SIGIR 5 Papers

09/19/2018

Xiaosong Jia

Sentiment Analysis of Peer Review Texts for Scholarly Papers

Ke Wang and Xiaojun Wan
Institute of Computer Science and Technology, Peking University
The MOE Key Laboratory of Computational Linguistics, Peking University
Beijing, China
wangke17@pku.edu.cn,wanxiaojun@pku.edu.cn

Author

Xiaojun Wan

Peking University

Verified email at pku.edu.cn - Homepage

Natural Language Processing   Text Mining   Artificial Intelligence

Cited by   VIEW ALL

	All	Since 2013
Citations	3764	2702
h-index	27	25
i10-index	68	57

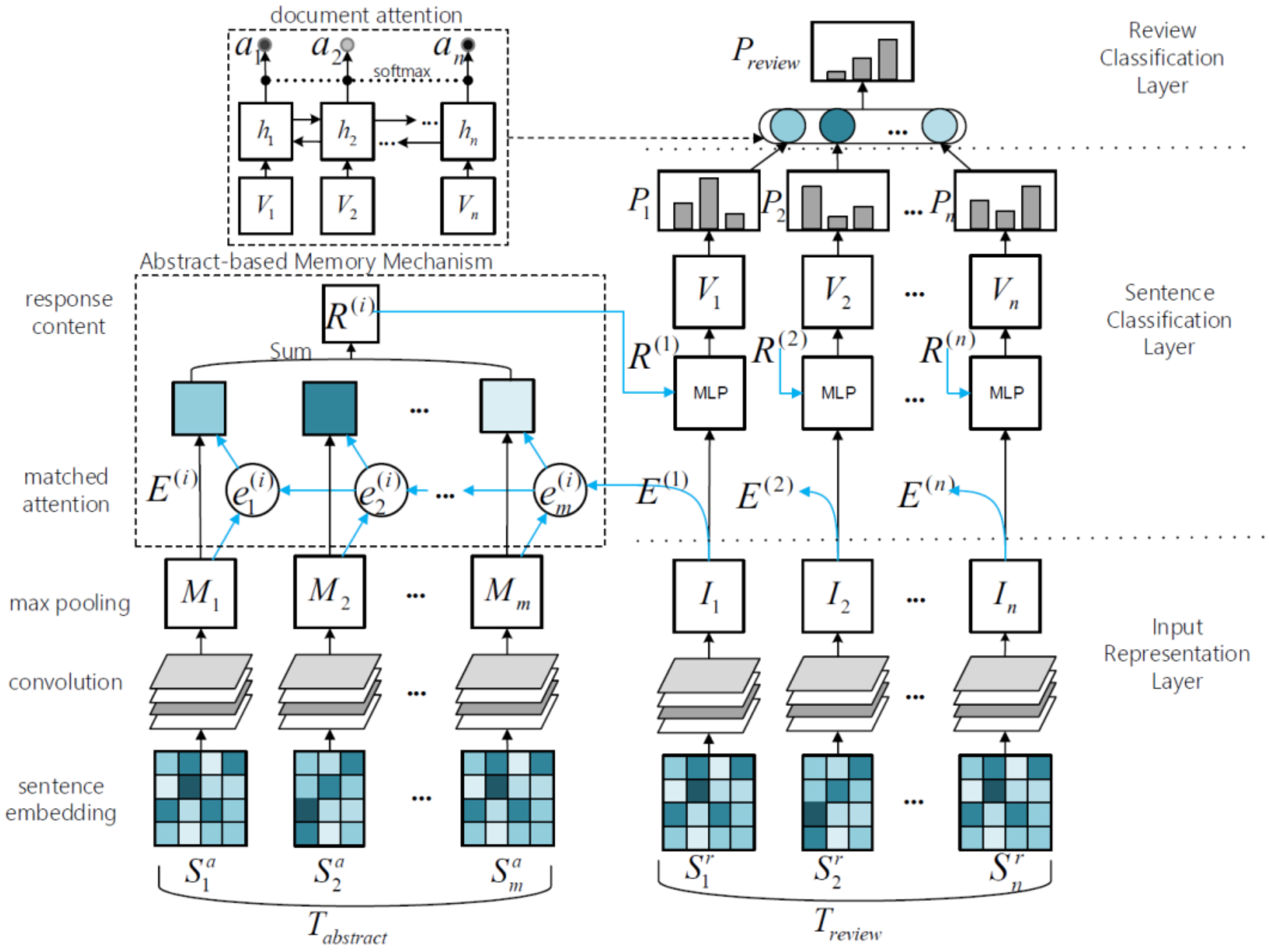
TITLE	CITED BY	YEAR
Co-training for cross-lingual sentiment classification X Wan Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL ...	405	2009
Multi-document summarization using cluster-based link analysis X Wan, J Yang Proceedings of the 31st annual international ACM SIGIR conference on ...	275	2008
Manifold-Ranking Based Topic-Focused Multi-Document Summarization. X Wan, J Yang, J Xiao IJCAI 7, 2903-2908	223	2007
Single Document Keyphrase Extraction Using Neighborhood Knowledge. X Wan, J Xiao AAAI 8, 855-860	219	2008

Motivation

- Whether the review texts and the recommendation scores are consistent with each other or not
- Help the chair to write a comprehensive meta-review
- Be convenient for authors to further improve their paper

- There is very few researches
- Long
- Mixture of non-opinionated and opinionated texts.
- Mixture of pros and cons.

Model



- Memory

$$e_t' = LSTM(\hat{h}_{t-1}, \boldsymbol{M}_t), (\hat{h}_0 = \boldsymbol{I}_i, t = 1, ..., m)$$

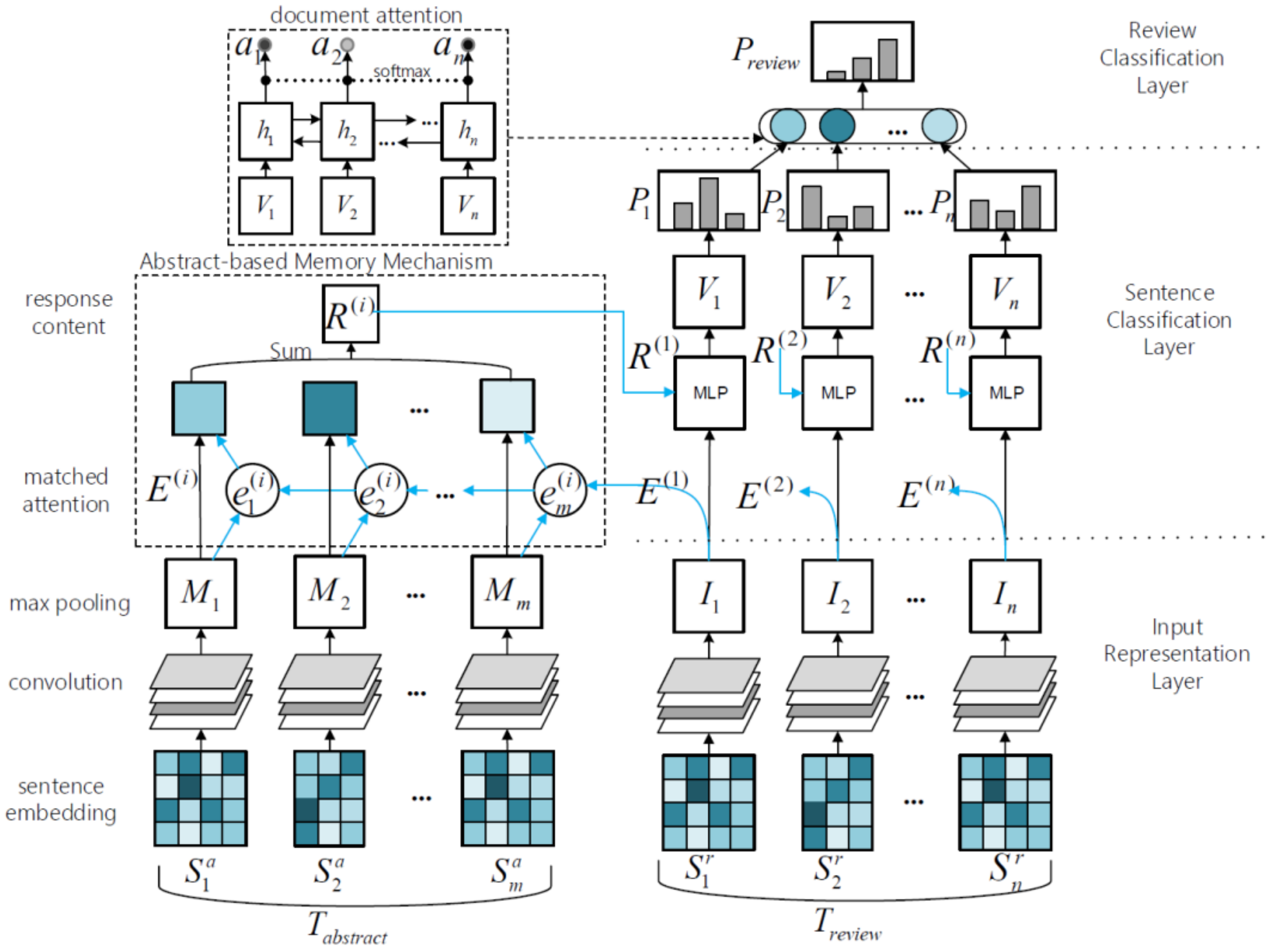
$$e_t^{(i)} = \frac{exp(e_t')}{\sum_j exp(e_j')}$$

