum_k \alpha_{k,t} \gamma_k, \qquad \tilde{\beta}_t = \sum_k \alpha_{k,t} \beta_k. \tag{5}$$

The shared latent representation is then modulated by feature-wise affine transformation:

$$z'_t = z_t \odot \big(1 + \tilde{\gamma}_t\big) + \tilde{\beta}_t \tag{6}$$

Equivalently, the modulation can be viewed as a weighted combination of slot-conditioned responses around the shared latent feature, where each slot response is defined as $z_t^{(k)} = z_t \odot (1 + \gamma_k) + \beta_k$. This formulation performs phase decomposition directly in the shared latent space, before decoding, while preserving the feature-wise affine structure of FiLM [40]. The aggregated representation z' is then passed to the reconstruction module together with the retained multi-scale features.

Physics-consistent reconstruction. Fig. 1**c** shows the physics-consistent reconstruction module. The decoder takes the modulated latent representation z' together with the retained multi-scale features and predicts $K_{\max}$ soft masks over the full angular domain. Denoting the decoder output by $U = D(z', \mathcal{S}) \in \mathbb{R}^{K_{\max} \times L}$, where $\mathcal{S}$ represents the retained multi-scale features, the soft mask associated with slot k is defined as $m_k = \sigma(U_k) \in (0, 1)^L$. The model then obtains the corresponding single-phase contribution by point-wise multiplication:

$$\hat{y}_k = m_k \odot x. \tag{7}$$

This mask-based parameterization is similar in form to mask-based separation methods in single-channel source separation. In the present setting, however, it is further constrained by the physical properties of the diffraction pattern, especially non-negativity and mixture consistency [33, 34]. Since the input diffraction pattern is non-negative and $m_k \in (0, 1)$, the model naturally satisfies $0 \le \hat{y}_k \le x$. As a result, each output remains non-negative and cannot exceed the input intensity at any position. This property helps suppress spurious peak responses, especially in low-background regions. The reconstructed mixed pattern is obtained by summing all candidate single-phase contributions, i.e., $\hat{x} = \sum_{k=1}^{K_{\max}} \hat{y}_k$. In this way, part of the mixture-consistency constraint is built directly into the output parameterization, rather than being enforced only through a posterior loss term.

4.3 Training and Inference

XDecomposer adopts a two-stage training strategy. **In Stage I**, an MAE-style masked reconstruction task [39] is constructed on single-phase diffraction patterns to pretrain the global-context encoder $G(\cdot)$ for transferable pure-phase structural representations. An auxiliary decoder, used only during pretraining, provides reconstruction supervision.**In Stage II**, the pretrained global-context encoder is transferred to the full decomposition model and kept frozen, while only the decomposition-related modules are optimized. This allows the model to learn single-phase separation, slot activity prediction, and mixture consistency from multiphase mixtures. Since the output slots are unordered, decomposition training uses permutation-invariant training (PIT) [31, 32]. Let $\{y_k\}_{k=1}^{K}$ denote the ground-truth single-phase contributions and $\{\hat{y}_k\}_{k=1}^{K_{\max}}$ the predicted set. The optimal assignment is

$$\pi^* = \arg\min_{\pi} \sum_{k=1}^{K} \mathcal{L}_{\text{sep}}(\hat{y}_{\pi(k)}, y_k). \tag{8}$$

Based on this assignment, the overall training objective is

$$\mathcal{L} = \mathcal{L}_{\text{sep}}^{\text{PIT}} + \lambda_{\text{act}} \mathcal{L}_{\text{act}} + \lambda_{\text{mix}} \mathcal{L}_{\text{mix}}, \tag{9}$$

where $\mathcal{L}_{\text{sep}}^{\text{PIT}}$ is the PIT-aligned separation loss, $\mathcal{L}_{\text{act}}$ supervises slot activity, and $\mathcal{L}_{\text{mix}}$ enforces consistency between the reconstructed mixture and the input pattern. The separation loss includes amplitude, shape, and geometric terms, with full definitions provided in Appendix A.1. **At inference**, XDecomposer decomposes the input pattern in a single forward pass and outputs separated components with activity scores.

5 Experiments

5.1 Benchmarks

Datasets Two large-scale benchmarks are employed in this work. **Simulated data** use the corpus described in Section 3.3. The dataset is split into training, validation, and test sets with an 8:1:1 ratio, ensuring no structural overlap across subsets (see Appendices B.4 and B.5). **Experimental data** consist of 662 measured single-phase PXRD patterns from the RRUFF database [47]. After baseline correction and background removal, mixed patterns are generated with controllable phase fractions. Model performance is evaluated using five-fold cross-validation (see Appendix B.3).

Baselines While related methods exist in BSS and PXRD analysis, none directly address prior-free, physically consistent decomposition under unknown phase cardinality (see Section 2). To ensure comprehensive evaluation, we include both domain-specific PXRD models and strong general-purpose sequence models that have demonstrated competitive performance on diffraction data.

- **Domain Models**: We include three representative domain-specific models developed for PXRD phase identification. XQueryer [9] is the first cross-chemical general model for single-phase identification based on classification. XRD_Proportion_Inference [48, 49] employs Vision Transformer (ViT) and CNN architectures to predict phase proportions, effectively performing regression over mixture ratios for a fixed set of candidate phases. XRDAutoAnalyzer [17, 50] is a classification-based model for single-phase identification; its multi-phase extension adopts a sequential phase identification strategy.
- **Sequence Models**: We further include strong general-purpose sequence models, including Transformer [38], iTransformer [51], and PatchTST [52]. These models have shown competitive performance on PXRD-related tasks [9, 10] and represent a class of flexible, domain-agnostic approaches capable of modeling complex diffraction patterns.

Baseline configurations differ between simulated and experimental settings. For simulated data, we report all baselines. For experimental data, XRDAutoAnalyzer is excluded because it is limited to fixed mixture component sets and its performance collapses on the RRUFF experimental dataset. We evaluate all models under a unified protocol to assess the extent to which their outputs can recover single-phase representations from mixture observations. Implementation details of all baselines are provided in Appendix B.6. Model size, inference cost, and deployment overhead are summarized in Table 4 of Appendix A.2.

Evaluation Metrics We report five evaluation metrics: the Pearson correlation coefficient (unitless), mean peak-position deviation $\overline{\Delta 2\theta}$ (in $^\circ$), full width at half maximum error $\Delta\mathrm{FWHM}$ (in $^\circ$), Top-1 retrieval accuracy (%), and Top-10 retrieval accuracy (%). These metrics evaluate global pattern-shape similarity, local peak-position accuracy, line-width fidelity, and downstream phase retrieval discriminability, respectively. Formal definitions are provided in Appendix B.7.

5.2 Results

Case Study Figure 2 presents a representative four-phase whole pattern decomposition example. XDecomposer accurately reconstructs the overall diffraction pattern while correctly identifying the contribution of each source phase, even in the most challenging cases with severe peak broadening. Additional quantitative cases for binary, ternary, and quaternary mixtures are provided in Appendix B.11, Figs. 6, 7, and 8. These results empirically indicate that XDecomposer achieves accurate whole-pattern decomposition with **near-refinement-level quality** [44].

Simulated Data Table 1 reports results on simulated data for $K = 2, 3, 4$. XDecomposer achieves the best overall performance across all settings. Among methods supporting Top-k retrieval evaluation,

Table 1: Baseline comparison on simulated PXRD mixtures with $K = 2, 3, 4$.

K	Method Type	Model	Pearson ↑	Peak Shift (°) ↓	ΔFWHM (°) ↓	Top-1 (%) ↑	Top-10 (%) ↑
2	Domain	XQueryer	0.6873	0.0363	5.0005	N/A	N/A
		XRD_Proportion_Inference	0.6330	0.0742	10.7957	30.44	43.90
		XRDAutoAnalyzer	0.0135	0.0897	27.5939	1.62	2.43
	Sequence	Transformer	0.9486	0.0322	3.1322	76.43	88.17
		iTransformer	0.9291	0.0347	3.6935	74.25	86.42
		PatchTST	0.9278	0.0336	3.3702	75.29	86.74
	Proposed	**XDecomposer**	**0.9691**	**0.0272**	**2.4330**	**87.92**	**92.50**
3	Domain	XQueryer	0.5943	0.0501	8.6083	N/A	N/A
		XRD_Proportion_Inference	0.5274	0.0773	12.2785	13.95	23.35
		XRDAutoAnalyzer	0.0043	0.0900	26.6406	0.52	0.81
	Sequence	Transformer	0.8608	0.0414	4.4464	64.60	78.62
		iTransformer	0.7949	0.0444	4.9595	58.19	72.45
		PatchTST	0.8010	0.0429	4.4322	60.21	73.94
	Proposed	**XDecomposer**	**0.9101**	**0.0389**	**3.7836**	**81.13**	**86.81**
4	Domain	XQueryer	0.5372	0.0612	12.9371	N/A	N/A
		XRD_Proportion_Inference	0.4092	0.0799	13.2168	5.68	11.25
		XRDAutoAnalyzer	0.0011	0.0909	25.5976	0.14	0.27
	Sequence	Transformer	0.7491	0.0499	5.7736	48.41	62.13
		iTransformer	0.7084	0.0527	6.4041	40.54	53.52
		PatchTST	0.7146	0.0519	5.7424	42.66	55.81
	Proposed	**XDecomposer**	**0.8322**	**0.0480**	**5.2410**	**69.68**	**76.21**

Note: Top-k measures whether the ground-truth phase appears within the top k retrieved candidates at the phase level. XQueryer is not included in the Top-k comparison because its framework does not support outputting retrieval results required for computing Top-k scores. The profile factors, R_p and R_{wp}, are reported in Table 9.

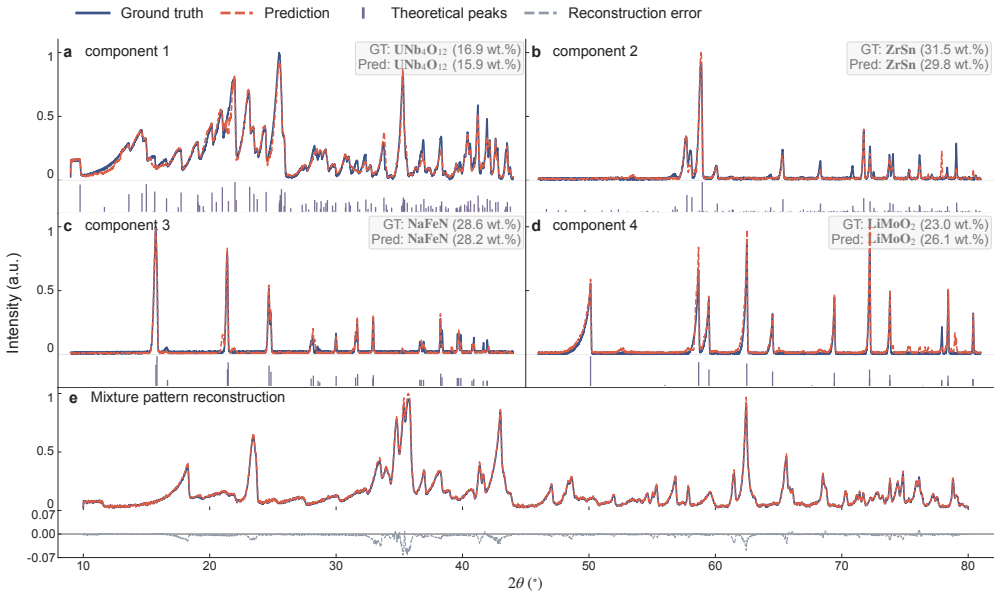


Figure 2: Representative quantitative results. **a**–**d**: true single-phase contributions, predicted patterns, and theoretical peak positions. **e**: mixture reconstruction consistency.

it consistently outperforms baselines in Pearson correlation, Peak Shift, ΔFWHM, Top-1, and Top-10, indicating more accurate recovery of overall shape, peak positions, and line profiles, and consequently better downstream phase retrieval.

As K increases from 2 to 4, all methods degrade due to stronger peak overlap and greater phase ambiguity. In contrast, XDecomposer degrades more gracefully, showing greater robustness in complex multiphase scenarios due to large-scale XRD pretraining. Although prior domain-specific models remain competitive with general sequence models on simulated data, their strong single-phase priors limit multiphase decomposition, especially in spectral fidelity, peak-geometry consistency, and retrieval accuracy.

Table 2: Baseline comparison on RRUFF real data with $K = 2, 3, 4$ (five-fold cross-validation).

