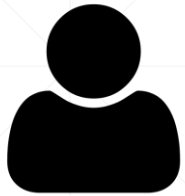
BERT4Rec : Sequential Recommendation with Bidirectional Encoder Representations from Transformer

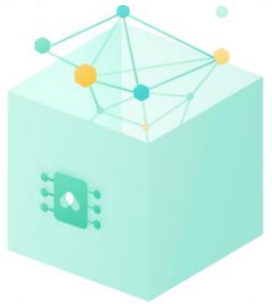
Advisor: Jia-Ling Koh
Presenter: You-Xiang Chen
Source: CIKM’19
Data: 2020/04/20

INTRODUCTION

Introduction

Sequential Recommendation





Recommender system

target item

historical subsequence

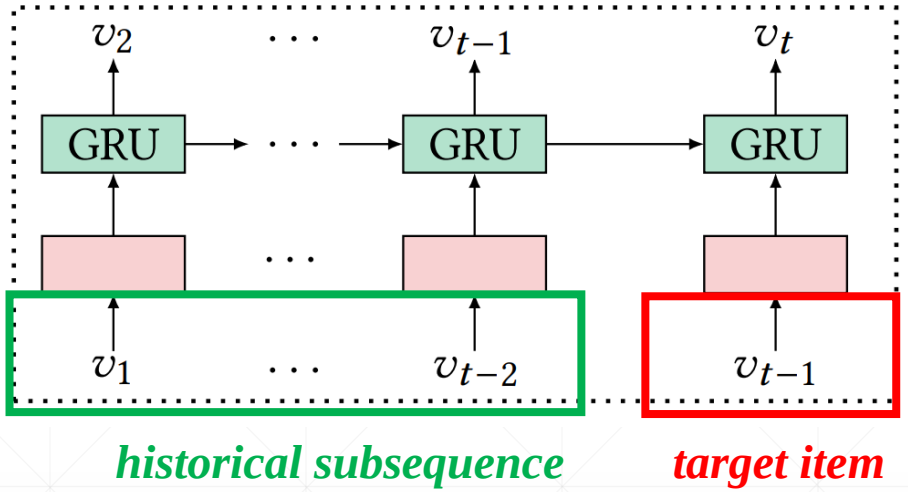
Motivation & Goal

- Unidirectional models often assume a rigidly ordered sequence over data which is not always true for user behaviors in real-world applications.

Proposing bidirectional self-attention network - BERT4Rec

Motivation & Goal

- Conventional bidirectional models encode each **historical subsequence** to predict the **target item**.
- This approach is **very time** and **resources consuming** since we need to create a new sample for each position in the sequence and predict them separately.



Introducing the Cloze task to produce more samples to train a more powerful model.

