DA626
UNIVERSAL CDR

Cross Domain Recommendation

By - Ankita Anand, Chhandam Pal, Anjum Ulazim

INTRODUCTION
CDR

- Some web services have abundant user data (source domains), while others lack data (target domains).
- CDR transfers knowledge from source to target domains to enhance recommendations.
- Data sparsity CDR addresses limited user interactions, while Cold-start CDR focuses on users with no interactions in the target domain.

PREVIOUS APPROACHES

Existing CDR methods can address either data sparsity or cold-start problems, but **not** both simultaneously

Prior works like **CoNet** and **Bi-TGCF** aim to transfer data across domains but focus on limited scenarios (either intra-domain or inter-domain recommendations)

PROPOSED FRAMEWORK

UNICDR

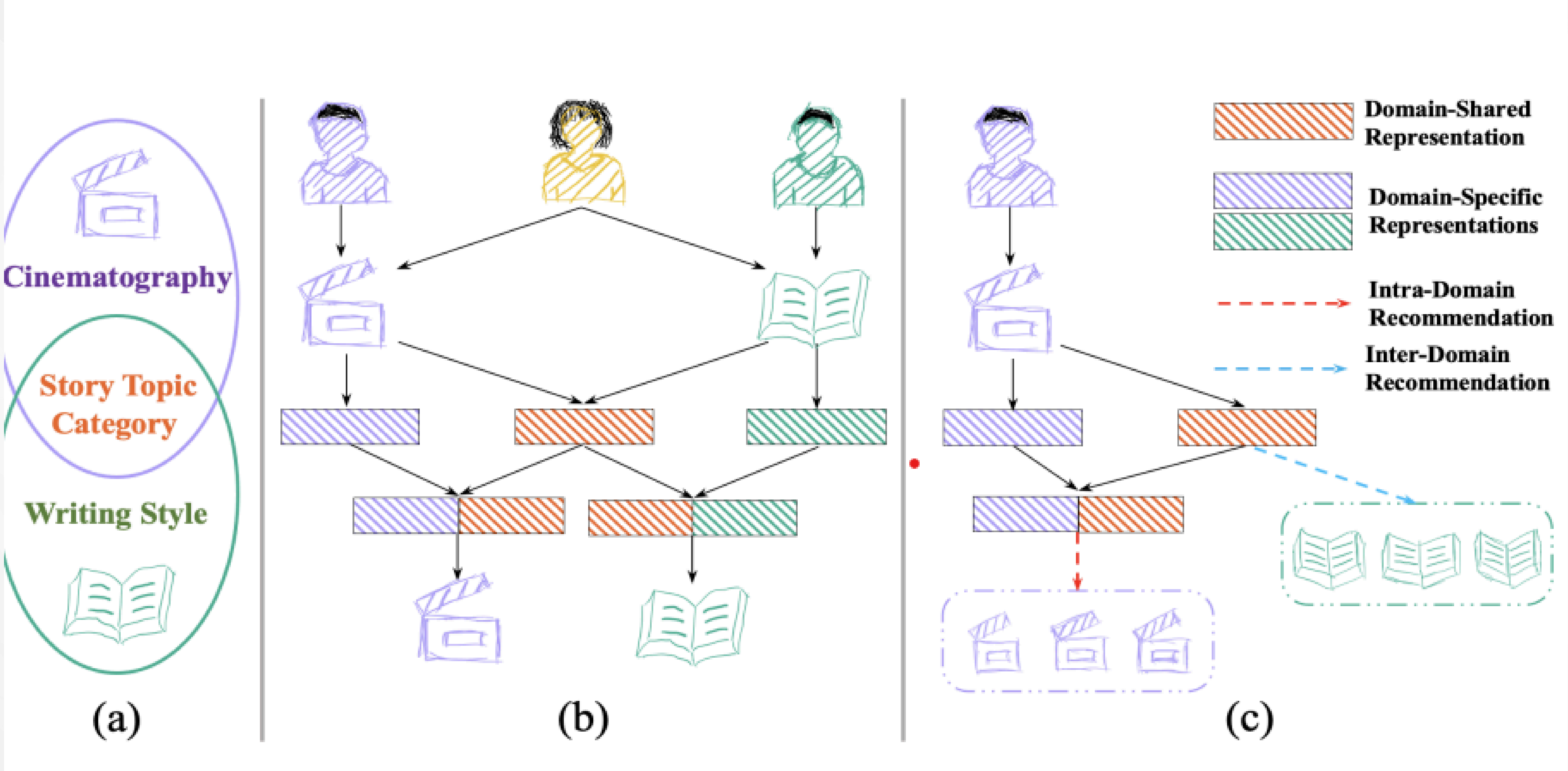
- UniCDR can handle multiple CDR scenarios, including data sparsity and cold-start, simultaneously.

- It Transfers shared information across domains, such as shared user preferences, while ignoring domain-specific preferences (e.g., writing style in books vs. cinematography in movies).

- The framework is trained to learn both domain-specific and domain-shared user/item representations. The shared representations are used for CDR.