$$\boldsymbol{E}^{(i)} = [e_t^{(i)}]_{t=1}^{m}$$

$$\boldsymbol{R}^{(i)} = \sum_{t=1}^{m} e_t^{(i)} \boldsymbol{M}_t$$

$$\boldsymbol{V}_i = f_{mlp}(\boldsymbol{I}_i || \boldsymbol{R}^{(i)}; \theta_{mlp}).$$

Model



- Review classifier

$$\boldsymbol{P_i} = softmax(\boldsymbol{W_p} \cdot \boldsymbol{V_i} + \boldsymbol{b_p}),$$

$$\boldsymbol{P_i} = [p_i^1, \ldots, p_i^C]$$

$$\overrightarrow{h_i} = \overrightarrow{LSTM}(\boldsymbol{V_i})$$

$$\overleftarrow{h_i} = \overleftarrow{LSTM}(\boldsymbol{V_i})$$

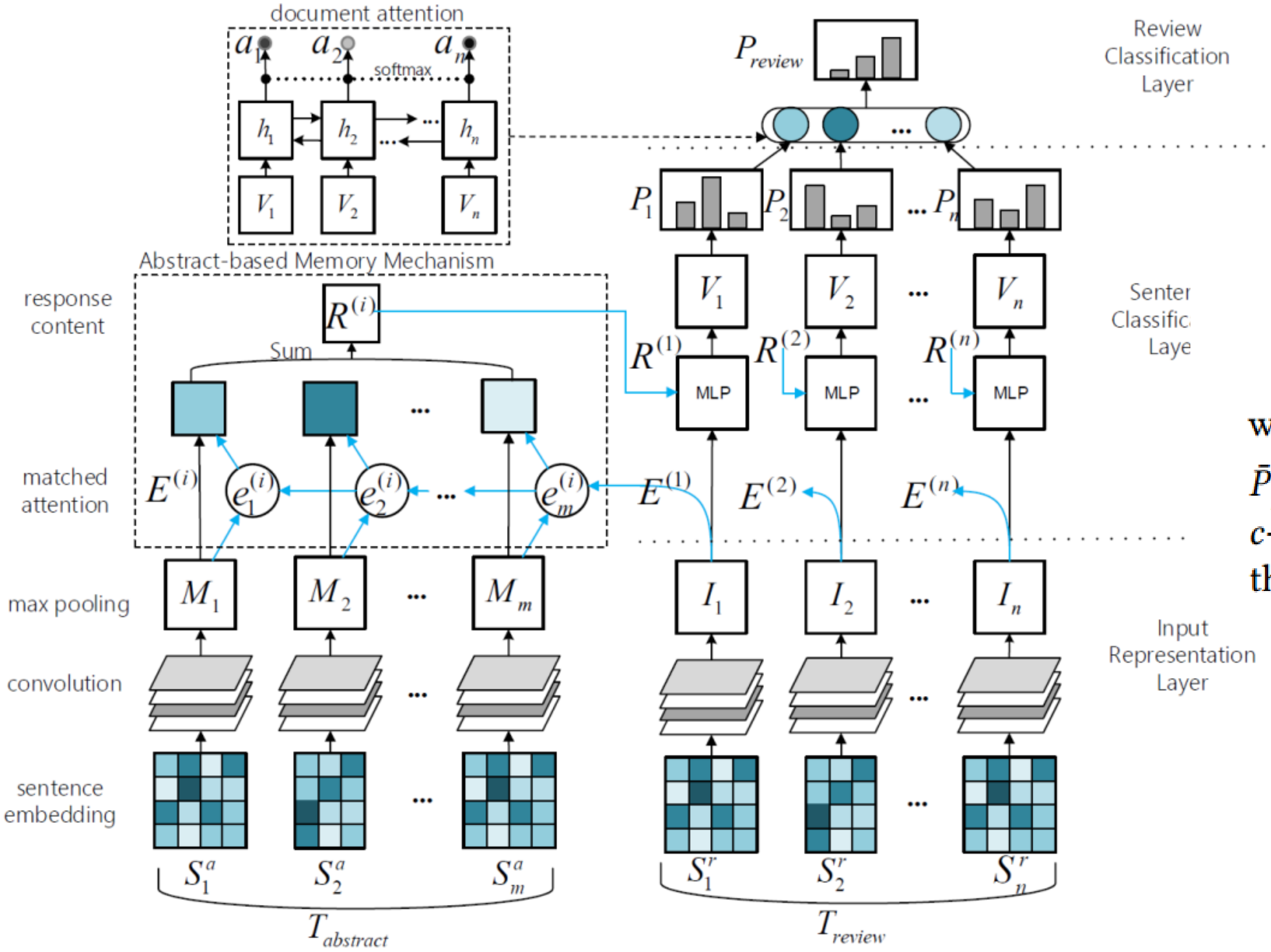
$$h_i = \overrightarrow{h_i} || \overleftarrow{h_i}$$

$$h_i' = tanh(\boldsymbol{W_a} \cdot h_i + \boldsymbol{b_a})$$

$$a_i = \frac{exp(h_i')}{\sum_j exp(h_j')}$$

$$P_{review}^{(c)} = \sum_i a_i P_i^{(c)}, c \in [1, C]$$

Model



- Objective Function

the categorical cross-entropy loss

$$L(\theta) = \sum_{T_{review}} \sum_{c=1}^{C} -P_{review}^{(c)} \log(\bar{P}_{review}^{(c)}) \quad (17)$$

where T_{review} is the review text in the training data, $P_{review}^{(c)}$ and $\bar{P}_{review}^{(c)}$ are the true and predicted probabilities of belonging to the c-th class, respectively. We use Adam [21] with minibatch to learn the model parameter θ.

Experiment

- ICLR, NIPS workshops, ICML workshops on the OpenReview website(https://openreview.net/)
- Only ICLR 2017 and ICLR 2018 provided both peer reviews and the corresponding overall recommendation scores for each submission
- K-fold cross validation(k=10)
- Google’s 300-dimensional word vectors

Result

This paper presents low-rank bilinear pooling that uses Hadamard product. The paper implements · · ·

· · · · · ·

I like the insights about low-rank bilinear pooling using Hadamard product presented in the paper. *However, it could not be justified that low-rank bilinear pooling leads to better performance than compact bilinear pooling. It does lead to reduction in number of parameters but it is justification of why low-rank bilinear pooling is better than other forms of pooling.*

Prediction: Accept

Summary:

[+0.19] I like the insights about low-rank bilinear pooling leads· · ·
[+0.12] The paper presents new insights into element-wise · · ·
[+0.06] The paper presents a new model for the task of VQA · · ·
[+· · ·] · · ·
[- 0.12] it could not be experimentally verified that low-rank · · ·
[- 0.11] I would like the authors to provide experimental · · ·
[- 0.05] It is not very clear from reading the paper.
[-· · ·] · · ·