METHOD

Problem Statement

Sets of user & item

$$\mathcal{U}=\{u_1, u_2, \ldots, u_{|\mathcal{U}|}\}$$

$$\mathcal{V}=\{v_1, v_2, \ldots, v_{|\mathcal{V}|}\}$$

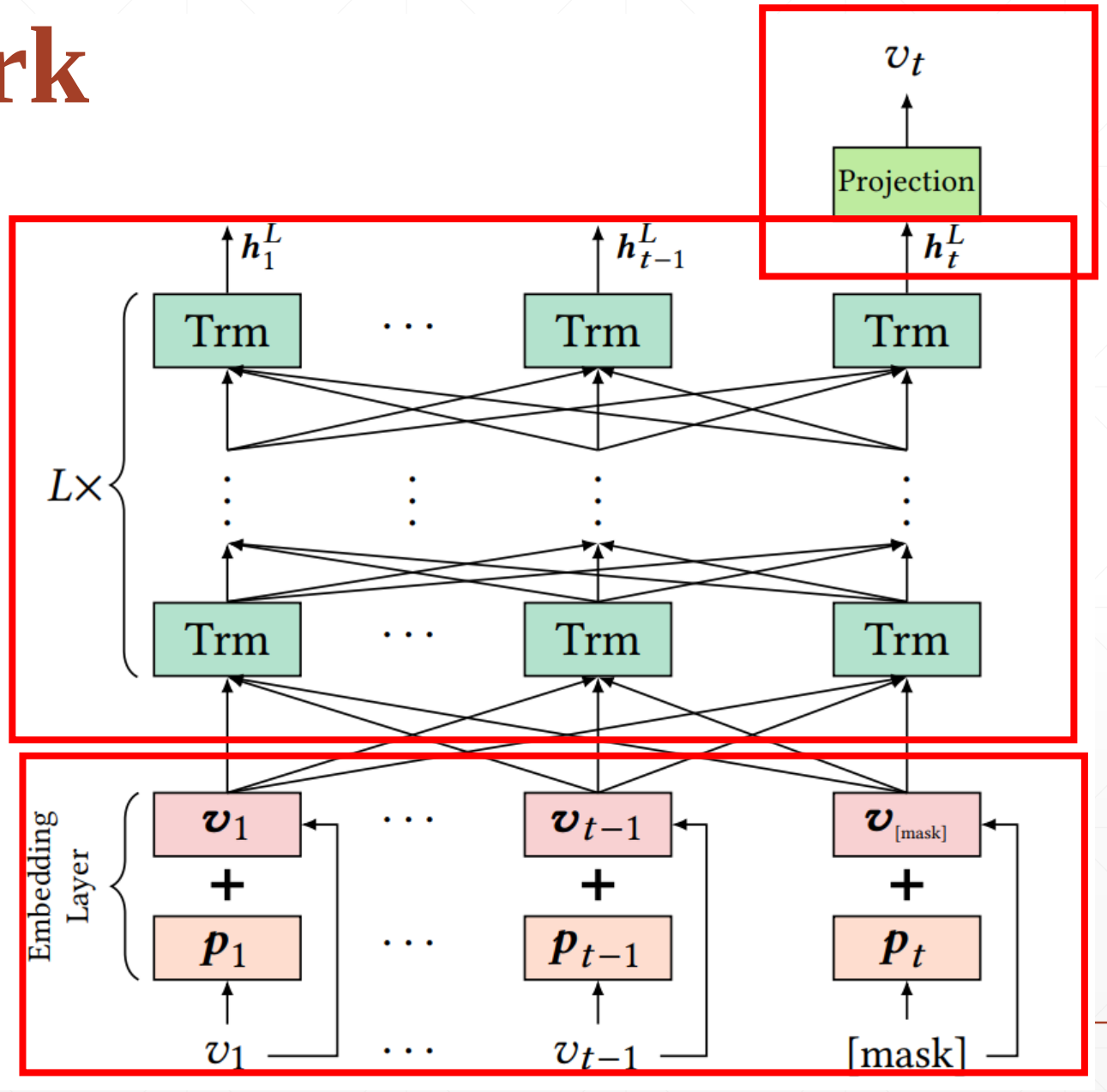
Interaction sequence

$$\mathcal{S}_u=[v_1^{(u)}, \ldots, v_t^{(u)}, \ldots, v_{n_u}^{(u)}]$$

Output

$$p(v_{n_u+1}^{(u)} = v | \mathcal{S}_u)$$

Framework


v_t
Projection
h_1^L
h_{t-1}^L
h_t^L
Trm
Trm
Trm
$L\times$
Trm
Trm
Trm
Embedding Layer
v_1
v_{t-1}
$v_{[mask]}$
+
+
+
p_1
p_{t-1}
p_t
v_1
v_{t-1}
[mask]

Embedding Layer

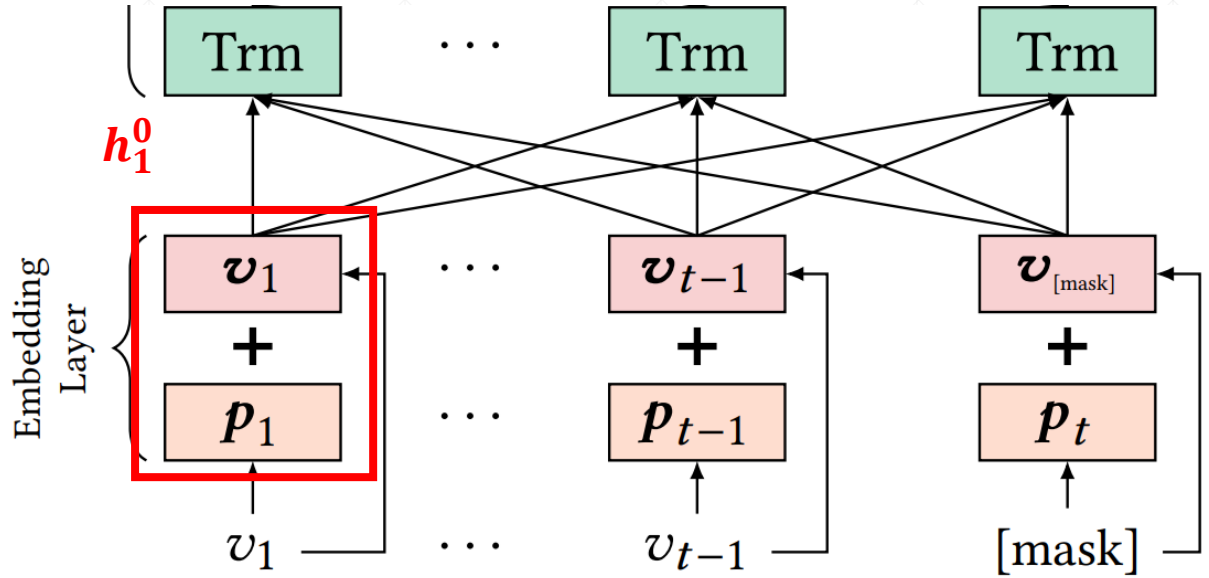
Input representation

$$h_i^0 = v_i + p_i$$

$v_i \in E$ — item embedding matrix

$p_i \in P$ — position embedding matrix

d-dim.