TECHNICAL ROUTE

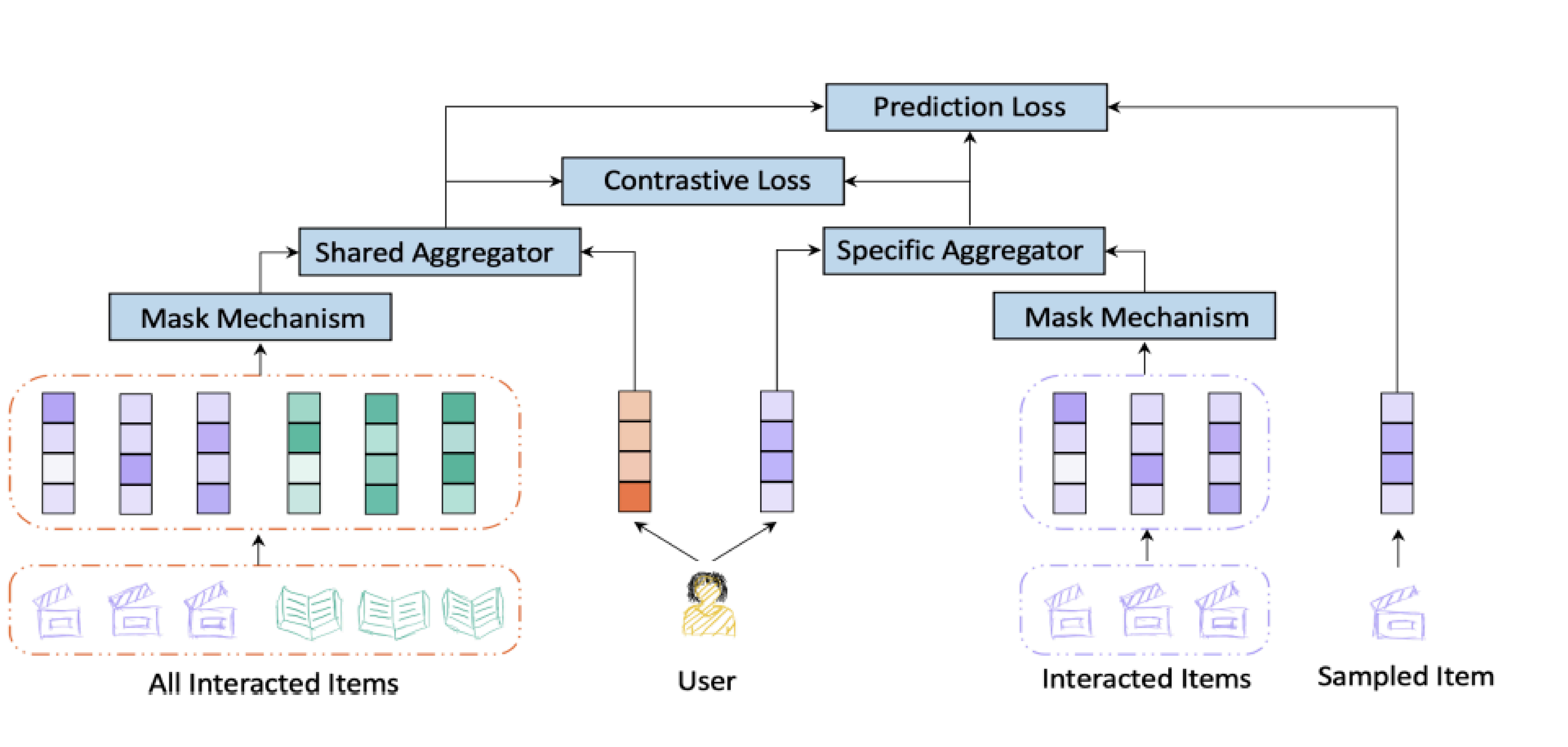
UNICDR



THE UNICDR MODEL

- **Embedding Layer:** It separates domain-specific and shared user/item representations.
- **Aggregator Architecture**: Three types of aggregators (mean-pooling, attention-based, and item-similarity-based) are used to combine user interactions with items.
- **Masking Mechanism & Contrastive Learning**: Augmented data is created by masking interactions and domains to encourage the model to learn invariant information between domains.
- **Contrastive Loss**: Helps the model distinguish between domain-shared and domain-specific representations by contrasting positive and negative pairs of interactions

THE UNICDR MODEL



EXPERIMENTS

CDR scenarios were explored in these two cases :

1. **Dual-user intra-domain recommendation**: Users interact with two different domains, and recommendations are made within each domain.
2. **Dual-user inter-domain recommendation**: Recommendations are made across two domains (e.g., recommending books based on movie preferences).
3. **Multi-item intra-domain recommendation:** Recommending items within same domain across multiple markets with overlapping items.
4. **Multi-user intra-domain recommendation:** Recommending items within same domain across multiple markets with overlapping userbase.