Methods	ICLR-2017	ICLR-2018
SVM(Uni)	70.14% (+/- 4.37%)	71.02% (+/- 3.81%)
SVM(Uni&Bi)	71.56% (+/- 2.52%)	73.23%(+/- 5.29%)
SVM(Uni&Bi&Senti)	74.23% (+/- 3.21%)	75.24% (+/- 2.36%)
CNN	68.93% (+/- 5.24%)	73.31% (+/- 2.70%)
LSTM	65.24% (+/- 4.21%)	69.25% (+/- 3.82%)
CNN+Bi-LSTM	74.35% (+/- 3.51%)	76.21% (+/- 4.08%)
CNN+Bi-LSTM+Att	75.64% (+/- 3.42%)	77.85% (+/- 3.72%)
MIL	75.02% (+/- 2.85%)	76.57% (+/- 2.31%)
MILAM	**78.24% (+/- 4.92%)**	**80.32% (+/- 5.43%)**

A Co-Memory Network for Multimodal Sentiment Analysis

Nan Xu
Institute of Automation, Chinese Academy of Sciences
University of Chinese Academy of Sciences, Beijing, China
xunan2015@ia.ac.cn

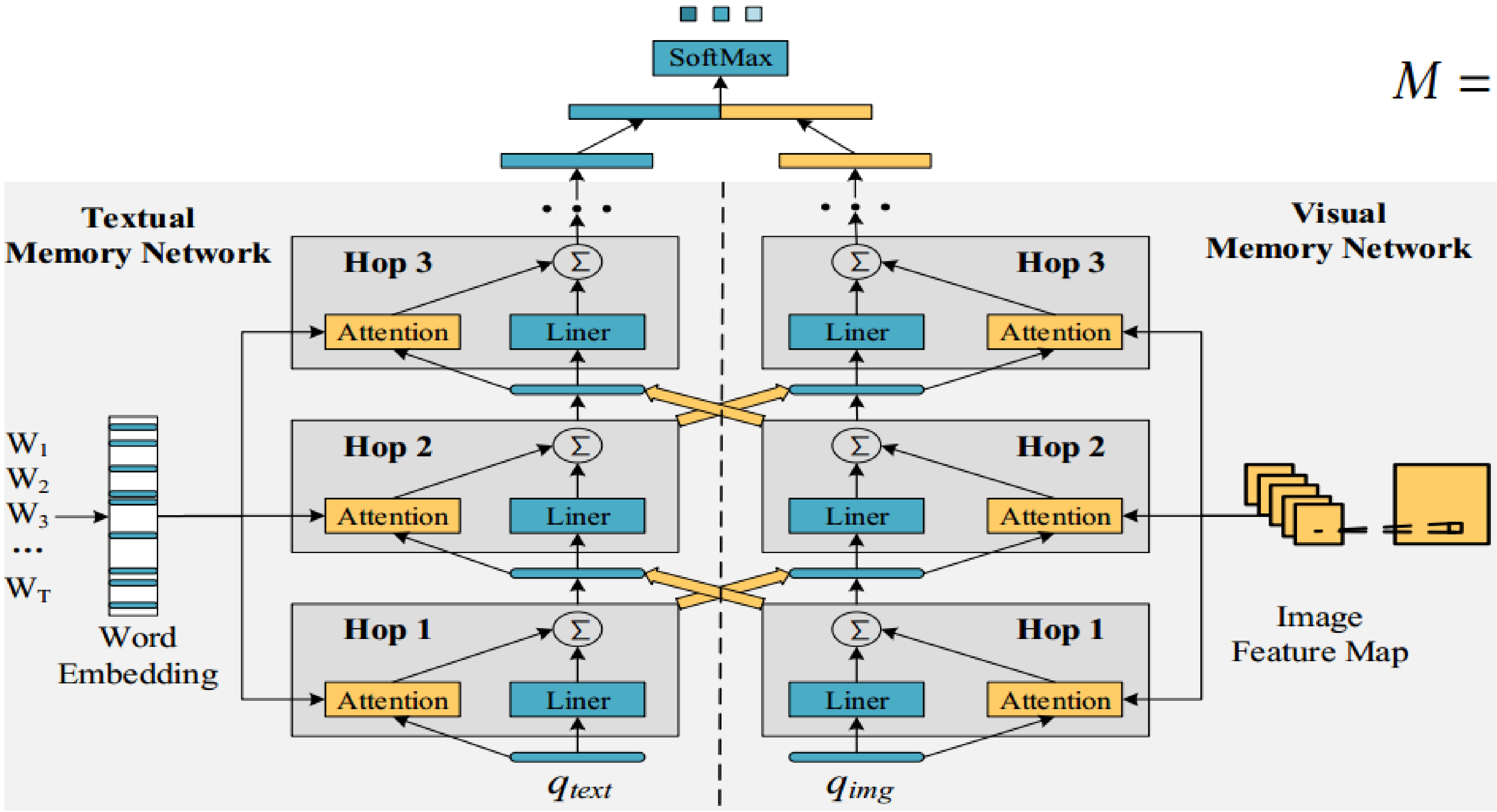
Wenji Mao
Institute of Automation, Chinese Academy of Sciences
University of Chinese Academy of Sciences, Beijing, China
wenji.mao@ia.ac.cn

Guandan Chen
Institute of Automation, Chinese Academy of Sciences
University of Chinese Academy of Sciences, Beijing, China
chenguandan2014@ia.ac.cn

Motivation

- As visual and textual information in multimodal data can mutually **reinforce and complement** each other in analyzing the sentiment of people, previous research all ignores this mutual influence between image and text.

Model



- Image Feature：

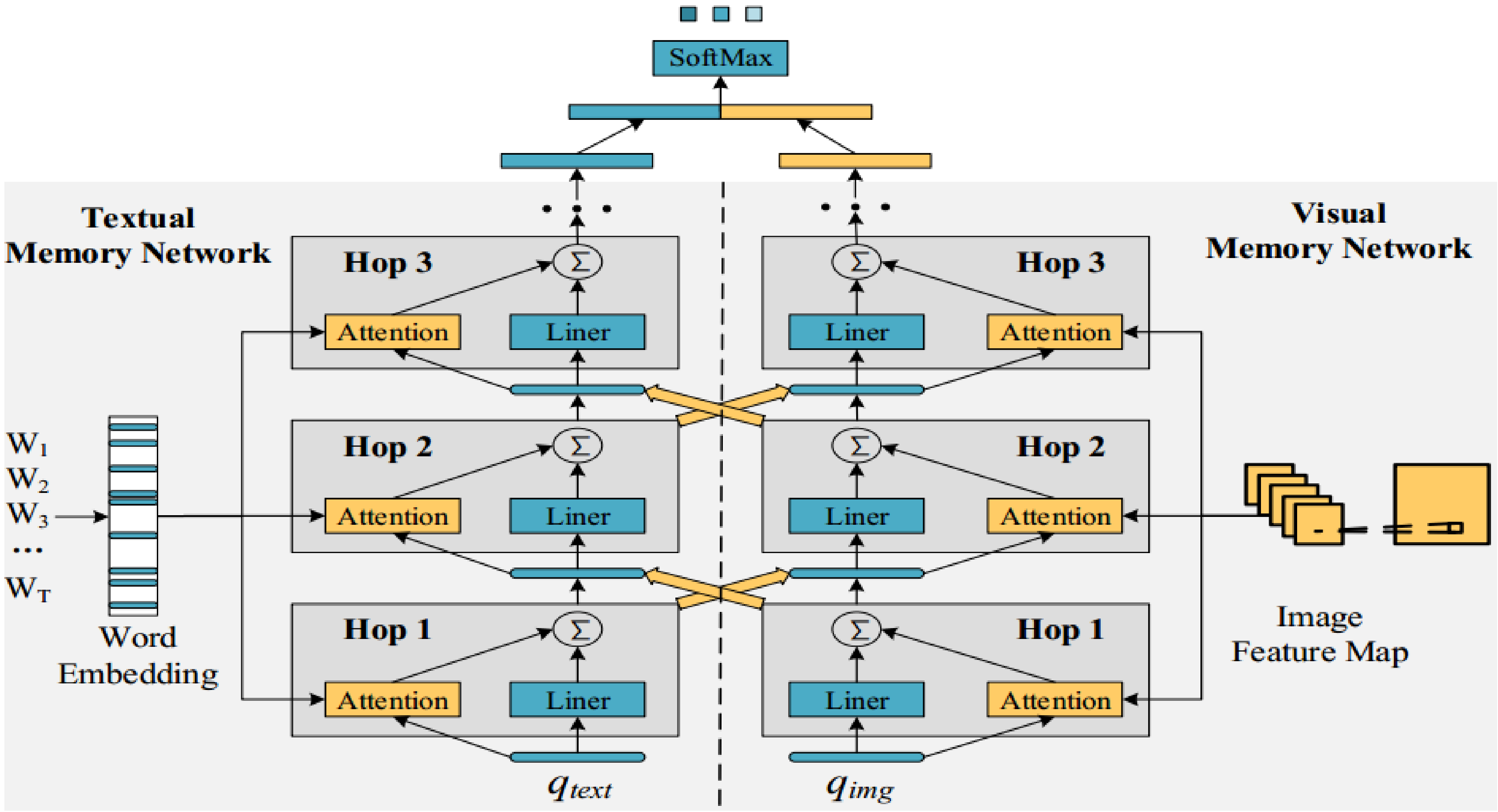
$$M = \{m_1, m_2, \ldots, m_L\}, m_i \in \mathbb{R}^{D_{img}}$$