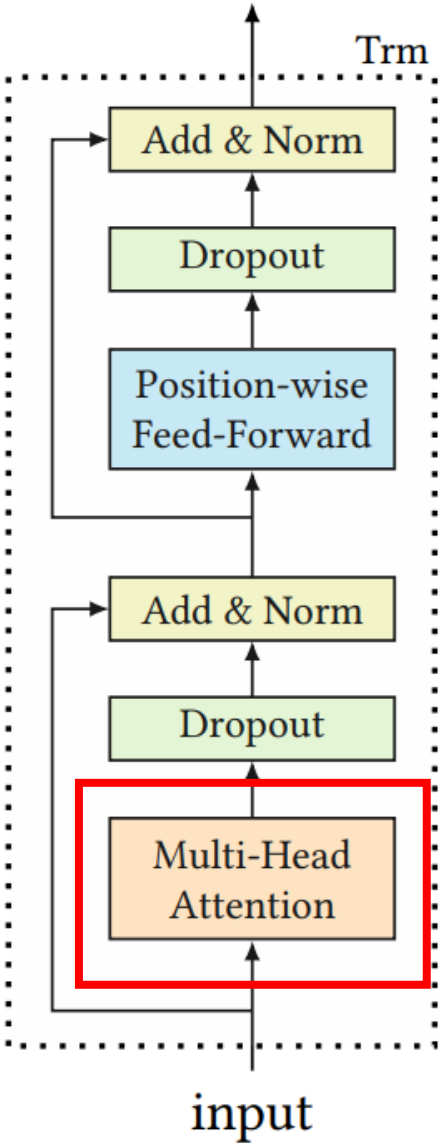
Transformer Layer

Multi-Head Self-Attention

$$H^l = [h_1^l, h_2^l, \ldots, h_t^l]$$

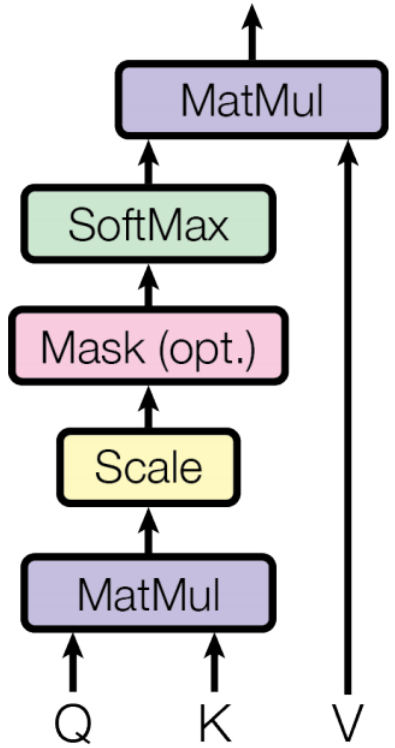
$$\mathrm{MH}(H^l) = [\mathrm{head}_1; \mathrm{head}_2; \ldots; \mathrm{head}_n] W^O$$