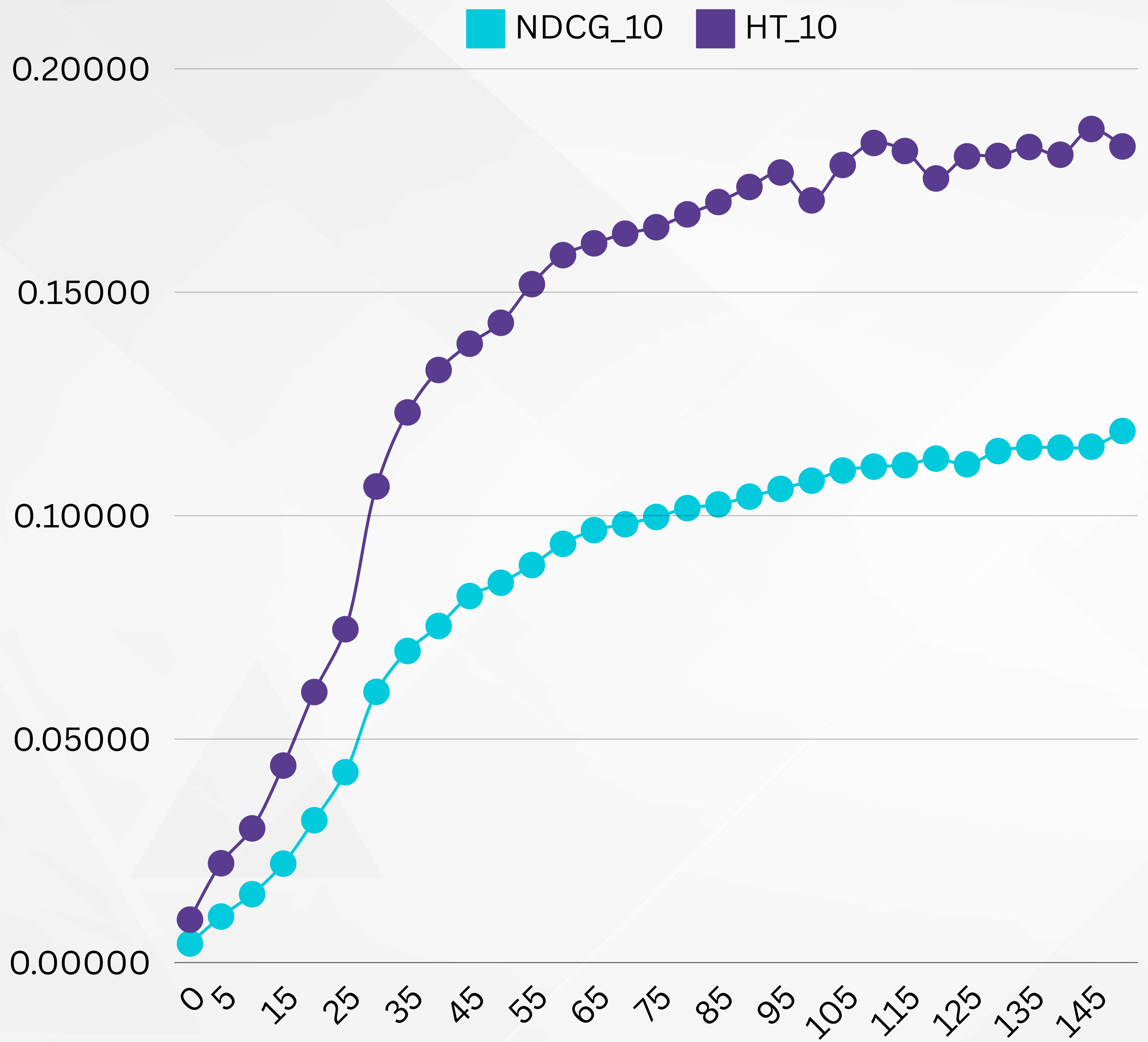
EVALUATION METRICS

The framework is evaluated using two common recommendation metrics:

- **Hit Ratio (HR):** Measures how often the correct item appears in the top-N recommendations.
- **Normalized Discounted Cumulative Gain (NDCG):** Measures the relevance of the recommended items, with higher relevance receiving more weight.

EVALUATION METRICS
(DUAL USER INTRA DOMAIN)

Metric	In Paper (100 epochs)	Our replication (150 epochs)
Hit Ratio (HR):	18.37	18.26
Normalized Discounted Cumulative Gain (NDCG):	10.98	11.89

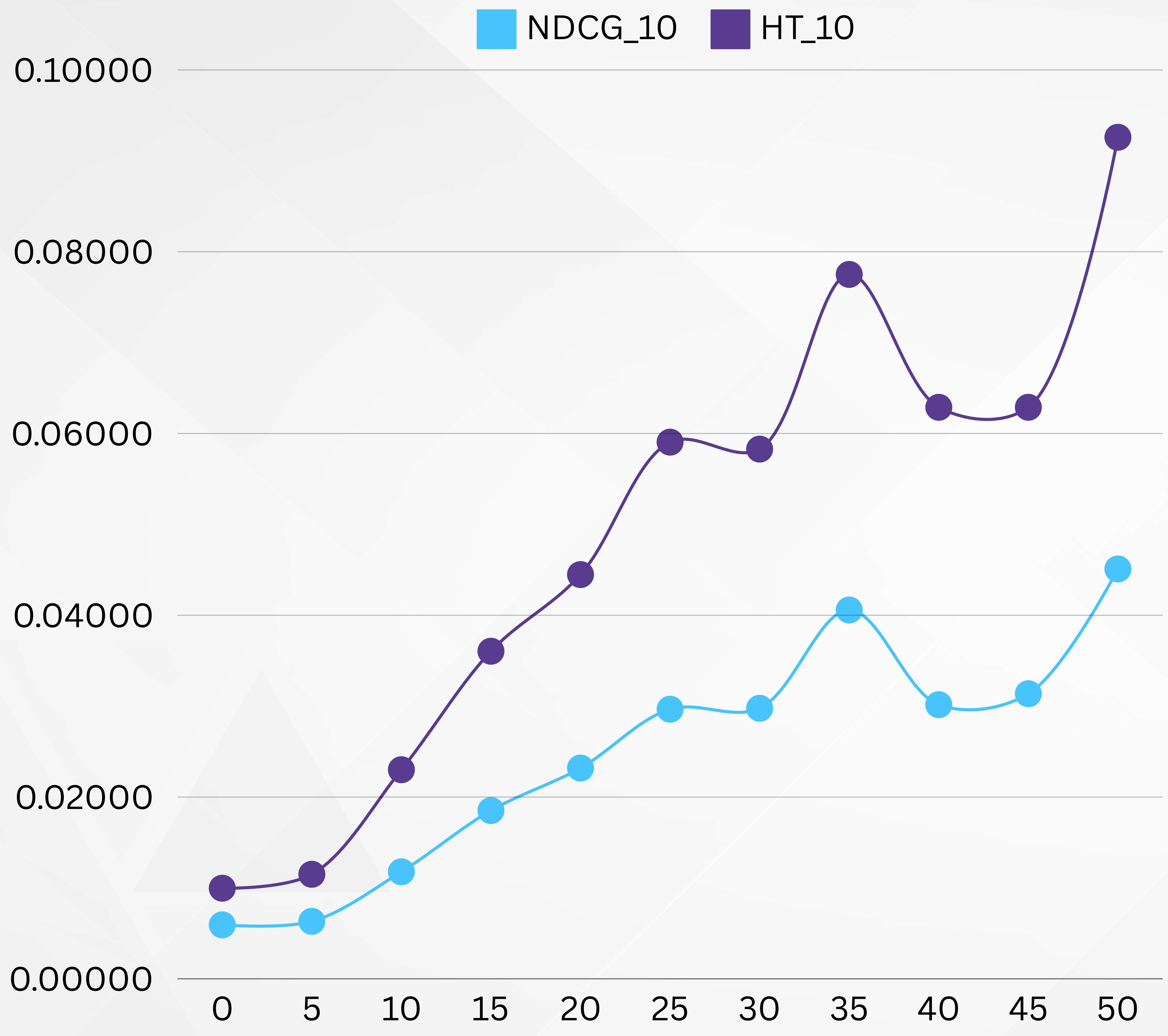


METRIC-SCORE VS EPOCH FOR

DUAL-USER INTRA-DOMAIN RECOMMENDATION

EVALUATION METRICS
(DUAL USER INTER DOMAIN)

Metric	In Paper (100 epochs)	Our replication (50 epochs)
Hit Ratio (HR):	8.78	9.25
Normalized Discounted Cumulative Gain (NDCG):	4.63	4.50

