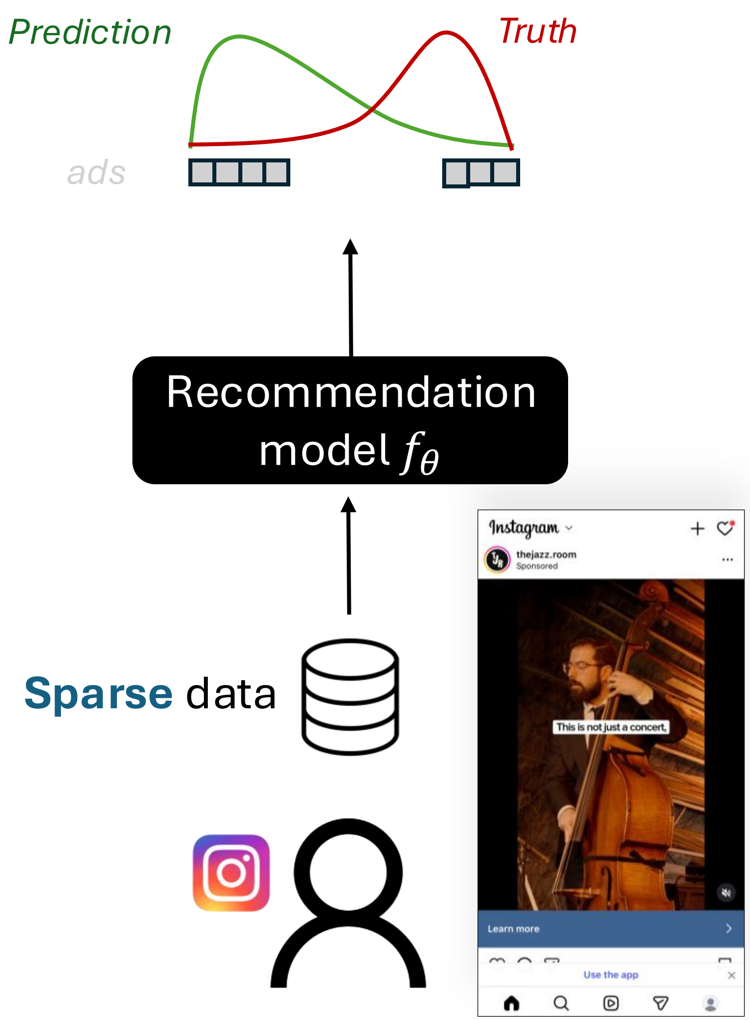
Multi-Modal Learning

[1] Cross Attention Secretly Performs Orthogonal Alignment in Recommendation models. ***Preprint***

Contents

1. Introduction



Problem: Building **recommendation models** that show advertisement to user. *(If user clicks an ad, that's how company earns money💰).*

Task: We train a model:

$$f_\theta(user_i - ads\ data) \in [0,1]$$

Which represents the probability that $user_i$ clicks on ad_j

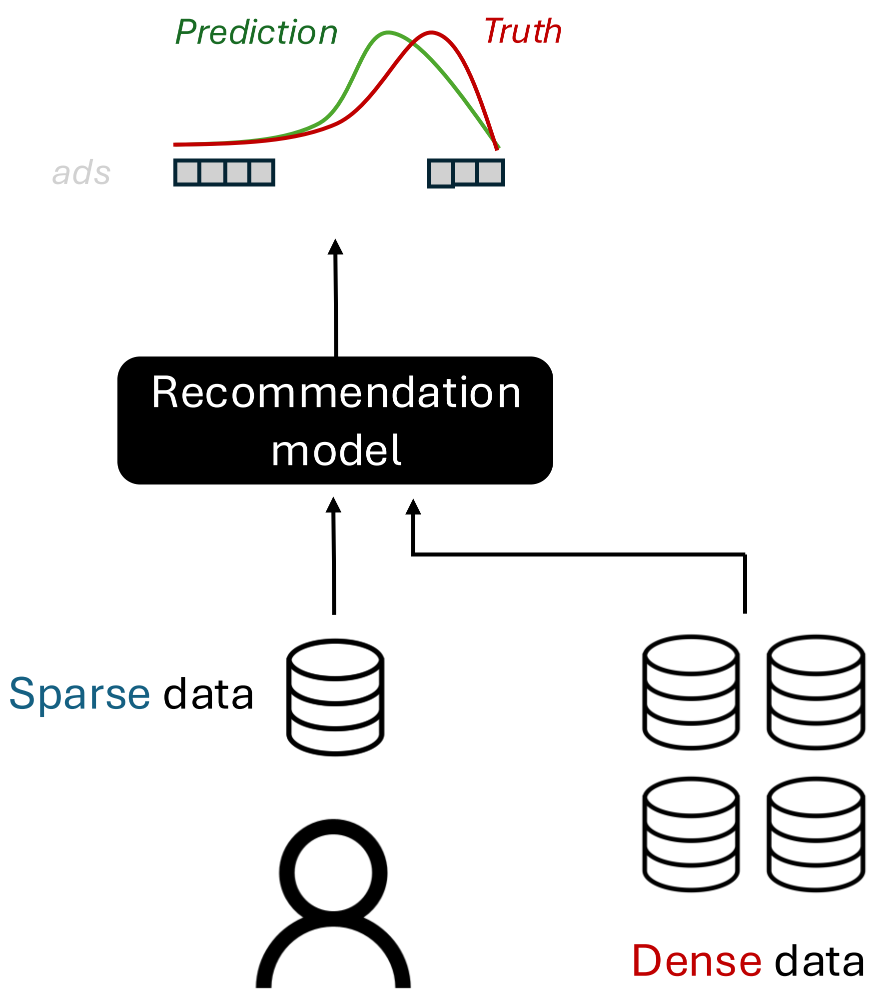
- Times series dataset

$$\boldsymbol{D} = \left\{\boldsymbol{user_i}, : \left\{\boldsymbol{time}, \boldsymbol{ad_j}, \{\boldsymbol{0}, \boldsymbol{1}\}\right\}\right\}_{i \in [I], j \in [J]}$$

Challenge: The key challenge was ads-domain data is **sparse**; users rarely click ads.

- Ex: {(14:00, ad-sport, **0**),
  (14:01 ,ad-movie, **0**),
  (14:02 ,ad-movie, **0**),
  ...(14:10, ad-Jazz, **0**)}

1. Introduction



Problem: Building recommendation models that display sponsored posts (ads) to user. *(If user clicks an ad, that's how company earns money💰).*

Challenge: The key challenge was ads-domain data is **sparse;** users rarely click ads.