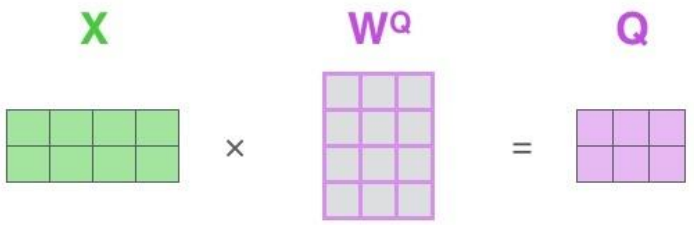
projects H^l into n subspaces

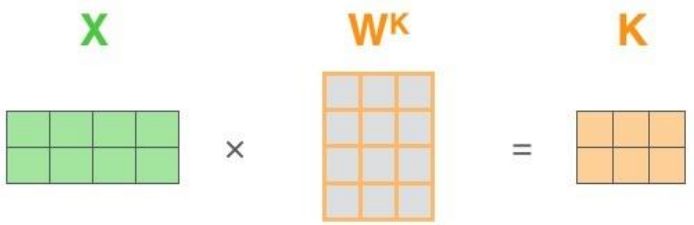


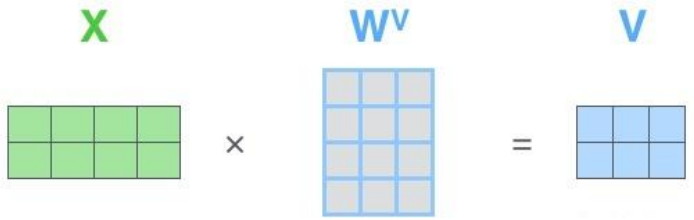
Transformer

Scaled Dot-Product Attention



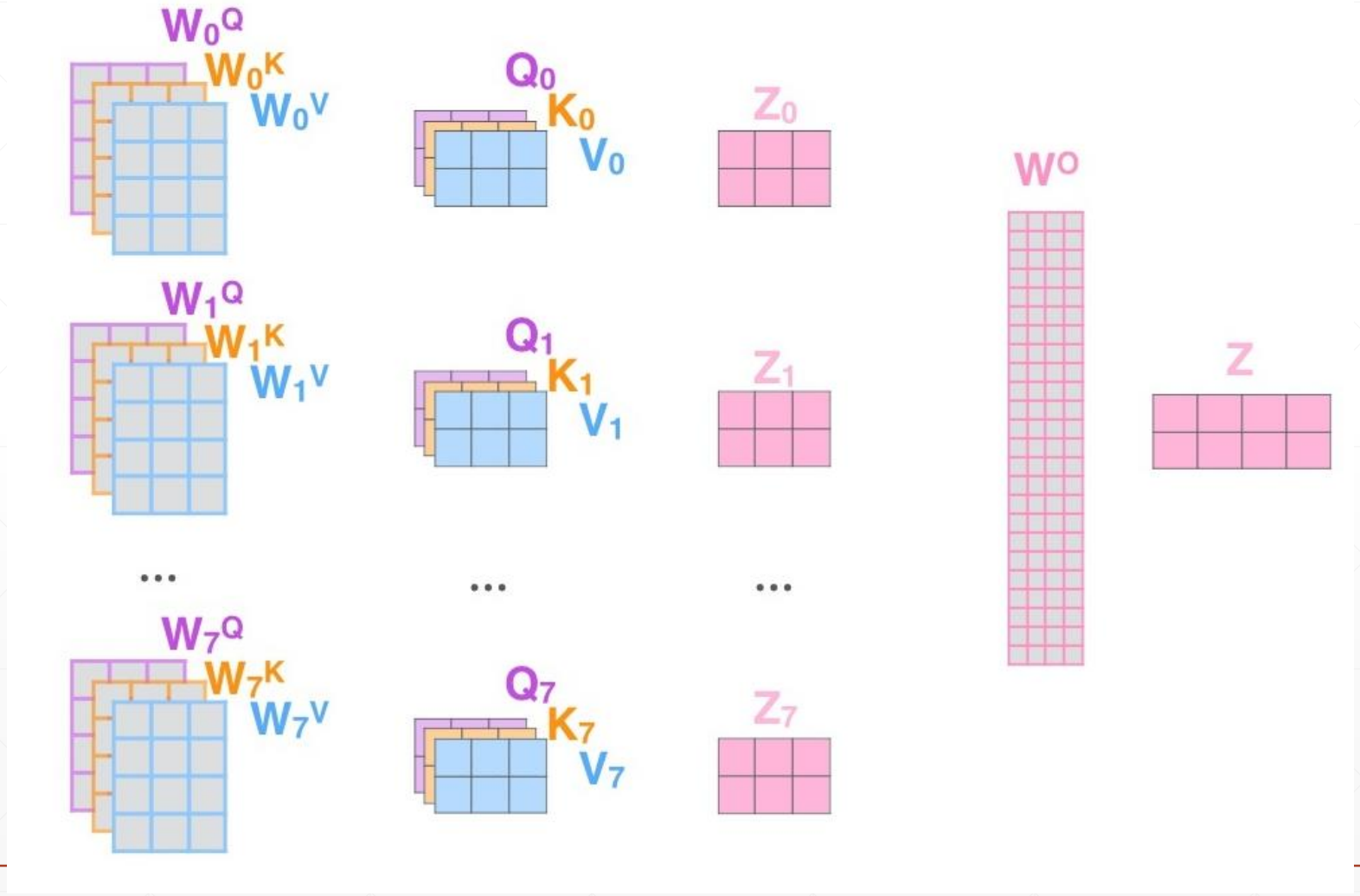






$$\text{softmax}\left(\frac{Q \times K^T}{\sqrt{d_k}}\right) V = Z$$

Multi-Head Attention



Transformer Layer

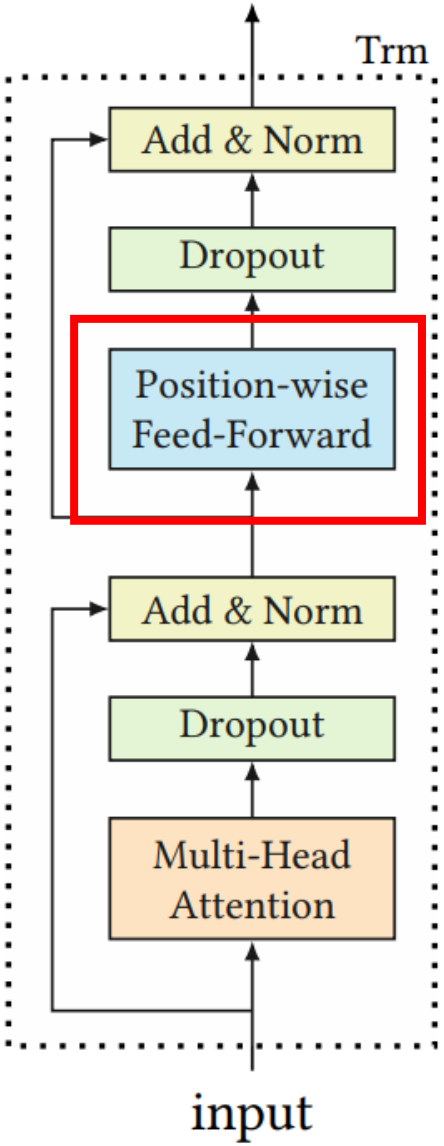
Position-wise Feed-Forward Network

