A Masked Mixture Model for Compact and Accurate Matrix Factorization

Yong-chan Park
Seoul National University
Seoul, Republic of Korea
wjdakf3948@snu.ac.kr

SeungJoo Lee
Seoul National University
Seoul, Republic of Korea
hera0131@snu.ac.kr

Jeongyoung Lee
Seoul National University
Seoul, Republic of Korea
ljklee@snu.ac.kr

U Kang
Seoul National University
Seoul, Republic of Korea
ukang@snu.ac.kr

Abstract

Matrix factorization (MF) is a widely used backbone for modeling large relational data due to its simplicity, scalability, and interpretability. However, classical MF uses a single shared latent basis, which can be overly restrictive for heterogeneous matrices. In this paper, we propose Masked Mixture Factorization (MMF), a lightweight yet effective MF variant that adapts to heterogeneous interactions through instance-wise latent gating, substantially improving accuracy under the same parameter budget while retaining MF's scalability. We provide theoretical results on MMF's expressivity and identifiability, clarifying when masking expands representational power and when the model is recoverable. Extensive experiments on matrix reconstruction, matrix completion, and Top-N recommendation show consistent gains over strong baselines.

CCS Concepts

• **Computing methodologies** → **Factorization methods**; • **Information systems** → *Recommender systems*; *Collaborative filtering*; • **Mathematics of computing** → *Dimensionality reduction.*

Keywords

Matrix factorization, Matrix completion, Low-rank models, Masking mechanisms, Representation learning, Data compression, Dimensionality reduction, Collaborative filtering, Recommender systems

ACM Reference Format:
Yong-chan Park, SeungJoo Lee, Jeongyoung Lee, and U Kang. 2026. A Masked Mixture Model for Compact and Accurate Matrix Factorization. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD 2026), August 9–13, 2026, Jeju Island, Republic of Korea.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3770855.3817710

1 Introduction

Matrix factorization (MF) is a widely used workhorse for uncovering low-rank structure in large relational datasets and has become a core tool in large-scale data mining, including graph learning [21, 22, 32, 33, 41], time-series analysis [13, 15, 42, 43, 53], tensor factorization [12, 18, 19, 30, 44, 45], speech and language modeling [10, 14, 20, 34], model compression [23, 63], and recommendation [17, 27, 28, 31, 62]. By representing the observed matrix X as a low-rank product $UV^\top$, MF provides compact embeddings and enables scalable optimization, making it a strong and practical baseline. However, MF also makes a restrictive structural assumption: all instances are represented using the *same* fixed set of latent dimensions. In heterogeneous data, this global sharing can be suboptimal: different users, items, or subpopulations may rely on different aspects of the latent space, and forcing every instance to use the same basis can blur structure and hurt accuracy unless the rank is increased substantially.

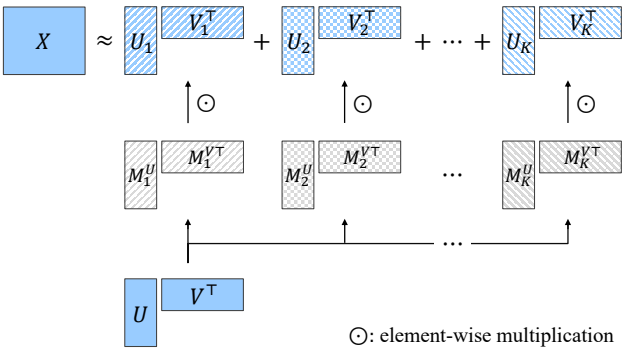


Figure 1: Overview of Masked Mixture Factorization (MMF). It approximates the target matrix X as a sum of K masked factorization components. Starting from shared base factors U and V, each component k applies instance-dependent masks M_k^U and M_k^V, producing masked factors U_k and V_k. The final result is obtained by summing the K masked products.

This motivates a basic question: *should all instances use the same latent dimensions to the same extent?* Standard MF implicitly answers "yes"—every prediction combines all dimensions through the same bilinear mechanism. Yet, when only a subset of dimensions is relevant for a given instance, uniformly using the entire basis wastes capacity and blurs structure. We argue that a better inductive bias is instance-wise allocation of representational capacity: for each instance, selectively emphasize the dimensions that matter and suppress those that do not. Such adaptive allocation should ideally be (i) differentiable and trainable end-to-end, (ii) parameter-efficient so that gains are not merely due to increased capacity, and (iii) compatible with standard MF training pipelines.

In this paper, we propose Masked Mixture Factorization (MMF), a lightweight yet effective generalization of MF that enables instance-wise dimension selection while preserving MF's bilinear structure (see Figure 1). The core idea is to replace a single global factorization with a *mixture of masked factorizations*. Drawing inspiration from the Mixture-of-Experts framework [11], our approach allows each component to use the same underlying latent factors but apply instance-dependent masks that gate active dimensions. The final

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.
KDD 2026, Jeju Island, Republic of Korea.
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2259-2/2026/08
https://doi.org/10.1145/3770855.3817710

reconstruction is obtained by summing over the contributions from all masked components. This design establishes a structured form of adaptive capacity allocation: rather than requiring every instance to express itself through the entire latent basis, MMF allows different instances to activate different subsets of latent dimensions—potentially differently across mixture components—while retaining the simplicity and scalability of factorization-based learning.

MMF occupies a distinct point in the design space of MF models. Prior approaches often increase expressivity by adding heavy nonlinear interaction networks or extensive side information, which complicates training and obscures the factorization structure [6, 59]. Instead, MMF directly targets the bottleneck of classical MF (i.e., the uniform sharing of latent dimensions) using a lightweight masking mechanism that remains optimization-friendly. Furthermore, MMF can be instantiated under capacity-controlled settings (e.g., parameter budgets matched to MF), so performance gains reflect a stronger inductive bias rather than a trivial increase in parameters.

We evaluate MMF in two complementary regimes. First, in matrix reconstruction, we isolate representational power and show that MMF achieves significantly lower reconstruction error than strong baselines. Second, in matrix completion and Top-N recommendation on standard benchmarks, MMF improves predictive accuracy over representative completion and ranking methods. Together, these results support the central hypothesis of this work: introducing instance-wise latent dimension selection within bilinear factorization substantially improves modeling of heterogeneous matrices without sacrificing the practical advantages of MF.

We summarize our contributions as follows:

- **Methodology.** We propose Masked Mixture Factorization (MMF), a principled extension of matrix factorization that endows each instance with instance-specific masks to gate latent dimensions, enabling adaptive allocation of representational capacity while preserving the simplicity of bilinear factorization.
- **Theory.** We provide theoretical properties of MMF, establishing its expressivity and identifiability, which clarify when and how masked mixtures expand the representational power of MF and under what conditions the model parameters are recoverable.
- **Experiments.** We conduct extensive experiments on matrix reconstruction, matrix completion, and Top-N recommendation, demonstrating that MMF yields consistent performance improvements over strong baselines across standard benchmarks.

Our source code and supplementary material are provided at https://github.com/snudatalab/MMF.

2 Related Work

Matrix Factorization. Matrix factorization (MF) is a dominant paradigm for collaborative filtering (CF) and rating prediction, built on the low-rank assumption that user-item interactions can be explained by a small number of latent factors. Classical approaches include SVD-based models [50] and their practical refinements, most notably bias-aware MF, which explicitly models user/item biases to improve prediction quality [25]. Probabilistic perspectives such as Probabilistic Matrix Factorization (PMF) [39] and Bayesian extensions [48] further formalize MF as a latent-variable model with uncertainty modeling, while alternative likelihoods (e.g., Poisson) have been explored to better match data characteristics [7].

A key limitation of standard MF is often overlooked: it offers no explicit mechanism for instance-wise selection of latent dimensions. Most variants retain a single global latent space and rely on regularization or larger rank to cope with heterogeneity, leaving the question of selective dimension usage largely unaddressed. Local low-rank methods such as LLORMA [29] partially mitigate this by combining multiple local factorizations to better fit heterogeneous regions. However, they do not directly realize instance-level dimension gating within a single model. Furthermore, they typically require maintaining multiple local models and partitioning schemes, adding complexity without addressing the core issue.

Neural and Adaptive Alternatives. Beyond linear factorization, a second major line of work replaces or augments MF with neural architectures. Early influential methods include RBM-based CF, which demonstrated strong performance but can be costly to train due to intractable inference, motivating more tractable neural constructions [49]. Autoencoder-based models such as AutoRec [51] and other neural interaction models (e.g., replacing inner products by feed-forward networks) have also been studied as scalable nonlinear alternatives [9]. Autoregressive formulations provide another neural route: CF-NADE [66] adapts NADE-style parameter sharing to collaborative filtering, handling variable-length user histories.

More recently, attention mechanisms (e.g., SASRec, BERT4Rec) have become the state-of-the-art for sequential recommendation, leveraging self-attention to dynamically weight relevant history items [16, 54]. Conceptually, attention offers adaptivity: it allows the model to focus on different information for different queries. However, these models often rely on heavy matrix multiplications and deep architectures, sacrificing the transparency and training efficiency of bilinear models. MMF occupies a unique middle ground: it achieves the adaptive capacity allocation characteristic of attention mechanisms but does so through a lightweight, element-wise masking operation that preserves the scalability of MF.

Matrix Completion. Matrix completion generalizes rating prediction to the recovery of missing entries in partially observed matrices and has been approached through a wide range of techniques, including (i) low-rank factorization families (MF/PMF/Bayesian variants and their extensions), (ii) approaches that exploit side information about users/items, and (iii) graph-based formulations that treat observed interactions as edges in a user-item graph [1, 39, 48].

Recent work on matrix completion increasingly leverages graph neural networks (GNNs) by viewing the rating matrix as a heterogeneous graph and casting prediction as link regression/classification. Representative models such as GC-MC [56] learn node embeddings and decode ratings, while inductive variants aim to generalize to unseen users/items by learning transferable local graph patterns [65] or by using graph autoencoder formulations [52]. In parallel, IDCF [60] proposes an inductive, model-based CF framework that generates embeddings for new users by leveraging relations to a set of key users. Finally, recent systems such as MoRGH [67] further explore recommendation on heterogeneous graphs by incorporating multiple relation types within a GNN framework.

These graph-based methods can be effective when high-quality side information is available, but they often introduce additional modeling and data requirements (feature engineering and graph construction). In contrast, MMF targets heterogeneity within the

factorization itself through instance-wise latent dimension gating while preserving MF's simplicity and scalability.