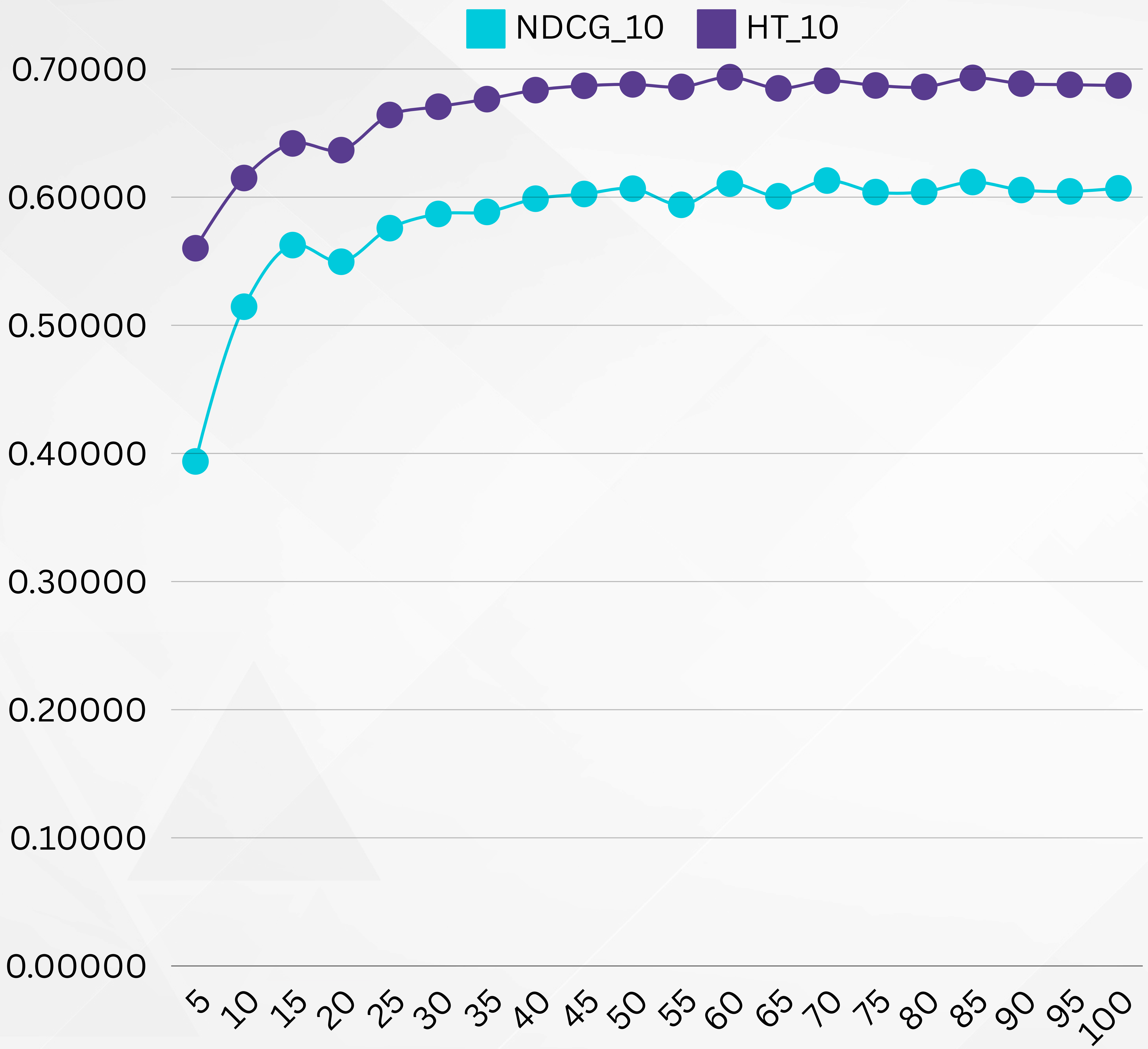


METRIC-SCORE VS EPOCH FOR

DUAL-USER INTER-DOMAIN RECOMMENDATION

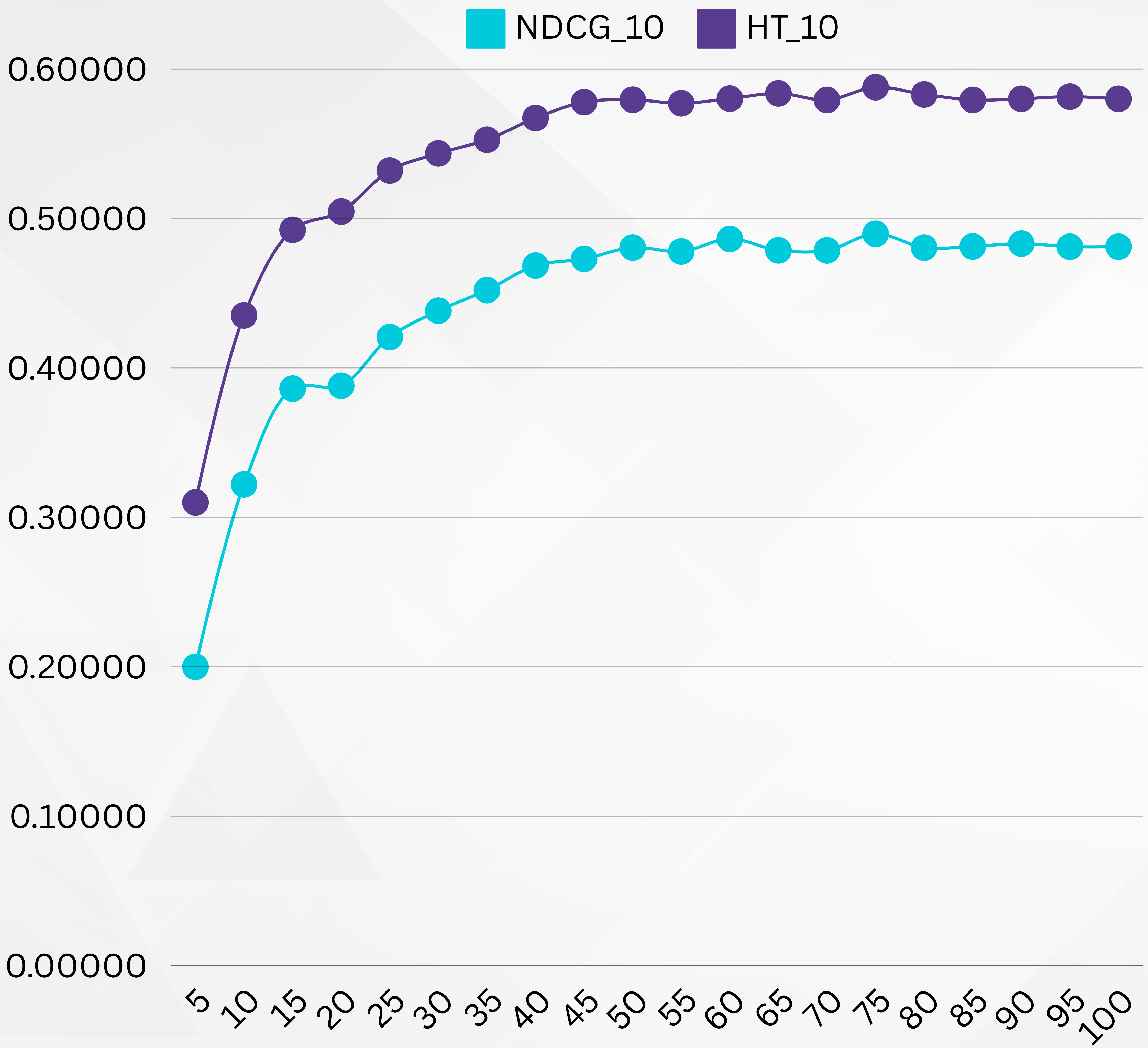
EVALUATION METRICS
(MULTI ITEM INTRA DOMAIN)