Cross-Domain Sequential Recommendation(CDSR) : Let's use **dense** data from other domain

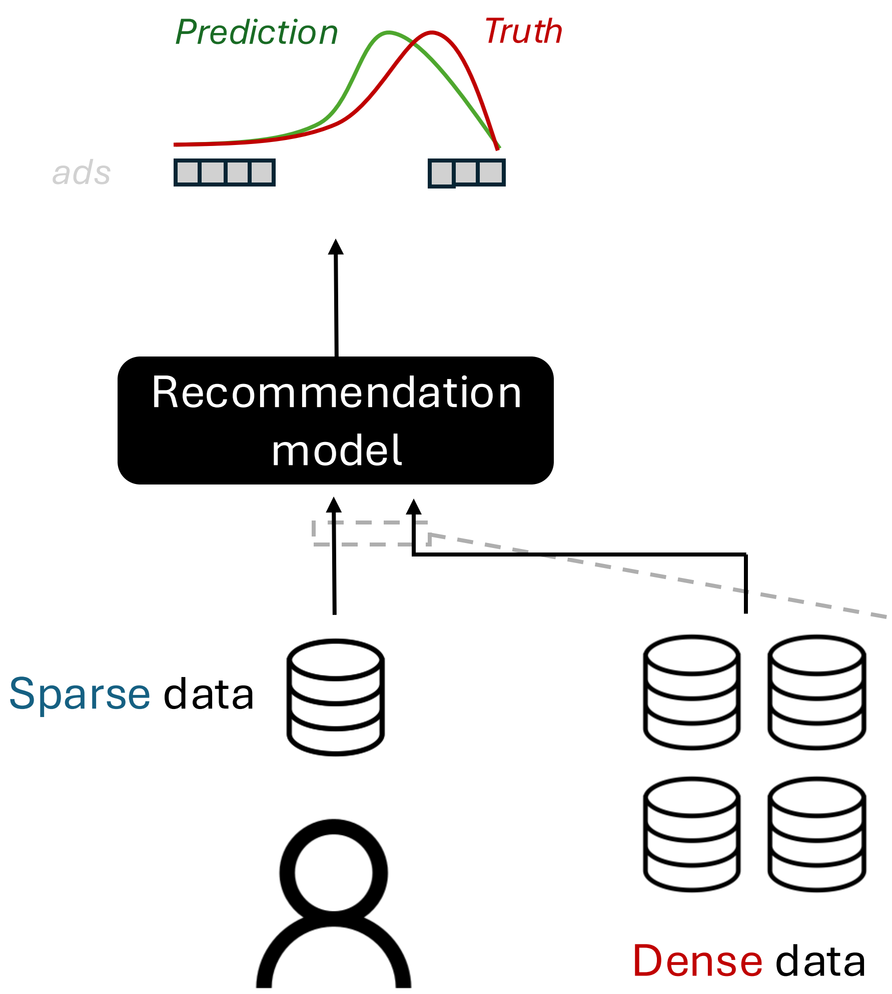
[Source Domain 1] Facebook app – post view duration.

- Ex: {(09/06, post-sport, 30s), (09/06, post-movie, 10s), (09/08, post-Jazz, 300s), ... }

[Source Domain 2] Instagram app – post likes.

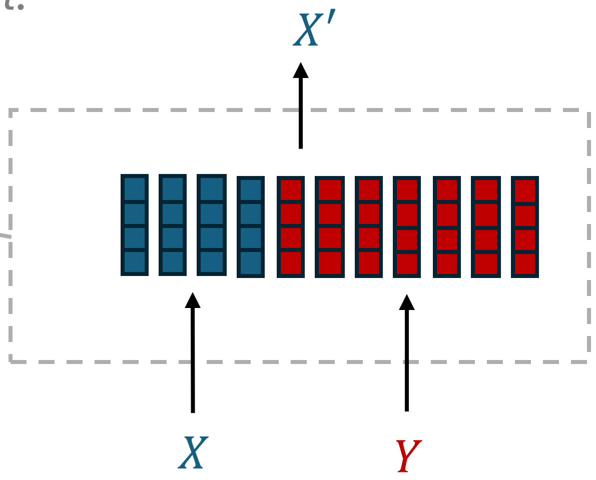
- Ex: {(09/06, post-sport, 1), (09/07, post-Jazz,1), (09/08, post-Jazz, 1), ... }

2. Research Question 1



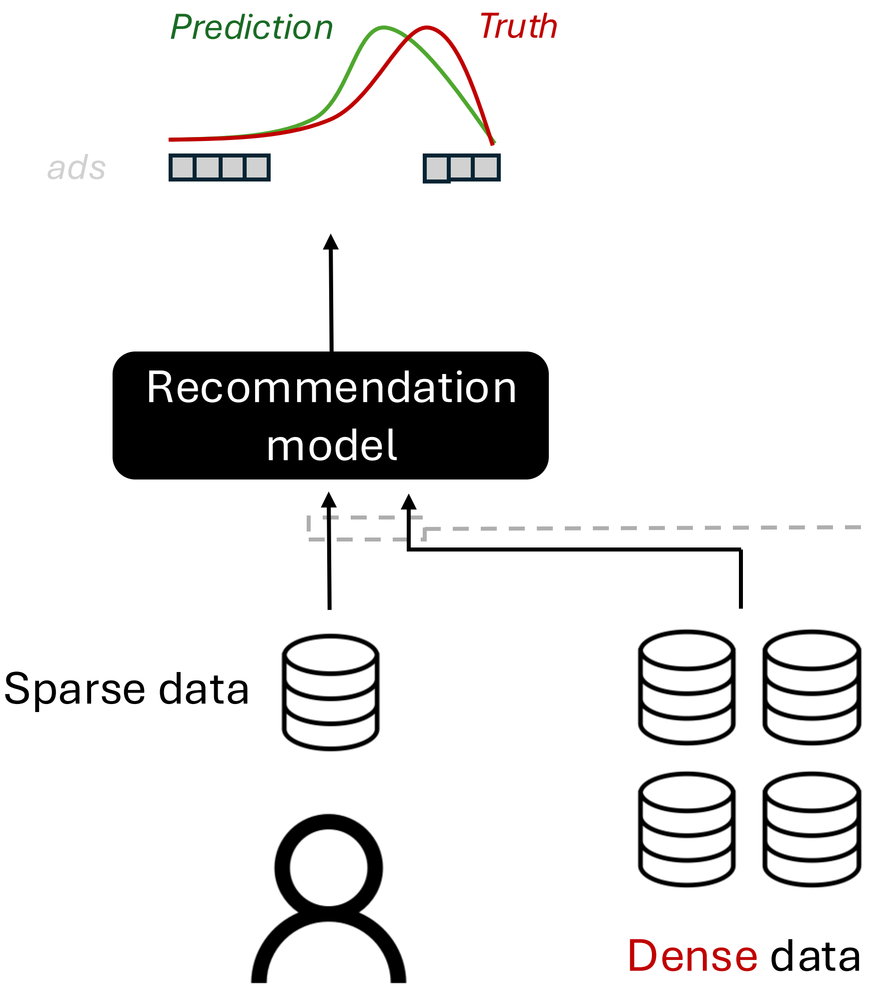
Research Question1: How to fuse those different modality data to improve the performance?

- Naive concatenation: Negative Transfer
  - Domain Noise
  - Preference conflicts: *(ex) user like to "view" video game post for a long time, but do not purchase since user just likes to watch streamer's game broadcast.*