$$h_i^1 = tanh(w_{img}^1 m_i + b_{img}^1)$$

$$\alpha_i^1 = \frac{exp({h_i^1}^T q_{img})}{\sum_{j=1}^{L} exp({h_j^1}^T q_{img})}$$

$$v_{img}^1 = \sum_{i=1}^{L} \alpha_i^1 m_i$$

Model



- Text Feature：

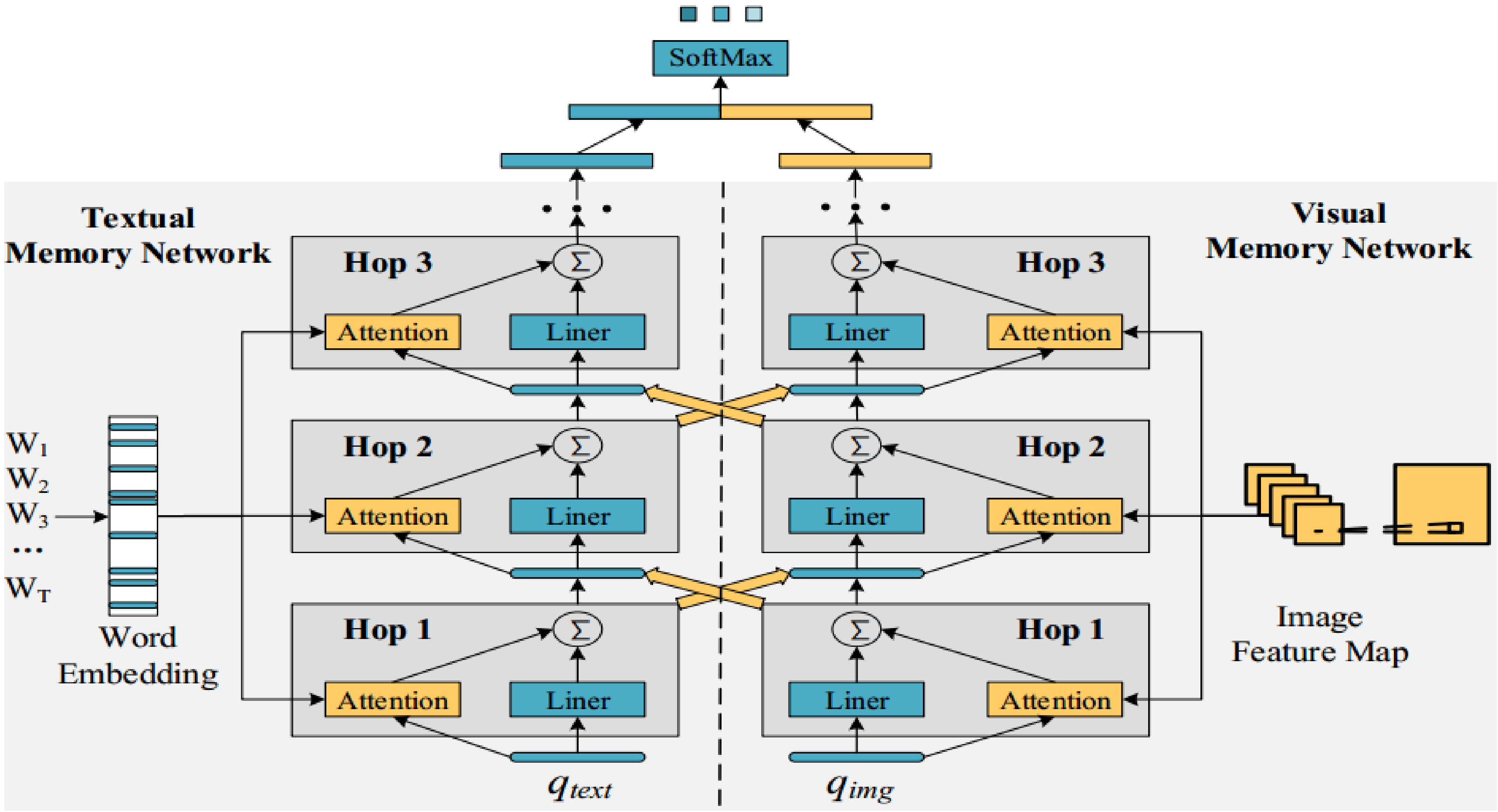
$$X = \{x_1, x_2, \dots, x_T\}, x_t \in \mathbb{R}^{D_{text}}$$

$$g_t^1 = tanh(w_{text}^1 x_t + b_{text}^1)$$

$$\beta_t^1 = \frac{exp({g_t^1}^T q_{text})}{\sum_{r=1}^T exp({g_r^1}^T q_{text})}$$

$$v_{text}^1 = \sum_{t=1}^T \beta_t^1 x_t$$

Model



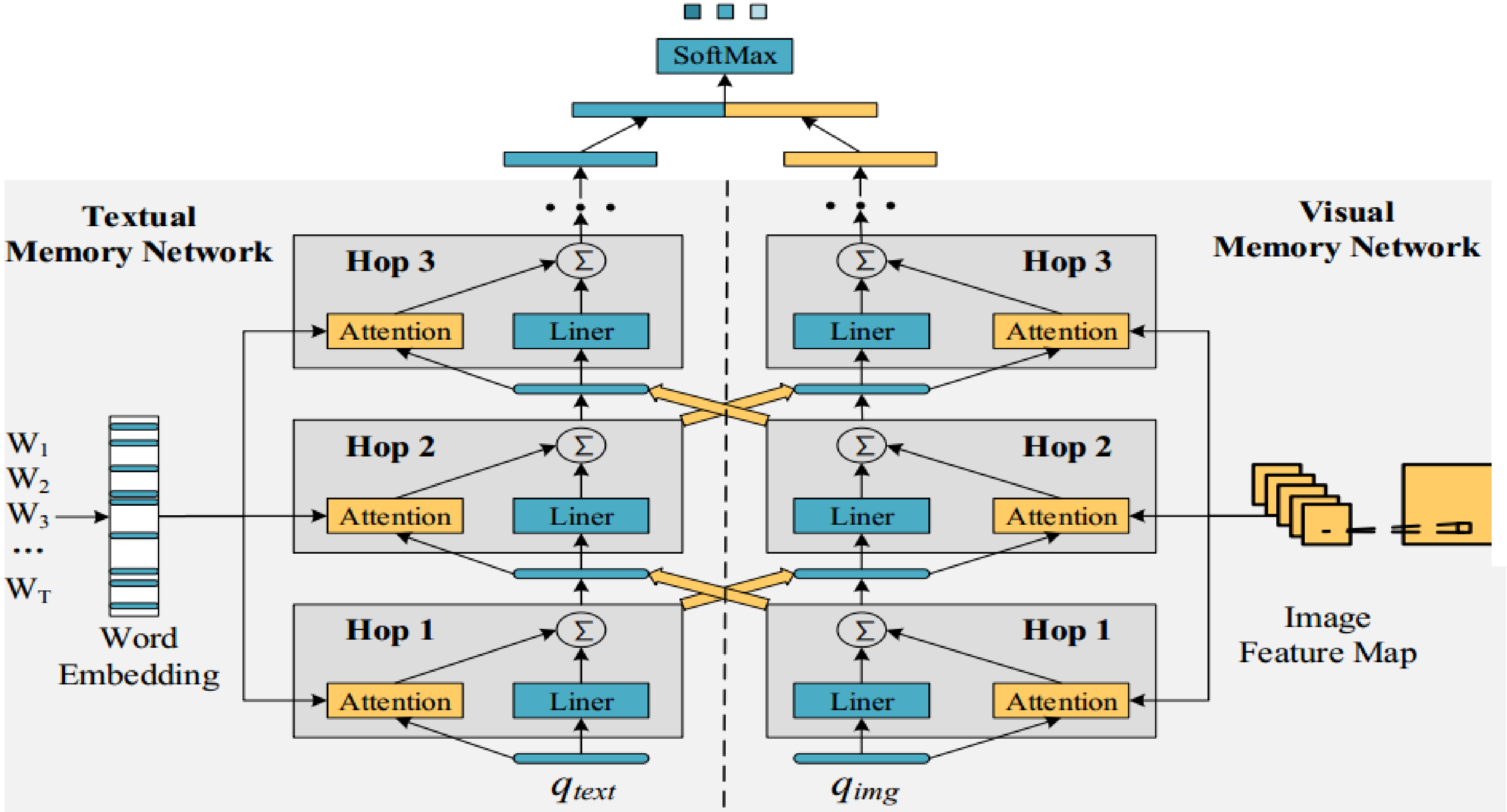
- Co-Memory Network:
- 1. Text-guided Visual Memory Network (TgVMN):

