DAN: Deep Attention Neural Network for News Recommendation

NELIST 信息内容安全技术国家工程实验室 National Engineering Laboratory for Information Security Technologies

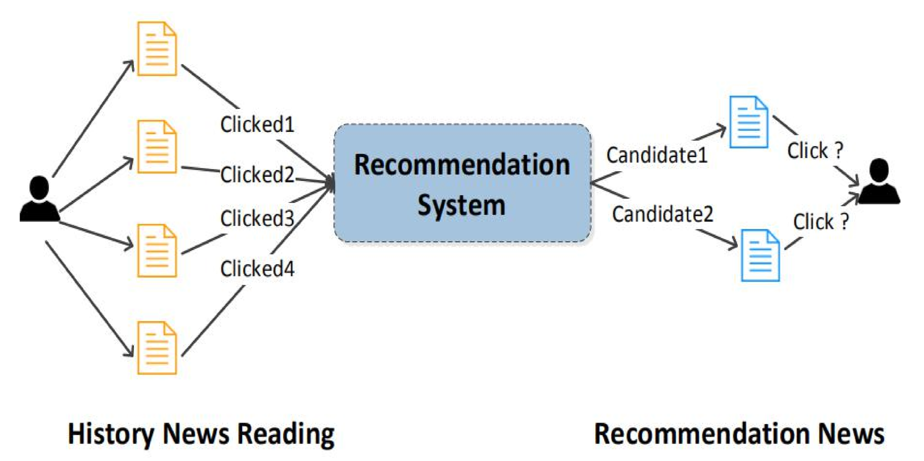
Qiannan Zhu[1,2], Xiaofei Zhou[1,2], Zengliang Song[1,2], Jianlong Tan[1,2] and Li Guo[1,2]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
{zhuqiannan, zhouxiaofei}@iie.ac.cn

Introduction

◆ **Task**

Using the given users' click history to predict whether user will click a candidate news that he has not seen before.



◆ **Existing problems**

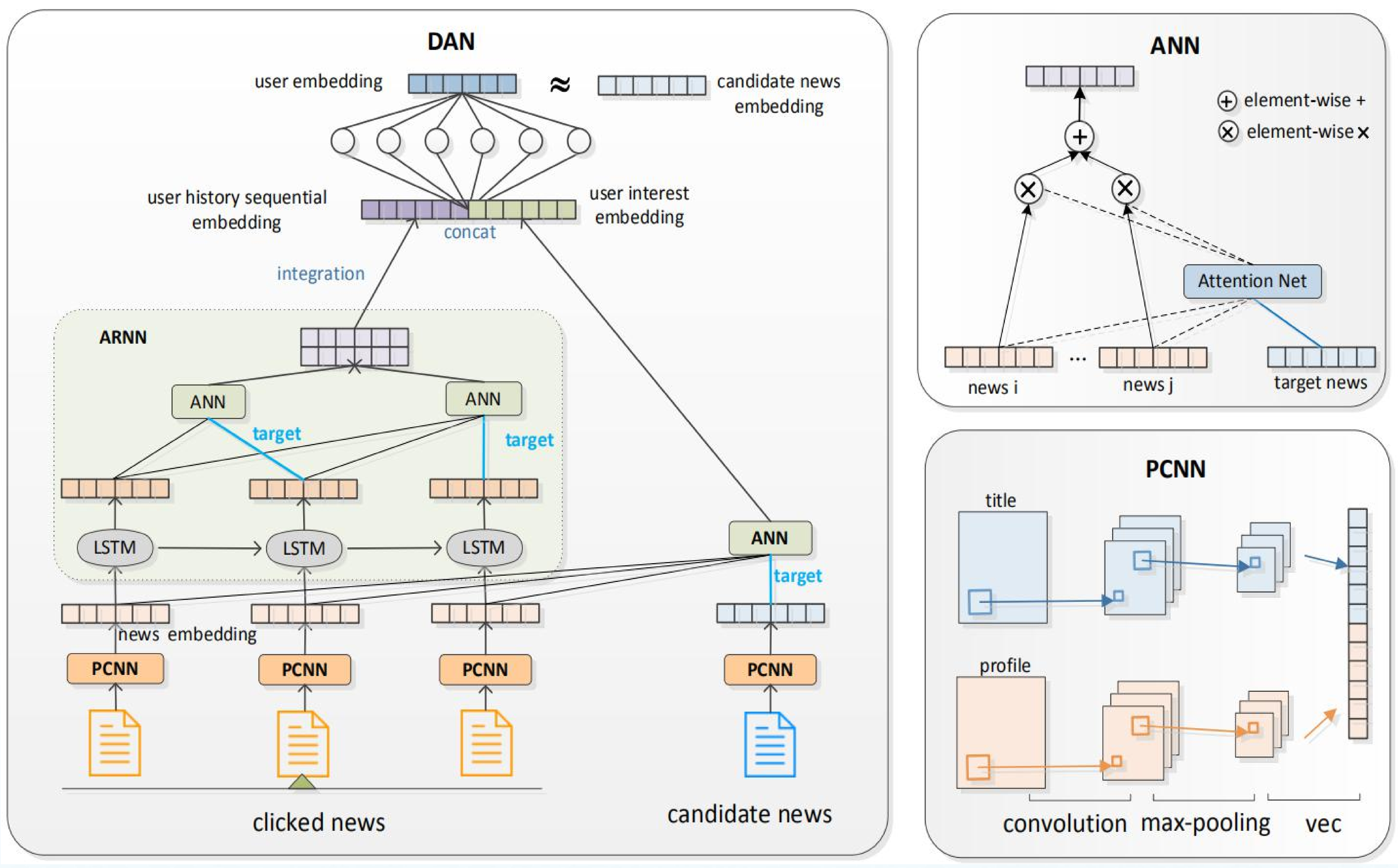
(1) Fail with the dynamic diversity of news and user's interests