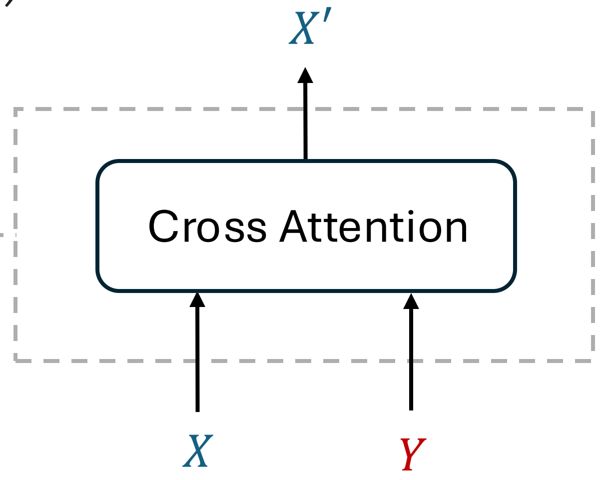
2. Research Question 1

2.1. Main Message 1



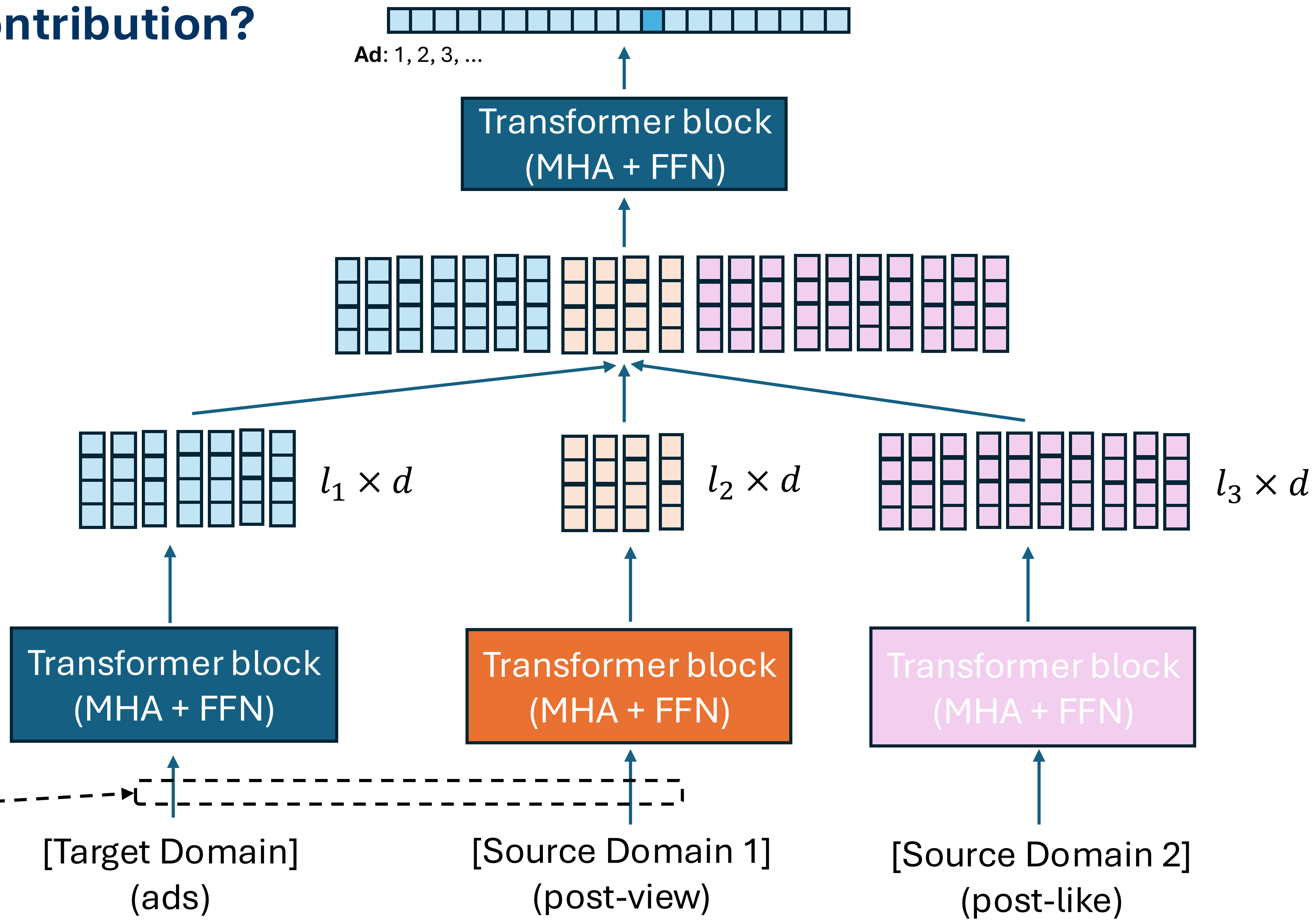
Research Question1: How to fuse those different modality data to improve the performance?

Main Message1: Using Gated Cross Attention at the Early Stage can improve the performance
(First Contribution)



2. Research Question 1

2.2. why contribution?

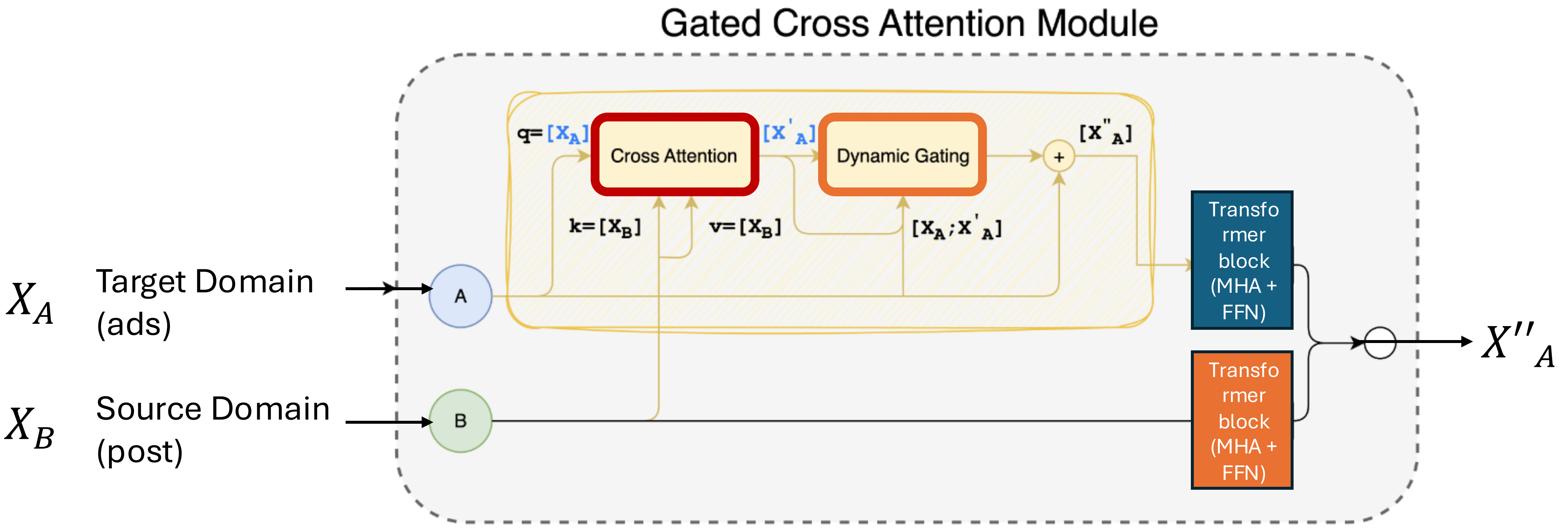


Main Message 1:

Early cross-sequence interactions improve performance

2. Research Question 1

2.3. Gated Cross Attention module



$X'_A = \mathrm{CrossAttention}(q=X_A,\ k=X_B,\ v=X_B)$

$X''_A = X_A + \mathrm{DyanmicGating}([X_A;X'_A]) \odot X'_A$

Soft update : Feed Forward Network. Gating depends on input.