Positioning MMF against mixture and local MF models. MMF is related to mixture-of-experts and local low-rank factorization in that it relaxes the rigid global-sharing assumption of MF. However, MMF differs in its shared-basis latent gating design: all components reuse the same base factors U and V, and instance-wise masks modulate how each row or column uses the shared latent coordinates. In contrast, local MF and ensemble-style approaches typically maintain multiple independent models, experts, or metric spaces, which may duplicate shared structure and require partitioning, anchor selection, or model aggregation. MMF therefore targets a different trade-off: it increases flexibility through coordinate-wise soft gating while preserving a single compact bilinear factorization.

3 Proposed Method

We propose Masked Mixture Factorization (MMF), a lightweight yet expressive generalization of matrix factorization (MF). MMF substantially increases the expressivity of MF while preserving MF's efficient bilinear structure. A central limitation of standard MF is its rigid global sharing: all instances are represented in the same latent coordinate system and must rely on the same set of latent dimensions. In heterogeneous data, however, the *useful* latent dimensions can vary drastically across rows and columns.

Our key idea is to enable instance-wise dimension emphasis—i.e., to adaptively highlight or suppress latent dimensions for each instance—so that a limited parameter budget is allocated where it is most effective. Specifically, MMF is built on two design principles:

(1) **Mixture of masked factorizations** (Section 3.1). Instead of a single global factorization, we approximate the target matrix by a sum of K masked factorization components. Each component shares base latent factors but applies instance-dependent masks to activate different subsets of dimensions. This preserves MF's bilinear structure while avoiding unnecessary global sharing.
(2) **Efficient mask parameterization** (Section 3.2). Naïvely learning a full mask matrix per component is prohibitively expensive and undermines MF's efficiency and interpretability. We therefore restrict masks to a predefined, differentiable function family and parameterize them with a small number of learnable parameters, enabling efficient training and clear semantics.

3.1 Mixture of Masked Factorizations

Let $X \in \mathbb{R}^{I\times J}$. Standard MF approximates X by $UV^\top$ with $U \in \mathbb{R}^{I\times R}$ and $V \in \mathbb{R}^{J\times R}$. In contrast, MMF represents X as a sum of masked components:

$$X \approx \sum_{k=1}^{K} (U \odot M_k^U)(V \odot M_k^V)^\top, \tag{1}$$

where $U \in \mathbb{R}^{I\times R}$ and $V \in \mathbb{R}^{J\times R}$ are base latent factor matrices, $M_k^U \in \mathbb{R}^{I\times R}$ and $M_k^V \in \mathbb{R}^{J\times R}$ are the k-th mask matrices, and $\odot$ denotes element-wise multiplication. Here R is the latent dimension (rank of the base factors), and K is the number of masked components.

This formulation admits a natural interpretation. For each component k, the masks M_k^U and M_k^V modulate the effective latent dimensions of U and V in an instance-specific manner: some dimensions are emphasized, others are suppressed. The product $(U \odot M_k^U)(V \odot M_k^V)^\top$ forms a single specialized factorization component, and the sum over $k \in [K] \coloneqq \{1,\dots,K\}$ aggregates such components to approximate X. Thus, MMF realizes a structured form of adaptive capacity allocation: different instances can utilize different subsets of latent dimensions, while retaining a bilinear factorization backbone.

3.2 Efficient Mask Parameterization

A crucial requirement is that the masking mechanism must remain lightweight. Learning each mask M_k^U, M_k^V as a free matrix would significantly increase the parameter count by $O(K(IR+JR))$, slow down training, and obscure interpretability. To avoid these issues, we parameterize masks using row-wise shift parameters combined with a predefined differentiable mask function family.

Specifically, for each $k \in [K] = \{1,\dots,K\}$, we define the mask matrices $M_k^U \in \mathbb{R}^{I\times R}$ and $M_k^V \in \mathbb{R}^{J\times R}$ by

$$\begin{aligned} M_k^U(i,r) &= f_k^U(r; s_{k,i}^U), \quad i \in [I],\ r \in [R], \\ M_k^V(j,r) &= f_k^V(r; s_{k,j}^V), \quad j \in [J],\ r \in [R], \end{aligned} \tag{2}$$

where $\boldsymbol{s}_k^U \in \mathbb{R}^I$ and $\boldsymbol{s}_k^V \in \mathbb{R}^J$ are learnable row-wise shifts, and $f_k^U(\cdot;\cdot)$, $f_k^V(\cdot;\cdot)$ are predefined differentiable mask functions. In our default implementation, we instantiate Eq. (2) using a shared base function $f : \mathbb{R} \to \mathbb{R}$ and a simple multi-scale parameterization:

$$f_k(r;s) \;=\; \frac{1}{K} f\!\left(\frac{k}{K}(r-s)\right), \quad k \in [K]. \tag{3}$$

The factor $1/K$ keeps the scale of the summed mixture comparable across K, while the scale $\frac{k}{K}$ encourages *diversity* by producing masks with different effective widths, enabling coarse-to-fine dimension gating without extra parameters. Here, f can be instantiated with common smooth families (e.g., Gaussian), and we quantify how different choices of f affect performance in Section 5.5.

Under this view, the base function f determines the *shape* of emphasis/suppression along r, while the shift s determines *where* each instance places that emphasis. For example, a sigmoid-shaped mask induces a monotone transition with s controlling the boundary, whereas a Gaussian-shaped mask emphasizes a localized region with s controlling the center. Figure 2 provides an intuitive example.

Note that classical MF is invariant to permutations and rotations of latent dimensions; for example, permuting the columns of U and V does not change $UV^\top$. Thus, the ordered index r in Eq. (3) should not be viewed as assuming a fixed semantic order in the true latent space. Instead, MMF imposes a soft locality bias on the *learned* latent coordinate system: because U, V, and the shifts are optimized jointly, the model can organize latent dimensions so that smooth masks become useful for instance-wise capacity allocation. Moreover, this bias is not a hard contiguous-block constraint, since multiple masked components can jointly emphasize different regions of the latent space.

A defining advantage of MMF is that it achieves substantially higher representational capacity under a limited parameter budget. In standard MF, the learnable parameters are the factor matrices U and V, totaling $(I+J)R$. In MMF, the base factors still contribute $(I+J)R$ parameters, and the shift parameters add $(I+J)K$ learnable scalars through $\boldsymbol{s}_k^U, \boldsymbol{s}_k^V$ for $k = 1,\dots,K$. Thus, the total number of trainable parameters of MMF is $(I+J)(K+R)$.

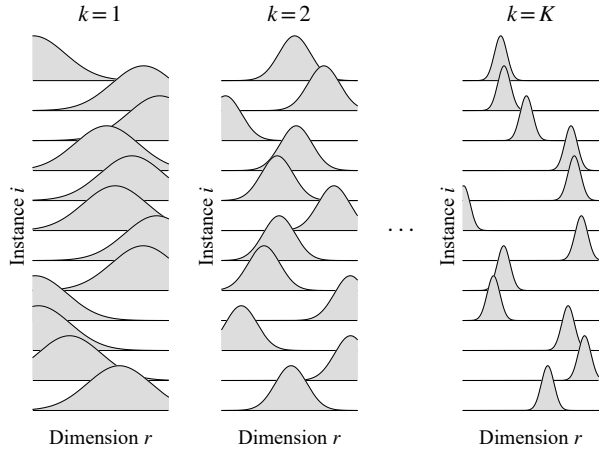


Figure 2: Illustration of the mask matrices when the mask function $f(\cdot)$ is chosen to have a Gaussian shape. Each column corresponds to a mixture component $k = 1, 2, \ldots, K$, and each horizontal ridge curve corresponds to one instance. Along the latent dimension axis, the Gaussian peak indicates the region of latent dimensions that is emphasized for that instance. The location of the peak varies across instances and components via learned shift parameters, showing how MMF enables instance-wise selection of different latent subspaces and combines multiple such masked components to represent heterogeneous structure.

Although MMF uses more parameters than rank-R MF due to the additional shifts, it remains highly parameter-efficient relative to the expressivity it gains. As shown in Theorem 1, the masked mixture form in Eq. (1) generically attains effective rank on the order of KR, i.e., it can represent matrices that would typically require rank KR under classical factorization. Achieving rank KR with standard MF would require $(I + J)KR$ parameters, which is much larger than $(I + J)(K + R)$ when $K, R \gg 1$. Hence, MMF amplifies expressivity without proportionally increasing parameters, yielding a more efficient use of modeling capacity.

Regarding optimization, MMF preserves the simplicity of MF training. Once the mask functions f_k^U, f_k^V are chosen to be differentiable, the full model is differentiable with respect to both the factor matrices U, V and the shift parameters $\{\boldsymbol{s}_k^U, \boldsymbol{s}_k^V\}_{k=1}^K$. Consequently, MMF can be trained end-to-end using standard gradient-based optimization under the same objectives as MF, with modest additional overhead compared to baseline MF.

4 Theoretical Analysis

We analyze two key theoretical properties of MMF that underpin its empirical gains. First, it significantly expands *expressivity*: while standard MF is bounded by rank R, our approach generically achieves rank KR with a similar budget, facilitating higher-fidelity modeling (Section 4.1). Second, MMF improves *identifiability* by using element-wise masking to break the rotational symmetries inherent in classical MF. Eliminating equivalent parameterizations yields a better-conditioned training objective, reducing drift along flat directions (Section 4.2).

4.1 Expressivity

We provide a simple yet strong expressivity guarantee for MMF. While standard MF with rank R produces outputs of rank at most R, MMF generically achieves rank up to KR. Concretely, under mild regularity and a mask linear-independence condition, MMF attains the maximal feasible rank for almost all parameter choices.

Notation. MMF parameterizes an $I \times J$ matrix using base factors $U \in \mathbb{R}^{I\times R}$, $V \in \mathbb{R}^{J\times R}$, and K masks per side. For parameters θ in a domain $\Theta \subseteq \mathbb{R}^{(I+J)(K+R)}$, define

$$U_k(\theta) \coloneqq U(\theta) \odot M_k^U(\theta), \quad V_k(\theta) \coloneqq V(\theta) \odot M_k^V(\theta),$$

and the MMF output

$$\widehat{X}(\theta) \;\coloneqq\; \sum\nolimits_{k=1}^{K} U_k(\theta) V_k(\theta)^\top. \tag{4}$$

Denote concatenations

$$U_{\text{cat}}(\theta) \coloneqq \big[\, U_1(\theta) \;\cdots\; U_K(\theta) \,\big] \in \mathbb{R}^{I\times KR},$$
$$V_{\text{cat}}(\theta) \coloneqq \big[\, V_1(\theta) \;\cdots\; V_K(\theta) \,\big] \in \mathbb{R}^{J\times KR},$$

so that

$$\widehat{X}(\theta) \;=\; U_{\text{cat}}(\theta)\, V_{\text{cat}}(\theta)^\top. \tag{5}$$

Hence, $\operatorname{rank}(\widehat{X}(\theta)) \le \min\{I, J, KR\}$ for all θ.