Dataset	NDCG@10 (Paper, 100 Epochs)	HR@10 (Paper, 100 Epochs)	NDCG@10 (our Results, 120 Epochs)	HR@10 (our Results, 120 Epochs)
M1	0.5957	0.6908	0.6086	0.6877
M2	0.4752	0.5801	0.4789	0.5771
M3	0.5324	0.6460	0.5263	0.6389
M4	0.4254	0.4752	0.4169	0.4646
M5	0.1704	0.1978	0.1483	0.1826



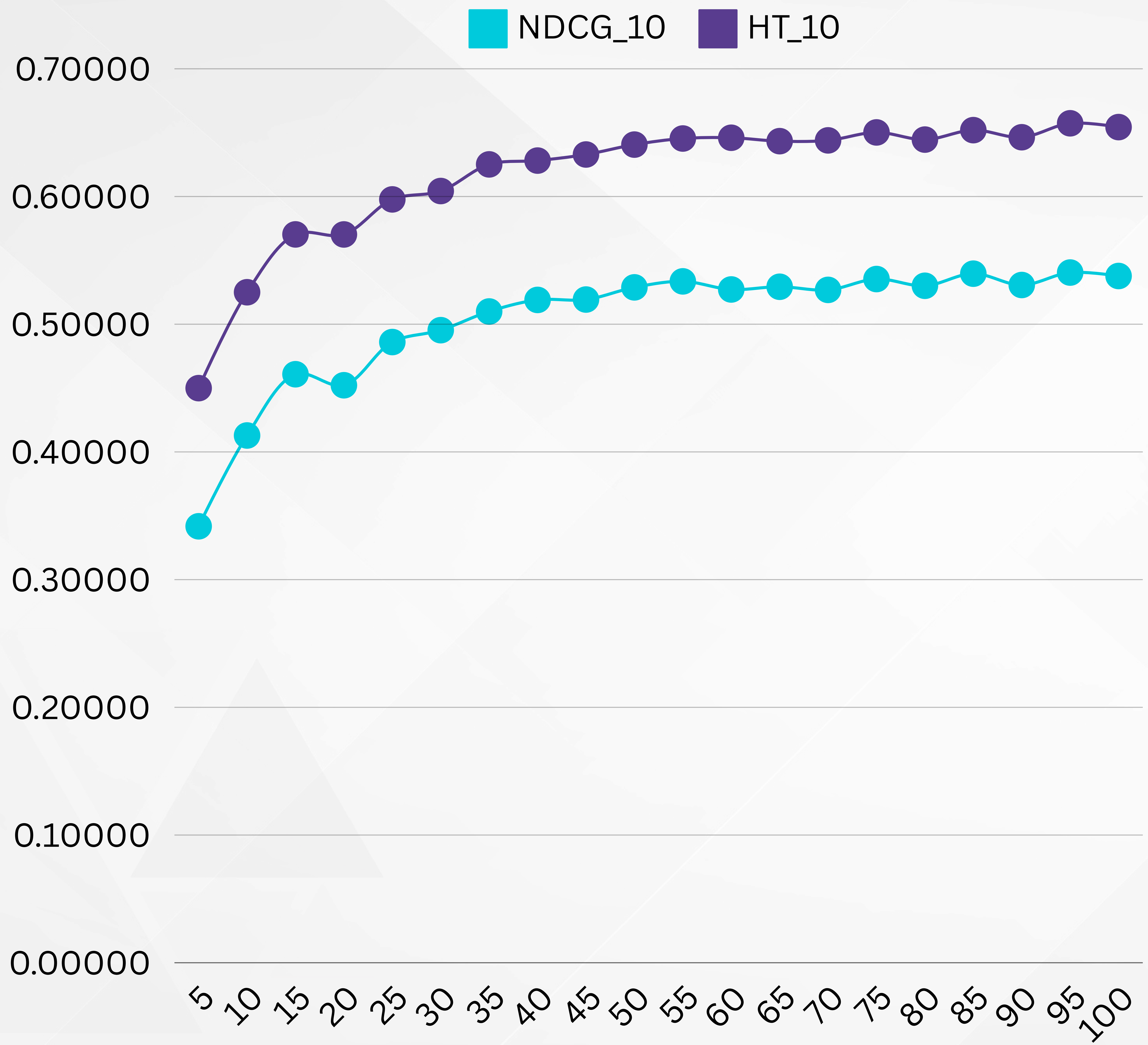
METRIC-SCORE VS EPOCH FOR

MULTI-ITEM INTRA-DOMAIN RECOMMENDATION (M1-DATASET)