2. Research Question 1

2.4. Experiment setting

- Baselines : CDSRNP, ABXI, LLM4CDSR
  - On top of baselines, we use Gated Cross Attention (GCA); that is skip connection: $X + \alpha X'$.
- Dataset: Four pairs from Amazon dataset (ex. Beauty-Electronic)

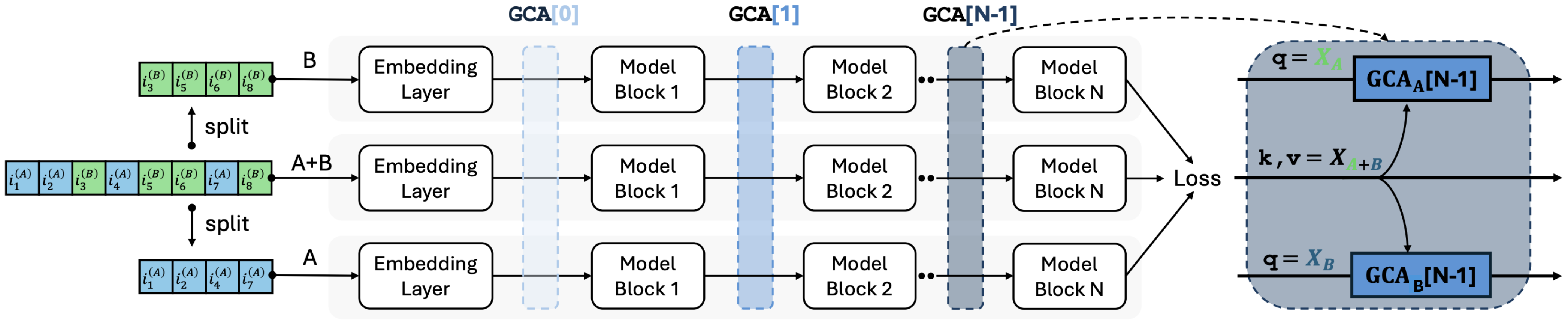


Figure 3 For each baseline model, we insert GCA modules at multiple vertical positions, denoted as $\mathtt{GCA}[i]$, where $i = 0$ corresponds to the module closest to the raw data and $i = N$ to the module farthest from the raw data. By design, $\mathtt{GCA}[0]$ is always placed immediately after the embedding layer, while $\mathtt{GCA}[1], \mathtt{GCA}[2], \ldots$ are positioned within intermediate layers of the backbone. Each $\mathtt{GCA}[i]$ comprises two parallel gated cross-attention modules, which respectively refine the representations of domains A and B.

2. Research Question 1

2.5. Observation 1 (Supports main message 1)

- GCA at the early stage consistently improves performance, but vertical stacking is not scalable.

GCA[0]

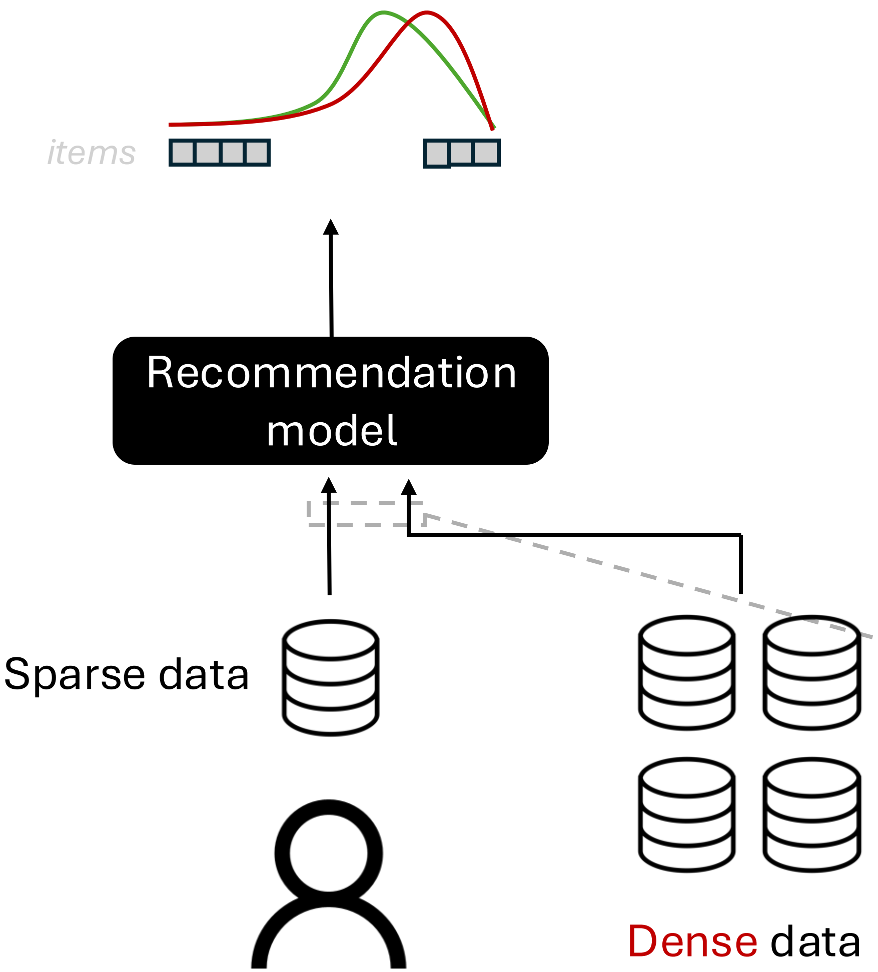
GCA[0,1]or GCA[0,2],..

Model	Dataset (A-B)	NDCG@1$_A$	NDCG@10$_A$	NDCG@1$_B$	NDCG@10$_B$	AUC$_A$	AUC$_B$
LLM4CDSR	Cloth-Sport	0.7157$\pm$0.0025	0.7821$\pm$0.0018	0.5870 $\pm$ 0.0051	0.6493 $\pm$ 0.002	0.9216 $\pm$0.0013	0.8621 $\pm$ 0.0054
+ GCA$_{early}$		0.7283$\pm$ 0.0027	0.8052$\pm$0.0014	0.5977 $\pm$ 0.0054	0.6560 $\pm$ 0.0046	0.9364 $\pm$0.0009	0.8655 $\pm$ 0.0038
+ GCA$_{stack}$		0.7310$\pm$ 0.0012	0.8056$\pm$0.0014	0.6112 $\pm$ 0.0032	0.6638 $\pm$ 0.0038	0.9370 $\pm$0.0010	0.8664 $\pm$ 0.0030
LLM4CDSR	Elec-Phone	0.2101$\pm$0.0030	0.3512$\pm$0.0009	0.1419 $\pm$ 0.0008	0.2608 $\pm$ 0.0010	0.7901 $\pm$0.0008	0.7197 $\pm$ 0.0011
+ GCA$_{early}$		0.2378$\pm$ 0.0011	0.3815$\pm$0.0018	0.1861 $\pm$ 0.0035	0.2845 $\pm$ 0.0027	0.7970 $\pm$0.0018	0.7218 $\pm$ 0.0026
+ GCA$_{stack}$		0.2410$\pm$ 0.0012	0.3800$\pm$0.0011	0.1994 $\pm$ 0.0054	0.3035 $\pm$ 0.0049	0.7937 $\pm$0.0013	0.7252 $\pm$ 0.0026
ABXI	Beauty-Elec	0.0730$\pm$0.0070	0.1724$\pm$0.0071	0.0548 $\pm$ 0.0038	0.1273 $\pm$ 0.0028	0.7216 $\pm$0.0027	0.7123 $\pm$ 0.0009
+ GCA$_{early}$		0.0727$\pm$ 0.0060	0.1793$\pm$0.0047	0.0544 $\pm$ 0.0044	0.1244 $\pm$ 0.0025	0.7410 $\pm$0.0025	0.7169 $\pm$ 0.0024
+ GCA$_{stack}$		0.0733$\pm$ 0.0042	0.1846$\pm$0.0057	0.0566 $\pm$ 0.0052	0.1271 $\pm$ 0.0042	0.7354 $\pm$0.0048	0.6973 $\pm$ 0.0051
ABXI	Food-Kitch	0.0593$\pm$0.0074	0.1541$\pm$0.0130	0.0416 $\pm$ 0.0058	0.1093 $\pm$ 0.0113	0.7205 $\pm$ 0.0015	0.7180 $\pm$ 0.0032
+ GCA$_{early}$		0.0703$\pm$ 0.0094	0.1757$\pm$0.0092	0.0548 $\pm$ 0.0053	0.1327 $\pm$ 0.0072	0.7317 $\pm$0.0039	0.7150 $\pm$ 0.0031
+ GCA$_{stack}$		0.0882$\pm$ 0.0052	0.1853$\pm$0.0013	0.0527 $\pm$ 0.0020	0.1282 $\pm$ 0.0028	0.7148 $\pm$0.0026	0.6924 $\pm$ 0.0009
CDSRNP	Elec-Phone (1M)	0.0499 $\pm$0.0087	0.1170$\pm$0.0079	0.0920 $\pm$ 0.0050	0.1935 $\pm$ 0.0021	-	-
+ GCA$_{early}$		0.0547$\pm$ 0.0010	0.1209 $\pm$0.0092	0.0980 $\pm$ 0.0010	0.1989 $\pm$ 0.0031	-	-
+ GCA$_{stack}$		0.0531$\pm$ 0.0078	0.1229$\pm$0.0125	0.0946$\pm$ 0.0022	0.1942 $\pm$ 0.0012	-	-