Theorem 1. *Let Θ be a connected open set and define $\widehat{X}(\theta)$ by Eq. (5). Assume the following two conditions hold.*

(C1) Analytic parameterization: All entries of $U(\theta)$, $V(\theta)$, and $\{M_k^U(\theta)\}_{k=1}^K, \{M_k^V(\theta)\}_{k=1}^K$ are real-analytic functions of $\theta \in \Theta$.

(C2) Linearly independent shift masks: Under the shift-mask form in Eq. (2), for each fixed $r \in [R]$ the family $\{s \mapsto f_k^U(r; s)\}_{k=1}^K$ is linearly independent on an open interval $\mathcal{S} \subset \mathbb{R}$, and the same holds for $\{s \mapsto f_k^V(r; s)\}_{k=1}^K$.

Then, for Lebesgue-almost every $\theta \in \Theta$,

$$\operatorname{rank}\big(\widehat{X}(\theta)\big) = \min\{I, J, KR\}.$$

In particular, $\operatorname{rank}(\widehat{X}(\theta)) = KR$ for almost all θ if $KR \le \min\{I, J\}$.

Proof. See Appendix A. □

Theorem 1 shows that MMF generically lifts the attainable rank from R to KR. Note that the required conditions are mild. Intuitively, (C1) just requires that the model parameters change smoothly with θ, ruling out pathological non-smooth or discontinuous parameterizations, and (C2) rules out degenerate mask choices where different components behave identically; as long as the K masks are not redundant copies, the condition holds. In the supplementary material, we verify (C2) for several canonical choices of the base mask (trigonometric, Gaussian, and logistic sigmoid), and the same template can be used to check additional smooth mask families.

4.2 Identifiability

A central challenge in matrix factorization is *non-identifiability*: many distinct parameter tuples can generate the same reconstructed matrix. This ambiguity is particularly severe for classical MF due to its continuous *rotation symmetry*. In standard MF, $X = UV^\top$ is invariant under the action of any invertible matrix $Q \in \mathrm{GL}_R$, where GL_R denotes the set of invertible matrices of size $R \times R$:

$$UV^\top = (UQ)(VQ^{-\top})^\top.$$

Hence, around any solution there exists a continuum of equivalent factorizations [4, 36].

In contrast, MMF reconstructs X as a sum of masked bilinear terms. Because masking is applied *element-wise* before the bilinear product, the MF invariance does not carry over. In Theorem 2, we formalize this by showing that MMF generically admits no infinitesimal non-diagonal rotations of the latent basis.

Notation. Fix a parameter point θ and abbreviate

$$U := U(\theta), \quad V := V(\theta), \quad U_k := U_k(\theta), \quad V_k := V_k(\theta).$$

Consider the change of basis $(U, V) \mapsto (UQ, VQ^{-\top})$ while keeping masks fixed. Define the resulting mapping

$$\widehat{X}_\theta(Q) := \sum_{k=1}^{K} \big((UQ) \odot M_k^U\big)\big((VQ^{-\top}) \odot M_k^V\big)^\top. \tag{6}$$

We say MMF admits an *infinitesimal non-diagonal rotation* at θ if there exists a differentiable function $Q : \mathbb{R}^{\geq 0} \to \mathrm{GL}_R$ with $Q(0) = I_R$ (identity), $Q'(0)$ having a nonzero off-diagonal entry, and $\widehat{X}_\theta(Q(t))$ constant for all sufficiently small $t \geq 0$. In other words, this means that we can slightly mix different latent dimensions (a non-diagonal "rotation") without changing the output $\widehat{X}_\theta$.

Theorem 2. *Let $K \geq 2$ and $I, J \geq R$. Assume that the MMF parameterization is real-analytic. For each θ, define the linear map*

$$\mathcal{L}_\theta : \{C \in \mathbb{R}^{R\times R} : \mathrm{diag}(C) = 0\} \to \mathbb{R}^{I\times J},$$
$$\mathcal{L}_\theta(C) := \frac{d}{dt}\widehat{X}_\theta(\exp(tC))\Big|_{t=0}.$$

Assume that the mask family is non-degenerate in the sense that there exists θ_0 such that $\mathcal{L}_{\theta_0}$ is injective. Then, for Lebesgue-almost every θ, MMF does not admit infinitesimal non-diagonal rotations. Thus, except for scaling each latent dimension (i.e., $U \to UD$, $V \to VD^{-1}$ with diagonal D), there is no local freedom that keeps $\widehat{X}_\theta$ unchanged.

Proof. See Appendix B. □

Theorem 2 shows that MMF removes the continuous rotational symmetry of standard MF by ruling out infinitesimal non-diagonal rotations. The elimination of these parameter redundancies results in a more constrained and well-defined parameter space. Such a formulation stabilizes the training process by preventing gradient drift among redundant, equivalent solutions. We further support this claim with empirical evaluation in Section 5.8.

Regarding the non-degeneracy condition, the assumption is mild in practice; it simply requires the mask family to be *rich enough* to detect off-diagonal perturbations, which typically holds for common real-analytic functions. For completeness, the supplementary material details a concrete instantiation that verifies the injectivity.

5 Experiments

We design experiments to answer the following questions:

- Q1 **Reconstruction Performance (Section 5.2).** Does MMF yield higher reconstruction accuracy than strong MF baselines under a matched parameter budget?
- Q2 **Matrix Completion Performance (Section 5.3).** Does MMF improve rating prediction accuracy on real-world benchmarks compared to strong baselines?
- Q3 **Recommendation Performance (Section 5.4).** Does MMF remain effective for Top-N recommendation under ranking-based evaluation?

Table 1: Datasets for the matrix reconstruction task.

Dataset	Type	Size	Density	Description
$\{S_n\}_{n=6}^{10}$	Synthetic	$2^n \times 2^n$	1	Random Dense
$\{H_b\}_{b=2}^{6}$	Synthetic	1080×1080	b^{-1}	Block-Diagonal
Yale-B[1]	Real	2016×2414	1	Face Images
Reuters[2]	Real	2000×3000	0.007	Text Documents

[1]https://vision.ucsd.edu/datasets
[2]https://archive.ics.uci.edu/dataset/137/

Table 2: Datasets for the matrix completion task.

Dataset	# Users	# Items	# Ratings	Density
Flixster[1]	3,000	3,000	26,173	0.0029
Douban[1]	3,000	3,000	136,891	0.0152
ML-100K[2]	943	1,682	100,000	0.0630
ML-1M[2]	6,040	3,706	1,000,209	0.0447
ML-10M[2]	69,878	10,677	10,000,054	0.0134

[1]https://github.com/muhanzhang/IGMC
[2]https://grouplens.org/datasets/movielens

- Q4 **Ablation Study (Section 5.5).** How do mask-related hyperparameters impact the performance of MMF?
- Q5 **Running Time (Section 5.6).** How fast is MMF in practice compared to MF and recent deep learning methods?
- Q6 **Expressivity and Identifiability (Sections 5.7–5.8).** Does MMF increase effective rank while yielding more stable and identifiable latent factors than standard MF?

5.1 Experiment Settings

Datasets. To ensure task-appropriate evaluation, we assess reconstruction on dense and sparse matrices and matrix completion on standard rating datasets. For the reconstruction task, we use both synthetic and real data described in Table 1: random matrices $\{S_n \in \mathbb{R}^{2^n\times 2^n}\}_{n=6}^{10}$ generated from a standard normal distribution, and block-diagonal matrices $\{H_b \in \mathbb{R}^{1080\times 1080}\}_{b=2}^{6}$, where each H_b is constructed by partitioning indices into b contiguous groups of size $m = 1080/b$ and placing b disjoint random blocks $B_1, \ldots, B_b \in \mathbb{R}^{m\times m}$ on the diagonal, mimicking b communities whose interactions depend on different latent subspaces. We also evaluate reconstruction on two real-world matrices: Yale-B (a face-image matrix) [5] and Reuters (a document-term matrix) [35]. These datasets are selected to test the model's capability to capture two distinct types of structural heterogeneity: non-linear visual variations and semantic sparsity.

For the matrix completion task, we evaluate rating prediction on five benchmarks described in Table 2; for Flixster and Douban, we use the preprocessed versions of [40]. On MovieLens, all methods are trained on 90% of observed ratings and tested on the remaining 10%. For Top-N recommendation, we additionally evaluate MMF on ML-1M with a BPR loss and standard Recall/NDCG metrics.

Baselines. We evaluate MMF against a comprehensive set of baselines. For reconstruction, we compare with SVD and MF variants (SGD, ridge, and Bias MF). For matrix completion, we include classical methods (PMF, SVD++, Bias MF, and LLORMA), neural approaches (RBM, AutoRec, NNMF, and CF-NADE), and graph-based models (sRMGCNN, GC-MC, STAR-GCN, IGMC, IDCF-GC, IMC-GAE, GHRS, and MoRGH). For Top-N recommendation, we compare with BPRMF, NeuMF, NGCF, LightGCN, and SGL.

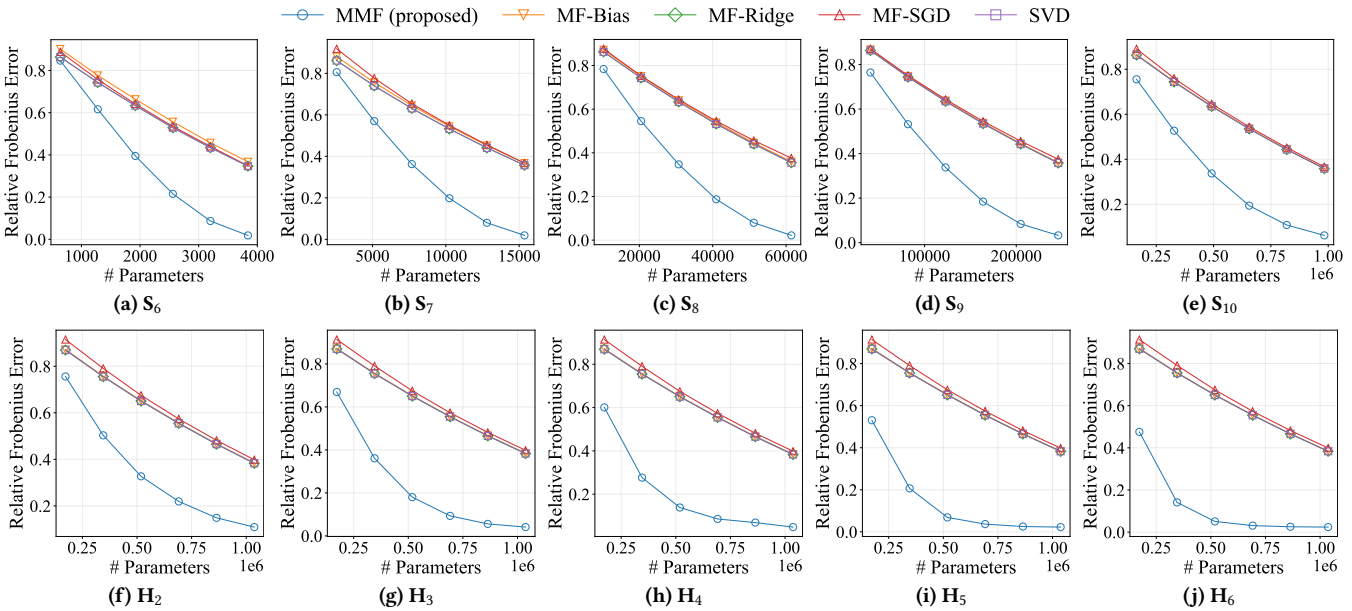


Figure 3: Reconstruction performance on synthetic matrices under matched parameter budgets. Across all scenarios, MMF achieves significantly lower reconstruction error than MF-based baselines, and notably outperforms SVD.