(2) Ignore the importance of sequential information of user's clicking selection

◆ **DAN**

Taking full advantages of CNN, RNN and attention mechanism, we propose a deep attention neural network DAN for personalized news recommendation. Three components PCNN, ARNN and ANN are included in DAN.

Architecture



◆ **PCNN** is composed of two parallel CNNs, which respectively take the title and profile of news as inputs and learn the title-level and profile-level representation.

◆ **ANN** is an attention-based neural network, which aggregates the user's current interest by matching clicked news with different weights about candidate news.

◆ **ARNN** is an attention-based RNN, which adds the attention mechanism on each state of RNN for getting richer sequential feature at different clicking time.

Conclusion

(1) We propose a deep attention neural network DAN for news recommendation. Three components are devised PCNN, ARNN and ANN are devised.

(2) DAN considers the user's history sequential information and user's current interest together to determine whether the user clicks on the candidate news.

(3) We conduct extensive experiments on a real world data set Adressa. The experimental results demonstrate the significant superiority of our DAN model.

Components

◆ **PCNN**

profile embedding $\mathrm{C} = [\mathrm{e}_1, f(\mathrm{g}_1), \mathrm{e}_2, f(\mathrm{g}_2) ... \mathrm{e}_m, f(\mathrm{g}_m)]$

title embedding $\mathbf{T} = [\mathbf{w}_1, \mathbf{w}_2, ... \mathbf{w}_n]$

(1) mapping: $f(\mathbf{g}) = \sigma(\mathbf{M}_t\, \mathbf{g} + \mathbf{b})$

(2) feature map: $\mathbf{m} = f(\mathbf{Z} \odot \mathbf{c} + b)$

(3) local pooling feature: $\mathbf{p} = maxpooling(\mathbf{m})$

(4) vector representation of input emd:

$$r(\mathbf{Z}) = [vec(\mathbf{p}^1); vec(\mathbf{p}^2); ... vec(\mathbf{p}^v)]$$

(5) news representation: $\mathbf{I} = [r(\mathbf{T}); r(\mathbf{C})]$

◆ **ANN**

user's clicked history news $\{\mathbf{I}_1, \mathbf{I}_2, ..., \mathbf{I}_{t-1}\}$ and the candidate(target) news $\mathbf{I}_t$

$$\mathbf{u}_j = tanh(\mathbf{W}_w \mathbf{I}_j + \mathbf{b}_w)$$
$$\mathbf{u}_t = tanh(\mathbf{W}_t \mathbf{I}_t + \mathbf{b}_t)$$
$$\alpha_{j,t} = \frac{exp(\mathbf{v}^T(\mathbf{u}_t + \mathbf{u}_j))}{\sum_j exp(\mathbf{v}^T(\mathbf{u}_t + \mathbf{u}_j))}$$
$$\mathbf{s}_t = \sum_j \alpha_{j,t} \mathbf{I}_j$$

◆ **ARNN**

user's clicked news $\{\mathbf{I}_1, \mathbf{I}_2, ..., \mathbf{I}_{t-1}\}$

(1) state: $\mathbf{h}_j = LSTM(\mathbf{h}_{j-1}, \mathbf{I}_j)$

(2) j-step sequential feature: $\mathbf{s}_j$

(3) feature matrix: $\mathbf{S} = [\mathbf{s}_2\ \mathbf{s}_3 ... \mathbf{s}_{t-1}]$

(4) integration: $\tilde{\mathbf{h}} = f(\mathbf{S}),\ f(\mathbf{S}) = cnn(\mathbf{S})$

◆ **Similarity**

user's history information $\tilde{\mathbf{h}}$ and user's current interest $\mathbf{s}_j$

(1) final user's embedding

$$\tilde{\mathbf{I}}_t = \mathbf{M}_g[\tilde{\mathbf{h}}; \mathbf{s}_t]$$

(2) probability

$$P = cosine(\tilde{\mathbf{I}}_t \cdot \mathbf{I}_t)$$

Experiment

We conduct experiments on a real-world online news dataset **Adressa to** evaluate the performance of our model.