$$h_i^2 = tanh(w_{img}^2[m_i, v_{text}^1] + b_{img}^2)$$

$$\alpha_i^2 = \frac{exp(h_i^2)}{\sum_{j=1}^{L} exp(h_j^2)}$$

$$v_{img}^2 = \sum_{i=1}^{L} \alpha_i^2 m_i$$

Model



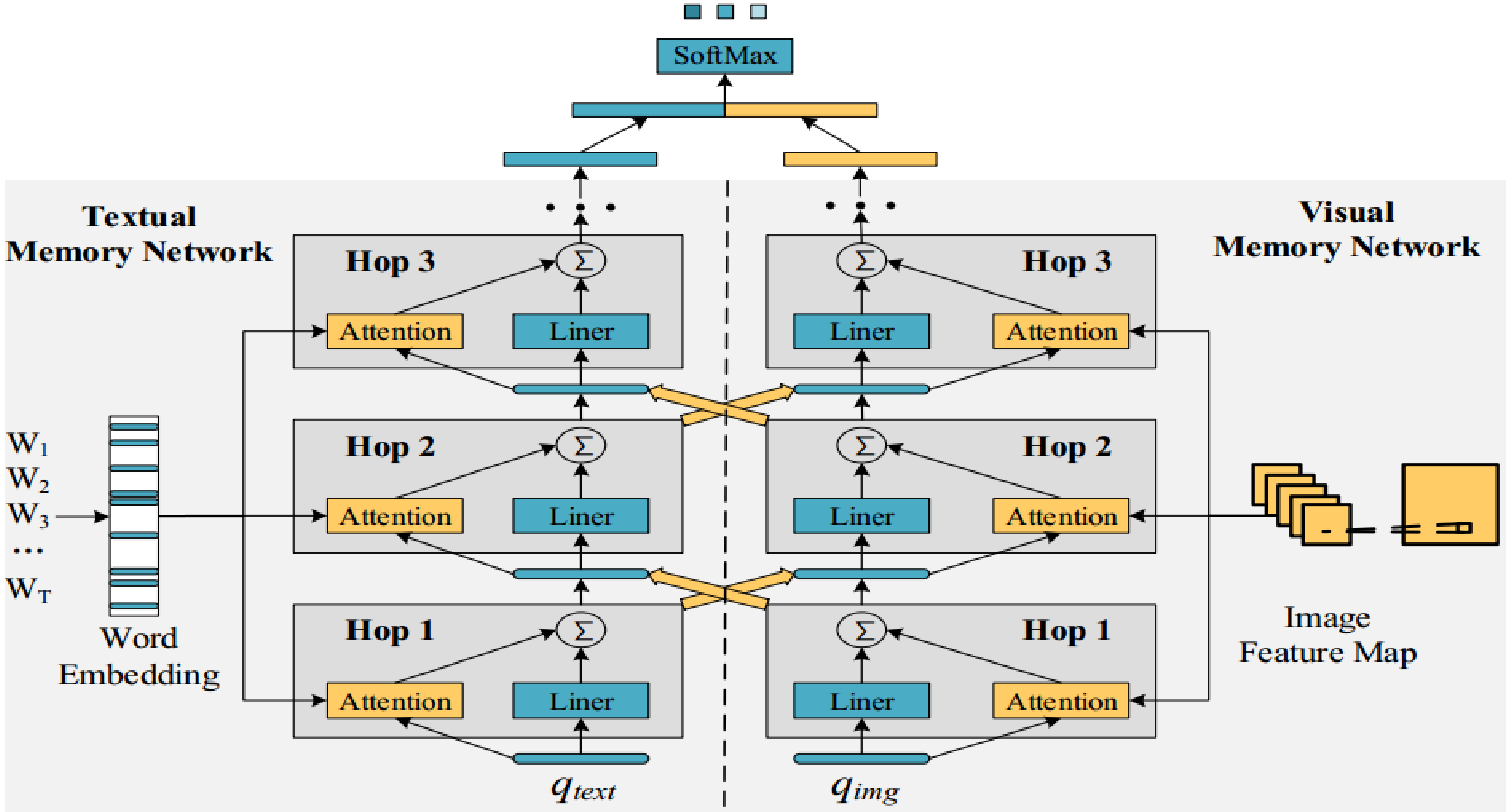
- 2. Image-guided Textual Memory Network (IgTMN):

$$g_t^2 = tanh(w_{text}^2[x_t, v_{img}^1] + b_{text}^2)$$

$$\beta_t^2 = \frac{exp(g_t^2)}{\sum_{r=1}^{T} exp(g_r^2)}$$

$$v_{text}^2 = \sum_{t=1}^{T} \beta_t^2 x_t$$

Model



- Stacked Co-Memory Network：

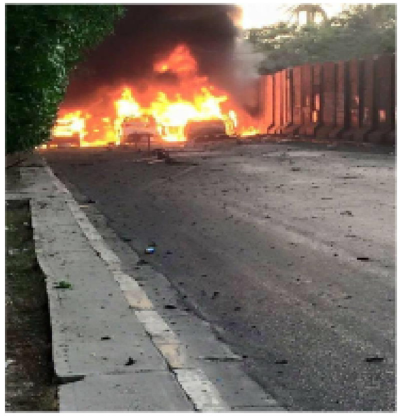
$$v_{text}^{k} = IgTMN([x_t, v_{img}^{k-1}])$$
$$v_{img}^{k} = TgVMN([m_i, v_{text}^{k-1}])$$

- Sentiment Classification：

$$y = Softmax(w_s[v_{text}^{K}, v_{img}^{K}] + b_s)$$

Experiment

Method	MVSA-Single		MVSA-Multi	
	Acc	*F1*	*Acc*	*F1*
SentiBank+SentiStrength	52.05	50.08	65.62	55.36
CBOW+DA+LR	63.86	63.52	64.22	63.73
CNN-Multi	61.20	58.37	66.39	64.19
DNN-LR	61.42	61.03	67.86	66.33
HSAN	66.83	66.9	68.16	67.76
MultiSentiNet	69.84	69.63	68.86	68.11
MN-Hop1	64.31	63.12	67.16	66.48
MN-Hop2	64.84	63.96	67.32	66.57
MN-Hop2+text2img	65.19	64.37	67.80	67.01
MN-Hop2+img2text	68.07	65.19	67.92	67.16
CoMN-Hop2	70.07	68.03	68.68	68.06
CoMN-Hop3	69.62	65.95	69.39	68.57
CoMN-Hop4	69.18	68.29	**69.92**	**69.83**
CoMN-Hop5	69.40	69.71	69.68	69.31
CoMN-Hop6	**70.51**	**70.01**	68.92	68.83



RT @IraqiSecurity: Car bomb in Aden Sq, Kadhimiyah, N. #Baghdad. Initial reports suggest at least 5 martyrs & 22 wounded. #ShiaGenocide httбн

RT @IraqiSecurity: Car bomb in Aden Sq, Kadhimiyah, N. #Baghdad. Initial reports suggest at least 5 martyrs & 22 wounded. #ShiaGenocide httбн