K	Method Type	Model	Pearson ↑	Peak Shift (°) ↓	ΔFWHM (°) ↓	Top-1 (%) ↑	Top-10 (%) ↑
2	Domain	XQueryer	0.6519 ± 0.0244	**0.0164 ± 0.0006**	**1.0097 ± 0.0842**	N/A	N/A
		XRD_Proportion_Inference	0.4808 ± 0.1448	0.0652 ± 0.0160	17.2678 ± 9.2924	44.60 ± 18.44	75.29 ± 19.06
	Sequence	Transformer	0.5117 ± 0.0114	0.0245 ± 0.0011	2.7504 ± 0.2616	40.68 ± 1.66	66.80 ± 2.32
		iTransformer	0.5984 ± 0.0150	0.0221 ± 0.0027	3.3806 ± 0.5232	49.34 ± 3.28	89.30 ± 4.36
		PatchTST	0.5544 ± 0.0099	0.0183 ± 0.0014	2.7112 ± 0.4100	49.08 ± 0.89	76.75 ± 1.62
	Proposed	XDecomposer	**0.6956 ± 0.0091**	0.0197 ± 0.0018	1.4538 ± 0.1293	**65.31 ± 2.62**	**97.51 ± 0.67**
3	Domain	XQueryer	0.5294 ± 0.0012	**0.0221 ± 0.0019**	**1.4311 ± 0.2081**	N/A	N/A
		XRD_Proportion_Inference	0.3769 ± 0.1326	0.0695 ± 0.0137	18.1328 ± 8.4472	31.29 ± 14.22	63.61 ± 21.25
	Sequence	Transformer	0.3304 ± 0.0134	0.0302 ± 0.0021	3.3232 ± 0.3127	26.49 ± 3.15	48.84 ± 2.57
		iTransformer	0.3961 ± 0.0063	0.0279 ± 0.0018	4.0545 ± 0.5033	34.42 ± 1.41	64.65 ± 0.76
		PatchTST	0.3856 ± 0.0061	0.0240 ± 0.0015	3.1564 ± 0.4406	35.02 ± 1.92	61.78 ± 1.07
	Proposed	XDecomposer	**0.5821 ± 0.0070**	0.0286 ± 0.0015	2.0912 ± 0.0545	**53.49 ± 1.48**	**92.96 ± 1.38**
4	Domain	XQueryer	0.4651 ± 0.0017	**0.0277 ± 0.0017**	**1.8599 ± 0.2528**	N/A	N/A
		XRD_Proportion_Inference	0.3158 ± 0.1249	0.0733 ± 0.0115	19.3598 ± 7.6530	23.47 ± 11.26	55.58 ± 20.95
	Sequence	Transformer	0.2461 ± 0.0097	0.0342 ± 0.0017	3.9567 ± 0.3040	21.02 ± 1.92	38.86 ± 2.28
		iTransformer	0.2834 ± 0.0107	0.0328 ± 0.0016	4.7631 ± 0.4952	25.86 ± 2.41	50.34 ± 1.02
		PatchTST	0.2806 ± 0.0090	0.0298 ± 0.0015	3.8275 ± 0.4475	26.25 ± 1.59	49.74 ± 1.72
	Proposed	XDecomposer	**0.5009 ± 0.0074**	0.0359 ± 0.0018	2.6735 ± 0.1577	**41.90 ± 1.74**	**87.36 ± 1.80**

Note: Results are reported as mean ± standard deviation over five folds. Top-1 and Top-10 are reported only for methods that provide retrieval outputs; otherwise the entries are marked as N/A.

Experimental Data We further evaluate generalization on the real-world RRUFF dataset, reporting all metrics as mean ± standard deviation over five-fold cross-validation. As shown in Table 2, all methods drop substantially on experimental data, indicating a clear sim-to-real gap, as multiphase modeling is more fragile since all constituent phases jointly contribute and single-phase variations can significantly influence the overall multiphase pattern, as discussed in Appendices B.10 and A.3.

Baselines show different strengths on experimental data. XQueryer achieves the best Peak Shift and ΔFWHM, indicating better preservation of local peak positions and widths, but this local advantage does not translate into better overall decomposition or retrieval performance. Pearson correlation and Top-k results show that existing domain-specific and general sequence models tend to absorb noise, background variations, and local measurement effects into their predictions, thereby degrading decomposition quality. This confirms that multiphase decomposition differs from single-phase identification, as it requires jointly modeling multiple component contributions rather than merely aligning local peaks. In contrast, XDecomposer achieves superior or competitive results across metrics and maintains relatively small standard deviations under five-fold cross-validation, demonstrating its robustness and generalization on real experimental data.

Ablation Study. We conduct ablation studies on simulated data with $K = 4$ to evaluate the key design components of XDecomposer under challenging multiphase conditions. We consider four structural ablations by removing multi-scale skip fusion, the geometry constraint, phase-guided modulation, or the global-context encoder $G(\cdot)$. Two alternative reconstruction strategies are also evaluated. Direct regression removes the mixture-aware masking constraint and directly predicts the component patterns, whereas hard-mask reconstruction binarizes the continuous assignment weights in overlapping regions. We further perform ablation experiments to quantify the effects of self-supervised pretraining, individual loss components, and loss weighting strategies.

Table 3: Ablation results on $K = 4$.

Variant	Pearson↑	Peak Shift↓	ΔFWHM↓	Top-1↑
Full XDecomposer	**0.8307**	**0.0486**	**5.2546**	**69.76**
w/o multi-scale skip fusion	0.4649	0.0635	17.7708	11.58
w/o geometry constraint	0.4273	0.0709	11.9110	15.24
w/o phase-guided modulation	0.4197	0.0744	12.2133	15.41
w/o global-context encoder $G(\cdot)$	0.4124	0.0813	13.5547	12.30
direct output	0.3439	0.0682	16.3048	14.24
hard mask output	0.1916	0.1276	22.3597	2.35

As shown in Table 3, the complete XDecomposer achieves the best performance across all decomposition and identification metrics. Removing multi-scale skip fusion produces the largest

structural-ablation decrease in Pearson correlation, demonstrating the importance of retaining high-resolution peak information during reconstruction. Removing the geometry constraint, phase-guided modulation, or global-context encoder also substantially reduces Pearson and Top-1 accuracy, confirming that these modules provide complementary geometric, phase-specific, and long-range structural information. Direct output performs worse than all structural variants, while hard-mask reconstruction causes the most severe degradation. Furthermore, Table 5 shows that removing mixture consistency causes a moderate performance drop, while removing SI-SDR results in larger degradation in profile similarity and identification accuracy. The amplitude-only variant achieves 51.16% Top-1 accuracy, confirming amplitude reconstruction as a strong base objective, while its lower performance than the full model demonstrates the value of auxiliary losses. Removing amplitude loss further decreases the Pearson correlation to 0.5982 and Top-1 accuracy to 47.24%. Removing self-supervised pre-training causes the largest degradation, highlighting its critical role in robust decomposition and representation learning. These results demonstrate that continuous mixture-aware masking is essential for resolving overlapping diffraction signals. Further analysis is provided in Appendix B.1.

6 Conclusion

We present XDecomposer, a prior-free deep learning framework for whole-pattern decomposition in multiphase identification. By casting multiphase decomposition as a blind source separation problem, XDecomposer enables joint phase inference through set prediction and recovers component patterns without prior phase information. Experiments on simulated and real-world datasets show consistent advantages in pattern reconstruction, peak position and width fidelity, and phase identifiability. These results establish XDecomposer as a new perspective beyond traditional phase identification pipelines, avoiding the limitations of sequential retrieval-based methods and offering strong potential for scalable high-throughput applications. **Limitations and Future Work.** XDecomposer relies on fixed output slots with activity prediction, which works well for the 2–4 phase settings studied here but may limit flexibility for mixtures with more variable phase counts. The current mask-based reconstruction improves mixture consistency and physical plausibility, yet may be less robust under complex experimental distortions (e.g., peak broadening, shifts, and background variation), suggesting the need for more adaptive reconstruction strategies. By producing standardized one-dimensional PXRD patterns, XDecomposer enables seamless integration with existing identification (XQueryer [9]), crystal structure prediction, refinement (WPEM [44]), and property prediction (PRDNet [53]) models via lightweight preprocessing, including interpolation and normalization. These structural outputs support downstream applications such as property prediction, inverse materials design, autonomous discovery, and experimental validation.

Acknowledgments and Disclosure of Funding

This work was supported by the Strategic Priority Research Program of the Chinese Academy of Sciences (XDA0480102) and the National Natural Science Foundation of China (62576339). The authors declare no competing interests.

References

[1] Robert E Dinnebier and Simon JL Billinge. *Powder diffraction: theory and practice*. Royal society of chemistry, 2015.

[2] Andrei A Bunaciu, Elena Gabriela UdriŞTioiu, and Hassan Y Aboul-Enein. X-ray diffraction: instrumentation and applications. *Critical reviews in analytical chemistry*, 45(4):289–299, 2015.

[3] Robert Alan Young. *The rietveld method*, volume 5. International union of crystallography, 1993.

[4] Bin Cao. *Physics-Constrained Learning of Crystal Structures and Properties from Powder Diffraction*. PhD thesis, PhD thesis, The Hong Kong University of Science and Technology, 2026.

[5] Hugo M Rietveld. A profile refinement method for nuclear and magnetic structures. *Applied Crystallography*, 2(2):65–71, 1969.

[6] George Stanley Pawley. Unit-cell refinement from powder diffraction scans. *Applied Crystallography*, 14(6):357–361, 1981.

[7] Armel Le Bail, Huguette Duroy, and Jean Louis Fourquet. Ab-initio structure determination of lisbwo6 by x-ray powder diffraction. *Materials Research Bulletin*, 23(3):447–452, 1988.

[8] Luca Lutterotti, Henry Pilliere, Christophe Fontugne, Philippe Boullay, and Daniel Chateigner. Full-profile search–match by the rietveld method. *Applied Crystallography*, 52(3):587–598, 2019.

[9] Bin Cao, Zinan Zheng, Yang Liu, Longhan Zhang, Lawrence WY Wong, Lu-Tao Weng, Jia Li, Haoxiang Li, and Tong-Yi Zhang. Xqueryer: an intelligent crystal structure identifier for powder x-ray diffraction. *National Science Review*, 12(12):nwaf421, 2025.

[10] Bin Cao, Yang Liu, Zinan Zheng, Ruifeng Tan, Jia Li, and Tong-yi Zhang. Simxrd-4m: big simulated x-ray diffraction data and crystal symmetry classification benchmark. In *The Thirteenth International Conference on Learning Representations*, 2025.

[11] Phillip M Maffettone, Lars Banko, Peng Cui, Yury Lysogorskiy, Marc A Little, Daniel Olds, Alfred Ludwig, and Andrew I Cooper. Crystallography companion agent for high-throughput materials discovery. *Nature Computational Science*, 1(4):290–297, 2021.

[12] Nathan J Szymanski, Christopher J Bartel, Yan Zeng, Mouhamad Diallo, Haegyeom Kim, and Gerbrand Ceder. Adaptively driven x-ray diffraction guided by machine learning for autonomous phase identification. *npj Computational Materials*, 9(1):31, 2023.

[13] Jan Schuetzke, Simon Schweidler, Friedrich R Muenke, Andre Orth, Anurag D Khandelwal, Ben Breitung, Jasmin Aghassi-Hagmann, and Markus Reischl. Accelerating materials discovery: Automated identification of prospects from x-ray diffraction data in fast screening experiments. *Advanced Intelligent Systems*, 6(3):2300501, 2024.

[14] Jerardo E Salgado, Samuel Lerman, Zhaotong Du, Chenliang Xu, and Niaz Abdolrahim. Automated classification of big x-ray diffraction data using deep learning models. *npj Computational Materials*, 9(1):214, 2023.

[15] Jin-Woong Lee, Woon Bae Park, Jin Hee Lee, Satendra Pal Singh, and Kee-Sun Sohn. A deep-learning technique for phase identification in multiphase inorganic compounds using synthetic xrd powder patterns. *Nature communications*, 11(1):86, 2020.

[16] Nam Q Le, Michael Pekala, Alexander New, Edwin B Gienger, Christine Chung, Timothy J Montalbano, Elizabeth A Pogue, Janna Domenico, and Christopher D Stiles. Deep learning models to identify common phases across material systems from x-ray diffraction. *The Journal of Physical Chemistry C*, 127(44):21758–21767, 2023.

[17] Nathan J Szymanski, Christopher J Bartel, Yan Zeng, Qingsong Tu, and Gerbrand Ceder. Probabilistic deep learning approach to automate the interpretation of multi-phase diffraction spectra. *Chemistry of Materials*, 33(11):4204–4215, 2021.

[18] Litao Chen, Bingxu Wang, Wentao Zhang, Shisheng Zheng, Zhefeng Chen, Mingzheng Zhang, Cheng Dong, Feng Pan, and Shunning Li. Crystal structure assignment for unknown compounds from x-ray diffraction patterns with deep learning. *Journal of the American Chemical Society*, 146(12):8098–8109, 2024.

[19] Qingsi Lai, Fanjie Xu, Lin Yao, Zhifeng Gao, Siyuan Liu, Hongshuai Wang, Shuqi Lu, Di He, Liwei Wang, Linfeng Zhang, et al. End-to-end crystal structure prediction from powder x-ray diffraction. *Advanced Science*, 12(8):2410722, 2025.

[20] Chenlei Xu, Tianhao Su, Jie Xiong, Yue Wu, Shuya Dong, Tian Jiang, Mengwei He, Shuai Chen, and Tong-Yi Zhang. Kan-enhanced contrastive learning: the accelerator of crystal structure identification from xrd patterns. *npj Computational Materials*, 2026.

[21] Xiao Zhang, Xitao Wang, and Shunbo Hu. Structural fingerprinting of crystalline materials from xrd patterns using atomic cluster expansion neural network and atomic cluster expansion. *Applied Sciences*, 15(11):5851, 2025.

[22] Qi Li, Rui Jiao, Liming Wu, Tiannian Zhu, Wenbing Huang, Shifeng Jin, Yang Liu, Hongming Weng, and Xiaolong Chen. Powder diffraction crystal structure determination using generative models. *Nature Communications*, 16(1):7428, 2025.

[23] Dongfang Yu, Zhewen Zhu, Fucheng Leng, and Yizhou Zhu. Equivariant diffusion solution for inorganic crystal structure determination from powder x-ray diffraction data. *Nature Communications*, 2026.

[24] Bin Feng, Bingxu Wang, Linpeng Lv, Mingzheng Zhang, Zhefeng Chen, Feng Pan, and Shunning Li. Interpreting x-ray diffraction patterns of metal–organic frameworks via generative artificial intelligence. *Journal of the American Chemical Society*, 148(1):869–878, 2025.