$$\mathrm{FFN}(\boldsymbol{x}) = \mathrm{GELU}\left(\boldsymbol{x}\boldsymbol{W}^{(1)} + \boldsymbol{b}^{(1)}\right)\boldsymbol{W}^{(2)} + \boldsymbol{b}^{(2)}$$

Gaussian Error Linear Unit (GELU) activation function

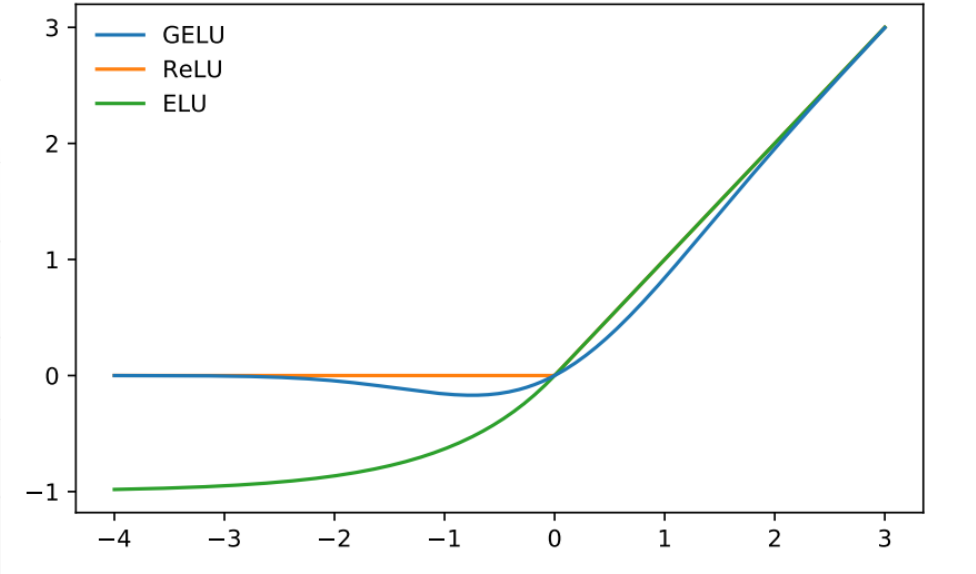
$$\mathrm{PFFN}(\boldsymbol{H}^{l}) = \left[\mathrm{FFN}(\boldsymbol{h}_1^{l})^{\top}; \ldots; \mathrm{FFN}(\boldsymbol{h}_t^{l})^{\top}\right]^{\top}$$