Detailed experimental configurations and hyperparameter settings are provided in the supplementary material.

5.2 Reconstruction Performance

We evaluate representational power in a controlled setting by measuring reconstruction accuracy. The goal is to isolate how effectively each method uses a given parameter budget to approximate a target matrix. For each synthetic dataset S_n and H_b, we fit all methods to minimize the standard squared reconstruction loss. The parameter budget is held constant across baselines via the following scheme: under a total budget $(I + J)B$ for an input $X \in \mathbb{R}^{I \times J}$,

- SVD / MF-Ridge / MF-SGD: use rank $R = B$.
- MF-Bias: uses rank $R = B - 1$ to account for the additional user/item bias parameters.
- MMF: with K masks, we reduce the base rank to $R = B - K$.

Figure 3*(a)–(e)* shows that across all matrices S_6–S_{10}, MMF consistently achieves substantially lower reconstruction error than the baselines. Notably, this improvement persists even when comparing against SVD, which is the optimal rank-B approximation in the fully observed case. This indicates that the advantage of MMF is not merely optimization-related: by relaxing the uniform global-sharing constraint, MMF better represents structures that a single factorization cannot capture under the same budget.

The performance gap becomes more pronounced on the heterogeneous matrices H_2–H_6, and the advantage of MMF grows as the number of blocks increases (Figure 3*(f)–(j)*). This trend is expected: as heterogeneity increases, a single global latent basis must simultaneously explain multiple distinct substructures, leading to a mismatch that is alleviated only by increasing rank. In contrast, MMF allocates capacity adaptively through multiple masked components (for an empirical analysis supporting this claim, see Section 5.7).

Table 3: Reconstruction error on real data (relative Frobenius error). MMF outperforms SVD under matched parameter budgets. Improvement indicates the relative reduction in error compared to SVD: $(\text{Error}_{\text{SVD}} - \text{Error}_{\text{MMF}})/\text{Error}_{\text{SVD}}$.

Dataset	Budget B	SVD	MMF (proposed)	Improvement
Yale-B	10	0.3902	**0.3616**	7.32%
	20	0.3334	**0.2969**	10.9%
	30	0.3004	**0.2640**	12.1%
	40	0.2770	**0.2374**	14.3%
	50	0.2586	**0.2166**	16.2%
	60	0.2436	**0.1994**	20.2%
Reuters	20	0.9256	**0.8584**	7.26%
	40	0.8907	**0.7395**	17.0%
	60	0.8627	**0.6428**	25.5%
	80	0.8383	**0.5718**	31.8%
	100	0.8165	**0.5169**	36.7%
	120	0.7964	**0.4751**	40.3%

As a result, MMF captures multiple coexisting low-rank patterns while maintaining a parameter budget comparable to standard MF.

To validate that these findings translate to natural data distributions, we conduct additional reconstruction experiments on real-world benchmarks. Table 3 summarizes the reconstruction performance; MMF consistently achieves lower reconstruction error than SVD across various rank budgets on both datasets. This performance advantage can be attributed to MMF's inductive bias, which is tailored to the data characteristics.

The gains on the real matrices reflect different types of instance heterogeneity. On Yale-B, lighting changes create image-specific shadows and local high-contrast variations; instead of representing all images through one static linear basis, MMF can gate latent dimensions according to the illumination pattern of each image. On

Table 4: Matrix completion results (RMSE). Best and second-best are marked in bold and underlined, respectively; O.O.M. denotes out-of-memory. MMF results are averaged over five runs (the standard deviations are ≤ 0.001 in all cases). MMF achieves competitive performance compared with state-of-the-art models despite using no side information.

Method	Flixster	Douban	ML-100K	ML-1M	ML-10M
PMF [39]	-	0.737	0.932	0.883	-
RBM [49]	-	-	-	0.854	0.825
SVD++ [24]	-	-	0.903	0.856	-
BiasMF [25]	0.945	0.758	0.917	0.845	0.803
LLORMA [29]	-	-	-	0.833	0.782
AutoRec [51]	-	-	-	0.831	0.782
NNMF [3]	-	0.729	0.907	0.843	-
CF-NADE [66]	-	-	-	<u>0.829</u>	0.771
sRMGCNN [40]	0.926	0.801	0.929	0.865	0.833
⋆GC-MC [56]	0.917	0.734	0.905	0.832	0.777
⋆STAR-GCN [64]	<u>0.879</u>	0.727	0.895	0.832	<u>0.770</u>
IGMC [65]	**0.872**	**0.721**	0.905	0.857	O.O.M.
⋆IDCF-GC [60]	0.910	0.733	0.893	0.835	O.O.M.
IMC-GAE [52]	0.884	**0.721**	0.897	<u>0.829</u>	O.O.M.
⋆GHRS [2]	-	-	0.887	0.833	-
⋆MoRGH [67]	-	-	**0.881**	**0.827**	-
MMF (proposed)	0.898	<u>0.726</u>	<u>0.884</u>	**0.827**	**0.769**

⋆ denotes methods using side information.

Reuters, documents are semantically sparse and typically cover only a few topics; masking helps allocate capacity to document-specific topic regions and reduces the semantic smearing that can occur when all latent dimensions are uniformly active. These results indicate that the benefits of MMF extend beyond synthetic settings to natural data with visual and semantic heterogeneity.

5.3 Matrix Completion Performance

We further evaluate MMF on rating prediction tasks to validate whether its inductive bias translates to improved generalization on real-world benchmark datasets. Table 4 summarizes the RMSE results. MMF consistently improves over standard MF baselines and remains competitive with recent neural and graph-based approaches. In particular, MMF obtains the best result on ML-1M and ML-10M, and the second-best result on ML-100K and Douban; it also scales to ML-10M, where several recent graph-based models are not reported due to computational overhead.

A key point is that MMF achieves these results while operating only on the interaction matrix. As indicated in Table 4, methods such as STAR-GCN, IDCF-GC, GHRS, and MoRGH exploit side information or graph structures, and graph-based models can be particularly advantageous on extremely sparse datasets such as Flixster [46, 61]. By contrast, MMF is a pure CF model that preserves the MF backbone and introduces only lightweight latent-coordinate masking. Thus, MMF provides a compact MF-style alternative that closes much of the gap to more complex architectures without graph construction, message passing, or external features.

We attribute the gains to MMF's ability to handle instance heterogeneity under a limited parameter budget. Standard MF forces all users and items to share the same set of active latent dimensions, which creates a trade-off between underfitting complex users and overfitting noisy observations. MMF relaxes this constraint by allowing different rows and columns to emphasize different subsets of a shared latent basis. This instance-wise capacity allocation improves rating prediction while retaining the simplicity and scalability of classical factorization.

Table 5: Top-N recommendation results on ML-1M. MMF is trained with the BPR loss and remains competitive with strong graph and self-supervised ranking baselines.

Method	Recall@10	NDCG@10	Recall@20	NDCG@20
BPRMF [47]	0.1745	0.2410	0.2624	0.2516
NeuMF [9]	0.1525	0.2183	0.2352	0.2285
NGCF [57]	0.1761	0.2467	0.2677	0.2576
LightGCN [8]	0.1782	0.2501	0.2699	0.2606
SGL [58]	0.1804	**0.2535**	0.2732	0.2644
MMF (proposed)	**0.1807**	0.2530	**0.2742**	**0.2647**

5.4 Recommendation Performance

We also evaluate MMF under Top-N ranking metrics, which are widely used in practical recommender systems and complement the RMSE-based rating prediction results. For this setting, we train MMF with the Bayesian Personalized Ranking (BPR) loss and evaluate Recall@10, NDCG@10, Recall@20, and NDCG@20 on ML-1M.

Table 5 shows that MMF performs competitively against representative ranking models, including graph-based and self-supervised recommenders. The comparison is informative because the strongest baselines use mechanisms beyond standard MF: LightGCN and NGCF exploit graph propagation, while SGL further leverages self-supervised contrastive learning. Despite using a simpler MF-style predictor, MMF slightly outperforms SGL on Recall@10, Recall@20, and NDCG@20, and is only marginally behind on NDCG@10. This suggests that instance-wise latent gating is useful not only for predicting explicit ratings but also for ranking unobserved items.

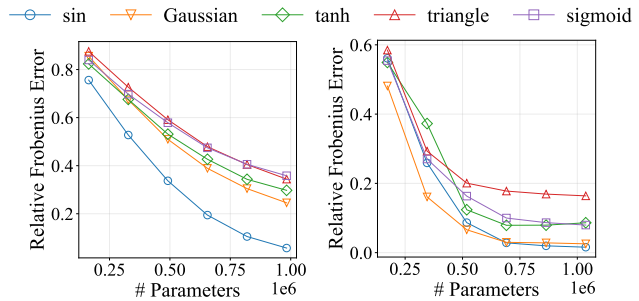
Note that these results should be interpreted as evidence of recommendation effectiveness rather than as a claim that MMF is a specialized ranking architecture. The current ranking version uses a straightforward BPR objective without graph augmentation, contrastive learning, or sequence modeling, suggesting that the MMF parameterization itself provides a useful inductive bias.

5.5 Ablation Study

We conduct ablation studies to isolate which design choices are responsible for MMF's gains.

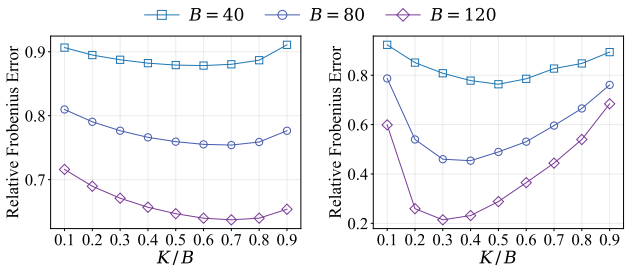
Mask family. We compare multiple mask function families, including smooth (sine, Gaussian), monotone (sigmoid, tanh), and non-smooth (triangle) ones. Figure 4 confirms that smooth masks consistently yield the strongest results, whereas monotone or non-smooth masks typically underperform. Notably, the optimal choice depends on data structure: on the random dense matrix S_{10}, sinusoidal masks perform the best, likely because they can periodically emphasize and suppress latent dimensions; in contrast, on the heterogeneous matrix H_6, Gaussian masks dominate, consistent with their ability to focus on localized regions of the latent space.