[25] Gabe Guo, Tristan Luca Saidi, Maxwell W Terban, Michele Valsecchi, Simon JL Billinge, and Hod Lipson. Ab initio structure solutions from nanocrystalline powder diffraction data via diffusion models. *Nature Materials*, 24(11):1726–1734, 2025.

[26] International Centre for Diffraction Data. Mdi jade software. `https://www.icdd.com/mdi-jade/`. Accessed: 2024-05-18.

[27] Yuxing Fei, Matthew J McDermott, Christopher L Rom, Shilong Wang, and Gerbrand Ceder. Dara: Automated multiple-hypothesis phase identification and refinement from powder x-ray diffraction. *Chemistry of Materials*, 2026.

[28] Jaimie Greasley and Patrick Hosein. Exploring supervised machine learning for multi-phase identification and quantification from powder x-ray diffraction spectra. *Journal of Materials Science*, 58(12):5334–5348, 2023.

[29] Ibrahim Olanrewaju Alade, Mojeed Opeyemi Oyedeji, Mohd Amiruddin Abd Rahman, and Tawfik A Saleh. Prediction of the lattice constants of pyrochlore compounds using machine learning: Io alade et al. *Soft Computing*, 26(17):8307–8315, 2022.

[30] Abdulgafor Alfares, Yusuf Abubakar Sha'aban, and Ahmed Alhumoud. Machine learning-driven predictions of lattice constants in abx3 perovskite materials. *Engineering Applications of Artificial Intelligence*, 141:109747, 2025.

[31] Dong Yu, Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen. Permutation invariant training of deep models for speaker-independent multi-talker speech separation. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 241–245. IEEE, 2017.

[32] Morten Kolbæk, Dong Yu, Zheng-Hua Tan, and Jesper Jensen. Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(10):1901–1913, 2017.

[33] Yi Luo and Nima Mesgarani. Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation. *IEEE/ACM transactions on audio, speech, and language processing*, 27 (8):1256–1266, 2019.

[34] Cem Subakan, Mirco Ravanelli, Samuele Cornell, Mirko Bronzi, and Jianyuan Zhong. Attention is all you need in speech separation. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 21–25. IEEE, 2021.

[35] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep sets. *Advances in neural information processing systems*, 30, 2017.

[36] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International conference on machine learning*, pages 3744–3753. PMLR, 2019.

[37] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[39] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[40] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[41] Vitalij K Pecharsky and Peter Y Zavalij. *Fundamentals of powder diffraction and structural characterization of materials*. Springer, 2003.

[42] Yusei Ito, Naoya Chiba, Tatsunori Taniai, Ryo Igarashi, Yuta Suzuki, Kotaro Saito, Yoshitaka Ushiku, and Kanta Ono. Enabling single-observation decomposition of multi-phase x-ray diffraction patterns via generative deep learning. *npj Computational Materials*, 2026.

[43] Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, et al. Commentary: The materials project: A materials genome approach to accelerating materials innovation. *APL materials*, 1(1), 2013.

[44] Bin Cao, Qian Zhang, Zhenjie Feng, Taolue Zhang, Jiaqiang Huang, Lu-Tao Weng, and Tong-Yi Zhang. Ai-driven structure refinement of x-ray diffraction. *arXiv preprint arXiv:2602.16372*, 2026.

[45] James A. Kaduk, Simon J. L. Billinge, Robert E. Dinnebier, Nathan Henderson, Ian Madsen, Radovan Černý, Matteo Leoni, Luca Lutterotti, Seema Thakral, and Daniel Chateigner. Powder diffraction. *Nature Reviews Methods Primers*, 1:77, 2021. doi: 10.1038/s43586-021-00074-7.

[46] PW Stephens. The physics of diffraction from powders. *International Tables for Crystallography, Volume H: Powder Diffraction*, page 252, 2019.

[47] Barbara Lafuente, Robert T Downs, Hexiong Yang, and Nate Stone. 1. the power of databases: The rruff project. In *Highlights in mineralogical crystallography*, pages 1–30. De Gruyter (O), 2015.

[48] Titouan Simonnet, Sylvain Grangeon, Francis Claret, Nicolas Maubec, Mame Diarra Fall, Rachid Harba, and Bruno Galerne. Phase quantification using deep neural network processing of xrd patterns. *IUCrJ*, 11(5), 2024.

[49] Titouan Simonnet, Mame Diarra Fall, Sylvain Grangeon, and Bruno Galerne. Vision transformers for x-ray diffraction patterns analysis. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[50] Nathan J Szymanski, Sean Fu, Ellen Persson, and Gerbrand Ceder. Integrated analysis of x-ray diffraction patterns and pair distribution functions for machine-learned phase identification. *Npj computational materials*, 10(1):45, 2024.

[51] Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. *CoRR*, abs/2310.06625, 2023.

[52] Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *ICLR*, 2023.

[53] Bin Cao, Yang Liu, Longhan Zhang, Yifan Wu, Zhixun Li, Yuyu Luo, Hong Cheng, Yang Ren, and Tong-Yi Zhang. Beyond structure: Invariant crystal property prediction with pseudo-particle ray diffraction. *arXiv preprint arXiv:2509.21778*, 2025.

[54] JH Hubbell. *X-ray cross sections and attenuation coefficients*. Chapman & Hall., 1974.

[55] BH Armstrong. Spectrum line profiles: the voigt function. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 7(1):61–88, 1967.

[56] G Caglioti, AT Paoletti, and FP Ricci. Choice of collimators for a crystal spectrometer for neutron diffraction. *Nuclear Instruments*, 3(4):223–228, 1958.

[57] MAR Miranda and JM Sasaki. The limit of application of the scherrer equation. *Acta Crystallographica Section A: Foundations and Advances*, 74(1):54–65, 2018.

[58] B Van Laar and WB Yelon. The peak in neutron powder diffraction. *Journal of Applied crystallography*, 17(2):47–54, 1984.

[59] Shyue Ping Ong, William Davidson Richards, Anubhav Jain, Geoffroy Hautier, Michael Kocher, Shreyas Cholia, Dan Gunter, Vincent L Chevrier, Kristin A Persson, and Gerbrand Ceder. Python materials genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science*, 68:314–319, 2013.

[60] Miroslav Morháč and Vladislav Matoušek. Peak clipping algorithms for background estimation in spectroscopic data. *Applied spectroscopy*, 62(1):91–106, 2008.

[61] University College London. X-ray powder diffraction quality assurance tables, date accessed. URL `http://pd.chem.ucl.ac.uk/pxrd/qa/smtable.htm`.

[62] Bin Cao. Xmatcher: An open-source framework for x-ray diffraction phase identification. *arXiv preprint arXiv:2607.17162*, 2026.

Appendix

A Methodology Details **15**

A.1 Training Objectives 15

A.2 Implementation Details 17

A.3 From Structure to PXRD 17

B Experimental Implementation **19**

B.1 Details of Ablation Study 19

B.2 Large-scale Simulated Single-phase Generation 20

B.3 RRUFF Experimental Dataset Construction and Preprocessing 20

B.4 Data Mixing Strategy 21

B.5 Data Split 22

B.6 Baseline Implementation 23

B.7 Metric Definitions 24

B.8 Robustness Analysis 24

B.9 Additional Robustness Analysis 25

B.10 Dataset Analysis 28

B.11 Quantitative Decomposition 29

C Additional Knowledge **31**

C.1 Search-Match Approach 31

C.2 In-library and Out-of-library Tasks 31

C.3 Crystal Structure Determination via PXRD 32

A Methodology Details

A.1 Training Objectives

We present the optimization objectives for the pretraining and decomposition stages separately. The two objectives are not simply the same loss function applied to different data. Instead, they share several reconstruction preferences while serving different purposes: the former targets structural representation learning of PXRD patterns, whereas the latter addresses multi-phase decomposition, slot alignment, and active-phase detection.

A.1.1 Pretraining Objective

In Stage I, the model performs masked reconstruction pretraining on single-phase diffraction patterns. This objective is used to learn the parameters of the global context encoder $G(\cdot)$, while the reconstruction supervision is provided by an auxiliary decoder that is used only during pretraining. Given a single-phase pattern x and its reconstruction $\hat{x}$, the pretraining loss is

$$\mathcal{L}_{\text{pre}} = \mathcal{L}_{\text{amp}}^{\text{pre}} + \lambda_{\text{shape}}^{\text{pre}} \mathcal{L}_{\text{shape}}^{\text{pre}} + \lambda_{\text{geo}}^{\text{pre}} \mathcal{L}_{\text{geo}}^{\text{pre}}. \tag{10}$$

The amplitude term is defined as

$$\mathcal{L}_{\text{amp}}^{\text{pre}} = \text{mean}[|\hat{x} - x| \cdot (1 + \alpha x)]\,. \tag{11}$$

This weighting increases supervision on major peak regions. The shape term is

$$\mathcal{L}_{\text{shape}}^{\text{pre}} = -\text{SI-SDR}(\hat{x}, x). \tag{12}$$

SI-SDR denotes the scale-invariant signal-to-distortion ratio. For a predicted pattern $\hat{\mathbf{x}}$ and reference pattern $\mathbf{x}$, it is defined as

$$\alpha = \frac{\langle \hat{\mathbf{x}}, \mathbf{x} \rangle}{\|\mathbf{x}\|_2^2}, \qquad \text{SI-SDR} = 10 \log_{10} \frac{\|\alpha \mathbf{x}\|_2^2}{\|\hat{\mathbf{x}} - \alpha \mathbf{x}\|_2^2}.$$

Equation12 minimizes negative SI-SDR and therefore maximizes profile-shape agreement.

The geometry term operates on first- and second-order differences in the square-root domain:

$$\mathcal{L}_{\text{geo}}^{\text{pre}} = \|\nabla \sqrt{\hat{x}} - \nabla \sqrt{x}\|_1 + \beta \|\nabla^2 \sqrt{\hat{x}} - \nabla^2 \sqrt{x}\|_1. \tag{13}$$

This stage is intended to capture local peak-shape statistics, long-range peak co-occurrence patterns, and transferable structural priors from single-phase data, rather than to directly learn multi-phase decomposition.

A.1.2 Decomposition Objective

In Stage II, the model transfers the pretrained global context encoder $G(\cdot)$ to the full decomposition model and keeps it frozen, while the decomposition-related modules are optimized on multiphase mixtures. Given the ground-truth single-phase contributions $\{y_k\}_{k=1}^{K}$ and the predicted contributions $\{\hat{y}_k\}_{k=1}^{K_{\max}}$, permutation-invariant training resolves the assignment ambiguity between output slots and target phases:

$$\pi^* = \arg\min_{\pi} \sum_{k=1}^{K} \mathcal{L}_{\text{decomp}}(\hat{y}_{\pi(k)}, y_k). \tag{14}$$

The slot-wise decomposition loss is defined as

$$\mathcal{L}_{\text{decomp}} = \mathcal{L}_{\text{amp}} + \lambda_{\text{shape}} \mathcal{L}_{\text{shape}} + \lambda_{\text{geo}} \mathcal{L}_{\text{geo}}. \tag{15}$$

The amplitude term is

$$\mathcal{L}_{\text{amp}} = \frac{1}{K} \sum_{k=1}^{K} \text{mean}\left[|\hat{y}_{\pi^*(k)} - y_k| \cdot (1 + \alpha y_k) \right]. \tag{16}$$

The shape term is defined as

$$\mathcal{L}_{\text{shape}} = -\text{SI-SDR}(\hat{y}, y). \tag{17}$$

where SI-SDR denotes the scale-invariant signal-to-distortion ratio, is defined as

$$\alpha = \frac{\langle \hat{y}, y \rangle}{\|y\|_2^2}, \qquad \text{SI-SDR} = 10 \log_{10} \frac{\|\alpha y\|_2^2}{\|\hat{y} - \alpha y\|_2^2}. \tag{18}$$

The geometry term is

$$\mathcal{L}_{\text{geo}} = \frac{1}{K} \sum_{k=1}^{K} \left(\|\nabla \hat{z}_{\pi^*(k)} - \nabla z_k\|_1 + \beta \|\nabla^2 \hat{z}_{\pi^*(k)} - \nabla^2 z_k\|_1 \right), \tag{19}$$

where $z_k = \sqrt{y_k}$ and $\hat{z}_{\pi^*(k)} = \sqrt{\hat{y}_{\pi^*(k)}}$. These terms jointly constrain amplitude fidelity, spectral contour agreement, and local geometric consistency.

Based on the optimal assignment, the full decomposition objective is

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{decomp}}^{\text{PIT}} + \lambda_{\text{act}} \mathcal{L}_{\text{act}} + \lambda_{\text{mix}} \mathcal{L}_{\text{mix}}. \tag{20}$$

The activity supervision term is

$$\mathcal{L}_{\text{act}} = \frac{1}{K_{\max}} \sum_{k=1}^{K_{\max}} \text{BCE}(p_k, c_k), \tag{21}$$

where $c_k \in \{0, 1\}$ indicates whether slot k is matched to a true active phase under the optimal PIT assignment. Unmatched slots are treated as inactive and assigned label 0. The mixture-consistency term is

$$\mathcal{L}_{\text{mix}} = \|\hat{x} - x\|_1, \qquad \hat{x} = \sum_{k=1}^{K_{\max}} \hat{y}_k. \tag{22}$$

In summary, although the pretraining and decomposition stages share the same general reconstruction preferences: amplitude, shape, and geometry, their optimization goals differ substantially. The former learns transferable pure-phase structure representations, whereas the latter additionally incorporates permutation-invariant matching, activity supervision, and mixture consistency for multi-phase decomposition.