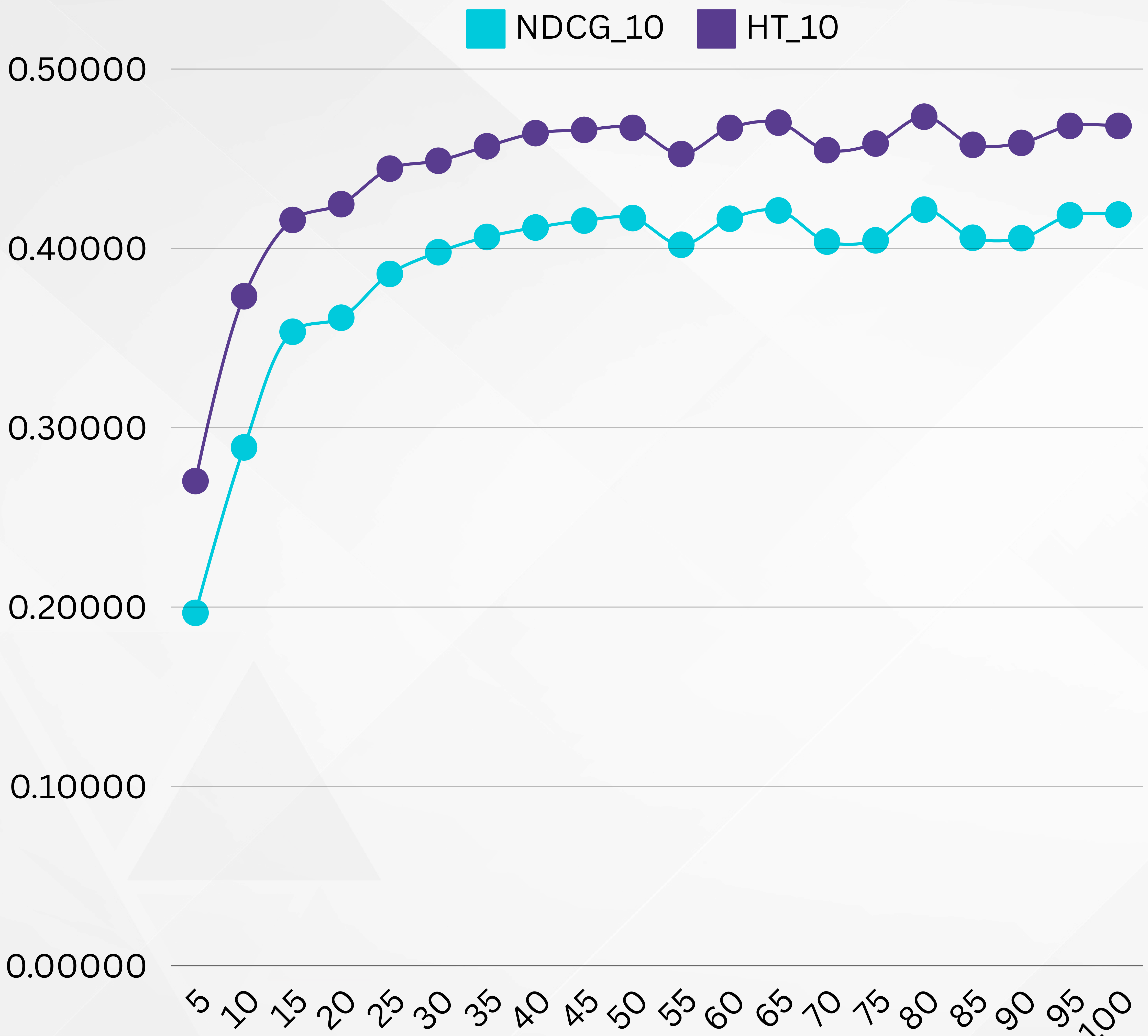
METRIC-SCORE VS EPOCH FOR

MULTI-ITEM INTRA-DOMAIN RECOMMENDATION (M2-DATASET)



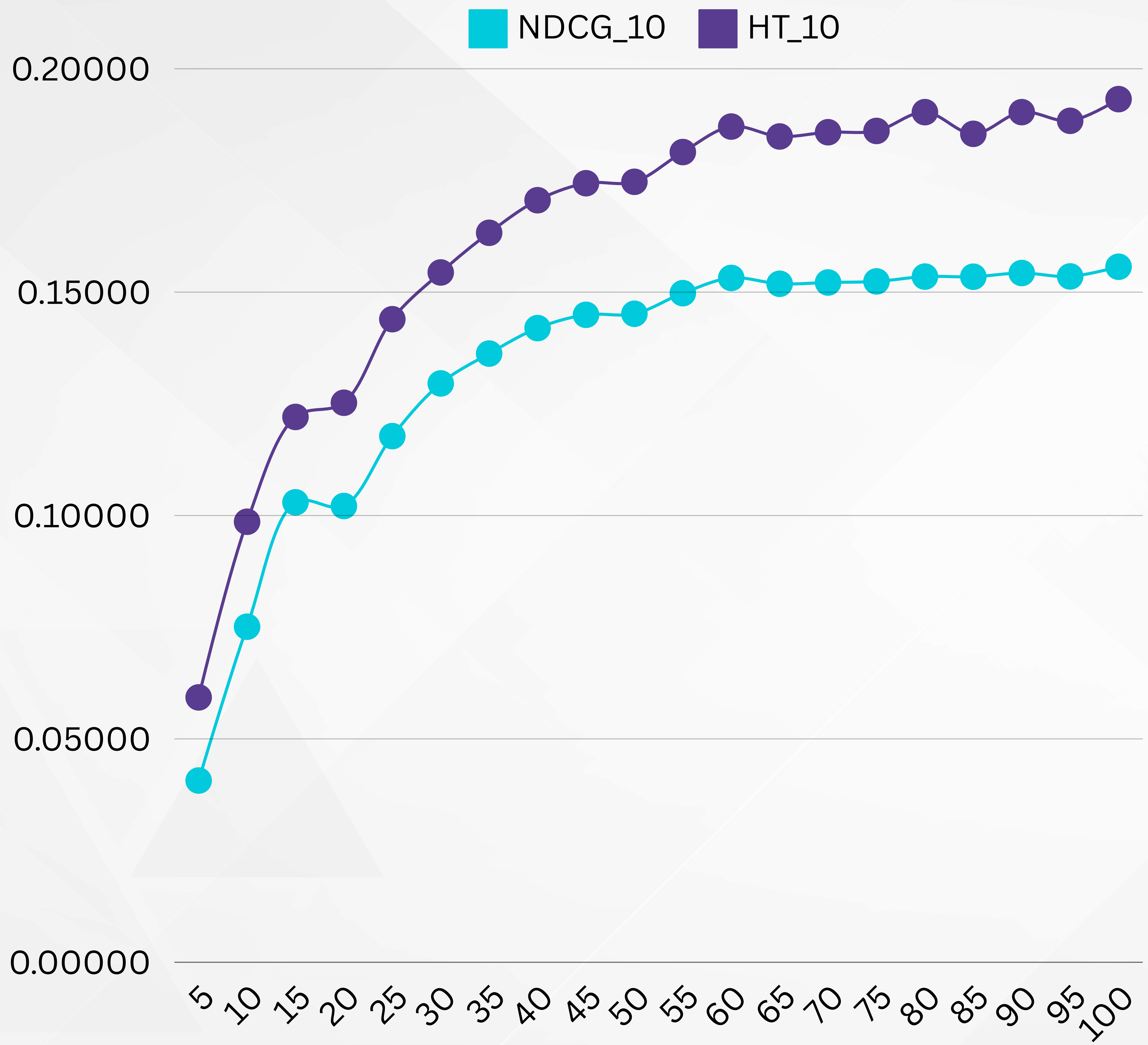
METRIC-SCORE VS EPOCH FOR

MULTI-ITEM INTRA-DOMAIN RECOMMENDATION (M3-DATASET)