Table 1 NCDG and AUC comparison with the three baselines and adhoc model with GCA. Elec stands for Electronic, and Kitch stands for Kitchen. GCA$_{early}$ denotes GCA[0] and GCA$_{stack}$ denotes GCA$[0, i_1, i_2, .., i_N]$ where $i_n > 1, n \in [N]$

3. Research Question 2 (Main)

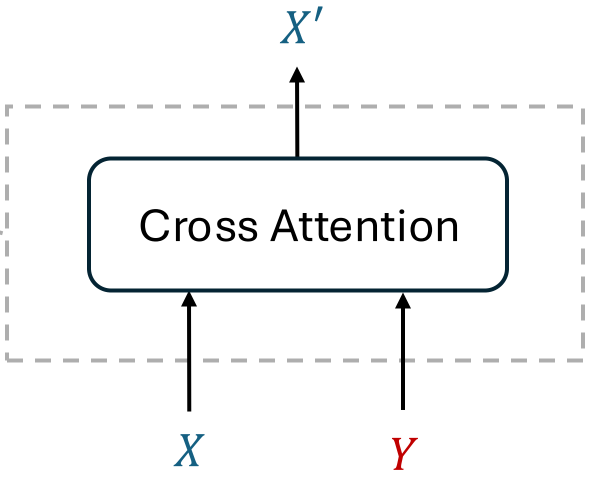
3.1. Research Question 2



So far:
Gated Cross Attention at the early stage can improve performance

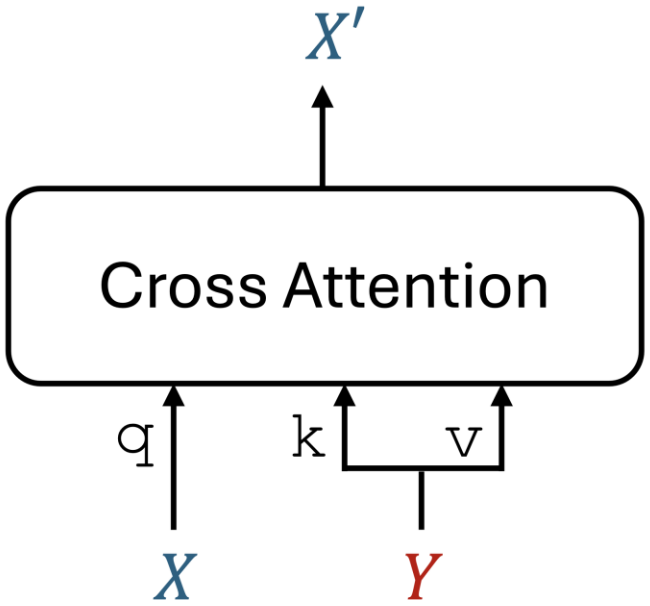
Research Question2:
Why gated cross attention improve the performance?

Observation by a chance:
As training goes by, $\cos(X, X')$ goes to zero.



3. Research Question 2 (Main)

3.2. Previous literature: conventional understanding on cross-attention as residual Alignment



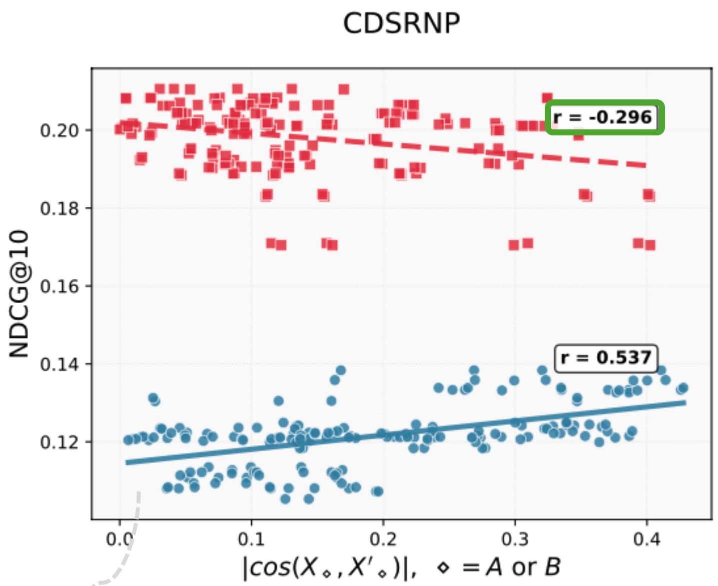
Residual Alignment:

- X' is generated by removing redundant information from X and preserving nonredundant information from X by referring to Y.
  - X : User likes to visit Milan, Italy
  - Y : User bought lots of sports players' uniforms
  - X' : User may likes to buy AC Millan's uniforms

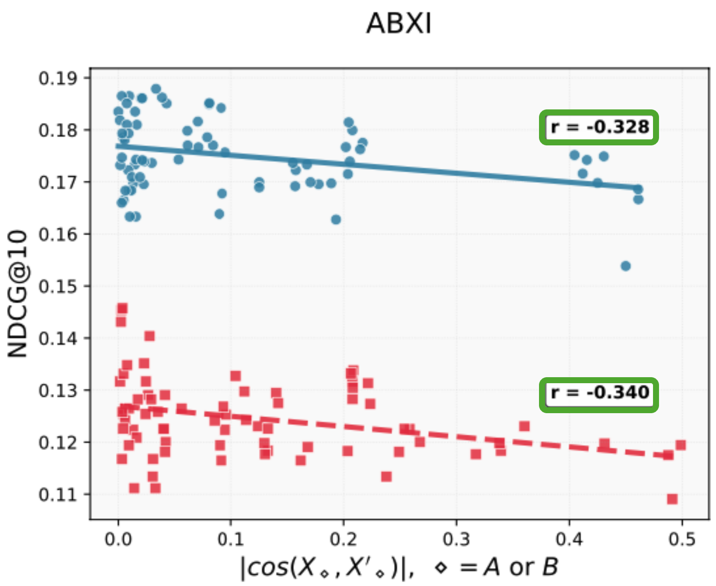
3. Research Question 2 (Main)

3.3. Observation 2 (Supports main message 2)

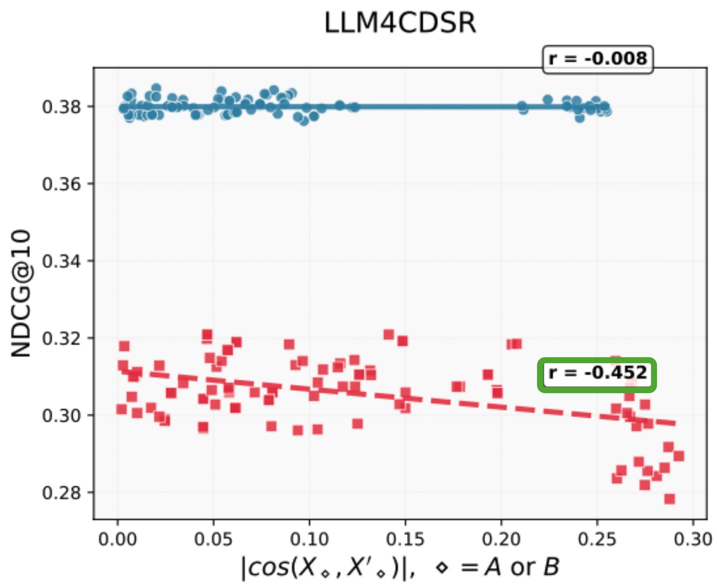
- We observe negative correlation between $\cos(X, X')$ and model performance regardless of dataset and baseline.