This behavior aligns with the intended role of masks: MMF emphasizes some latent dimensions while suppressing others. Gaussian and sinusoidal families yield localized or structured emphasis



(a) S_{10} $(K/B = 0.6)$ (b) H_6 $(K/B = 0.4)$

Figure 4: Effect of mask families on MMF's performance. Smooth localized or oscillatory masks demonstrate advantages over strictly monotone or non-differentiable families.



(a) S_{10} (mask = sine) (b) H_6 (mask = Gaussian)

Figure 5: Impact of mask count on performance under a fixed parameter budget. The optimal balance between mask count K and base rank $B-K$ shifts depending on the data structure.

patterns, enabling fine-grained instance-specific gating. Monotone masks can separate dimensions into "early vs. late" groups, but their monotonicity is restrictive; non-smooth masks can also hinder optimization.

Number of masks. We vary K while keeping the total parameter budget fixed at $(I + J)B$. Since allocating capacity to K masks consumes $(I + J)K$, the remaining parameters support a base rank of $R = B - K$. According to Theorem 1, the expressive rank scales as $KR = K(B - K)$, which is theoretically maximized at $K \approx B/2$. Figure 5 reveals that the empirical optimum shifts depending on data characteristics. On the random dense matrix S_{10}, the performance peaks at a slightly higher K, suggesting that differentiating complex patterns requires greater combinatorial flexibility from the masks. Conversely, on the heterogeneous matrix H_6 with clear community structures, a smaller K suffices to isolate latent regions; thus, allocating more budget to the base rank yields better results.

5.6 Running Time

To evaluate the computational efficiency of MMF, we conduct (i) scalability tests on synthetic matrices and (ii) end-to-end runtime benchmarks on real-world datasets. For the synthetic study, we fix $B = 40$ and measure per-epoch runtime while varying both the matrix size and the number of masks. Table 6 confirms that MMF maintains comparable runtime to MF baselines. This is because MMF preserves the MF structure and primarily adds element-wise masking and a sum over K components, which can be efficiently implemented via batch operations and parallelization. Consequently,

Table 6: Running time (ms) comparison. MMF scales efficiently with respect to both data size and the mask count.

Method	S_6	S_7	S_8	S_9	S_{10}
MMF $(K/B = 0.1)$	1.118	1.129	1.163	1.195	1.264
MMF $(K/B = 0.2)$	1.062	1.076	1.091	1.120	1.213
MMF $(K/B = 0.3)$	1.174	1.200	1.221	1.288	1.295
MMF $(K/B = 0.4)$	1.209	1.211	1.228	1.293	1.344
MMF $(K/B = 0.5)$	1.223	1.218	1.231	1.206	1.301
MMF $(K/B = 0.6)$	1.215	1.222	1.233	1.204	1.299
MMF $(K/B = 0.7)$	1.215	1.229	1.237	1.285	1.295
MMF $(K/B = 0.8)$	1.226	1.213	1.242	1.280	1.292
MMF $(K/B = 0.9)$	1.242	1.232	1.240	1.270	1.274
MF-Bias	0.902	0.914	0.915	0.928	0.967
MF-Ridge	0.857	0.886	0.889	0.897	0.929
MF-SGD	0.529	0.592	0.594	0.605	0.682
SVD*	4.504	9.008	23.18	43.03	84.07

*For SVD, the total running time is reported.

Table 7: Running time (sec) comparison on real-world benchmarks. MMF attains markedly faster end-to-end training time while maintaining comparable inference latency.

	ML-100K		ML-1M	
Method	Training	Inference	Training	Inference
sRMGCNN [40]	46.969	0.0116	407.80	0.0221
GC-MC [56]	43.597	0.0071	384.26	**0.0159**
IGMC [65]	984.03	13.682	7467.8	54.861
IDCF-GC [60]	74.389	0.1016	876.45	4.0528
IMC-GAE [52]	58.942	0.0153	412.11	0.0312
MMF	**15.158**	**0.0052**	**67.340**	0.0448

MMF delivers substantially improved accuracy without incurring prohibitive training overhead.

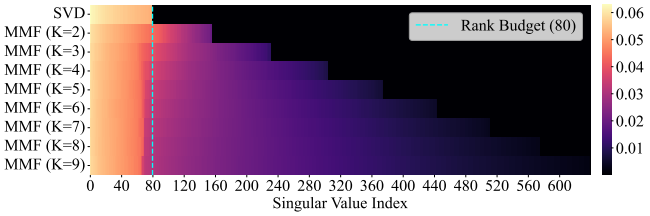
We further benchmark the end-to-end computational efficiency of MMF against recent state-of-the-art matrix completion models on the ML-100K and ML-1M datasets. We report both *training time* and *inference time*: training time is measured as the total wall-clock time required to train a model until convergence, and inference time is measured as the wall-clock time of a single forward pass over the test set. All runtimes are averaged over five independent runs to reduce variance.

Table 7 shows that MMF achieves substantially shorter training times than modern graph-based deep learning frameworks on both datasets. This is consistent with their computational characteristics: graph methods repeatedly perform message passing or neighborhood propagation, whereas MMF retains bilinear MF computation and adds only lightweight masking. Although some graph models have comparable inference latency, their optimization overhead remains much larger during training.

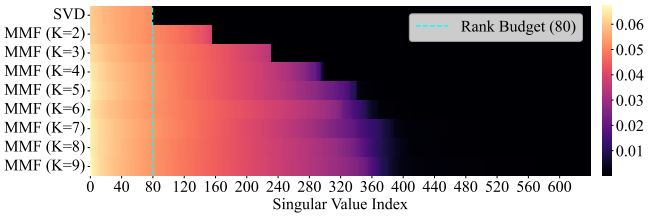
5.7 Expressivity Analysis

We provide empirical evidence that MMF increases effective expressivity in an adaptive manner predicted by our theory. Figure 6 visualizes the singular value spectra of the reconstructed matrices $\widehat{X}$ produced by SVD and MMF under a fixed parameter budget.

Standard SVD (rank $B = 80$) exhibits a sharp spectral cutoff, confirming its expressivity is strictly bounded by the predefined budget.



(a) S_{10} (Random Dense Matrix)



(b) H_6 (Heterogeneous Block-Diagonal Matrix)

Figure 6: Singular value spectra for SVD vs. MMF under a fixed budget $B = 80$ (cyan line). (a) On random dense data, MMF expands effective rank as K increases. (b) On structured heterogeneous data, the effective rank tends to saturate, suggesting adaptation to the intrinsic block structure rather than blind rank inflation.

In contrast, MMF (rank $B - K$ with K masks) yields a spectrum where significant singular values extend well beyond the rank-B threshold (cyan line), confirming that the masked mixture mechanism allows the model to generate a high-rank approximation using limited parameters. Crucially, the spectral expansion patterns suggest that MMF adapts to the intrinsic complexity of the data. On the unstructured random matrix S_{10}, the effective rank increases almost linearly with the mask count K, indicating that additional masks are used to model additional non-negligible modes. On the block-diagonal matrix H_6, the expansion is less aggressive and tends to saturate, suggesting that MMF allocates sufficient capacity to the underlying block subspaces rather than blindly inflating rank. This contrast supports the view that MMF expands expressivity when necessary while converging toward a compact structure when such structure exists.

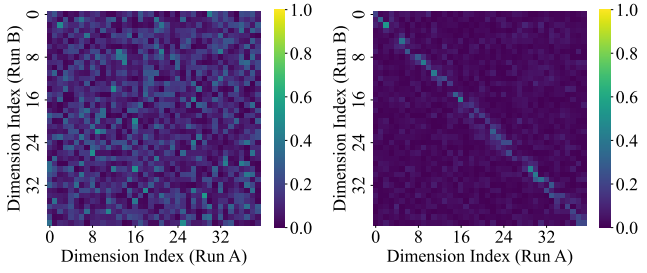
5.8 Identifiability Analysis

We validate the theoretical identifiability established in Theorem 2 by measuring factor stability on S_{10} and H_6. We set $R = 40$ and $K = 8$ with Gaussian masks, fix the randomly initialized masks, and train the model twice with different random seeds (A and B). To compare the resulting left factors $U^{(A)}$ and $U^{(B)}$, we compute the similarity matrix $S \in [0, 1]^{R \times R}$:

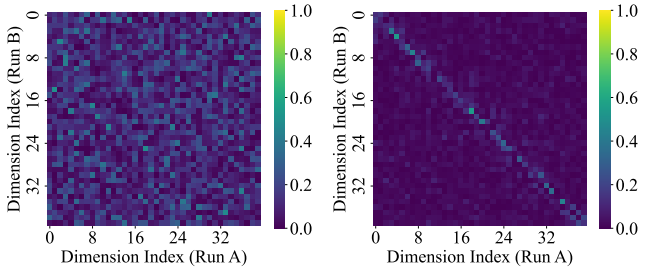
$$S_{ij} = \frac{\left|(u_i^{(A)})^\top u_j^{(B)}\right|}{\|u_i^{(A)}\|_2 \|u_j^{(B)}\|_2},$$

where $u_i^{(X)}$ denotes the i-th column of $U^{(X)}$. A strongly diagonal S indicates one-to-one correspondence between latent dimensions across runs, up to column-wise scaling.

As shown in Figure 7, standard MF yields disordered correlation maps due to rotational symmetry, whereas MMF exhibits sharp



(a) Standard MF on S_{10} **(b) MMF on S_{10}**



(c) Standard MF on H_6 **(d) MMF on H_6**

Figure 7: Empirical verification of MMF's identifiability via cosine similarity between factors learned from two independent runs. Standard MF exhibits scattered correlations, whereas MMF shows strong diagonal alignment, indicating more consistent latent dimensions across runs.

diagonal alignment. This confirms that instance-wise masking leads to more reproducible and structurally stable latent factors than conventional bilinear MF.

6 Conclusion

In this paper, we propose MMF, a principled generalization of MF that resolves the bottleneck of rigid global sharing through adaptive instance-wise masked mixtures. By dynamically allocating representational capacity, MMF achieves a strong balance between parameter efficiency and expressivity, consistently outperforming standard MF baselines in reconstruction and completion tasks while also showing competitive Top-N recommendation performance. Our theoretical analysis confirms that this masking mechanism not only expands the effective rank but also enhances identifiability by breaking rotational symmetry. Future directions include exploring richer mask function families and extending the framework to inductive settings to handle cold-start scenarios effectively.