Convolution-based Memory Network for Aspect-based Sentiment Analysis

Chuang Fan*
Shenzhen Graduate School,
Harbin Institute of Technology

Qinghong Gao*
Shenzhen Graduate School,
Harbin Institute of Technology

Jiachen Du
Shenzhen Graduate School,
Harbin Institute of Technology

Lin Gui
Aston University

Ruifeng Xu†
Shenzhen Graduate School,
Harbin Institute of Technology
xuruifeng@hit.edu.cn

Kam-Fai Wong
The Chinese University of Hong Kong

Author

kam-Fai Wong

chinese university of hong kong
Verified email at se.cuhk.edu.hk - Homepage
natuural language processing   database

Cited by   VIEW ALL

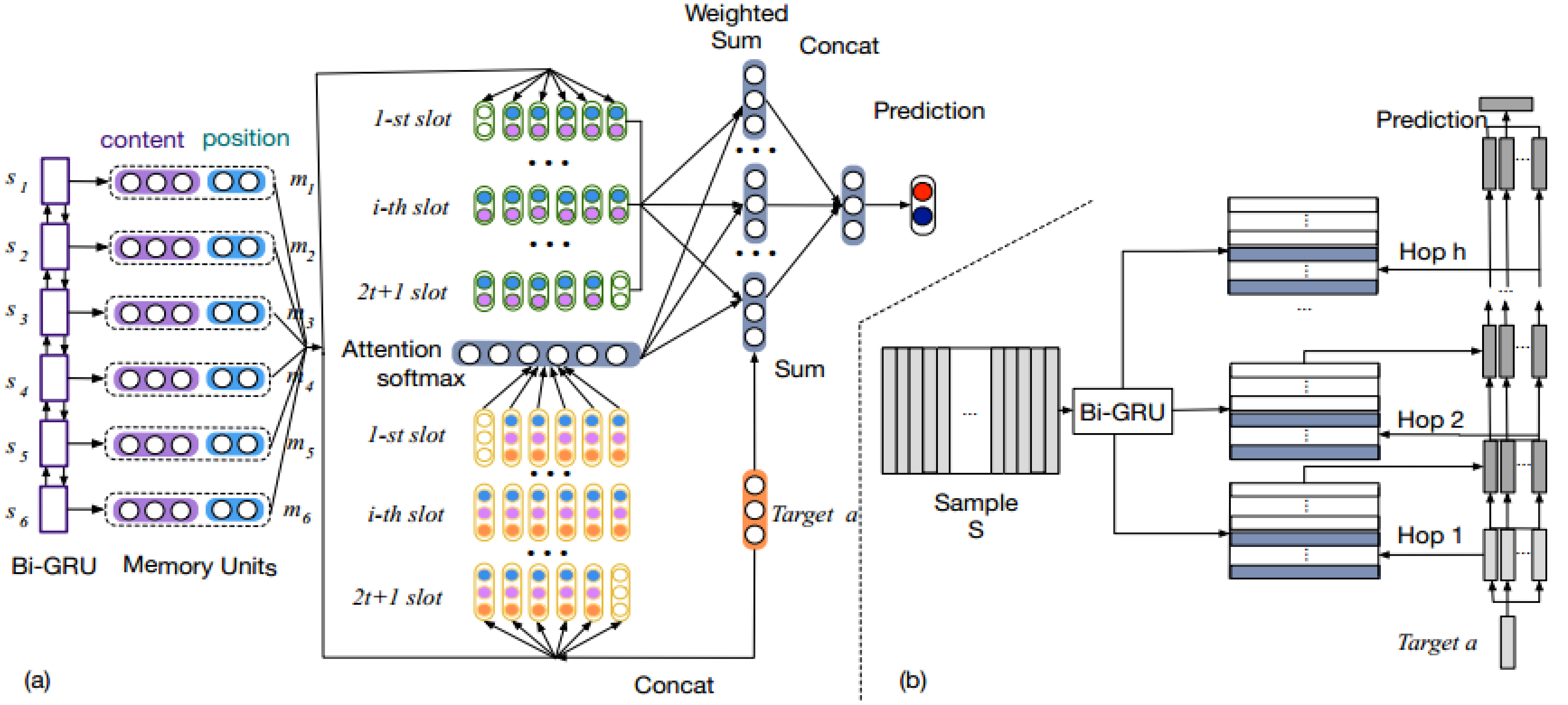
	All	Since 2013
Citations	4250	2252
h-index	32	23
i10-index	95	56

TITLE	CITED BY	YEAR
Interpreting tf-idf term weights as making relevance decisions HC Wu, RWP Luk, KF Wong, KL Kwok ACM Transactions on Information Systems (TOIS) 26 (3), 13	478	2008
Component-based software engineering: technologies, development frameworks, and quality assurance schemes X Cai, MR Lyu, KF Wong, R Ko Software Engineering Conference, 2000. APSEC 2000. Proceedings. Seventh Asia ...	215	2000
Extractive summarization using supervised and semi-supervised learning KF Wong, M Wu, W Li Proceedings of the 22nd International Conference on Computational ...	148	2008
A genetic algorithm-based clustering approach for database partitioning CH Cheng, WK Lee, KF Wong IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and ...	130	2002

Motivation

- Ordinary memory networks only capture word-level information and lack the capacity for modeling complicated expressions which consist of multiple words

Model



- Pre-processing：

$$M = \{m_1, m_2, \ldots, m_n\}$$

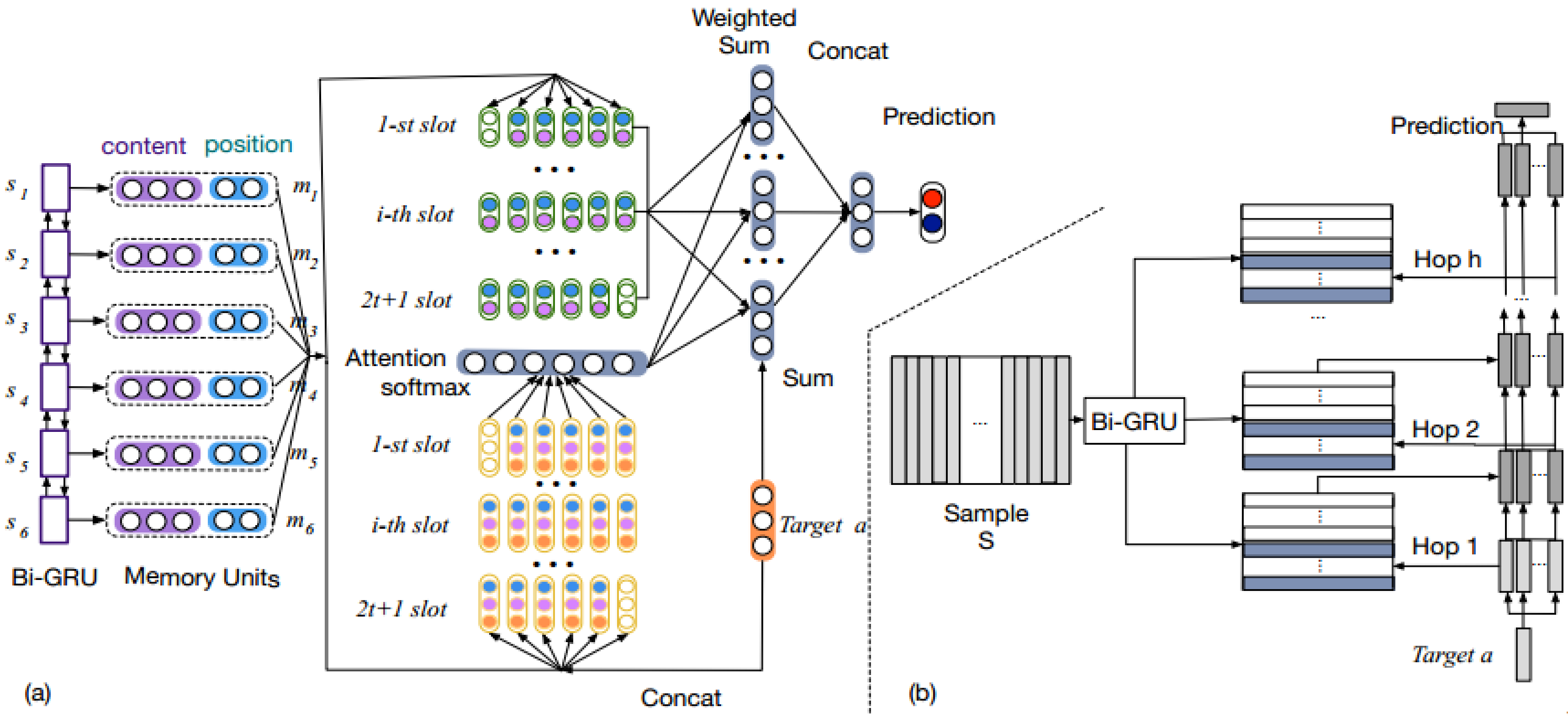
- Attention：

$$g_i = w^T \left(\sum_{j=i-t}^{i+t} (m_j \oplus T) \right)$$

$$\alpha_i = \frac{\exp(g_i)}{\sum_{i'} \exp(g_{i'})}$$

$$o_j = T + \sum_{i=1}^{n} \alpha_i \cdot m_{i+j} \qquad (j = -t, \ldots, 0, \ldots, t)$$

