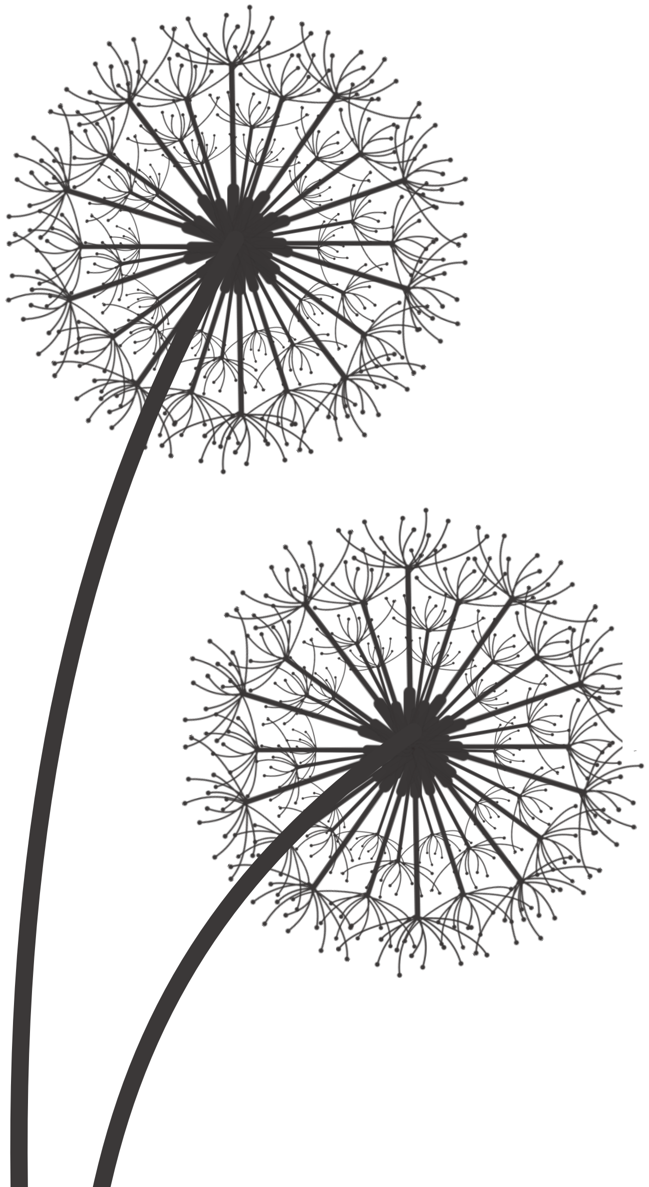


Topic-Aware Neural Keyphrase Generation for Social Media Language

Yue Wang[1*] **Jing Li**[2†] **Hou Pong Chan**[1] **Irwin King**[1] **Michael R. Lyu**[1] **Shuming Shi**[2]

[1]Department of Computer Science and Engineering
The Chinese University of Hong Kong, HKSAR, China
[2]Tencent AI Lab, Shenzhen, China
[1]{yuewang, hpchan, king,lyu}@cse.cuhk.edu.hk
[2]{ameliajli,shumingshi}@tencent.com

https://arxiv.org/pdf/1906.03889.pdf

• Motivation：

- 现阶段很多关键词方面的研究都是从句子中抽取现有的关键词无法预测语句中没有关键词

Source post with keyphrase "*super bowl*":
[S]: Somewhere, a wife that is not paying attention to the *game*, says "I want the *team* in *yellow pants* to *win*."
Relevant tweets:
[T_1]: I been a *steelers fan* way before *black* & *yellow* and this *super bowl*!
[T_2]: I will bet you the *team* with *yellow pants wins*.
[T_3]: Wiz Khalifa song '*black* and *yellow*" to spur the *pittsburgh steelers* and Lil Wayne is to sing "*green* and *yellow*' for the *packers*.

Table 1: Sample tweets tagged with "*super bowl*" as their keyphrases. *Blue and italic words* can indicate the topic of super bowl.