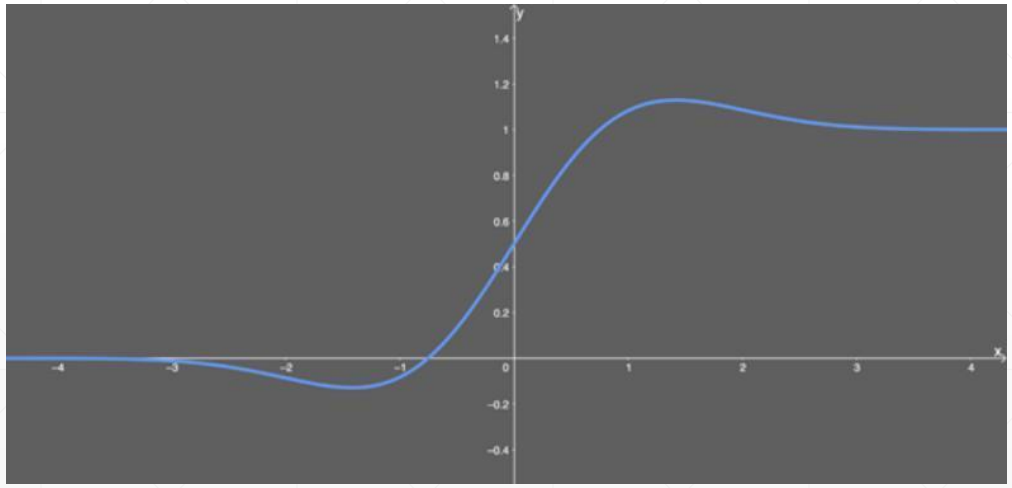
separately and identically at each position



Gaussian Error Linear Units

$$\text{GELU}(x) = xP(X \leq x) = x\Phi(x).$$
$$= 0.5x(1 + \tanh[\sqrt{2/\pi}(x + 0.044715x^3)])$$





https://arxiv.org/pdf/1606.08415.pdf

Transformer Layer

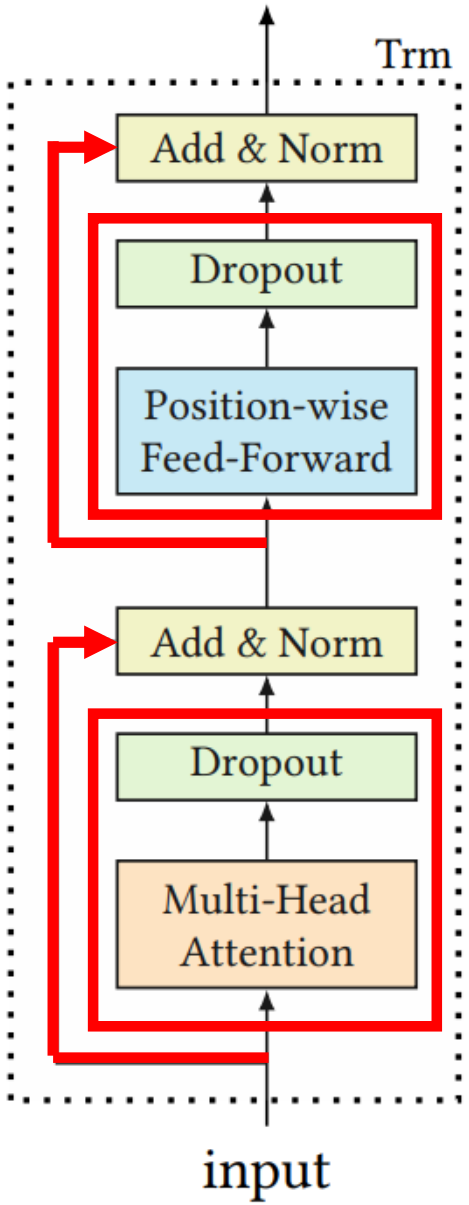
Stacking Transformer Layer

$$A^{l-1} = \mathrm{LN}\left(H^{l-1} + \mathrm{Dropout}\left(\mathrm{MH}(H^{l-1})\right)\right)$$

$$\mathrm{Trm}(H^{l-1}) = \mathrm{LN}\left(A^{l-1} + \mathrm{Dropout}\left(\mathrm{PFFN}(A^{l-1})\right)\right)$$