(a) CDSRNP



(b) ABXI

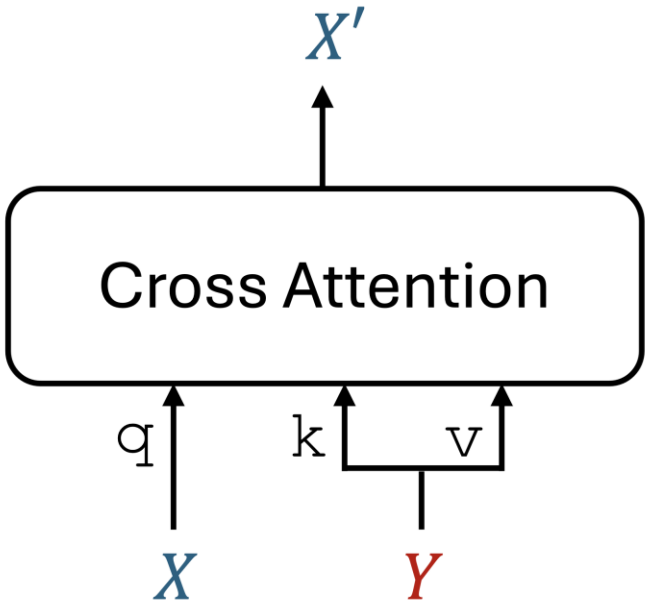


(c) LLM4CDSR

Vertical Stacking: {GCA[0], GCA[0,1], GCA[0,2],...}
Dataset : {Cloth-sports, Elec-phone, ..}
Hidden dimension: {64, 128, ...}
Num of attention heads: {4, 8, ...}

3. Research Question 2 (Main)

3.4. Main Message 2



Main Message 2

Orthogonal Alignment:

A phenomenon such that as X' and X getting orthogonal, then the model performance increases

- X' can be trained to contrain information irrelevant to X by referring to Y.
  - X : User likes to visit Milan, Italy
  - Y : User bought lots of sports players' uniforms
  - X' : User may likes to visit Mancherster, England

3. Research Question 2 (Main)

3.5. Observation 3 (Supports main message 2)

- GCA induces orthogonalization independently of how similar X and Y happens to be.

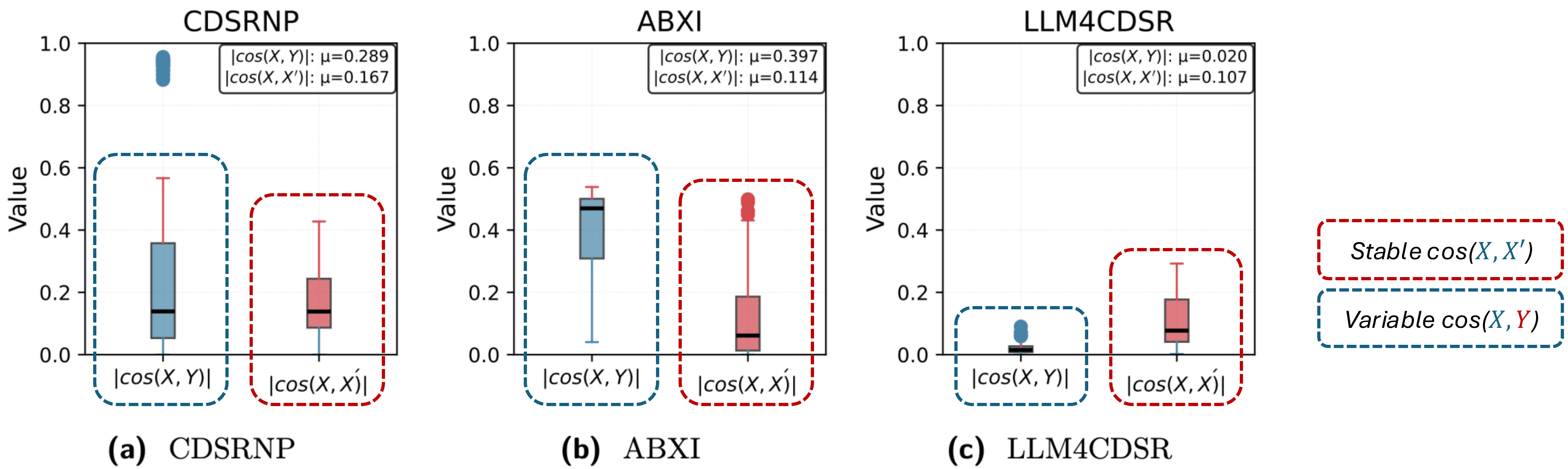
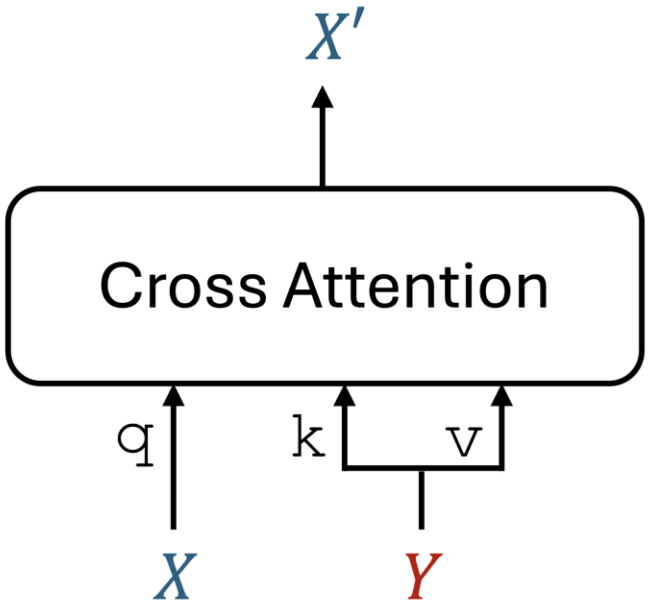


Figure 8 Boxplots of cosine similarities $|\cos(X, Y)|$ and $|\cos(X, X')|$. While $|\cos(X, X')|$ remains stable across models (median $\approx\in [0.1, 0.2]$), $|\cos(X, Y)|$ varies substantially depending on the dataset, highlighting that `GCA` induces a consistent degree of orthogonalization regardless of underlying X(query)–Y(key,value) similarity. μ represents a median.