A.2 Implementation Details

All experiments were conducted on a single node with dual AMD EPYC 7742 CPUs and 8× NVIDIA GeForce RTX 3090 24GB GPUs, using Python 3.10 and PyTorch 2.5.1 with Distributed Data Parallel (DDP).Table 4 summarizes the training configurations and model efficiency statistics.

In the decomposition stage, the pretrained global context encoder $G(\cdot)$ is frozen. Accordingly, only 3.09M out of the 9.20M total parameters are trainable.

Table 4: Training configurations and model efficiency summary.

Parameter	**Pretraining**	**Decomposition**
Backbone	Transformer	
Embedding dimension	768	
Attention heads	12	
Transformer layers	4	
Optimizer	AdamW	
Weight decay	0.05	
Batch size	128	
LR scheduler	Cosine decay	
Learning rate	5×10^{-4}	2×10^{-4}
Epochs	2000	1000
Warmup epochs	50	20
Patch size / stride	50 / 25	–
Mask ratio	70%	–
Conv channel widths	–	`{48,96,192,384}`
Conv kernel sizes	–	`{15,8,8,10}`
Conv strides	–	`{1,2,2,5}`
Downsampling ratio	–	×20
Max slots $K_{\max}$	–	4
EMA at inference	–	Enabled
Total params (M)	6.11	9.20
Trainable params (M)	6.11	3.09
Est. GFLOPs	1.2–2.0	2.5–3.5
CPU latency (ms)	∼15.0	∼30.0
FP32 model size (MB)	∼24	∼36

A.3 From Structure to PXRD

The physical process by which a PXRD pattern is generated from a crystal is detailed below:

The diffraction vector $\mathbf{Q}$ is defined as

$$|\mathbf{Q}| = \frac{2\pi \sin\theta}{\lambda}, \tag{23}$$

where λ is the wavelength of the scattered X-ray and $\theta = 2\theta/2$, with 2θ being the angle between the incident and diffracted beams. In three dimensions, scattering occurs only at a discrete set of reciprocal vectors, $\mathbf{K}$, forming the reciprocal lattice,

$$\mathbf{K} = h\mathbf{a}^* + k\mathbf{b}^* + l\mathbf{c}^*, \tag{24}$$

where $\mathbf{a}^*$, $\mathbf{b}^*$, and $\mathbf{c}^*$ are the reciprocal lattice vectors, and h, k, and l are constants. The diffraction condition is defined as

$$\mathbf{Q} = \mathbf{K}. \tag{25}$$

The diffraction intensity I on each diffraction vector $\mathbf{Q}$ for a single phase is determined by

$$I(\mathbf{Q}) = SF^*F\varnothing LPOD + I^{\mathrm{BG}}, \tag{26}$$

where S denotes the scale factor, F is the structure factor, F^* is the complex conjugate of F, $\varnothing$ is the profile function, L is the Lorentz-polarization factor, P is the multiplicity, O is the preferred orientation factor, D is the Debye-Waller factor, and I^{BG} is the background intensity.

The structure factor is computed as:

$$F = \sum_{j=1}^{N} f_j e^{i\mathbf{Q}\cdot\mathbf{R}}, \tag{27}$$

where f_j is the form factor [54] in XRD, $\mathbf{R}$ is the lattice coordinate of atom j, and N is the total number of atoms in a lattice cell.

The Lorentz-polarization L is calculated by,

$$L = \frac{1 + \cos^2 2\theta}{\sin^2\theta \cos\theta}, \tag{28}$$

where $\theta = \arcsin\left(\frac{\lambda|\mathbf{Q}|}{2\pi}\right)$. The multiplicity P is determined by counting the number of diffraction vectors present within an Ewald diffraction sphere. The Debye-Waller factor D is calculated by,

$$D = e^{-2M}, \tag{29}$$

where $M = \frac{6h^2T}{mk\Theta^2}\left(\phi\left(\frac{\Theta}{T}\right) + \frac{\Theta}{4T}\right)\left(\sin^2\theta\right)/\lambda$, h is Planck's constant, m is atom mass, k is Boltzmann constant, Θ is the average characteristic temperature, T is absolute temperature, and $\phi(\Theta/T)$ is the Debye function.

The profile function ($\varnothing$) is a convolution of various factors, including diffraction effects, detector geometry, and noise contributions. The peaks arise from:

$$y(x) = W \otimes G \otimes S. \tag{30}$$

Here, $\otimes$ denotes the convolution process, and W, G, and S represent the contributions to the observed XRD pattern from diffraction emission, instrumental factors, and the noise mixture, respectively. S is modeled as a Gaussian peak.

The W is a Voigt function [55],

$$W = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} \left[\frac{\gamma}{(2\theta - t)^2 + \gamma^2}\right] \exp\left(-\frac{(2\theta - t)^2}{2\sigma^2}\right) dt. \tag{31}$$

Peak broadening is correlated with the Full Width at Half Maximum (FWHM, Γ), where $2\gamma = 2\sqrt{2\ln 2}\sigma = \Gamma$ [56]. Γ is calculated by Scherrer's equation [57] related to finite grain size.

The geometrical factor[58], G, accounts for the actual dimensions of the detectors and the powder specimen. It is defined as follows:

$$G(2\alpha, 2\theta) = \frac{L}{4HSh\cos(2\alpha)} \int dz, \tag{32}$$

where 2α represents the Bragg angle. The detector is slit-shaped, with a height of $2H$, and the sample has a height of $2S$. L denotes the distance between the specimen and the detector.

For multiphase systems, the observed diffraction intensity arises from the additive superposition of the scattering contributions from all constituent phases, which is a fundamental property of PXRD under the assumption that different phases scatter independently without significant interaction effects. The measured intensity can be expressed as

$$I(\mathbf{Q}_{\mathrm{mult}}) = \sum_{i=1}^{N} w_i\, I_i(\mathbf{Q}) + \sigma(\mathbf{Q}), \tag{33}$$

where $I_i(\mathbf{Q})$ denotes the diffraction intensity of the i-th phase at scattering vector $\mathbf{Q}$, w_i represents its relative proportion (mixture coefficient) in the sample, and N is the total number of constituent phases.

The term $\sigma(\mathbf{Q})$ accounts for experimental noise, background signals, instrumental fluctuations, trace impurities, and minor unidentified components whose contributions are typically weak and lack clear physical guidance.

Each single-phase pattern $I_i(\mathbf{Q})$ is influenced by multiple physical factors, including crystal structure, preferred orientation, crystallite size, strain broadening, peak overlap, and instrumental broadening etc. In multiphase systems, these effects accumulate through linear superposition, making the observed pattern substantially more sensitive to small variations in any individual phase. Even slight deviations in peak position, intensity, or line shape from one component phase can propagate to the final mixed pattern and affect phase identification and quantitative decomposition.

As a result, multiphase modeling is inherently more fragile than single-phase analysis: errors introduced at the single-phase level are amplified after superposition, while overlapping peaks further increase ambiguity in phase separation. This explains the significant sim-to-real performance gap observed in experimental datasets, where real measurements contain stronger structural variation and more complex noise sources than simulated data.

B Experimental Implementation

B.1 Details of Ablation Study

We conduct ablation studies on simulated four-phase mixtures with $K = 4$ to evaluate the contribution of the main design components of XDecomposer under severe peak overlap and increased combinatorial ambiguity. This setting provides a challenging testbed for assessing spectral reconstruction quality, peak-structure preservation, and downstream identification accuracy.

We first examine the architectural and reconstruction components of XDecomposer. The evaluated variants include removing multi-scale skip fusion, removing the geometry constraint, removing phase-guided modulation, and removing the global-context encoder $G(\cdot)$. Two alternative reconstruction forms are also considered. In the *direct regression output* variant, the mixture-aware masking pathway is replaced by direct component-pattern regression. In the *hard mask output* variant, the continuous assignment weights are binarized to approximate discrete allocation in overlapping regions. The results are reported in Table 3.

The full XDecomposer model achieves the best performance across both decomposition and identification metrics, indicating that its advantage arises from the joint effect of its architectural modules and reconstruction strategy. All ablated variants exhibit clear degradation at $K = 4$, confirming that each design choice is important for handling severe peak overlap. Among the structural variants, *w/o multi-scale skip fusion* remains closest to the full model in most metrics; however, its ΔFWHM increases sharply from 5.2546 to 17.7708. This substantial increase indicates that multi-scale skip connections are critical for preserving local peak details and accurate line-width information. Removing the *geometry constraint* or *phase-guided modulation* leads to consistent declines in spectral fidelity, peak-position accuracy, and retrieval performance, suggesting that the geometry constraint promotes physically plausible peak structures, while phase-guided modulation improves the separation of component-specific representations. The *w/o global-context encoder* $G(\cdot)$ variant performs worst among the structural ablations, underscoring the importance of global-context modeling for recovering coherent and identifiable single-phase patterns.

The comparison of reconstruction forms further demonstrates the importance of the output pathway. The *direct regression output* variant underperforms the full model, indicating that direct component regression does not fully exploit the structural information contained in the input mixture. The *hard mask output* variant yields the lowest overall performance among the reconstruction variants. This result shows that discretizing continuous assignments disrupts shared spectral structure in overlapping regions, impairing both reconstruction accuracy and downstream phase identification.

We further analyze the effects of self-supervised pre-training and the individual components of the composite learning objective. For the pre-training ablation, the self-supervised stage is removed and the model is trained directly from initialization using the same architecture and supervised training protocol. For the objective-component ablations, one component is removed at a time while all other settings are kept unchanged. These results are summarized in Table 5.

Table 5: Ablation results for self-supervised pre-training and objective components on simulated four-phase mixtures with $K = 4$.

Variant	Pearson ↑	Peak Shift (°) ↓	ΔFWHM (°) ↓	Top-1 (%) ↑
Full model	**0.8322**	**0.0480**	**5.2410**	**69.68**
w/o mixture-consistency loss	0.7912	0.0524	5.4364	60.34
w/o SI-SDR loss	0.7612	0.0514	5.4952	58.35
amplitude loss only	0.6287	0.0573	5.6198	51.16
w/o amplitude loss	0.5982	0.0609	5.9110	47.24
w/o activity loss	0.4073	0.0777	13.1825	14.34
w/o pre-training	0.4067	0.0787	13.5071	14.00

The loss-component ablations show that each part of the learning objective contributes to the final performance. Removing the mixture-consistency term moderately degrades all metrics, reducing Top-1 accuracy from 69.68% to 60.34%. Removing the SI-SDR term causes a larger reduction in profile similarity and identification accuracy, indicating that scale-invariant spectral matching is beneficial for recovering distinguishable component patterns. The *amplitude loss only* variant still achieves 51.16% Top-1 accuracy, showing that amplitude reconstruction provides a strong basic training signal. Nevertheless, its gap from the full model confirms that auxiliary constraints are necessary for accurate phase separation under heavy overlap. Removing the amplitude loss further reduces Pearson correlation to 0.5982 and Top-1 accuracy to 47.24%, demonstrating that amplitude-level supervision remains indispensable for faithful profile recovery.

The most pronounced degradation occurs when activity supervision or self-supervised pre-training is removed. Without activity supervision, Top-1 accuracy drops to 14.34% and ΔFWHM increases to $13.1825°$, indicating that explicit activity-related supervision is essential for associating recovered patterns with identifiable phases. Similarly, removing self-supervised pre-training reduces Top-1 accuracy to 14.00% and produces the largest ΔFWHM among these variants. This result suggests that pre-training provides a critical initialization for learning robust spectral representations before supervised decomposition, especially in the highly ambiguous $K = 4$ setting.

B.2 Large-scale Simulated Single-phase Generation

The simulated dataset comprises a large-scale corpus of single-phase XRD patterns generated using PySimXRD [10] and subsequently mixed online during the training stage. Pattern generation was conducted on a high-performance computing cluster with SLURM job scheduling and a multi-process parallel framework, enabling the generation of multiple simulated patterns for each crystal ID under varied parameter configurations. This design improves coverage of diverse sample states and experimental conditions encountered in real-world settings. A fixed random seed is used to ensure reproducibility. The objective is not to produce idealized template patterns, but to introduce sufficient variation in peak positions, widths, backgrounds, and instrumental geometry, thereby enabling the model to learn transferable diffraction structure representations under realistic perturbations. Table 6 summarizes the main parameter configurations used for simulated single-phase PXRD pattern.

B.3 RRUFF Experimental Dataset Construction and Preprocessing

To establish a cross-domain zero-shot generalization benchmark, we constructed a unified dataset of 662 high-quality experimental PXRD patterns with corresponding crystal structure information from the publicly available RRUFF mineral database [47]. The raw data consist of experimentally measured diffraction patterns and separately stored CIF structure files. The construction pipeline comprises four stages: raw data acquisition, structure–pattern matching with deduplication, WPEM [44] baseline correction, and database serialization with quality control.

Structure–pattern matching Experimental patterns were matched to CIF files using mineral names as the primary key. When multiple PXRD samples existed for the same mineral species, only one representative sample was retained to ensure a unique structure–pattern pair. All CIF structures were subsequently parsed using `pymatgen`'s `SpacegroupAnalyzer` (`symprec = 0.1`) [59], and

Table 6: Main parameter configurations for simulated single-phase XRD pattern

Category	Parameter	Value / Range
Random perturbations	Crystallite size (nm)	10–120
	Thermal vibration coefficient	0.01–0.2
	Zero shift	0–0.2
	Detector–sample distance (mm)	300–600
	Detector slit half-height (mm)	3–8
	Sample half-height (mm)	1–4
Fixed parameters	Preferred orientation factor	0.15
	Background polynomial order	6
	Mixture noise ratio	0.02
	Lattice extinction / torsion ratio	0.01 / 0.01
Data control	2θ scan range	10°–80°
	Step size	0.02°
	Spectra per crystal ID	20

structures containing site disorder were strictly excluded, yielding 665 naturally ordered mineral structures.