METRIC-SCORE VS EPOCH FOR

MULTI-ITEM INTRA-DOMAIN RECOMMENDATION (M4-DATASET)



METRIC-SCORE VS EPOCH FOR

MULTI-ITEM INTRA-DOMAIN RECOMMENDATION (M5-DATASET)

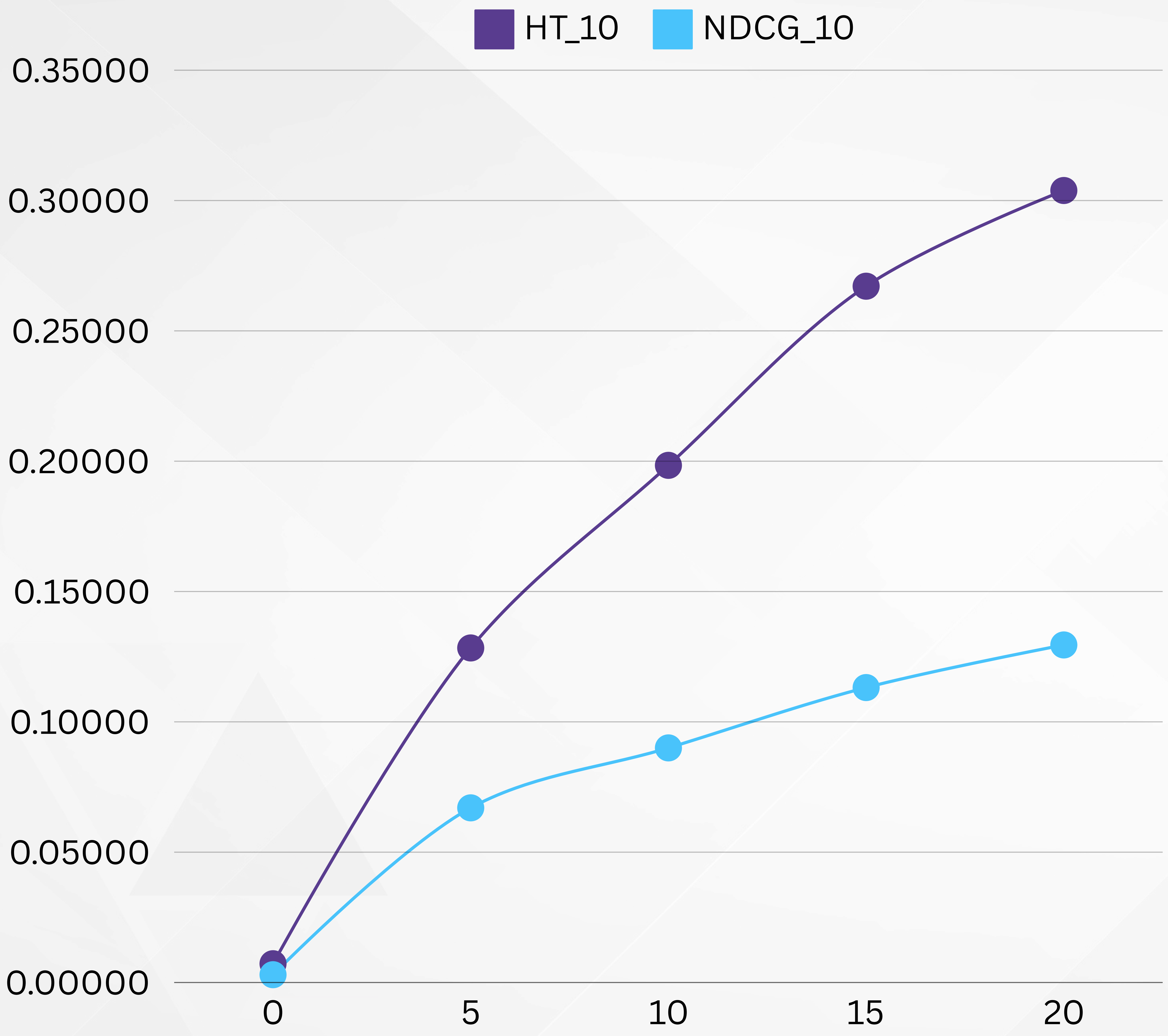
EVALUATION METRICS
(MULTI ITEM INTRA DOMAIN)

Key Observations:

1. M1: Slight improvement in NDCG@10 in Our results, with a minor decrease in HR@10.
2. M2: Nearly identical results in both metrics.
3. M3: Our results are very close, with minor declines.
4. M4: Small decreases in both metrics.
5. M5: Slightly lower performance in both NDCG@10 and HR@10.