3. Research Question 2 (Main)

3.6. Main Message 3



So far

Orthogonal Alignment:

A phenomenon such that as X' and X getting orthogonal, then the model performance increases

Main Message 3

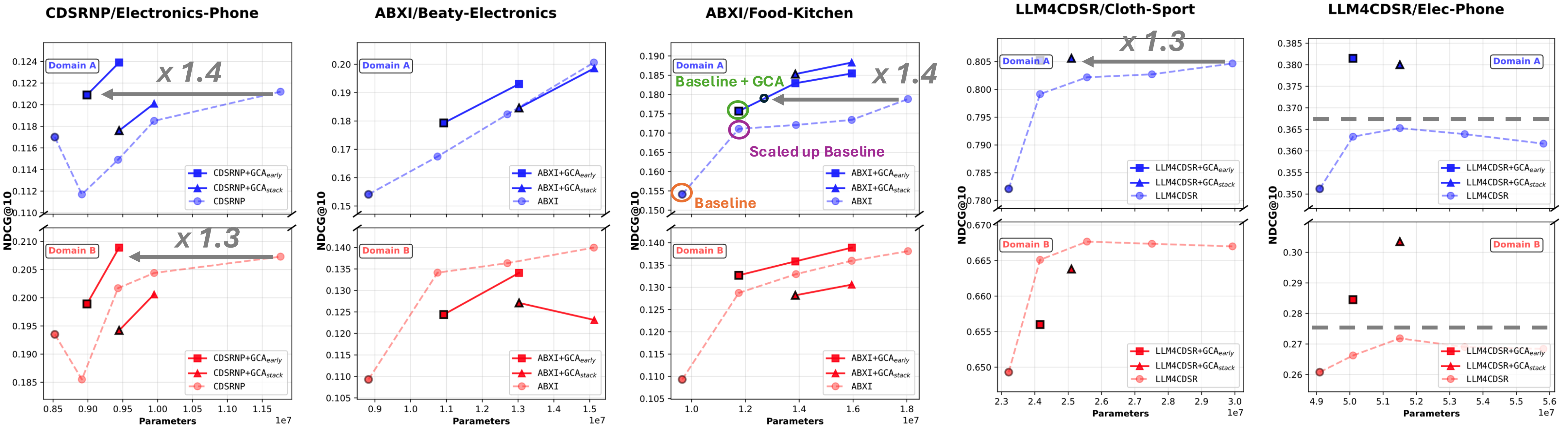
Orthogonal Alignment ***emerges naturally***, since it improves scaling law

- High level : As an perspective of model, feeding $X + \alpha X'$ where $X' \perp X$ is better quality signal than X' as denoised X .

3. Research Question 2 (Main)

3.7. Observation 4 (Supports main message 3)

- Ex: Scaled up Baseline (3M) vs. Baseline (2M) + GCA (1M)
- Orthogonal Alignment provides ~ *x1.4* parameter efficient scaling up.



(a) CDSRNP **(b)** ABXI (Beaty-Elec) **(c)** ABXI (Food-Kitch) **(d)** LLM4CDSR (Cloth-Sports) **(e)** LLM4CDSR (Elec-Phone)

4. How Orthogonal alignment improve Google's Product

General Message

Orthogonal Alignment improves scaling law in multi-modal model.

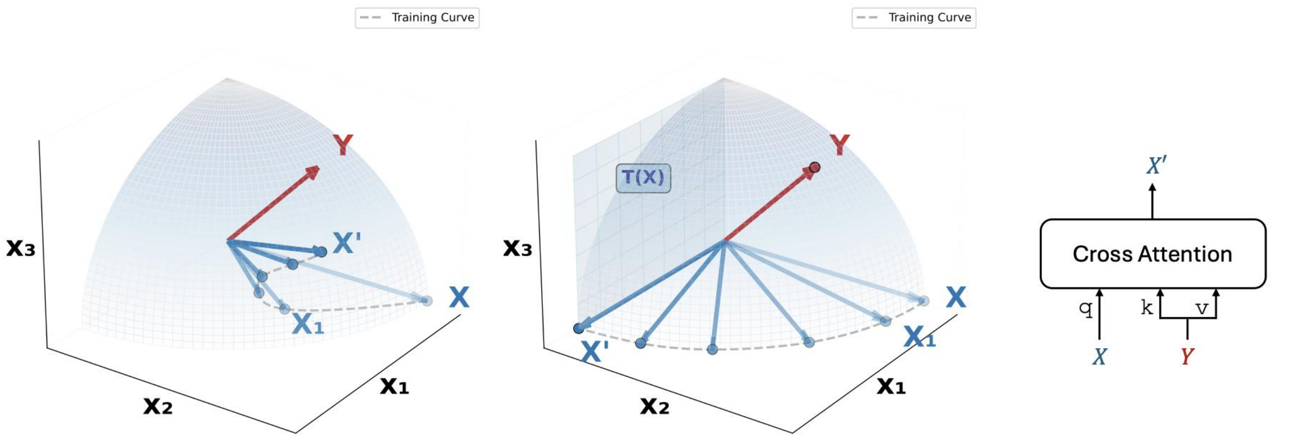
- **Recommendation algorithm**
  - Orthogonal alignment is providing an irrelevant information from input X but may fall in true user preference. This may partially solve closed-loop recommendation
- **Gemini model**
  - Orthogonal Alignment can also improve scaling law of vison-language model.
  - + may also improve with my RL experience on post-training!

Summary

Motivation: In recommendation models, learning a universal user preference from different modality user behavior data due to some sparse interaction data.

Research Question: Cross-attention is widely used mechanism to fuse different modality data, but it’s inner mechanism is poorly understood.

Main Message:
1. Orthogonal Alignment: If input (X) and output (X’) of cross-attention is getting orthogonal, then performance increases
2. Orthogonal alignment naturally happens since it improves the scaling law.



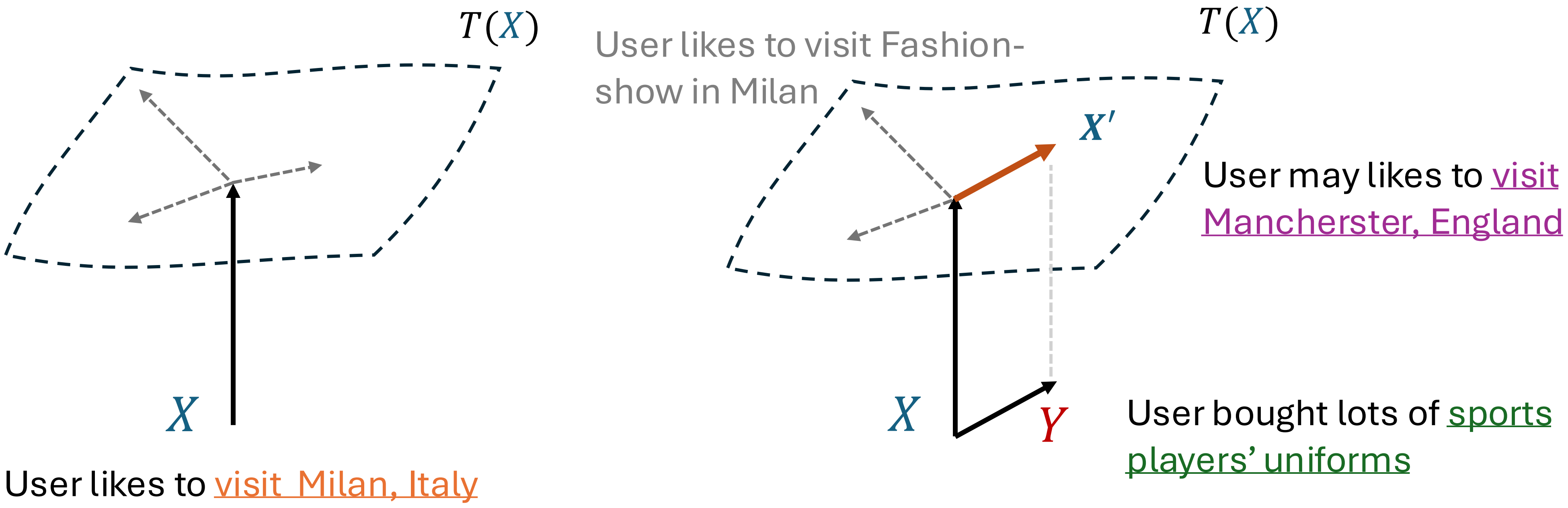
(a) Residual alignment **(b)** Orthogonal alignment **(c)** Cross-attention

[P1] Cross-Attention Secretly Performs Orthogonal Alignment in Recommendation Models.