Baseline correction Since experimentally measured PXRD patterns inherently contain background baselines (see Appendix A.3), and these non-crystalline background signals can interfere with phase decomposition, we applied a unified automated baseline stripping procedure to all experimental patterns using the WPEM module [44]. This method combines frequency-domain decomposition, piecewise polynomial fitting, and Savitzky–Golay smoothing to estimate and subtract the background baseline. The resulting baseline-corrected net patterns were serialized as NumPy arrays in the database. Three samples were discarded due to insufficient data points or anomalous angular ranges, resulting in a final database of 662 valid records.

Database serialization During model training and evaluation, the data loader further applies sensitive nonlinear iterative peak (SNIP) clipping algorithms[60] interpolation to a standard angular grid of 3500 equidistant points over $2\theta \in [10^\circ, 80^\circ]$, followed by max-value normalization. This ensures comparability across samples on a unified input grid.

B.4 Data Mixing Strategy

During decomposition training, we construct training samples on the fly using an anchor-driven online mixing protocol, rather than relying on a pre-generated fixed dataset of multiphase mixtures. For each online sample, one single-phase pattern from the current data split is first selected as the anchor component, and the remaining components are then sampled from the same split. A simulated mixture is constructed from these components using randomly sampled mixing weights. Its physical basis follows the linear superposition approximation [41], under which a multiphase diffraction pattern is approximated as the weighted sum of the diffraction contributions from its constituent phases.

The pre-prepared single-phase set covers 100,315 crystal samples, and each crystal sample is simulated under 20 different realistic conditions, yielding a total of $100{,}315 \times 20$ single-phase PXRD patterns. For each crystal sample, we randomly select one pattern from the 20 candidates and use the resulting set as the basis for subsequent online mixing. Let the single-phase library corresponding to the current split be denoted by $\mathcal{D}^{(\mathrm{split})} = \{x^{(m)}\}_{m=1}^{M_{\mathrm{split}}}$, where $x^{(m)} \in \mathbb{R}_+^L$, M_{split} is the number of single-phase patterns in that split, and L is the spectrum length on the common sampling grid. Online mixing is always performed after the train, validation, and test splits have been defined at the single-phase level. As a result, mixture samples for training, validation, and testing are generated only from the single-phase patterns within their respective splits, with no cross-split mixing of components.

For each online sample, the number of components is first drawn as

$$N \sim \mathrm{Unif}\{2, 3, 4\}. \tag{34}$$

An anchor single-phase pattern is then selected from $\mathcal{D}^{(\mathrm{split})}$, with index a, and the corresponding anchor component is written as $x_{\mathrm{anchor}} = x^{(a)}$. Given the anchor, the remaining $N-1$ single-phase patterns are sampled without replacement from $\mathcal{D}^{(\mathrm{split})} \setminus \{x^{(a)}\}$. Let their index set be $R = \{r_1, \ldots, r_{N-1}\} \subset \{1, \ldots, M_{\mathrm{split}}\} \setminus \{a\}$ with $|R| = N-1$. The resulting candidate component set is $\{x_i\}_{i=1}^{N} = \{x_{\mathrm{anchor}}, x^{(r_1)}, \ldots, x^{(r_{N-1})}\}$.

The mixing weights are sampled from a Dirichlet distribution,

$$w \sim \mathrm{Dir}(\alpha_1, \ldots, \alpha_N), \tag{35}$$

where $w = (w_1, \ldots, w_N)$ satisfies $w_i \geq 0$ and $\sum_{i=1}^{N} w_i = 1$. In the default setting, we use a symmetric Dirichlet prior, i.e., $\alpha_1 = \cdots = \alpha_N = \alpha$, with $\alpha = 1$. To prevent any component from becoming nearly unobservable after mixing, we further require

$$w_i \geq 0.15, \qquad i = 1, \ldots, N. \tag{36}$$

If a sampled weight vector does not satisfy this condition, it is discarded and resampled.

Given the sampled components and weights, the mixture pattern is constructed as

$$x_{\mathrm{mix}} = \sum_{i=1}^{N} w_i x_i. \tag{37}$$

We then add low-amplitude Gaussian noise to the mixture,

$$\tilde{x}_{\mathrm{mix}} = x_{\mathrm{mix}} + \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, \sigma^2 I), \tag{38}$$

where σ is the preset noise level and I is the identity matrix. To keep the model input and supervision targets on the same numerical scale, we apply max normalization with a shared scaling factor. Specifically, we define

$$c = \max\left(\max_{1 \leq t \leq L} \tilde{x}_{\mathrm{mix}}(t), \ \max_{1 \leq i \leq N} \max_{1 \leq t \leq L} x_i(t) \right), \tag{39}$$

and obtain the normalized input and targets as

$$\hat{x}_{\mathrm{mix}} = \frac{\tilde{x}_{\mathrm{mix}}}{c}, \qquad \hat{x}_i = \frac{x_i}{c}, \quad i = 1, \ldots, N. \tag{40}$$

Each online sample is therefore represented as $(\hat{x}_{\mathrm{mix}}, \{\hat{x}_i\}_{i=1}^{N}, \{w_i\}_{i=1}^{N})$.

Three considerations motivated our adoption of the online mixing strategy. First, the space of multiphase mixtures grows rapidly with component choice, composition ratio, and noise perturbation, making it difficult for a fixed dataset to cover the resulting continuous combination space. Second, online mixing continuously generates new component combinations, weight configurations, and noise realizations during training, which increases sample diversity. Third, on-demand generation avoids the additional storage and data-management cost of pre-generating large numbers of mixture patterns, while allowing unified control over the component range, weight lower bound, and noise level.

B.5 Data Split

For the simulated data, the single-phase pattern library is partitioned at the crystal ID level rather than the pattern file level. This prevents patterns generated from the same crystal under different parameter perturbations from appearing simultaneously in the training, validation, and test sets, thereby eliminating potential data leakage. In other words, the split unit is the "crystal instance," not the "pattern sample," ensuring that the model does not merely memorize near-replicate patterns of the same structure under varying perturbations.

For the real-domain experiments, evaluation is uniformly based on the 662 RRUFF experimental single-phase patterns using 5-fold cross-validation. The dataset is partitioned into five disjoint subsets, with each subset serving as the test set in turn. The mean and standard deviation of all metrics are reported across the five folds.

B.6 Baseline Implementation

This section describes how the comparative baselines are instantiated within our experimental framework. Since the original objectives of these methods are not fully aligned, the comparison does not focus on whether they share exactly the same task definition as XDecomposer. Instead, we evaluate whether they can recover single-phase representations with physical interpretability and downstream identifiability under a unified evaluation protocol.

We divide the baselines into two categories: **domain-specific baselines** and **sequence baselines**. The former includes XQueryer, XRDAutoAnalyzer, and XRD_Proportion_Inference, all of which were originally developed for PXRD phase identification or quantitative analysis. The latter includes Transformer, iTransformer, and PatchTST, which are general-purpose sequence models adapted here to the PXRD decomposition setting.

XQueryer XQueryer is the first cross-chemical general model for single-phase powder XRD structure identification and is therefore used as a baseline in our study [9]. Since the original XQueryer targets single-phase XRD identification, we adapt it into a multi-phase baseline by introducing a slot-based prediction head on mixture inputs. Specifically, instead of producing one identification result for a single pattern, the adapted model takes a mixed XRD pattern as input and predicts S unordered phase slots; each slot contains a reconstructed component pattern, an estimated mixing ratio, and a classification logit over candidate phases. We train this variant with online-synthesized mixtures and use Hungarian matching to associate predicted slots with ground-truth phases. This modification preserves the main XQueryer backbone while enabling direct comparison under our unified multi-phase separation and identification benchmark.

XRDAutoAnalyzer XRDAutoAnalyzer is based on the automated XRD analysis framework of Szymanski *et al.* [17, 50], whose original goal is phase identification and refinement within a reference-driven analysis pipeline rather than blind multi-phase source separation. In our benchmark, we adapt it through an external wrapper instead of modifying its core model. Specifically, we first synthesize mixture patterns from multiple single-phase references, then feed each mixture to the official `SpectrumAnalyzer`, which returns candidate phases, confidence scores, refinement scalings, and reconstructed phase-wise spectra. We map these outputs to our protocol by aligning predicted phases to the ground truth, forming a fixed-size set of predicted component patterns, and deriving compatible activity signals for unified metric computation. This adaptation allows us to compare XRDAutoAnalyzer under the same separation, identification, and quantification metrics as other baselines, while making clear that it remains a retrieval-and-refinement pipeline rather than an end-to-end decomposition network. Owing to this design, XRDAutoAnalyzer is evaluated only in the simulated-domain setting and is not included in the main experimental-domain comparison.

XRD_Proportion_Inference XRD_Proportion_Inference is based on the deep-learning framework for XRD quantitative analysis introduced by Simonnet *et al.*, together with its Vision Transformer extension [48, 49]. Since the original formulation is designed for proportion inference rather than source-resolved separation, we adapt it into a multi-phase benchmark baseline by changing the prediction target from global composition regression to component-pattern reconstruction. Concretely, instead of using the original output head that predicts phase proportions from a single mixed pattern, we employ a separation-oriented encoder–decoder variant that preserves the overall convolutional parameterization of the baseline while mapping an input mixture pattern to a fixed set of reconstructed component patterns. Training samples are generated by online mixing of multiple single-phase XRD references, with mixture cardinality and proportions sampled on the fly, and the model is supervised against the corresponding component-wise target patterns. The adapted outputs are then evaluated under our unified protocol for pattern reconstruction, active-phase detection, database retrieval, and quantitative estimation, allowing us to assess how well a proportion-oriented baseline can be repurposed for multi-phase decomposition.

Transformer The Transformer baseline uses a standard attention-based sequence-to-sequence architecture [38]. It shares the same pretrained MAE encoder-side backbone as the other sequence baselines, including patch embedding, positional encoding, and the encoder. A Transformer decoder is then attached after the shared backbone to predict a fixed set of component patterns and their corresponding activity logits.

iTransformer The iTransformer baseline is built on the same pretrained MAE encoder backbone [51]. Unlike the standard Transformer variant, it applies an inverted Transformer structure after the shared encoder and performs component-wise prediction in latent space. The model outputs a fixed set of reconstructed component patterns together with their activity logits.

PatchTST The PatchTST baseline also uses the same pretrained MAE encoder backbone and attaches an encoder-only PatchTST-style prediction head after the shared latent representation [52]. It directly predicts a fixed set of component patterns and the corresponding activity logits from the encoded latent features.

B.7 Metric Definitions

Pearson Correlation Coefficient Given a predicted pattern $\hat{y}$ and target pattern y, the Pearson correlation coefficient is defined as

$$\rho(\hat{y}, y) = \frac{\sum_i (\hat{y}_i - \bar{\hat{y}})(y_i - \bar{y})}{\sqrt{\sum_i (\hat{y}_i - \bar{\hat{y}})^2}\sqrt{\sum_i (y_i - \bar{y})^2}}. \tag{41}$$

This metric measures overall pattern shape consistency but does not explicitly distinguish peak shift from peak width distortion.

Mean Peak-Position Deviation Let the positions of the m-th matched peak pair be $\hat{\theta}_m$ and θ_m. The mean peak-position deviation is defined as

$$\overline{\Delta 2\theta} = \frac{1}{M}\sum_{m=1}^{M} |\hat{\theta}_m - \theta_m|. \tag{42}$$

This metric directly reflects local peak geometry fidelity.

FWHM Error Let the full widths at half maximum of a matched peak pair be $\widehat{\mathrm{FWHM}}_m$ and FWHM_m. The FWHM error is

$$\Delta\mathrm{FWHM} = \frac{1}{M}\sum_{m=1}^{M} |\widehat{\mathrm{FWHM}}_m - \mathrm{FWHM}_m|. \tag{43}$$

This metric reflects peak width recovery and is directly related to crystallographic line shape fidelity.

Top-1 / Top-10 Retrieval Accuracy Let $\hat{s}_n$ be a predicted single-phase pattern and r_j a reference pattern in the database. An initial similarity function is first employed for coarse candidate recall; we use cosine similarity as the first-stage retrieval score:

$$\mathrm{sim}_{\mathrm{base}}(\hat{s}_n, r_j) = \frac{\hat{s}_n^\top r_j}{\|\hat{s}_n\|_2 \, \|r_j\|_2}. \tag{44}$$

Based on this score, the top M candidates are selected to form a candidate set $\mathcal{C}_M(\hat{s}_n)$. A reranking module then recomputes scores for entries in the candidate set:

$$\mathrm{sim}_{\mathrm{rerank}}(\hat{s}_n, r_j) = f_{\mathrm{rerank}}(\hat{s}_n, r_j), \quad r_j \in \mathcal{C}_M(\hat{s}_n), \tag{45}$$

and the final retrieval list is obtained by reordering according to $\mathrm{sim}_{\mathrm{rerank}}$.

B.8 Robustness Analysis

We evaluate the robustness of our method in two challenging scenarios: resolving low-content phases and separating highly overlapping component patterns.

Table 7 reports the minimum phase proportions considered, ranging from 0.15 to 0.05. As the proportion of minor phases decreases, Pearson correlation drops by 1.4%, 4.1%, and 6.8% for $K = 2, 3, 4$, respectively, while Top-1 accuracy decreases by 2.03, 6.02, and 8.24 percentage points. The larger degradation at higher K arises from the increased difficulty in resolving weak reflections,

Table 7: Effect of the minimum phase proportion.