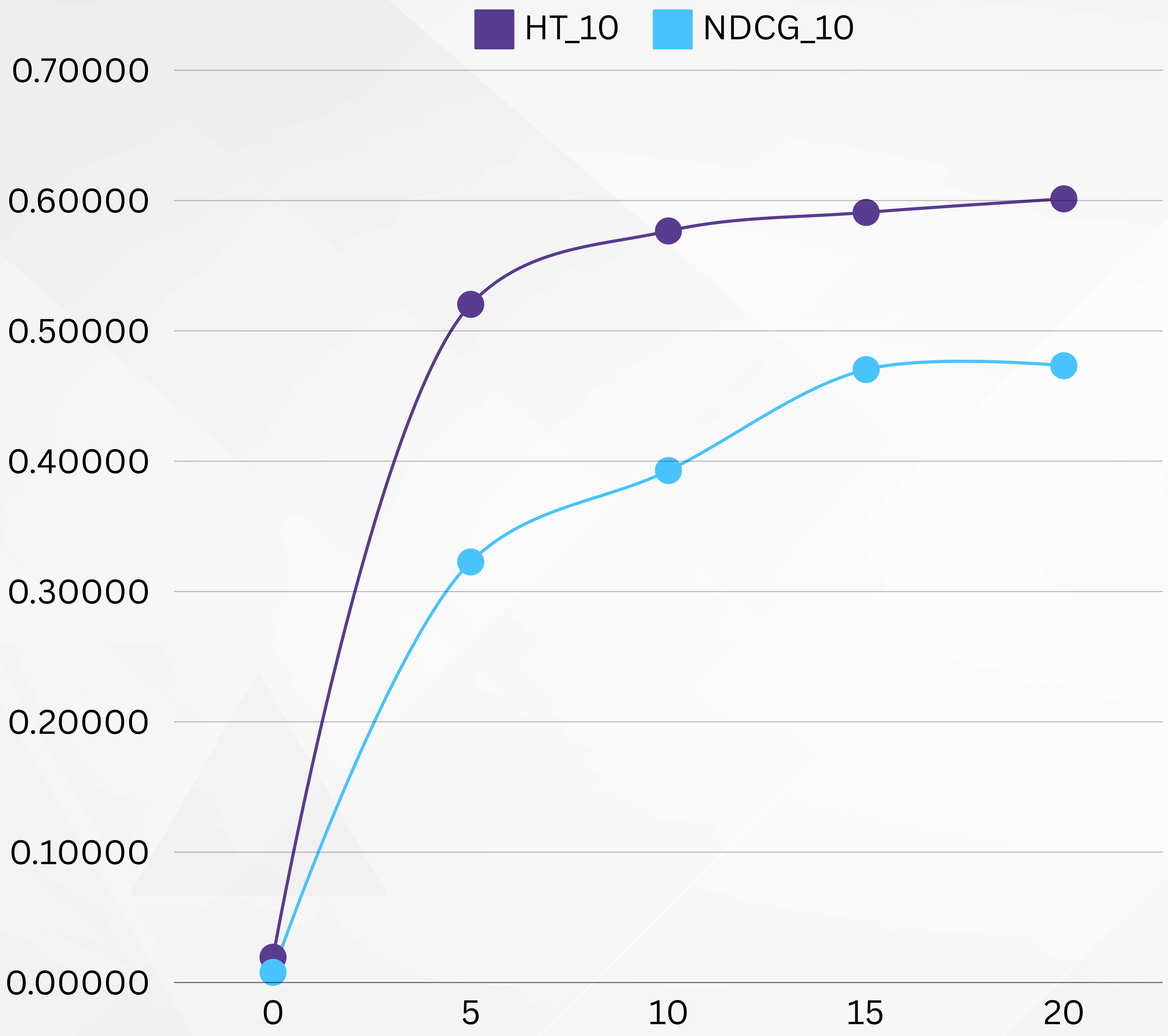
EVALUATION METRICS
(MULTI USERS INTRA DOMAIN)

Dataset	NDCG@10 (Paper, 100 Epochs)	HR@10 (Paper, 100 Epochs)	NDCG@10 (our Results, 20 Epochs)	HR@10 (our Results, 20 Epochs)
D1	0.1356	0.3260	0.1300	0.3024
D2	0.5048	0.6437	0.4750	0.6018



METRIC-SCORE VS EPOCH FOR

MULTI-USER INTRA-DOMAIN RECOMMENDATION (D1-DATASET)



METRIC-SCORE VS EPOCH FOR

MULTI-USER INTRA-DOMAIN RECOMMENDATION (D2-DATASET)

EVALUATION METRICS
(MULTI USERS INTRA DOMAIN)

Key Observations:

- For D1, we observed that our results are slightly lower than the paper in both NDCG@10 and HR@10.
- In D2, our HR@10 result is marginally higher than the paper’s, but our NDCG@10 score is a bit lower.
- For D3, although our NDCG@10 is somewhat lower than the paper’s, we see a slight improvement in HR@10.

PROBLEMS FACED IN PROJECT SETUP

Environment Setup Issues:

1. The GitHub repository instructions for package and Python versions were incomplete and conflicting.
2. Correct CUDA and CuDNN versions were essential but not specified.
3. We resolved conflicts by following official documentation and setting up the environment with:
   a. Python 3.7.9, PyTorch 1.6.0 with CUDA 10.1, and additional necessary packages (e.g., jsonlib-python3, pickle-mixin).

Challenges with Google Colab:

1. Google Colab doesn't support virtual environments well; changing Python and CUDA versions wasn't feasible.
2. We attempted using Miniconda with Google Colab, but the setup was unstable, forcing us to switch to our laptops for the tasks.

Hardware Constraints:

1. Google Colab's Tesla T4 GPU was powerful but limited by compatibility issues.
2. Switched to personal laptops with NVIDIA GTX 1650 GPUs, but encountered memory limitations during training on larger datasets.

CONCLUSION AND PERFORMANCE OF UNICDR

UniCDR Performance:

1. Showed strong performance on cross-domain recommendation tasks, with results close to those in the original study.
2. Effectively tackled challenges like limited data (data sparsity) and new users/items (cold start).

Key Observations:

1. High accuracy in scenarios with overlapping users/items between domains, showcasing effective preference transfer.
2. Minor performance gaps in complex datasets, likely due to dataset-specific complexity.
3. Faced memory constraints, which required adjusting batch sizes.

ANALYSIS OF AGGREGATING METHODS IN CDR

Experimenting with aggregating methods mean, user attention, item similarity,etc, to determine which one is better to work with in CDR case.

Mean-Pooling Aggregator:

- Best for Inter-Domain Recommendations
- Results (Sport): HR: 11.20, NDCG: 7.04
- Results (Cloth): HR: 12.48, NDCG: 7.52

User-Attention Pooling Aggregator:

- Best for Intra-Domain Recommendations
- Results (Sport): HR: 18.37, NDCG: 10.98
- Results (Cloth): HR: 17.85, NDCG: 11.20

Item-Similarity Pooling Aggregator:

- Best for Item-Overlapped Multi-Intra-Domain
- Results (M1): HR: 73.13, NDCG: 55.83
- Results (M3): HR: 66.53, NDCG: 43.35

Conclusion:

- Mean-Pooling: Effective for general inter-domain tasks.
- User-Attention: Enhances personalized intra-domain recommendations.
- Item-Similarity: Best for scenarios with item overlap.

THANK YOU