Appendix

[1] What is role of Y?

- **Y** functions as a guide that identifies which direction on $T(X)$ correspond to positive transfer signal. Intuitively, **Y** acts as a positive, negative transfer classifier.



Appendix

[2] What is NCDG@10, AUC@10?

$$NDCG@10 = \frac{1}{\log_2(r+1)}, \qquad AUC@10 = \frac{10-r}{10}$$

- For given final user representation $h \in R^n$, compute the cosine similarity between $e_i^A \in R^n, i \in [|A|]$ and $e_i^B \in R^n, i \in [|B|]$, then model outputs its softmax – the probability of choosing next item.
- Then suppose after the sorting, we have the following outputs:

A

Item 4	Item 8	Item 3	Item 1	Item 6	Item 7	Item 5	Item 2	Item 10	Item 9
0.2	0.18	0.16	0.14	0.12	0.08	0.06	0.04	0.02	0.0

B

Item 9	Item 6	Item 10	Item 8	Item 2	Item 5	Item 7	Item 4	Item 1	Item 4
0.2	0.18	0.16	0.14	0.12	0.08	0.06	0.04	0.02	0.0

$$NDCG@10_A = \frac{1}{\log_2(6+1)}, \qquad AUC@10_A = \frac{10-6}{10},$$

$$NDCG@10_B = \frac{1}{\log_2(3+1)}, \qquad AUC@10_B = \frac{10-3}{10}$$

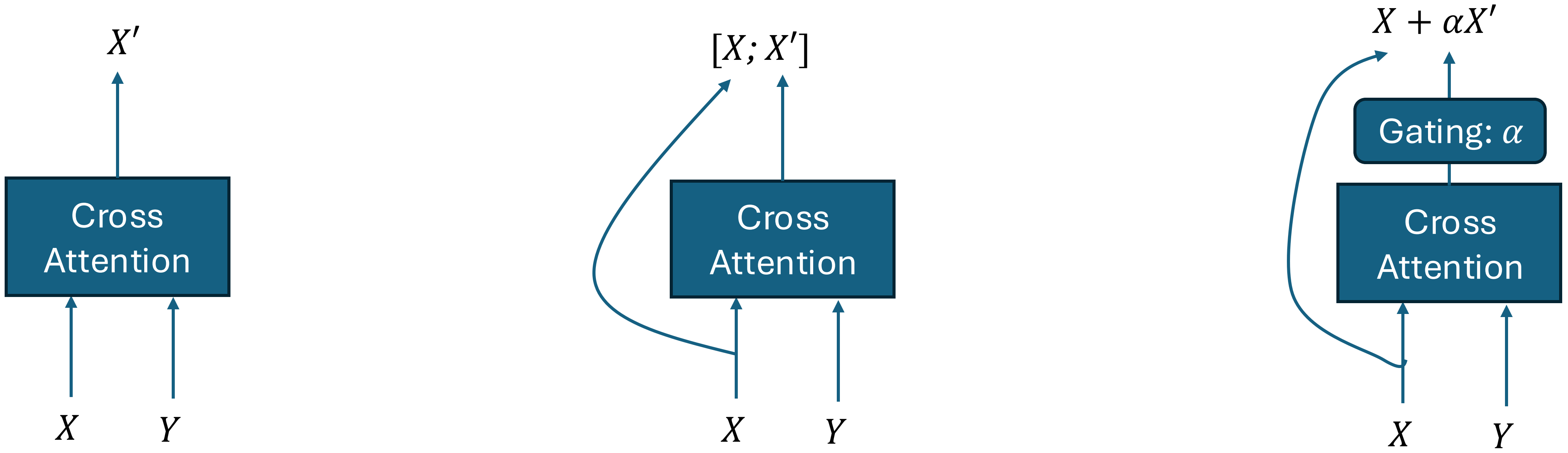
Appendix

[3] Why did you use gated cross attention?

(1) Replacement : X to X'

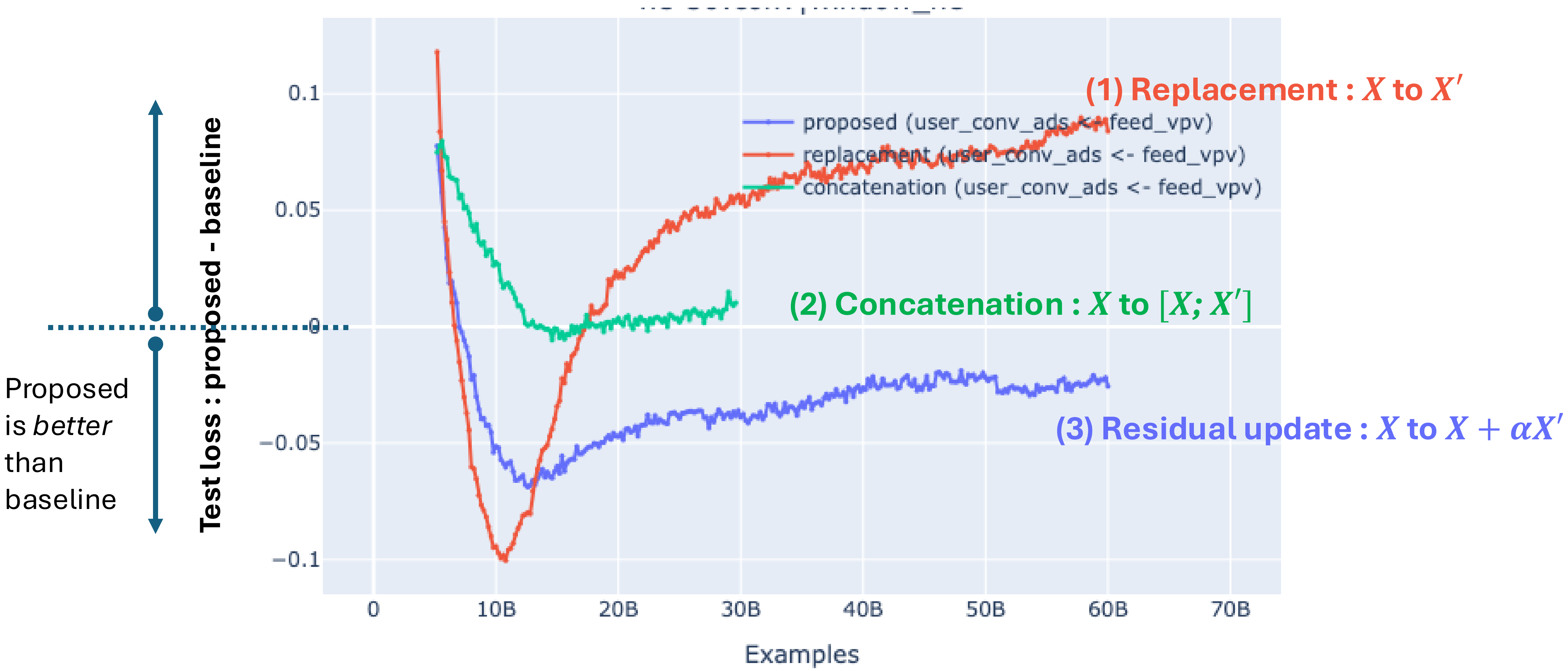
(2) Concatenation : X to $[X; X']$

(3) Gated Residual update : X to $X + \alpha X'$



Appendix

[3] Why did you use gated cross attention?



Appendix

Further research question

- Is cross-attention is best (parameter-efficient) mechanism to induce orthogonal alignment?
- Can Transformer model with large feedforward can benefit from this?
- Is orthogonal alignment related with better back-prop gradient flow?