Model



- Multi-hop：

$$g_i = w^T(\sum_{j=i-t}^{i+t}(m_j \oplus o_{j-i}^{h-1}))$$

- Prediction：

$$o = o_{-t} \oplus o_{-t+1} \ldots o_t$$

$$\hat{o} = softmax(W_m \cdot o)$$

- Objective function:

$$L = \sum_{(x,y)\in D} \sum_{c\in C} y^c log f^c(x;\theta)$$

Experiment

Table 1: Experimental performance

Method	Laptop		Restaurant		Tweet	
	ACC	Macro-F1	ACC	Macro-F1	ACC	Macro-F1
SVM	0.7049	NA	0.8016	NA	0.6340	0.6330
TD-LSTM	0.7183	0.6843	0.7800	0.6673	0.6662	0.6401
MemNet	0.7033	0.6409	0.7816	0.6583	0.6850	0.6691
RAM	0.7449	0.7135	**0.8023**	**0.7080**	0.6936	0.6730
Conv-Memnet	**0.7637**	**0.7210**	0.7826	0.6838	**0.7211**	**0.7080**

No. of Hops	Laptop	Restaurant	Tweet
Hop 1	0.6775	0.6689	0.6735
Hop 2	0.7000	0.6752	0.6767
Hop 3	**0.7210**	**0.6838**	**0.7080**
Hop 4	0.6869	0.6741	0.6948
Hop 5	0.6694	0.6406	0.6795
Hop 6	0.6655	0.6226	0.6631

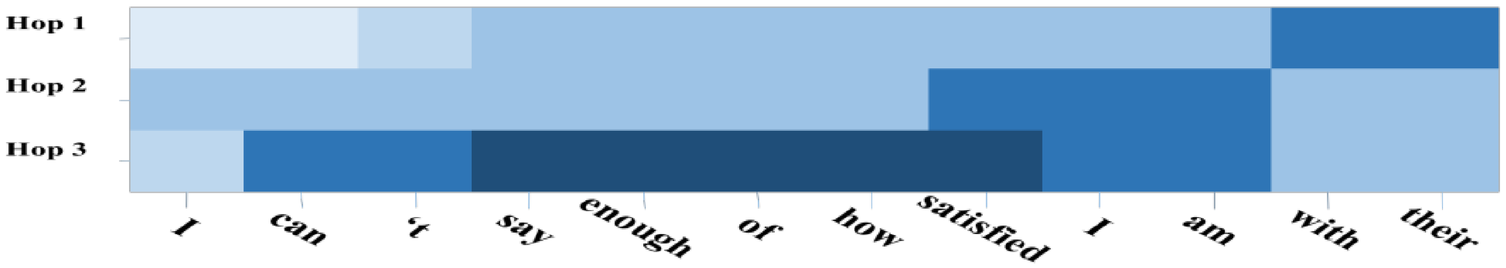


Figure 2: The changes in each hop of attention

SAAN: A Sentiment-Aware Attention Network for Sentiment Analysis

Zeyang Lei
Graduate School at Shenzhen,
Tsinghua University
leizy16@mails.tsinghua.edu.cn

Yujiu Yang
Graduate School at Shenzhen,
Tsinghua University
yang.yujiu@sz.tsinghua.edu.cn

Min Yang
Shenzhen Institutes of Advanced Technology,
Chinese Academy of Sciences
min.yang1129@gmail.com

Motivation

- make full use of the prior sentiment knowledge (e.g., sentiment lexicon, negation words, intensity words)

Word-level correlation modeling

- Pre-processing:

$$x^c = [x_1^c, x_2^c, \ldots x_i^c, \ldots, x_n^c]$$

$$x^s = [x_1^s, x_2^s, \ldots, x_j^s, \ldots, x_m^s]$$

$$R = (x^c)^T \cdot x^s \in \mathbb{R}^{n \times m} \quad (1)$$

where each element $R_{i,j}$ refers to the correlation between the i-th word x_i^c in the context and j-th word x_j^s in the sentiment resource words.

- Attention: $\phi = softmax(\frac{\sum_{h=1}^{m} R[:,h]}{m})$ $\quad \varphi = softmax(\frac{\sum_{k=1}^{n} R[k,:]}{n})$

$$\mathbf{v}^{\phi} = \sum_{i=1}^{n} \phi_i \mathbf{x}_i^c. \qquad \mathbf{v}^{\varphi} = \sum_{i=1}^{m} \varphi_i \mathbf{x}_i^s$$

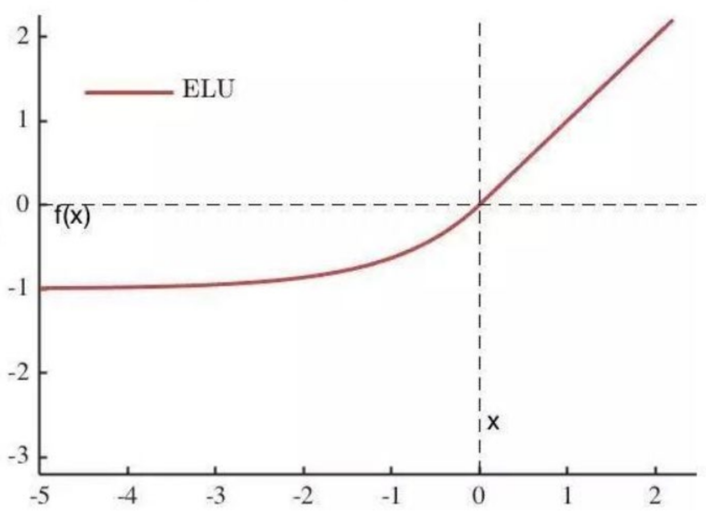
Word-level correlation modeling

- Enhanced representation:

$$E^c = \tilde{f}_{semantics}(x^c + \mathbf{e_n} \otimes \mathbf{v}^{\boldsymbol{\phi}}) \quad (2)$$

$$\tilde{f}_{semantics}(x) = \begin{cases} x & x > 0 \\ \xi(exp(x) - 1) & x \leq 0, \xi > 0 \end{cases} \quad (3)$$

$$E^s = \tilde{f}_{semantics}(x^s + \mathbf{e_m} \otimes \mathbf{v}^{\boldsymbol{\varphi}}) \quad (4)$$



Phrase-Level Correlation Modeling

- CNN

$$P^c = P^c_{\ell_1} \oplus P^c_{\ell_2} \cdots \oplus P^c_{\ell_q} \tag{5}$$

$$P^s = P^s_{\ell_1} \oplus P^s_{\ell_2} \cdots \oplus P^s_{\ell_q} \tag{6}$$

$$P^c_{\ell_i} = \gamma(E^c * W^{\ell_i} + b) \in \mathbb{R}^{(n-\ell_i+1)\times k} \tag{7}$$

$$P^s_{\ell_i} = \gamma(E^s * W^{\ell_i} + b) \in \mathbb{R}^{(m-\ell_i+1)\times k}, i = 1, 2, ..., q \tag{8}$$