Acknowledgments

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) [No. RS-2020-II200894, Flexible and Efficient Model Compression Method for Various Applications and Environments], [No. 2022-0-00641, XVoice: Multi-Modal Voice Meta Learning], [No. RS-2024-00509257, Global AI Frontier Lab], [No. RS-2021-II211343, Artificial Intelligence Graduate School Program (SNU)], and [No. RS-2025-25442338, AI Star Fellowship Support Program (SNU)]. The Institute of Engineering Research and the ICT at Seoul National University provided research facilities for this work. U Kang is the corresponding author.

References

[1] Emmanuel Candes and Benjamin Recht. 2012. Exact Matrix Completion via Convex Optimization. *Commun. ACM* 55, 6 (2012).

[2] Zahra Zamanzadeh Darban and Mohammad Hadi Valipour. 2022. GHRS: Graph-Based Hybrid Recommendation System with Application to Movie Recommendation. *Expert Syst. Appl.* 200 (2022).

[3] Gintare Karolina Dziugaite and Daniel M Roy. 2015. Neural Network Matrix Factorization. *arXiv preprint arXiv:1511.06443* (2015).

[4] Rong Ge, Chi Jin, and Yi Zheng. 2017. No Spurious Local Minima in Nonconvex Low Rank Problems: A Unified Geometric Analysis. In *ICML*.

[5] Athinodoros S. Georghiades, Peter N. Belhumeur, and David J. Kriegman. 2002. From Few to Many: Illumination Cone Models for Face Recognition Under Variable Lighting and Pose. *TPAMI* 23, 6 (2002).

[6] Shivangi Gheewala, Shuxiang Xu, and Soonja Yeom. 2025. In-Depth Survey: Deep Learning in Recommender Systems—Exploring Prediction and Ranking Models, Datasets, Feature Analysis, and Emerging Trends. *Neural Comput. Appl.* (2025).

[7] Prem Gopalan, Jake M Hofman, and David M Blei. 2013. Scalable Recommendation with Poisson Factorization. *arXiv preprint arXiv:1311.1704* (2013).

[8] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. In *SIGIR*.

[9] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In *WWW*.

[10] Ka Hyun Park, Junghun Kim, and U Kang. 2025. Accurate Semi-Supervised Automatic Speech Recognition for Ordinary and Characterized Speeches via Multi-Hypotheses-Based Curriculum Learning. *PLOS One* 20, 10 (2025).

[11] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. 1991. Adaptive Mixtures of Local Experts. *Neural Comput.* 3, 1 (1991).

[12] Jun-Gi Jang, Jeongyoung Lee, Yong-chan Park, and U Kang. 2023. Fast and Accurate Dual-Way Streaming PARAFAC2 for Irregular Tensors—Algorithm and Application. In *KDD*.

[13] Jun-Gi Jang, Yong-chan Park, and U Kang. 2024. Fast and Accurate PARAFAC2 Decomposition for Time Range Queries on Irregular Tensors. In *CIKM*.

[14] Jihyeong Jeon, Jiwon Lee, Cheol Ryu, and U Kang. 2025. Entity-Aware Generative Retrieval for Personalized Contexts. In *CIKM*.

[15] Nam Kyu Kang, Yong-chan Park, and U Kang. 2026. Fast and Accurate Temporal Super-Resolution via Residual-Aware Coupled Tensor Factorization. In *ICASSP*.

[16] Wang-Cheng Kang and Julian McAuley. 2018. Self-Attentive Sequential Recommendation. In *ICDM*.

[17] Jongjin Kim and U Kang. 2025. Sequentially Diversified and Accurate Recommendations in Chronological Order for a Series of Users. In *WSDM*.

[18] Junghun Kim, Ka Hyun Park, Jun-Gi Jang, and U Kang. 2024. Fast and Accurate Domain Adaptation for Irregular Tensor Decomposition. In *KDD*.

[19] Junghun Kim, Ka Hyun Park, Jun-Gi Jang, and U Kang. 2026. Fast and Accurate Domain Adaptation for Irregular and Regular Tensor Decomposition. *TKDE* (2026).

[20] Junghun Kim, Ka Hyun Park, and U Kang. 2024. Accurate Semi-Supervised Automatic Speech Recognition via Multi-Hypotheses-Based Curriculum Learning. In *PAKDD*.

[21] Junghun Kim, Shihyung Park, and U Kang. 2026. Dual-Level Reweighting for Positive-Unlabeled Graph Classification. In *WWW*.

[22] Junghun Kim, Hoyoung Yoon, Ka Hyun Park, and U Kang. 2025. Accurate Graph-Based Multi-Positive Unlabeled Learning via Disentangled Multi-View Feature Propagation. In *KDD*.

[23] Minjun Kim, Jaeri Lee, Jongjin Kim, Jeongin Yun, Yongmo Kwon, and U Kang. 2026. LampQ: Towards Accurate Layer-Wise Mixed Precision Quantization for Vision Transformers. In *AAAI*, Vol. 40.

[24] Yehuda Koren. 2008. Factorization Meets the Neighborhood: A Multifaceted Collaborative Filtering Model. In *KDD*.

[25] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *Computer* 42, 8 (2009).

[26] Steven G Krantz and Harold R Parks. 2002. *A Primer of Real Analytic Functions*. Springer Science & Business Media.

[27] Jaeri Lee and U Kang. 2025. Context-Aware Sequential Bundle Recommendation via User-Specific Representations. In *CIKM*.

[28] Jaeri Lee, Jongjin Kim, and U Kang. 2026. CatDive: A Simple yet Effective Method for Maximizing Category Diversity in Sequential Recommendation. *PLOS One* 21, 1 (2026).

[29] Joonseok Lee, Seungyeon Kim, Guy Lebanon, and Yoram Singer. 2013. Local Low-Rank Matrix Approximation. In *ICML*.

[30] Jeongyoung Lee, SeungJoo Lee, and U Kang. 2026. Fast and Accurate Element-Level Streaming CP Decomposition for Higher-Order Tensors. In *ICDE*.

[31] Jaeri Lee, Jeongin Yun, and U Kang. 2024. Towards True Multi-Interest Recommendation: Enhanced Scheme for Balanced Interest Training. In *BigData*.

[32] SeungJoo Lee, Yong-chan Park, and U Kang. 2024. Accurate Coupled Tensor Factorization with Knowledge Graph. In *BigData*.

[33] SeungJoo Lee, Yong-chan Park, and U Kang. 2025. Offline and Online Coupled Tensor Factorization with Knowledge Graph. *PLOS One* 20, 11 (2025).

[34] SeungJoo Lee, Yong-chan Park, and U Kang. 2025. SwaGNER: Leveraging Span-Aware Grid Transformers for Accurate Nested Named Entity Recognition. In *CIKM*.

[35] David D Lewis, Yiming Yang, Tony G Rose, and Fan Li. 2004. RCV1: A New Benchmark Collection for Text Categorization Research. *JMLR* 5, Apr (2004).

[36] Qiuwei Li, Zhihui Zhu, and Gongguo Tang. 2019. The Non-Convex Geometry of Low-Rank Matrix Optimization. *Inf. Inference* 8, 1 (2019).

[37] Jan R Magnus and Heinz Neudecker. 2019. *Matrix Differential Calculus with Applications in Statistics and Econometrics*. John Wiley & Sons.

[38] Boris Mityagin. 2015. The Zero Set of a Real Analytic Function. *arXiv preprint arXiv:1512.07276* (2015).

[39] Andriy Mnih and Russ R Salakhutdinov. 2007. Probabilistic Matrix Factorization. *NeurIPS* (2007).

[40] Federico Monti, Michael Bronstein, and Xavier Bresson. 2017. Geometric Matrix Completion with Recurrent Multi-Graph Neural Networks. *NeurIPS* (2017).

[41] Ka Hyun Park, Junghun Kim, Jinhong Jung, and U Kang. 2025. PiGLeT: Probabilistic Message Passing for Semi-Supervised Link Sign Prediction. In *ICDM*.

[42] Yong-chan Park, Jun-Gi Jang, and U Kang. 2021. Fast and Accurate Partial Fourier Transform for Time Series Data. In *KDD*.

[43] Yong-chan Park, Jongjin Kim, and U Kang. 2024. Fast Multidimensional Partial Fourier Transform with Automatic Hyperparameter Selection. In *KDD*.

[44] Yong-chan Park, Kisoo Kim, and U Kang. 2025. PuzzleTensor: A Method-Agnostic Data Transformation for Compact Tensor Factorization. In *KDD*.

[45] Yong-chan Park, SeungJoo Lee, and U Kang. 2026. Fast and Accurate Online Coupled Matrix-Tensor Factorization via Frequency Regularization. In *KDD*.

[46] Nikhil Rao, Hsiang-Fu Yu, Pradeep K Ravikumar, and Inderjit S Dhillon. 2015. Collaborative Filtering with Graph Information: Consistency and Scalable Methods. *NeurIPS* (2015).

[47] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *UAI*.

[48] Ruslan Salakhutdinov and Andriy Mnih. 2008. Bayesian Probabilistic Matrix Factorization Using Markov Chain Monte Carlo. In *ICML*.

[49] Ruslan Salakhutdinov, Andriy Mnih, and Geoffrey Hinton. 2007. Restricted Boltzmann Machines for Collaborative Filtering. In *ICML*.

[50] Badrul Sarwar, George Karypis, Joseph Konstan, and John T Riedl. 2000. Application of dimensionality reduction in recommender system-a case study. (2000).

[51] Suvash Sedhain, Aditya Krishna Menon, Scott Sanner, and Lexing Xie. 2015. AutoRec: Autoencoders Meet Collaborative Filtering. In *WWW*.

[52] Wei Shen, Chuheng Zhang, Yun Tian, Liang Zeng, Xiaonan He, Wanchun Dou, and Xiaolong Xu. 2021. Inductive Matrix Completion Using Graph Autoencoder. In *CIKM*.

[53] Sangjun Son, Yong-chan Park, Minyong Cho, and U Kang. 2022. DAO-CP: Data-Adaptive Online CP Decomposition for Tensor Stream. *PLOS One* 17, 4 (2022).

[54] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. In *CIKM*.

[55] Gerald Teschl. 2012. *Ordinary Differential Equations and Dynamical Systems*. Vol. 140. American Mathematical Soc.

[56] Rianne Van Den Berg, N Kipf Thomas, and Max Welling. 2018. Graph Convolutional Matrix Completion. *KDD* (2018).

[57] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In *SIGIR*.

[58] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised graph learning for recommendation. In *SIGIR*.