LN(·) : layer normalization fun

$$H^{l} = \mathrm{Trm}(H^{l-1}), \quad \forall i \in [1, \ldots, L]$$



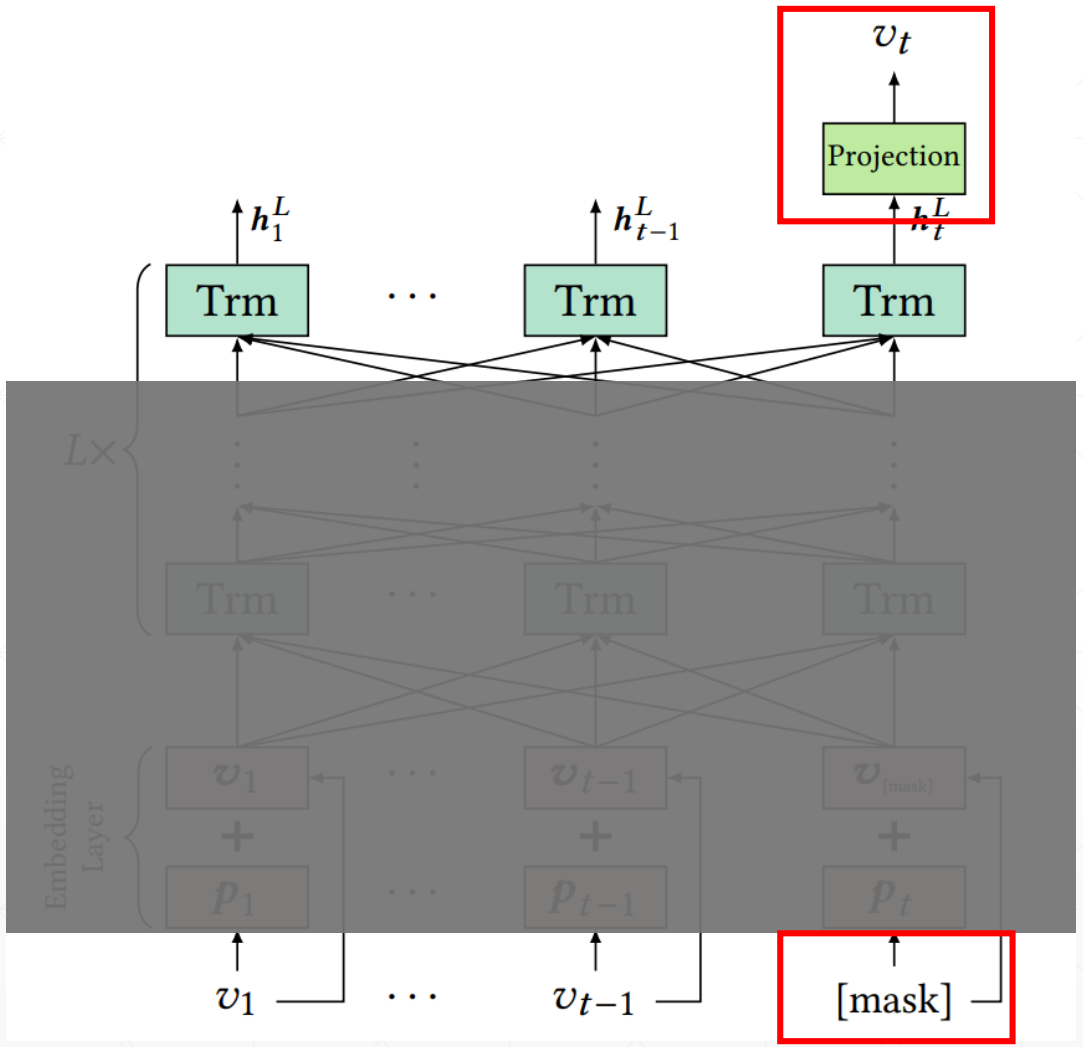
https://arxiv.org/pdf/1607.06450.pdf

Output Layer

$$P(v) = \text{softmax}\left(\text{GELU}(\boldsymbol{h}_t^L \boldsymbol{W}^P + \boldsymbol{b}^P)\boldsymbol{E}^\top + \boldsymbol{b}^O\right)$$

$\boldsymbol{W^P}$: ***Leanable projection matrix***

$\boldsymbol{E}$: ***Embedding Matrix of items***



Model Learning

Input: $[v_1, v_2, v_3, v_4, v_5] \xrightarrow{\text{randomly mask}} [v_1, [\text{mask}]_1, v_3, [\text{mask}]_2, v_5]$

Labels: $[\text{mask}]_1 = v_2, \quad [\text{mask}]_2 = v_4$

$$\mathcal{L} = \frac{1}{|S_u^m|} \sum_{v_m \in S_u^m} -\log P(v_m = v_m^* | S_u')$$

the masked items

masked version for user behavior

EXPERIMENT

Datasets

Datasets	#users	#items	#actions	Avg. length
Beauty	40,226	54,542	0.35m	8.8
Steam	281,428	13,044	3.5m	12.4
ML-1m	6040	3416	1.0m	163.5
ML-20m	138,493	26,744	20m	144.4

Baselines

- POP
- BPR-MF
- NCF

$non - sequential$

- FPMC markov chain
- GRU4Rec +
- Caser
- SASRec

$sequential$

Evaluation metrics

Hit Ratio

$$HR@K = \frac{Number\ of\ Hits\ @\ K}{|GT|}$$

Normalized Discounted cumulative gain

$$DCG_k = \sum_{i=1}^{k} \frac{2^{rel_i} - 1}{\log_2(i+1)}$$

$$NDCG@K = \frac{DCG@K}{IDCG}$$

Mean Reciprocal Rank

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i}$$

Query	Top 3 Returns	Correct response	Rank	Reciprocal rank
cat	catten, cati, **cats**	cats	3	1/3
torus	torii, **tori**, toruses	tori	2	1/2
virus	**viruses**, virii, viri	viruses	1	1