K	Min.w_k	Pearson ↑	Peak Shift ↓	ΔFWHM ↓	Top-1 (%) ↑
2	0.15	0.9691	0.0272	2.4330	87.92
	0.10	0.9644	0.0260	2.3560	87.79
	0.05	0.9553	0.0245	2.2186	85.89
3	0.15	0.9101	0.0389	3.7836	81.13
	0.10	0.8971	0.0369	3.5879	78.69
	0.05	0.8727	0.0343	3.3141	75.11
4	0.15	0.8322	0.0480	5.2410	69.68
	0.10	0.8107	0.0457	4.8147	66.19
	0.05	0.7755	0.0422	4.3668	61.44

which are progressively masked by dominant and overlapping peaks from other components. This behavior is physically expected, as low-content phases produce weaker diffraction signals that are more susceptible to interference from dominant phases, background contributions, and experimental uncertainties.

For the similarity-controlled benchmark, additional components are selected from the farthest, intermediate, or nearest 10% of test phases ranked by cosine similarity. Table 8 shows that the medium condition closely matches the original random-mixture benchmark. From medium to high similarity, Pearson decreases from 0.9097 to 0.8739 for $K = 3$ and from 0.8275 to 0.7945 for $K = 4$, while Top-1 decreases by 4.01 and 5.95 percentage points. For $K = 2$, Top-1 accuracy remains nearly unchanged, although reconstruction and peak-based metrics show noticeable degradation. These results indicate that both low phase proportions and high pattern similarity become increasingly challenging as the number of constituent phases grows.

Table 8: Effect of inter-phase pattern similarity.

K	Sim.	Cosine	Pearson ↑	Peak Shift ↓	ΔFWHM ↓	Top-1 (%) ↑
2	Low	0.1439	0.9839	0.0222	1.5880	85.94
	Med.	0.2866	0.9692	0.0272	2.4665	88.43
	High	0.5208	0.9542	0.0301	2.9770	88.39
3	Low	0.1508	0.9554	0.0298	2.2315	81.95
	Med.	0.2942	0.9097	0.0395	3.8291	81.34
	High	0.5221	0.8739	0.0441	4.7655	77.33
4	Low	0.1535	0.9065	0.0358	2.7895	75.42
	Med.	0.3000	0.8275	0.0494	5.2740	69.40
	High	0.5233	0.7945	0.0550	6.5606	63.45

Additional results demonstrate that the model effectively suppresses spurious peaks in inactive slots (Appendix B.9.2), achieves the best decomposition and identification performance when using the full 2θ range (Appendix B.9.3), and remains robust to threshold selection (Appendix B.9.4).

B.9 Additional Robustness Analysis

This section provides additional sensitivity analyses concerning inactive output slots, the angular range of the input pattern, and the activity threshold used to select predicted phases.

B.9.1 Profile-Factor Evaluation

To further quantify the agreement between the predicted component profiles and their corresponding ground-truth patterns, we report the conventional profile factor R_p and the weighted profile factor R_{wp}. For a matched predicted profile $\hat{y}$ and ground-truth profile y, sampled at N diffraction-angle

positions, these metrics are defined as

$$R_p = \frac{\sum_{i=1}^{N} |y_i - \hat{y}_i|}{\sum_{i=1}^{N} |y_i|} \times 100\%, \qquad R_{wp} = \sqrt{\frac{\sum_{i=1}^{N} w_i \left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{N} w_i y_i^2}} \times 100\%, \tag{46}$$

where y_i and $\hat{y}_i$ denote the ground-truth and predicted intensities, respectively, at the i-th diffraction-angle position, and w_i denotes the corresponding weight. While R_p measures the normalized absolute profile discrepancy, R_{wp} places greater emphasis on larger intensity deviations through its squared-error formulation.

Table 9: Profile factors for mixtures containing different numbers of phases.

K	R_p (%) $\downarrow$	$R_{wp}^{(w=1)}$ (%) $\downarrow$
2	13.20	10.17
3	27.74	23.60
4	45.04	38.89

As K increases, both profile factors increase, reflecting the greater difficulty of decomposing mixtures containing a larger number of strongly overlapping diffraction patterns. Nevertheless, the profile-factor results are consistent with the performance trends observed using Pearson correlation, peak-position error, and peak-width error.

B.9.2 Behavior of Inactive Output Slots

We first examine whether slots classified as inactive retain peak-like signals and whether such signals should be interpreted as additional crystalline phases. After permutation-invariant matching, each output slot is associated either with a real phase target or with a zero-padding target. The latter corresponds to an extra output slot that should ideally be suppressed, whereas a real-phase target classified as inactive represents a missed phase.

To quantify the residual structure in these outputs, peaks are detected using a normalized height threshold of $h > 0.05$ and a non-maximum suppression (NMS) window of 10 samples. Table 10 reports the inactive rate, median relative energy, and peak-count statistics for each PIT-matched target category.

Table 10: Statistics of output slots classified as inactive after PIT matching. Padding denotes an extra slot matched to a zero target, whereas a real phase denotes a true component missed by the activity prediction. Peaks are detected using $h > 0.05$ and an NMS window of 10 samples.

K	PIT-matched target	Inactive rate (%)	Median relative energy	No peak (%)	≥ 2 peaks (%)
2	Padding (extra slot)	96.70	0.0011	94.50	3.29
	Real phase (missed)	1.66	0.0246	54.35	33.93
3	Padding (extra slot)	73.35	0.0210	67.11	18.31
	Real phase (missed)	5.95	0.0329	45.09	39.56

Most padding-matched inactive outputs are strongly suppressed. For $K = 2$, 94.50% of these outputs contain no detectable peaks, and their median relative energy is only 0.0011. The $K = 3$ setting is more challenging and retains more weak peak fragments; nevertheless, 67.11% of the corresponding outputs remain peak-free, and the median relative energy is still low at 0.0210. In contrast, inactive outputs matched to real phases exhibit higher relative energy and substantially more multi-peak structure, indicating that they correspond to activity-detection false negatives rather than hallucinated additional phases.

This behavior is also constrained by the mask-based output parameterization:

$$m_k = \sigma(U_k), \qquad \hat{y}_k = m_k \odot x, \qquad 0 \leq \hat{y}_k \leq x. \tag{47}$$

Each predicted contribution is therefore a pointwise attenuated component of the observed mixture and cannot introduce intensity beyond the input pattern. Weak structures retained by inactive slots

must originate from intensity already present in the mixture. They are thus most plausibly attributed to unresolved residual intensity, background or noise retained in the input, or duplicated allocation of reflections from existing phases, rather than to independently generated crystalline phases.

B.9.3 Sensitivity to the Input Angular Range

We next evaluate the effect of the available diffraction-angle range. In addition to the complete 2θ interval of $10°$–$80°$, we consider four representative restricted windows: $10°$–$20°$, $30°$–$40°$, $50°$–$60°$, and $70°$–$80°$. The results are reported in Table 11.

Table 11: Decomposition and phase-identification performance under different 2θ input ranges. Bold values denote the best result for each mixture cardinality K.

K	2θ range	Pearson ↑	Peak shift (°) ↓	ΔFWHM (°) ↓	Top-1 (%) ↑	Top-10 (%) ↑
2	Full ($10°$–$80°$)	**0.9691**	**0.0272**	**2.4330**	**87.92**	**92.50**
	$10°$–$20°$	0.6503	0.0664	7.3907	12.10	20.17
	$30°$–$40°$	0.7485	0.0383	5.7770	30.73	41.27
	$50°$–$60°$	0.7143	0.0389	6.5156	39.40	51.52
	$70°$–$80°$	0.6929	0.0433	6.0569	45.83	59.59
3	Full ($10°$–$80°$)	**0.9101**	**0.0389**	**3.7836**	**81.13**	**86.81**
	$10°$–$20°$	0.5505	0.0716	9.9861	9.29	15.10
	$30°$–$40°$	0.6084	0.0454	7.2331	16.91	23.66
	$50°$–$60°$	0.5637	0.0468	8.3543	22.83	30.58
	$70°$–$80°$	0.5465	0.0527	8.3102	27.74	37.13
4	Full ($10°$–$80°$)	**0.8322**	**0.0480**	**5.2410**	**69.68**	**76.21**
	$10°$–$20°$	0.4903	0.0713	11.3767	7.22	11.63
	$30°$–$40°$	0.5043	0.0506	9.0105	8.87	13.00
	$50°$–$60°$	0.4671	0.0524	9.5928	12.53	17.90
	$70°$–$80°$	0.4516	0.0567	9.7295	16.12	22.23

For $K = 2$–4, the complete $10°$–$80°$ input consistently achieves the best performance across all decomposition and identification metrics. This result demonstrates that the angular regions contain complementary information that cannot be recovered from any isolated window.

Among the restricted ranges, $30°$–$40°$ consistently provides the highest Pearson correlation and the lowest peak-position and peak-width errors, indicating the best preservation of the overall pattern profile and local peak geometry. By contrast, $70°$–$80°$ consistently produces the highest Top-1 and Top-10 retrieval accuracy among the restricted ranges. The latter observation may be attributed to the greater angular sensitivity of high-angle reflections to small differences in lattice spacing, despite the smaller number and generally lower intensity of reflections in this region. Because no restricted window is optimal for both reconstruction and retrieval, the complete diffraction range is required for robust multiphase analysis.

B.9.4 Sensitivity to the Activity Threshold

The activity probability p_k is used to select the final set of predicted phases:

$$\hat{\mathcal{Y}}^* = \{\hat{y}_k \,|\, p_k > \tau\}\,, \tag{48}$$

where τ denotes the activity threshold. To evaluate sensitivity to this parameter, we vary τ from 0 to 0.8 and report the corresponding performance in Table 12.

In the controlled fixed-K experiments, we use $\tau = 0$ so that no candidate component is discarded by activity thresholding. This protocol prevents phase-number selection errors from being conflated with the intrinsic decomposition performance and retains weak but valid phases under severe peak overlap and noise.

As τ increases, Pearson correlation decreases because an increasing number of low-confidence but valid components are removed. Retrieval accuracy remains approximately stable at low thresholds and subsequently decreases, whereas the peak-shift and FWHM errors vary only marginally. This

Table 12: Sensitivity to the activity threshold τ. Bold values indicate the setting used in the fixed-K experiments rather than the numerical optimum of each individual column.

K	τ	Pearson ↑	Peak shift (°) ↓	ΔFWHM (°) ↓	Top-1 (%) ↑	Top-10 (%) ↑
2	**0.0**	**0.9691**	**0.0272**	**2.4330**	**87.92**	**92.50**
	0.2	0.9668	0.0274	2.4673	88.10	92.28
	0.4	0.9630	0.0274	2.4687	87.90	92.02
	0.5	0.9609	0.0274	2.4670	87.74	91.83
	0.6	0.9580	0.0273	2.4645	87.51	91.58
	0.8	0.9477	0.0272	2.4536	86.69	90.63
3	**0.0**	**0.9101**	**0.0389**	**3.7836**	**81.13**	**86.81**
	0.2	0.8959	0.0389	3.7997	80.57	86.17
	0.4	0.8800	0.0389	3.8013	79.71	85.05
	0.5	0.8710	0.0389	3.8015	79.14	84.37
	0.6	0.8598	0.0388	3.8008	78.36	83.39
	0.8	0.8345	0.0387	3.7834	76.31	81.06
4	**0.0**	**0.8322**	**0.0480**	**5.2410**	**69.68**	**76.21**
	0.2	0.8272	0.0484	5.2352	69.70	76.07
	0.4	0.8206	0.0484	5.2358	69.45	75.73
	0.5	0.8150	0.0484	5.2367	69.20	75.42
	0.6	0.8075	0.0483	5.2329	68.88	75.02
	0.8	0.7794	0.0481	5.2056	67.53	73.38

indicates that thresholding primarily affects phase retention rather than the local peak geometry of the components that remain selected.

When the true number of phases is known, returning the top-K slots ranked by p_k avoids the need to select a global threshold. When K is unknown, the range $\tau \in [0, 0.2]$ provides a practical operating region that suppresses low-confidence outputs while retaining most valid phases. For deployment under substantial domain shift, the threshold should be calibrated using a representative validation set.

B.10 Dataset Analysis

Table 13 summarizes the basic statistics of the simulated dataset *UniqCryLabeled* and the experimental dataset *UniqRruffCrystal*. The experimental subset contains only 662 samples, whereas the simulated library contains 100,315 structures. The experimental subset also has narrower elemental coverage, with 55 unique elements compared with 84 in the simulated library. In addition, the experimental subset has lower median density (3.60 vs. 4.44 g/cm^3), smaller median volume per atom (12.3 vs. 15.1 Å^3), and a larger median number of atoms per unit cell (36 vs. 20). These statistics indicate that the experimental set is not a uniform sample of the simulated structure space, but exhibits clear biases in both chemical composition and structural complexity.

Table 13: Statistical comparison of the simulated and experimental datasets.

Metric	*UniqCryLabeled*	*UniqRruffCrystal*
Total entries	100,315	662
Unique elements	84	55
Median density (g/cm^3)	4.44	3.60
Median Vol/Atom (Å^3)	15.1	12.3
Median N_{atoms}	20	36
Dominant crystal system	Monoclinic	Monoclinic
Most frequent space group	1	14

Figure 3 further shows systematic differences in categorical and continuous attributes. Both datasets are dominated by monoclinic structures, but the experimental subset has higher proportions of cubic and trigonal systems and lower proportions of triclinic, tetragonal, and hexagonal systems. The high-frequency space-group distribution also changes substantially: space group 1 is the most common in

the simulated library, whereas space group 14 dominates the experimental subset. The continuous attributes follow the same trend. The experimental subset is shifted toward lower densities, smaller volumes per atom, and larger numbers of atoms per unit cell. These results indicate that the two domains differ not only in sample scale, but also in symmetry composition and local structural scale.