[59] Le Wu, Xiangnan He, Xiang Wang, Kun Zhang, and Meng Wang. 2022. A Survey on Accuracy-Oriented Neural Recommendation: From Collaborative Filtering to Information-Rich Recommendation. *TKDE* 35, 5 (2022).

[60] Qitian Wu, Hengrui Zhang, Xiaofeng Gao, Junchi Yan, and Hongyuan Zha. 2021. Towards Open-World Recommendation: An Inductive Model-Based Collaborative Filtering Approach. In *ICML*.

[61] Shiwen Wu, Fei Sun, Wentao Zhang, Xu Xie, and Bin Cui. 2022. Graph Neural Networks in Recommender Systems: A Survey. *ACM Comput. Surv.* 55, 5 (2022).

[62] Jeongin Yun, Jaeri Lee, and U Kang. 2025. DART: Diversified and Accurate Long-Tail Recommendation. In *PAKDD*.

[63] Jeongin Yun, Jaeri Lee, Jongjin Kim, Minjun Kim, Jinho Song, and U Kang. 2026. SharVeT: Similarity-Aware Parameter Sharing with Vector-Based Tuning for Efficient LLM Compression. In *ACL*.

[64] Jiani Zhang, Xingjian Shi, Shenglin Zhao, and Irwin King. 2019. Star-GCN: Stacked and Reconstructed Graph Convolutional Networks for Recommender Systems. *arXiv preprint arXiv:1905.13129* (2019).

[65] Muhan Zhang and Yixin Chen. 2020. Inductive Matrix Completion Based on Graph Neural Networks. *ICLR* (2020).

[66] Yin Zheng, Bangsheng Tang, Wenkui Ding, and Hanning Zhou. 2016. A Neural Autoregressive Approach to Collaborative Filtering. In *ICML*.

[67] Seyed Sina Ziaee, Hossein Rahmani, and Mohammad Nazari. 2024. MoRGH: Movie recommender system using GNNs on heterogeneous graphs. *KAIS* (2024).

A Proof of Theorem 1

We use the following lemmas for the proof.

Lemma 3. *Let $f_1, \dots, f_K$ be real-analytic and linearly independent on an open interval $\mathcal{S} \subset \mathbb{R}$. Then there exist $t_1, \dots, t_K \in \mathcal{S}$ such that*

$$\det\left[f_k(t_p)\right]_{p,k=1}^{K} \neq 0.$$

Proof. Define the Wronskian $W(t) \coloneqq \det\left[f_k^{(p-1)}(t)\right]_{p,k=1}^{K}$. For real-analytic functions, linear independence implies $W \not\equiv 0$, hence there exists $t_0 \in \mathcal{S}$ with $W(t_0) \neq 0$ [55]. For small h, set $t_p = t_0 + (p-1)h \in \mathcal{S}$ and consider

$$D(h) \coloneqq \det\left[f_k(t_0 + (p-1)h)\right]_{p,k=1}^{K}.$$

By analyticity, $D(h)$ is analytic in h. A Taylor expansion around t_0 shows that the lowest-order nonzero term of $D(h)$ is proportional to $W(t_0) \cdot h^{K(K-1)/2}$ (times a nonzero Vandermonde constant), hence $D(h) \not\equiv 0$. Therefore, $D(h) \neq 0$ for some sufficiently small h, giving the desired $t_1, \dots, t_K$. □

Lemma 4. *Let $\Theta \subset \mathbb{R}^n$ be connected and open. If $g : \Theta \to \mathbb{R}$ is real-analytic and not identically zero, then $\{\theta \in \Theta : g(\theta) = 0\}$ has Lebesgue measure zero.*

Proof. This is a standard fact from real-analytic geometry: a nontrivial real-analytic function cannot vanish on a set with non-empty interior [26, 38]; its zero set is contained in a countable union of lower-dimensional analytic submanifolds, hence has Lebesgue measure zero. □

We now present the proof of Theorem 1.

Proof. Let $d \coloneqq \min\{I, J, KR\}$. We show that $\operatorname{rank}(\widehat{X}(\theta)) = d$ for Lebesgue-almost every parameter θ. The proof follows a standard *(existence)* + *(genericity)* template:

(1) *Existence.* The goal of this step is to exhibit *one* parameter θ_0 for which $\widehat{X}(\theta_0)$ is non-degenerate, namely, some $d \times d$ submatrix of $\widehat{X}(\theta_0)$ has nonzero determinant. Once such a witness exists, the generic claim follows immediately in Step 2. We construct θ_0 by selecting shifts that make appropriate submatrices invertible via (C2) and Lemma 3 (full-column construction when $KR \leq \min\{I, J\}$, and a block-matrix construction otherwise).
(2) *Genericity.* Step 1 yields θ_0 for which some $d \times d$ submatrix of $\widehat{X}(\theta_0)$ has nonzero determinant. Because this determinant is a real-analytic function of θ under (C1), it can vanish only on a Lebesgue-null set (Lemma 4). This leads to $\operatorname{rank}(\widehat{X}(\theta)) = d$ for almost all θ, which completes the proof.

Step 1: Existence of a parameter achieving rank d. We first present a concrete construction in the common case $KR \leq \min\{I, J\}$; the case $KR > \min\{I, J\}$ follows by restricting to d active columns.

[Case 1: $KR \leq \min\{I, J\}$] In this regime, $d = KR$. Assume without loss of generality that $I \leq J$. Choose disjoint index sets $G_1, \dots, G_R \subset [I]$ with $|G_r| = K$, and define $S \coloneqq \bigcup_{r=1}^{R} G_r$, so $|S| = KR$. We construct a "block one-hot" base factor $U \in \mathbb{R}^{I \times R}$ by

$$U_{i,r} \coloneqq \mathbf{1}\{i \in G_r\} = \begin{cases} 1, & i \in G_r, \\ 0, & \text{otherwise.} \end{cases}$$

Fix $r \in [R]$. By condition (C2), the K real-analytic functions $s \mapsto f_1^U(r; s), \dots, f_K^U(r; s)$ are linearly independent on $\mathcal{S}$. Then, applying Lemma 3, we can choose $t_{r,1}, \dots, t_{r,K} \in \mathcal{S}$ such that the matrix

$$W_r \coloneqq \left[f_k^U(r; t_{r,p})\right]_{p,k=1}^{K} \in \mathbb{R}^{K \times K}$$

is invertible. Enumerate $G_r = \{i_{r,1}, \dots, i_{r,K}\}$ and set $s_{i_{r,p}}^U \coloneqq t_{r,p}$ for all $p \in [K]$ (shifts for $i \notin S$ can be arbitrary).

Now consider the $KR \times KR$ submatrix $(U_{\text{cat}})_{S,:}$. For any $i \in G_r$ and $r' \neq r$, we have $U_{i,r'} = 0$, hence $(U \odot M_k^U)(i, r') = 0$ for all k. Thus, up to row/column permutations (grouping the selected rows by G_r and the concatenated columns by r), the matrix $(U_{\text{cat}})_{S,:}$ is block-diagonal across r. Here each "block" refers to the submatrix indexed by the row group G_r and the corresponding column group. Within the r-th block, the (p, k) entry equals

$$(U \odot M_k^U)(i_{r,p}, r) = U(i_{r,p}, r)\, M_k^U(i_{r,p}, r) = f_k^U(r; s_{i_{r,p}}^U) = f_k^U(r; t_{r,p}).$$

Consequently, after suitable permutations,

$$(U_{\text{cat}})_{S,:} \sim \operatorname{blockdiag}(W_1, \dots, W_R),$$

which is invertible, so $\operatorname{rank}(U_{\text{cat}}) = KR$. By the symmetric construction, $\operatorname{rank}(V_{\text{cat}}) = KR$. Thus, $\operatorname{rank}(\widehat{X}) = \operatorname{rank}(U_{\text{cat}} V_{\text{cat}}^\top) = KR$.

[Case 2: $KR > \min\{I, J\}$]. In this regime, $d = \min\{I, J\}$. Assume without loss of generality that $I \leq J$, so $d = I$. It suffices to construct an $I \times I$ submatrix of $\widehat{X}$ with nonzero determinant.

Choose integers $m_1, \dots, m_R$ such that $0 \leq m_r \leq K$ and $\sum_{r=1}^{R} m_r = I$ (existence holds since $KR \geq I$). Pick disjoint sets $G_r \subset [I]$ with $|G_r| = m_r$ and $\bigcup_{r=1}^{R} G_r = [I]$. Define $U \in \mathbb{R}^{I \times R}$ and $V \in \mathbb{R}^{J \times R}$ by

$$U_{i,r} \coloneqq \mathbf{1}\{i \in G_r\}, \qquad V_{j,r} \coloneqq \mathbf{1}\{j \in G_r\} \text{ for } j \in [I],$$

and set $V_{j,r} = 0$ for $j \notin [I]$. Then for $i \in G_r$ and $j \in G_{r'}$ with $r \neq r'$, we have $U_{i,r'} = 0$ and $V_{j,r} = 0$, so cross-component contributions vanish on $[I] \times [I]$. Hence, up to row/column permutations (grouping indices by G_r), the $I \times I$ submatrix $\widehat{X}_{[I],[I]}$ is block-diagonal across r, with diagonal blocks

$$\widehat{X}_{G_r,G_r} = \sum_{k=1}^{K} \left(U_{G_r,r} \odot (M_k^U)_{G_r,r}\right)\left(V_{G_r,r} \odot (M_k^V)_{G_r,r}\right)^\top =: A_r B_r^\top,$$

where $A_r, B_r \in \mathbb{R}^{m_r \times K}$ have entries $(A_r)_{p,k} = f_k^U(r; s_{i_{r,p}}^U)$ and $(B_r)_{q,k} = f_k^V(r; s_{i_{r,q}}^V)$, with $G_r = \{i_{r,1}, \dots, i_{r,m_r}\}$.