Top 3 MMR=(1/3 + 1/2 + 1)/3 = 0.61

Performance

worst *non – sequential* *sequential* *Transformer*

Datasets	Metric	POP	BPR-MF	NCF	FPMC	GRU4Rec	GRU4Rec$^+$	Caser	SASRec	BERT4Rec	Improv.
Beauty	HR@1	0.0077	0.0415	0.0407	0.0435	0.0402	0.0551	0.0475	0.0906	**0.0953**	5.19%
	HR@5	0.0392	0.1209	0.1305	0.1387	0.1315	0.1781	0.1625	0.1934	**0.2207**	14.12%
	HR@10	0.0762	0.1992	0.2142	0.2401	0.2343	0.2654	0.2590	0.2653	**0.3025**	14.02%
	NDCG@5	0.0230	0.0814	0.0855	0.0902	0.0812	0.1172	0.1050	0.1436	**0.1599**	11.35%
	NDCG@10	0.0349	0.1064	0.1124	0.1211	0.1074	0.1453	0.1360	0.1633	**0.1862**	14.02%
	MRR	0.0437	0.1006	0.1043	0.1056	0.1023	0.1299	0.1205	0.1536	**0.1701**	10.74%
Steam	HR@1	0.0159	0.0314	0.0246	0.0358	0.0574	0.0812	0.0495	0.0885	**0.0957**	8.14%
	HR@5	0.0805	0.1177	0.1203	0.1517	0.2171	0.2391	0.1766	0.2559	**0.2710**	5.90%
	HR@10	0.1389	0.1993	0.2169	0.2551	0.3313	0.3594	0.2870	0.3783	**0.4013**	6.08%
	NDCG@5	0.0477	0.0744	0.0717	0.0945	0.1370	0.1613	0.1131	0.1727	**0.1842**	6.66%
	NDCG@10	0.0665	0.1005	0.1026	0.1283	0.1802	0.2053	0.1484	0.2147	**0.2261**	5.31%
	MRR	0.0669	0.0942	0.0932	0.1139	0.1420	0.1757	0.1305	0.1874	**0.1949**	4.00%
ML-1m	HR@1	0.0141	0.0914	0.0397	0.1386	0.1583	0.2092	0.2194	0.2351	**0.2863**	21.78%
	HR@5	0.0715	0.2866	0.1932	0.4297	0.4673	0.5103	0.5353	0.5434	**0.5876**	8.13%
	HR@10	0.1358	0.4301	0.3477	0.5946	0.6207	0.6351	0.6692	0.6629	**0.6970**	4.15%
	NDCG@5	0.0416	0.1903	0.1146	0.2885	0.3196	0.3705	0.3832	0.3980	**0.4454**	11.91%
	NDCG@10	0.0621	0.2365	0.1640	0.3439	0.3627	0.4064	0.4268	0.4368	**0.4818**	10.32%
	MRR	0.0627	0.2009	0.1358	0.2891	0.3041	0.3462	0.3648	0.3790	**0.4254**	12.24%
ML-20m	HR@1	0.0221	0.0553	0.0231	0.1079	0.1459	0.2021	0.1232	0.2544	**0.3440**	35.22%
	HR@5	0.0805	0.2128	0.1358	0.3601	0.4657	0.5118	0.3804	0.5727	**0.6323**	10.41%
	HR@10	0.1378	0.3538	0.2922	0.5201	0.5844	0.6524	0.5427	0.7136	**0.7473**	4.72%
	NDCG@5	0.0511	0.1332	0.0771	0.2239	0.3090	0.3630	0.2538	0.4208	**0.4967**	18.04%
	NDCG@10	0.0695	0.1786	0.1271	0.2895	0.3637	0.4087	0.3062	0.4665	**0.5340**	14.47%
	MRR	0.0709	0.1503	0.1072	0.2273	0.2967	0.3476	0.2529	0.4026	**0.4785**	18.85%

Analysis on Bidirection and Cloze

Model	Beauty			ML-1m		
	HR@10	NDCG@10	MRR	HR@10	NDCG@10	MRR
SASRec	0.2653	0.1633	0.1536	0.6629	0.4368	0.3790
BERT4Rec (1 mask)	0.2940	0.1769	0.1618	0.6869	0.4696	0.4127
BERT4Rec	0.3025	0.1862	0.1701	0.6970	0.4818	0.4254

CONCLUSION

- We introduce a deep bidirectional sequential model called **BERT4Rec** for sequential recommendation.
- For model training, we introduce the **Cloze task** which predicts the masked items using both left and right context.
- Extensive experimental results on four real-world datasets show that our model outperforms state-of-the-art baselines.