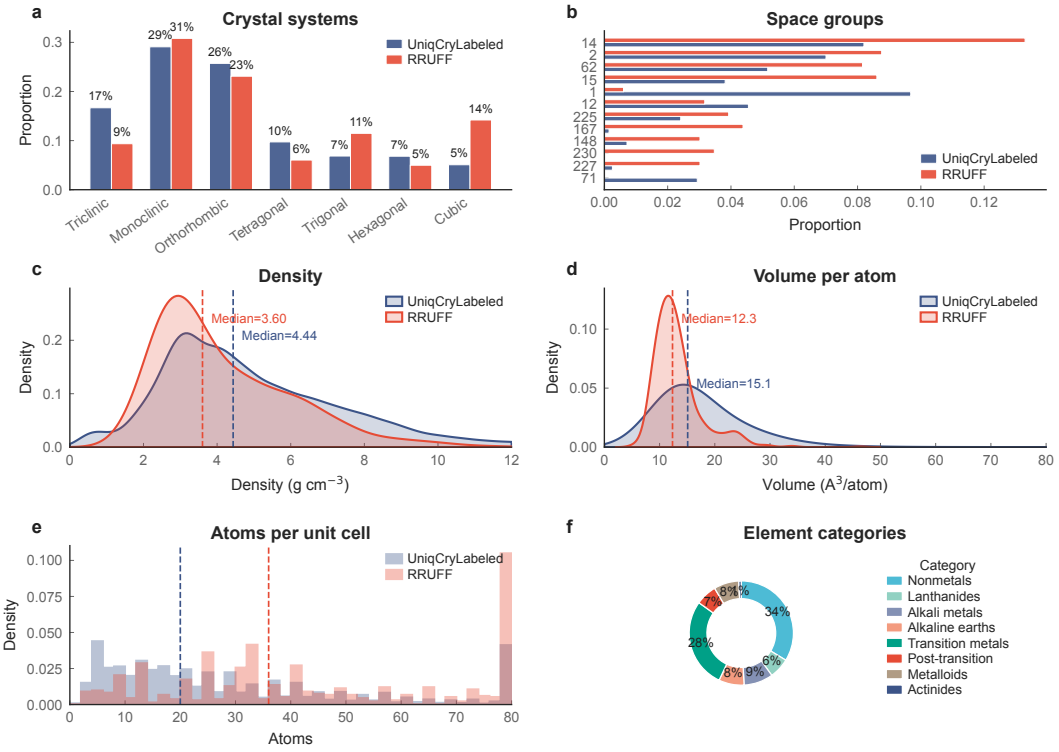


Figure 3: Comparison of categorical and physical attribute distributions between simulated and experimental datasets. Top: crystal system and high-frequency space group distributions; Bottom: density, volume per atom, and number of atoms per unit cell. The experimental subset exhibits systematic shifts in both symmetry composition and structural scales.

The chemical distributions show a similar pattern. The simulated library spans a broader elemental range, whereas the experimental subset is concentrated in typical mineral chemistries. O and Si are frequent in both datasets, but the experimental subset is relatively enriched in common mineral-forming elements such as Ca, Mg, H, P, Na, and Fe. This indicates that the experimental domain occupies a more localized region of chemical space rather than broadly covering the simulated compositional distribution.

Figure 5 provides a joint manifold view of the two domains. The experimental samples overlap with the global simulated manifold, but they do not cover it uniformly. Instead, they concentrate in several localized high-density regions. The crystal-system labels and the density and volume-per-atom gradients remain spatially organized on the same manifold, which suggests that the experimental subset occupies a localized region of the full simulated structure space rather than forming a uniform subset. Taken together, the observed domain gap is reflected in symmetry composition, structural scale, chemical composition, and joint manifold coverage.

B.11 Quantitative Decomposition

This section presents quantitative whole-pattern decomposition results obtained by XDecomposer. Figures 6, 7, and 8 show representative examples for mixture complexities of $K = 2$, 3, and 4, respectively.

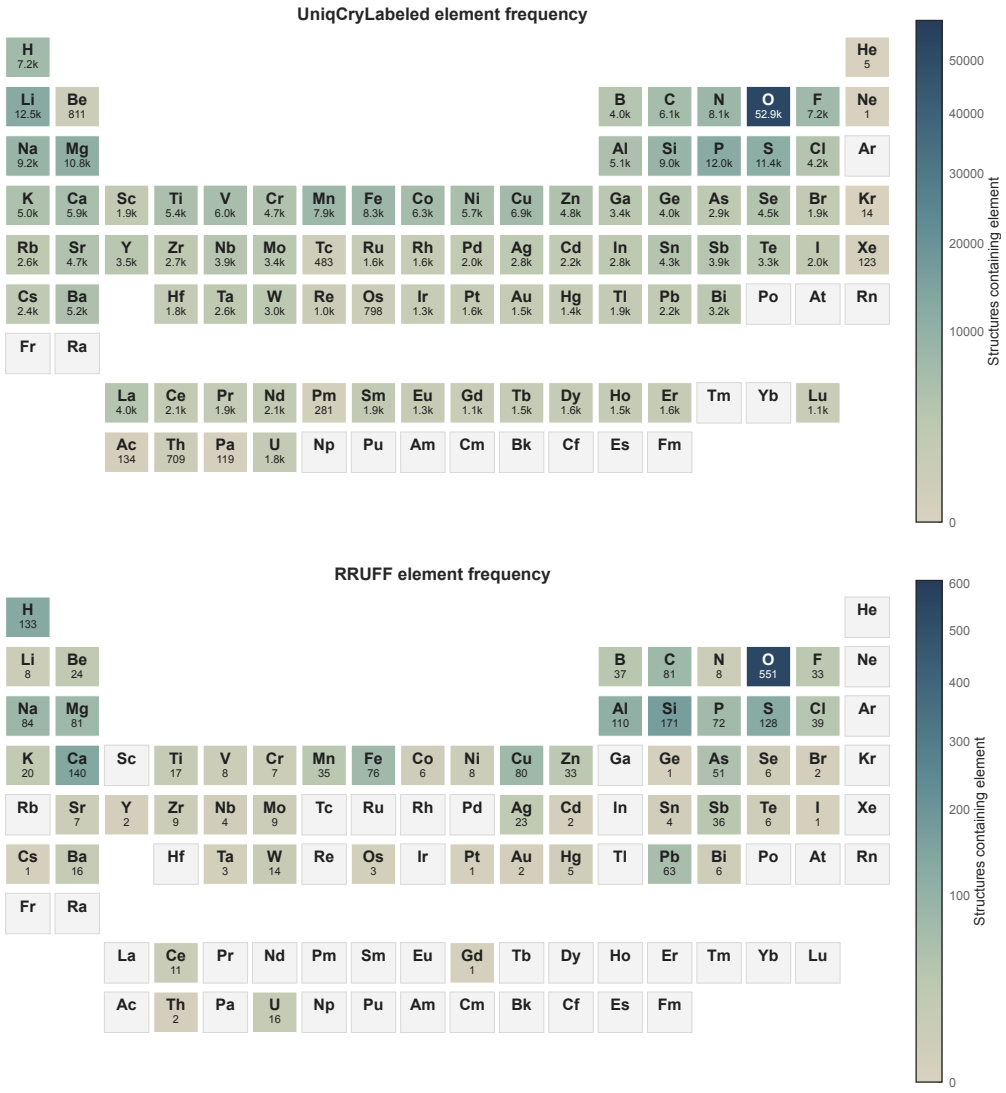


Figure 4: Element frequency in the simulated library (top) and the experimental RRUFF subset (bottom). The experimental subset covers a narrower chemical range and is concentrated around common mineral-forming elements.

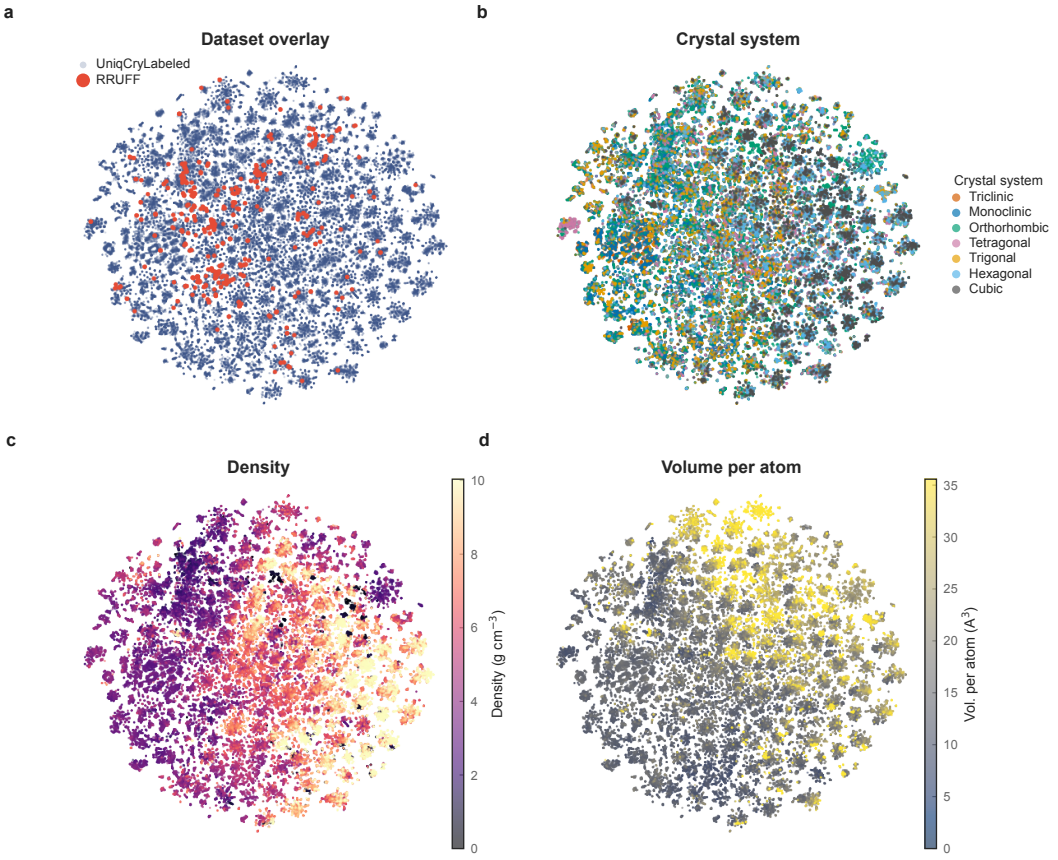


Figure 5: Joint manifold visualization of the simulated and experimental datasets. The experimental subset overlaps with the global simulated manifold but concentrates in several localized regions rather than covering the full theoretical structure space uniformly.

C Additional Knowledge

C.1 Search-Match Approach

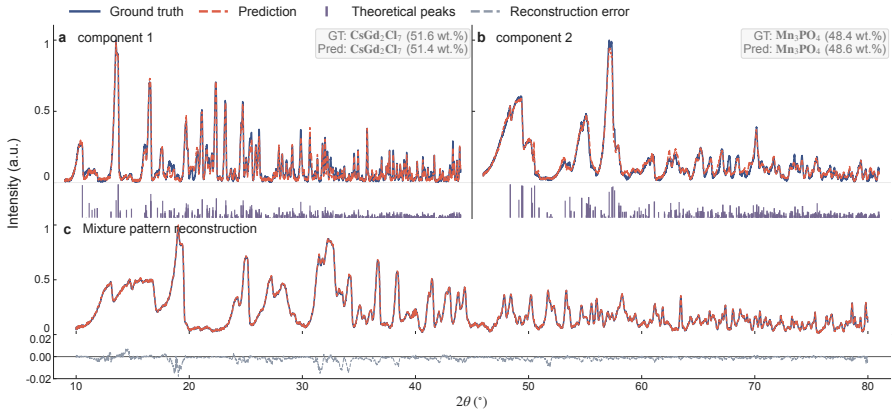
The predominant method for conventional phase identification in X-ray powder diffraction patterns is the search-match approach, which involves three key steps.

First, (d, I) values, representing interplanar spacing (d) and peak intensity (I), are extracted from the diffraction pattern. Next, potential phases are identified by searching diffraction databases for matching d values, such as those in the Hanawalt indexes [61]. Candidate phases are then compared to the pattern's (d, I) values using a scoring system, helping to select the best match. This process continues until satisfactory alignment is achieved for most (d, I) values. Despite advances in computer technology enabling novel qualitative methods, the core goal remains to match experimental data with database entries and compute a corresponding score.

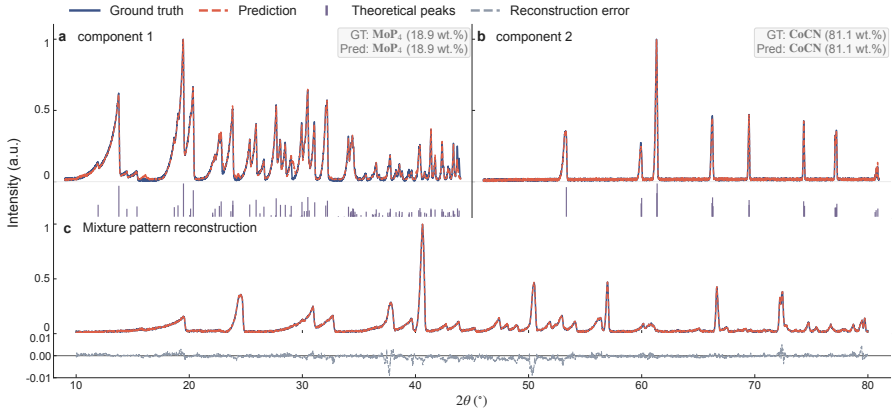
C.2 In-library and Out-of-library Tasks

For phase identification, the ultimate goal is to map an experimental diffraction pattern, or a decomposed phase pattern in the case of multiphase materials, to a specific crystal structure. Depending on whether the target structure has been previously recorded, this task can be categorized into two distinct scenarios: in-library identification and out-of-library identification, as defined in our previous work [10].