◆ **Dataset**

(1) String the news article sequence into a session according to time-stamp

(2) Split longer news sequence into shorter sequence

Number	*Adressa-1week*	*Adressa-10week*
#users	640,503	3,614,911
#news	20,428	81,018
#events	3,101,991	35,244,078
#entity	160,559	417,572
#entity-type	19	19
#average words per title	6.57	6.64
#average entities per news	27.7	26.5
#average entity-types per news	12.6	12.5

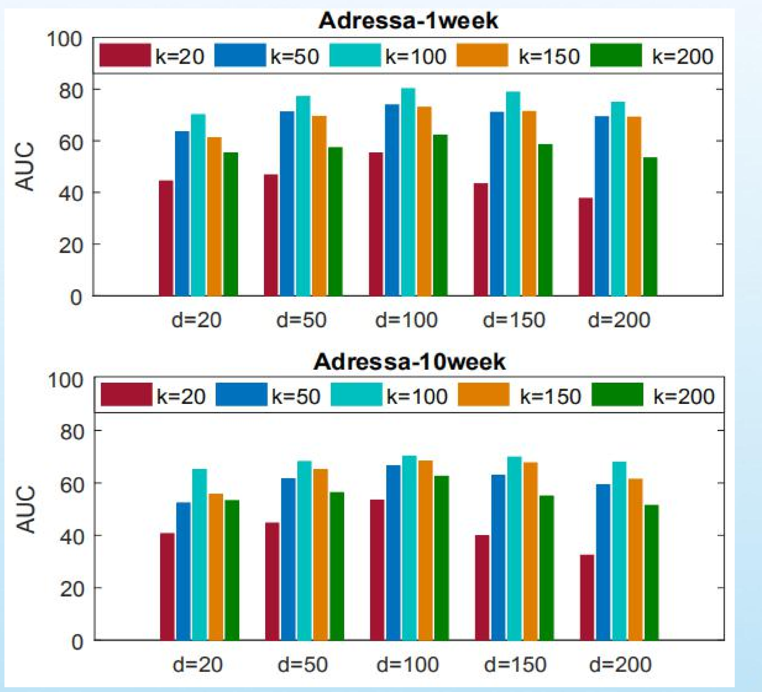
◆ **Results on Adressa**

Model	Adressa-1week		Adressa-10week	
	F1	AUC	F1	AUC
LibFM(-)	63.93	61.79	55.75	53.83
LibFM	70.69	69.53	64.44	61.41
DSSM(-)	69.36	68.25	62.24	60.74
DSSM	74.78	72.71	69.11	67.57
DeepWide(-)	67.39	64.83	59.98	57.98
DeepWide	73.94	71.07	67.87	66.80
DeepFM(-)	66.09	64.83	58.47	57.03
DeepFM	72.47	70.33	64.71	63.60
DMF	63.43	61.49	55.43	53.47
DKN(-)	75.38	73.45	68.57	60.57
DKN	79.97	77.24	70.39	67.53
DAN	**82.32**	**80.18**	**73.58**	**70.17**

◆ **Discussion on Different DAN Variants**

Model	Adressa-1week		Adressa-10week	
	F1	AUC	F1	AUC
DAN without entity and entity-type	76.01	72.51	69.37	58.19
DAN with entity-type	77.21	74.49	70.76	65.42
DAN with entity	80.46	78.64	71.93	67.63
DAN with entity and entity-type	**82.32**	**80.18**	**73.58**	**70.17**
DAN without mapping	78.06	75.36	67.13	65.49
DAN with linear mapping	80.17	77.24	70.57	67.81
DAN with no-linear mapping	**82.32**	**80.18**	**73.58**	**70.17**
DAN without attention	78.13	75.44	70.23	68.76
DAN with attention	**82.32**	**80.18**	**73.58**	**70.17**
DAN with mul	72.46	68.54	60.39	58.73
DAN with sum	80.17	78.29	69.98	67.04
DAN with vec	79.91	77.74	69.46	66.29
DAN with cnn	**82.32**	**80.18**	**73.58**	**70.17**
DAN without ARNN	81.59	77.27	71.25	69.61
DAN with ARNN	**82.32**	**80.18**	**73.58**	**70.17**

◆ **Different Dimension of Embedding**



◆ **Analysis**

(1) DAN consistently outperforms all baselines on both datasets.

(2) Incorporating entity-types, attention mechanism, mapping function, integration function into our model excellently improves performance.

(3) DAN achieves best performance at d=k=100 setting.

The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)
January 27- February 1 - 2019, Honolulu, Hawaii,USA

中国科学院信息工程研究所 INSTITUTE OF INFORMATION ENGINEERING,CAS