- Attention

$$S = P^s \odot \{(P^c)^T softmax(P^c(P^s)^T)\}$$

$$\bar{s} = \frac{\sum_{j=1}^{m} S[:, j]}{m}$$

Final Model

- Sentence-level Semantic Modeling

$$\rho = Q^T tanh(\tilde{U}(P^c + e_n \otimes \bar{s}))$$

$$B = \frac{exp(\rho)}{\sum_{i=1}^{n} exp(\rho)}$$

$$\tilde{\mathbf{o}} = flatten(BP^s)$$

- Sentence Classifer

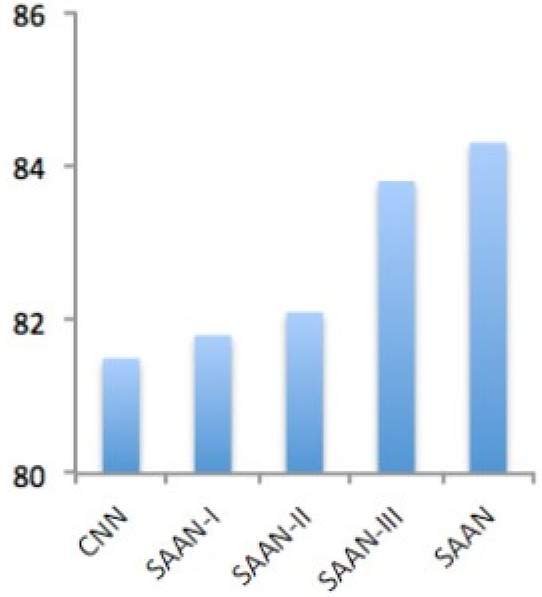
$$\hat{y} = \frac{exp(U_o^T \tilde{\mathbf{o}} + b_o)}{\sum_{k=1}^{C} exp(U_o^T \tilde{\mathbf{o}} + b_o)}$$

- Loss function

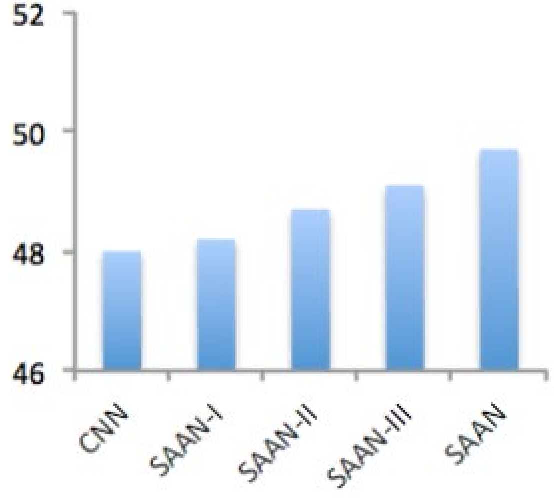
$$L(\hat{y}, y) = -\sum_{i=1}^{N} \sum_{j=1}^{C} y_i^j log(\hat{y}_i^j) + \lambda(\sum_{\theta \in \Theta} \theta^2)$$

Experiment

Methods	MR	SST(sent.-level)
RNTN [10]	75.9%	45.7%
LSTM [2]	77.4%	45.6%
BiLSTM	79.3%	46.5%
Tree-LSTM [11]	80.7%	48.1%
CNN [4]	81.5%	48.0%
NSCL [12]	82.9%	47.1%
LR-LSTM [8]	81.5%	48.3%
LR-Bi-LSTM [8]	82.1%	48.6%
Self-attention [6]	82.5%*	48.7%*
SAAN (our model)	**84.3%**	**49.7%**



(a) MR



(b) SST

Measuring Influence on Instagram: a Network-oblivious Approach

Noam Segev, Noam Avigdor, Eytan Avigdor

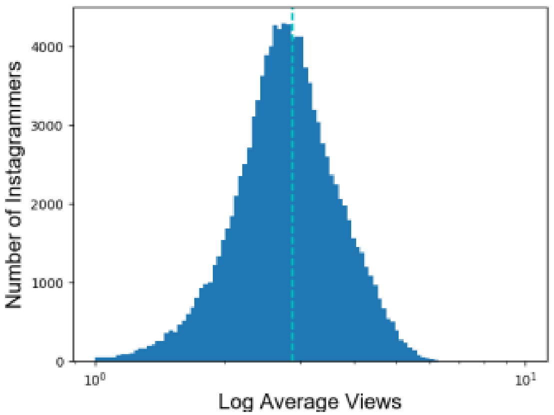
Motivation

- Graphs information are not always available, and building them for Instagram users require a great deal of resources, e.g., crawling time and computing costs.
- Infer the underlying network structure using the user activity logs
- Instagram content, however, is “pulled”

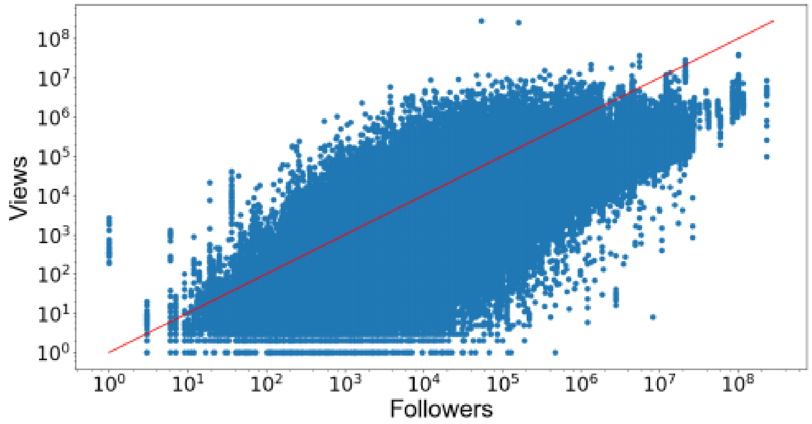
Method

- Collected feature
  - $likes_c$ - Number of likes awarded to post c.
  - $comments_c$ - Number of comments given to post c.
  - v_c - Number of Instagrammers who watched part of the video.

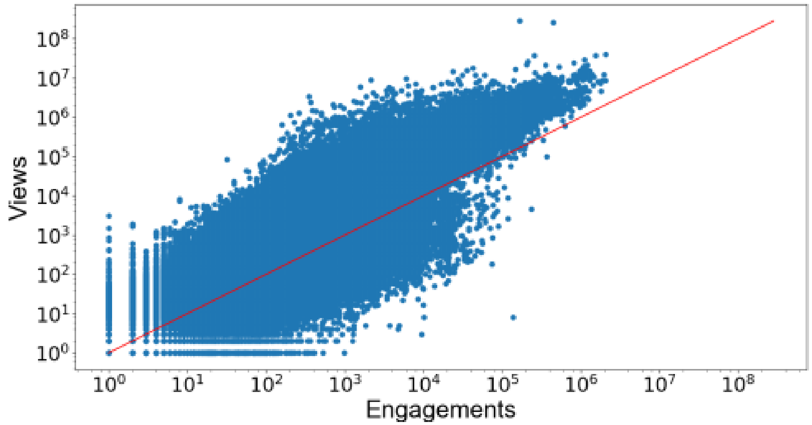
- Instagram statistics



(a) Views Histogram



(b) Views per Followers



(c) Views per Likes

Method

- Features controlled

$likes$ - The average number of user post likes.
$comments$ - The average number of comments per user post.
$followers$ - The users audience size.
$\sqrt{likes \cdot followers}$ - Geometric mean of likes and followers, taken as neither statistic is an exact representation of influence.
$\frac{followers}{post}$ - Used to suggest odd behavior as same level influencers should have similar ratios.
$\frac{comments}{likes}$ - Another odd behavior indicator as bought engagements tend to effect likes more than comments.

$focus$ - The difference and ratio between most and least engaged post, these features were designed to test the variance and stability of a user engagement level.

Model

- Regression model

  Ridge Regression(RR)

  Random Forests(RF)

- Tricks

  K-means

$$f\left(x\right)=\frac{x}{\ln x}$$

Experiment

	Regression		Multi-Regression	
	R^2	r_s	R^2	r_s
full Ridge Regression	**0.725**	0.848	**0.727**	0.821
full Random Forest	0.626	**0.869**	0.621	**0.861**
minimal Ridge Regression	0.723	0.818	0.727	0.818
minimal Random Forest	0.616	0.864	0.611	0.859
Followers Baseline	0.211	0.757	0.204	0.725
Likes Baseline	0.666	0.859	0.654	0.853

Thanks!