In-library identification aims to identify unknown materials by matching their XRD patterns against a library of known crystal structures. This represents the most widely used scenario in practical



a Balanced mixture: $CsGd_2Cl_7$–$MnPO_4$



b Highly imbalanced mixture: MoP_4–CoCN

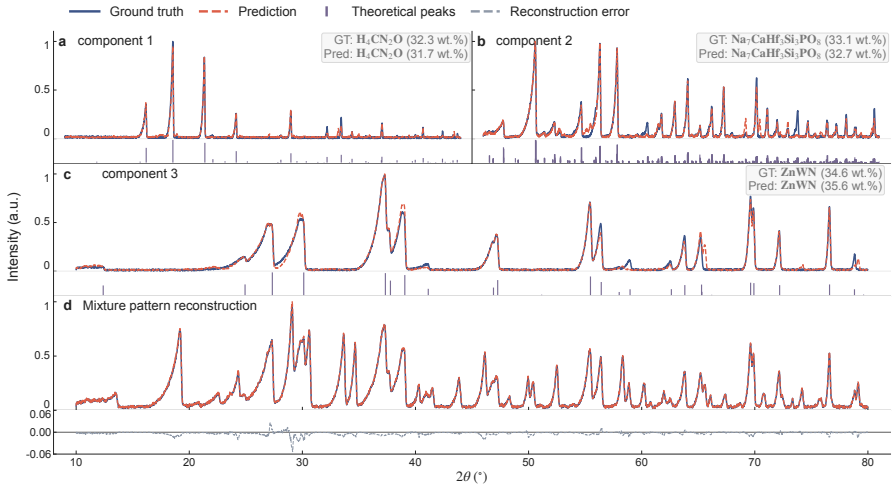
Figure 6: Whole-pattern decomposition results for binary mixtures with $K = 2$.

crystallographic analysis. Since the establishment of the Powder Diffraction File (PDF) database in 1938, experimentally determined crystal structures have been systematically collected and archived, enabling structure identification through diffraction pattern retrieval and comparison. With the development of computational methods, numerous search-match software packages [62] have been developed to accelerate this process. The primary challenge in this scenario is not discovering new structures, but achieving robust identification under realistic experimental variations, including instrumental noise, preferred orientation, lattice strain, peak broadening, and other measurement-related factors.

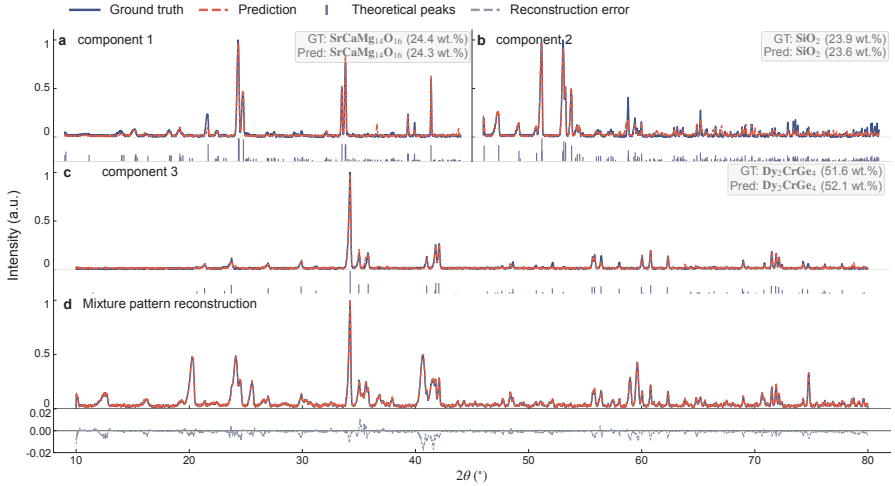
In contrast, out-of-library identification refers to the identification of crystal structures that are absent from existing diffraction databases. Since no reference patterns are available for direct matching, this task requires solving the inverse crystallographic problem, where diffraction patterns must be interpreted to recover an unknown three-dimensional atomic arrangement. Typically, this involves multiple stages, including symmetry determination, estimation of lattice parameters and space groups, chemical composition inference, structure solution, and subsequent refinement and validation. Although recent advances in generative crystal structure prediction have shown promising capabilities for exploring unknown materials, experimentally validated predictions of entirely novel crystal structures remain extremely limited. Therefore, out-of-library identification represents a much more challenging and less frequently encountered problem compared with in-library identification.

C.3 Crystal Structure Determination via PXRD

Crystal structure determination via PXRD typically involves two main steps:



a Balanced mixture: H_4CN_2O–$Na_7CaHf_3Si_3PO_4$–ZnWN



b Highly imbalanced mixture: $SrCaMg_{14}O_{16}$–SiO_2–Dy_2CrGe_4

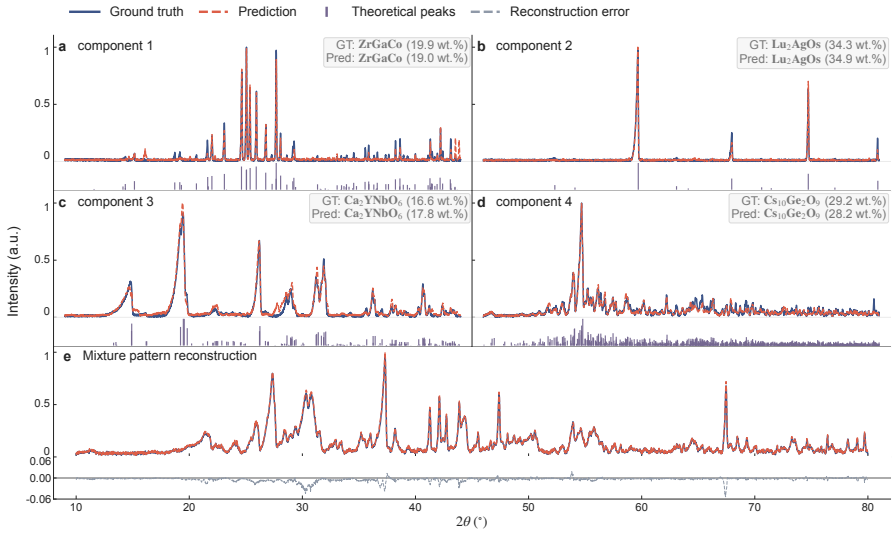
Figure 7: Whole-pattern decomposition results for ternary mixtures with $K = 3$.

1. **Initial structure identification:** interpreting the experimental PXRD pattern to propose a plausible (partial or approximate) structure.
2. **Structure refinement:** using the proposed structure as input for further relaxation and optimization to best fit the experimental data.

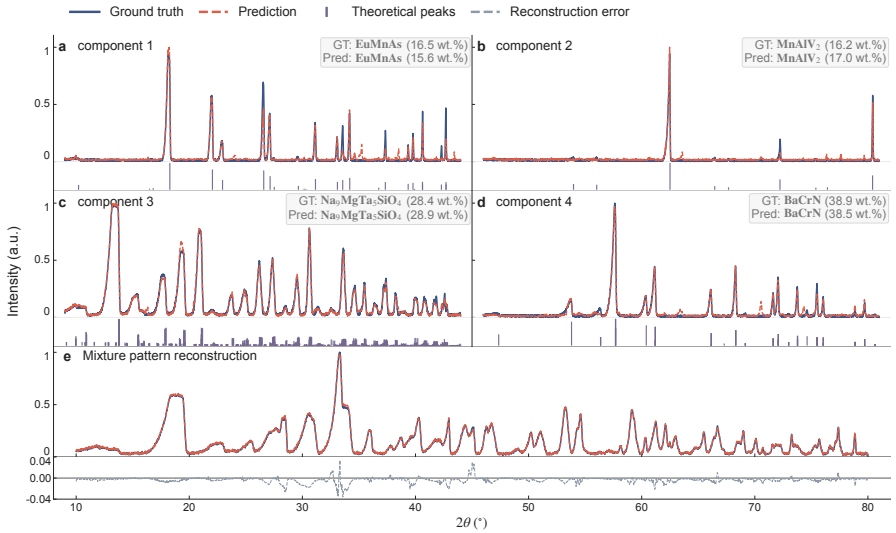
The first step focuses on identifying a rough structural model, while the second step fine-tunes this model to capture structural details. It is well understood that no two crystals are exactly identical, even for the same compound. For example, NaCl crystals synthesized under different conditions can exhibit structural variations and defects that must be resolved through refinement.

However, refinement cannot begin from scratch. It works by minimizing the differences between the observed PXRD profile and the theoretical pattern calculated from a structural model. Once this fitting process converges, the refined parameters represent the only physically meaningful and experimentally determinable information. Therefore, an accurate initial guess of the crystal structure is essential for successful refinement. This task—*structure identification*—is the focus of this paper.

The purpose of structure identification is to recognize known reference crystals from their PXRD patterns. For instance, identifying that a given pattern belongs to NaCl, without needing to determine precise lattice parameters or defects. For the past few decades, a **search-match strategy** has been



a $ZrGaCo–Lu_2AgOs–Ca_2YNbO_6–Cs_{10}Ge_2O_9$



b $EuMnAs–MnAlV_2–Na_9MgTa_5SiO_4–BaCrN$

Figure 8: Whole-pattern decomposition results for quaternary mixtures with $K = 4$.

widely adopted. By comparing experimental patterns against large crystal structure databases, this approach retrieves candidate structures that best match the observed data and recommends the most similar ones. This process essentially confirms the tested sample based on historical crystallographic knowledge and is referred to as *in-library identification*.

Only when all known structures fail to match does *out-of-library identification* begin. Traditionally, this is a labor-intensive process involving the determination of lattice geometry, ideal chemical composition, and iterative refinement with trial and error. In recent years, generative methods, those that propose structures from scratch, have been increasingly applied in such situations. However, experimental crystal structure determination remains highly challenging, and generation from scratch carries a high risk of producing incorrect structures.

Current methods mainly focus on single-phase identification, where a PXRD pattern contains only one dominant phase. However, for the multiphase scenario considered in this work, most existing approaches still rely on an iterative subtract-and-match strategy: a candidate phase is first identified from the observed mixture, then its contribution is estimated and subtracted, and the residual pattern is used for the next round of matching and identification. This strategy is essentially a trial-and-error process and is highly sensitive to errors in early-stage phase identification. In multiphase mixtures, strong peak overlap and mutual interference among components greatly reduce the reliability of prior phase identification. Even when a candidate phase is correctly retrieved, its subtraction is often inaccurate because the true phase proportion is unknown, and the theoretical reference pattern may differ significantly from its actual experimental contribution due to peak broadening, background effects, preferred orientation, and instrumental variation.

As a result, error accumulation is common, and the residual pattern may become increasingly unreliable for subsequent identification steps. This limitation highlights that multiphase identification should not be treated as a recursive single-phase matching problem. Instead, it requires joint decomposition of all constituent single-phase signals directly from the mixed pattern, enabling simultaneous recovery of component contributions and more reliable downstream phase identification.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Justification: The main claims in the abstract and introduction are aligned with the proposed method and the reported experiments. We clearly state that XDecomposer addresses prior-free multiphase XRD decomposition and evaluate this claim on both simulated and experimental datasets.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the limitations of the current framework, including the use of a fixed maximum number of output slots, the difficulty of handling complex experimental distortions, and the remaining sim-to-real gap. These limitations are discussed in the limitations and future work section.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer:[N/A]

   Justification: The paper does not present formal theoretical results, theorems, or proofs. The contribution is mainly methodological and empirical.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides the information needed to reproduce the main experimental results, including the model design, data construction protocol, training setup, evaluation metrics, and baseline settings. Additional details are provided in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We provide code, configuration files, and evaluation scripts for reproducing the main experiments. For datasets that cannot be redistributed directly, we describe the data sources, preprocessing steps, and generation protocol needed to reconstruct the experimental setup.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies the key training and testing details, including data splits, online mixing strategy, model hyperparameters, optimizer settings, batch sizes, and evaluation metrics. Full implementation details are further provided in the appendix and released code.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: For experimental data, we report results using five-fold cross-validation and provide standard deviations across folds. These statistics quantify the stability of the method under different data splits.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper reports the computational setup used for training and evaluation, including the GPU-based training environment and the main training configuration. This information is sufficient to estimate the resources needed to reproduce the experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research follows the NeurIPS Code of Ethics. It uses crystallographic and XRD data and does not involve personal data, human subjects, surveillance, or sensitive demographic information.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses the potential positive impact of improving automated multiphase XRD analysis for materials discovery and high-throughput characterization. It also acknowledges that incorrect phase decomposition may affect downstream scientific interpretation and should be validated by domain experts.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The released model and data processing pipeline do not pose a high risk for misuse comparable to generative models, language models, or scraped personal datasets. The work is focused on scientific XRD pattern analysis.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper cites the existing datasets, software, and baseline methods used in the experiments. We also describe the use of external crystallographic and XRD resources and follow their access conditions and citation requirements.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The paper introduces XDecomposer together with code, configuration files, and evaluation scripts. The released materials include documentation and instructions for training, testing, and reproducing the main results.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing experiments or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve human subjects or personally identifiable data.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method does not use LLMs as an important, original, or non-standard component.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.