It remains to choose shifts so that each $A_r B_r^\top$ is invertible. By (C2) and Lemma 3, we can select points $t_{r,1}, \dots, t_{r,K} \in \mathcal{S}$ such that $W_r^U \coloneqq \left[f_k^U(r; t_{r,p})\right]_{p,k=1}^{K}$ is invertible; similarly, we can select $\tau_{r,1}, \dots, \tau_{r,K} \in \mathcal{S}$ such that $W_r^V \coloneqq \left[f_k^V(r; \tau_{r,q})\right]_{q,k=1}^{K}$ is invertible. Define $C_r \coloneqq W_r^U (W_r^V)^\top \in \mathbb{R}^{K \times K}$. Since W_r^U and W_r^V are invertible, so is C_r, and thus $\operatorname{rank}(C_r) = K$. Therefore, there exist index sets $P_r, Q_r \subset [K]$ with $|P_r| = |Q_r| = m_r$ such that $\det\left((C_r)_{P_r,Q_r}\right) \neq 0$. Assign the shifts by

$$s_{i_{r,p}}^U \coloneqq t_{r,P_r(p)}, \qquad s_{i_{r,q}}^V \coloneqq \tau_{r,Q_r(q)}, \qquad p, q \in [m_r],$$

where $P_r(p)$ (resp. $Q_r(q)$) denotes the p-th (resp. q-th) element in P_r (resp. Q_r). With this choice, the resulting block satisfies $A_r B_r^\top = (C_r)_{P_r,Q_r}$, and hence is invertible. Thus, every diagonal block of $\widehat{X}_{[I],[I]}$ is invertible, so $\det(\widehat{X}_{[I],[I]}) \neq 0$. This implies $\operatorname{rank}(\widehat{X}) \geq I = d$, and since always $\operatorname{rank}(\widehat{X}) \leq d$, we conclude $\operatorname{rank}(\widehat{X}) = d$.

Step 2: Genericity via analytic determinants. Under (C1), each entry of $\widehat{X}(\theta)$ is real-analytic in θ, hence every $d\times d$ submatrix determinant is real-analytic. From Step 1, choose θ_0 with $\operatorname{rank}(\widehat{X}(\theta_0)) = d$. Then there exist $S \subset [I]$, $T \subset [J]$ with $|S| = |T| = d$ such that

$$g(\theta_0) \coloneqq \det\big(\widehat{X}_{S,T}(\theta_0)\big) \neq 0.$$

Here, we write $\widehat{X}_{S,T}$ for the $|S|\times|T|$ submatrix obtained by restricting $\widehat{X}$ to rows S and columns T. Define $g(\theta) \coloneqq \det(\widehat{X}_{S,T}(\theta))$, which is real-analytic and not identically zero. By Lemma 4, the set $Z(g) \coloneqq \{\theta : g(\theta) = 0\}$ has measure zero. If $\operatorname{rank}(\widehat{X}(\theta)) < d$, then all $d \times d$ submatrix determinants vanish, in particular $g(\theta) = 0$, hence

$$\{\theta : \operatorname{rank}(\widehat{X}(\theta)) < d\} \subseteq Z(g),$$

which has measure zero, so $\operatorname{rank}(\widehat{X}(\theta)) = d$ for almost every θ. □

B Proof of Theorem 2

Proof. We show that $\widehat{X}_\theta(Q(t))$ cannot stay constant along any *non-diagonal* infinitesimal change of basis. More precisely, consider any differentiable function $Q(t) \in \mathrm{GL}_R$ with $Q(0) = I_R$. If $\widehat{X}_\theta(Q(t))$ were constant for all small $t \geq 0$, then necessarily its derivative at $t = 0$ must vanish. Thus, we proceed with the proof as follows:

(1) *Linearization.* Write $Q(t) = \exp(tC)$ so that the tangent is $C = Q'(0)$. Then define the linear map $\mathcal{L}_\theta(C) = \frac{d}{dt}\widehat{X}_\theta(\exp(tC))\Big|_{t=0}$. Constancy of $\widehat{X}_\theta(Q(t))$ implies $\mathcal{L}_\theta(C) = 0$.

(2) *Off-diagonal restriction.* Diagonal D corresponds to the unavoidable scaling symmetry $U \mapsto UD$, $V \mapsto VD^{-1}$. Therefore, the key is to show: for *almost every* θ,

$$\mathcal{L}_\theta(C) = 0 \text{ and } \operatorname{diag}(C) = 0 \quad \Longrightarrow \quad C = 0.$$

Equivalently, we have to show that $\mathcal{L}_\theta|_{\mathfrak{off}}$ is an injective map, where $\mathfrak{off} \subset \mathbb{R}^{R\times R}$ denotes the off-diagonal subspace.

(3) *Matricization of $\mathcal{L}_\theta$.* Choose an explicit basis of the off-diagonal subspace $\mathfrak{off}$ and represent $\mathcal{L}_\theta|_{\mathfrak{off}}$ by a matrix $L(\theta)$. Then:

$$\mathcal{L}_\theta|_{\mathfrak{off}} \text{ is injective} \quad \Longleftrightarrow \quad L(\theta) \text{ has full column rank.}$$

(4) *Generic injectivity.* The non-degeneracy assumption says this holds for at least one θ_0. Because entries of $L(\theta)$ are real-analytic in θ, the set of θ where $L(\theta)$ *fails* to have full column rank has Lebesgue measure zero.

Step 1: Linearization. Let $Q(t)$ be differentiable with $Q(0) = I_R$. Because we only care about the first-order behavior at $t = 0$, we may parameterize the function as

$$Q(t) = \exp(tC) \quad \text{for some } C \in \mathbb{R}^{R\times R},$$

so that $Q'(0) = C$ [37]. Also, using $\frac{d}{dt}Q(t)^{-1}\big|_{t=0} = -C$, we have $\frac{d}{dt}Q(t)^{-\top}\big|_{t=0} = -C^\top$. Recall that $\widehat{X}_\theta(Q)$ is obtained by applying the change of basis $(U, V) \mapsto (UQ,\ VQ^{-\top})$ inside the MMF reconstruction. Differentiating $\widehat{X}_\theta(Q(t))$ at $t = 0$ and collecting terms linear in C, we define

$$\mathcal{L}_\theta(C) \coloneqq \frac{d}{dt}\widehat{X}_\theta(Q(t))\Big|_{t=0},$$

where explicitly

$$\mathcal{L}_\theta(C) = \sum\nolimits_{k=1}^{K}\Big(\big((UC)\odot M_k^U\big)V_k^\top - U_k\big((VC^\top)\odot M_k^V\big)^\top\Big).$$

Here $\frac{d}{dt}(UQ(t))|_{t=0} = UC$ and $\frac{d}{dt}(VQ(t)^{-\top})|_{t=0} = -VC^\top$. This indicates that if $\widehat{X}_\theta(Q(t))$ is constant for all sufficiently small t, then its derivative must vanish:

$$\mathcal{L}_\theta(C) = 0.$$

Hence, any infinitesimal symmetry must lie in the kernel of $\mathcal{L}_\theta$.

Step 2: Off-diagonal restriction. A diagonal transform corresponds to coordinate-wise rescalings, which always preserve a bilinear factorization. Thus, we aim to rule out *off-diagonal* infinitesimal symmetries. Define the off-diagonal subspace

$$\mathfrak{off} \coloneqq \{C \in \mathbb{R}^{R\times R} : \operatorname{diag}(C) = 0\}.$$

Our goal is to show $\ker\big(\mathcal{L}_\theta|_{\mathfrak{off}}\big) = \{0\}$ for Lebesgue-almost every θ. Since $\mathcal{L}_\theta$ is linear in C, the following are equivalent:

$$\mathcal{L}_\theta|_{\mathfrak{off}} \text{ is injective} \iff \Big(\forall C \in \mathfrak{off},\ \mathcal{L}_\theta(C) = 0 \Rightarrow C = 0\Big)$$
$$\iff \ker\big(\mathcal{L}_\theta|_{\mathfrak{off}}\big) = \{0\}.$$

Step 3: Matricization of $\mathcal{L}_\theta$. We now represent $\mathcal{L}_\theta|_{\mathfrak{off}}$ as a matrix. For $p \neq q$, let $E_{p,q} \in \mathbb{R}^{R\times R}$ be the matrix with $(E_{p,q})_{p,q} = 1$ and all other entries 0. Then, the set

$$\{E_{p,q} : 1 \leq p, q \leq R,\ p \neq q\}$$

is a concrete basis of $\mathfrak{off}$, and $\dim(\mathfrak{off}) = R(R-1)$.

Define the coordinate map $\mathrm{vec}_{\mathfrak{off}} : \mathfrak{off} \to \mathbb{R}^{R(R-1)}$ by

$$C = \sum\nolimits_{p\neq q} c_{p,q}E_{p,q} \quad \mapsto \quad \mathrm{vec}_{\mathfrak{off}}(C) = (c_{1,2}, c_{1,3}, \ldots, c_{R,R-1})^\top,$$

in any fixed ordering of pairs (p, q) with $p \neq q$. Also vectorize the output $\mathbb{R}^{I\times J}$ as $\mathrm{vec}(\cdot) \in \mathbb{R}^{IJ}$. Then, because $\mathcal{L}_\theta$ is linear, there exists a matrix $L(\theta) \in \mathbb{R}^{IJ\times R(R-1)}$ such that for all $C \in \mathfrak{off}$,

$$\mathrm{vec}\big(\mathcal{L}_\theta(C)\big) = L(\theta)\,\mathrm{vec}_{\mathfrak{off}}(C).$$

Note that a linear map $x \mapsto Ax$ is injective iff the null space of A is trivial. This is equivalent to A having linearly independent columns. Hence, it follows that

$$\mathcal{L}_\theta|_{\mathfrak{off}} \text{ is injective} \quad \Longleftrightarrow \quad L(\theta) \text{ has full column rank.}$$

Step 4: Generic injectivity. By the real-analytic parameterization assumption, every entry of $L(\theta)$ is a real-analytic function of θ. The non-degeneracy premise says that there exists θ_0 such that $\mathcal{L}_{\theta_0}$ is injective on $\mathfrak{off}$, equivalently $L(\theta_0)$ has full column rank.

Thus, among the rows of $L(\theta)$, we can select $R(R-1)$ rows so that the resulting square submatrix has nonzero determinant. Concretely, there exists a choice of $R(R-1)$ row indices $\mathcal{I} \subset [IJ]$ such that the square matrix $L_{\mathcal{I}}(\theta) \in \mathbb{R}^{R(R-1)\times R(R-1)}$ (obtained by taking rows $\mathcal{I}$) satisfies

$$g(\theta) \coloneqq \det\big(L_{\mathcal{I}}(\theta)\big), \qquad g(\theta_0) \neq 0.$$

Because determinants are polynomial expressions of matrix entries, $g(\theta)$ is real-analytic in θ. Moreover $g \not\equiv 0$ since $g(\theta_0) \neq 0$. Therefore, by Lemma 4, the zero set $\{\theta : g(\theta) = 0\}$ has Lebesgue measure zero. Hence, for Lebesgue-almost every θ, we have $g(\theta) \neq 0$, which implies $L_{\mathcal{I}}(\theta)$ is invertible, so $L(\theta)$ has full column rank and $\mathcal{L}_\theta|_{\mathfrak{off}}$ is injective.

Consequently, for almost every θ, the condition $\mathcal{L}_\theta(C) = 0$ forces $C \in \mathfrak{off}$ to be zero. Therefore, $C = Q'(0)$ must be diagonal. This proves that MMF admits no infinitesimal non-diagonal rotations for Lebesgue-almost every θ. □