- Topic-Aware Neural Keyphrase Generation Model

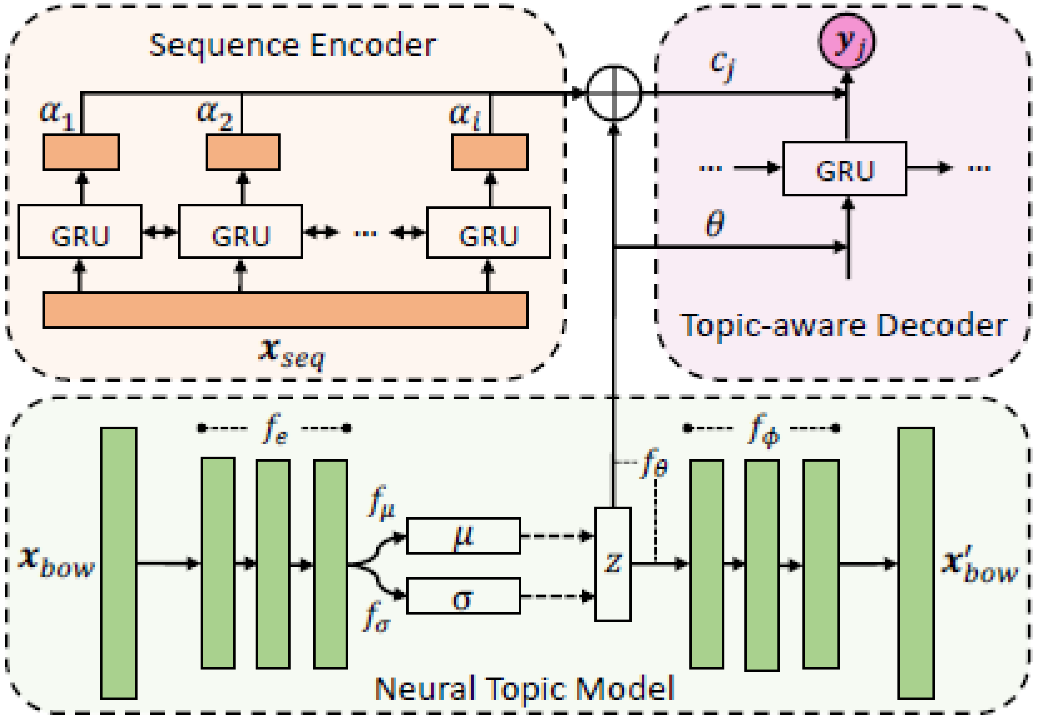
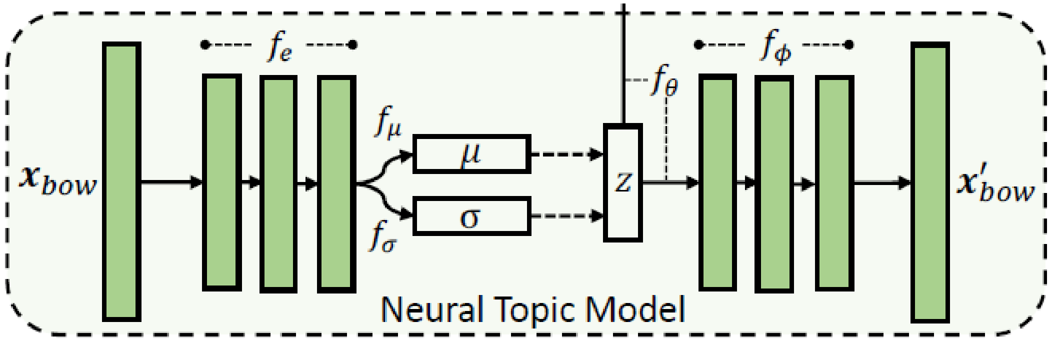


Figure 1: Our topic-aware neural keyphrase generation framework (§3).

- A neural topic model for exploring latent topics
- A seq2seq-based model for keyphrase generation

- Neural topic model



Each post： Xbow （V-dim vector）

$$\mu = f_\mu(f_e(\mathbf{x}_{bow})),\ log\,\sigma = f_\sigma(f_e(\mathbf{x}_{bow})),$$

BoW Encoder

Draw latent topic variable： $\mathbf{z} \sim \mathcal{N}(\mu, \sigma^2)$

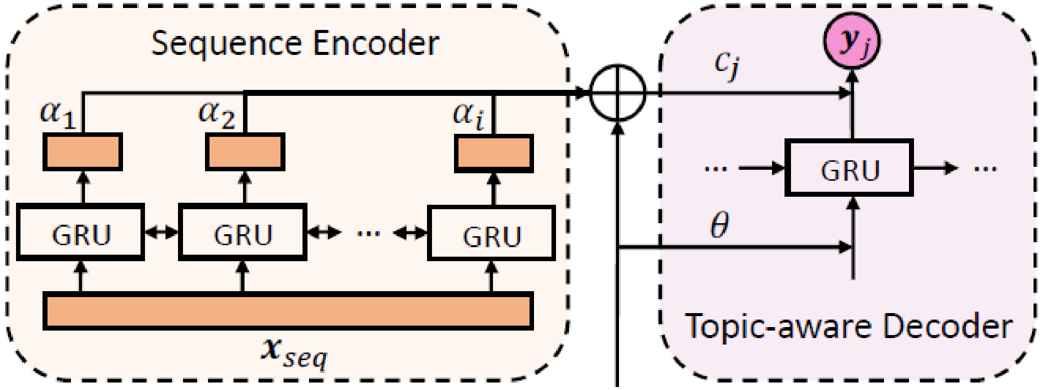
$\theta = softmax(f_\theta(\mathbf{z}))$ （K-dim distributional vector）

For each word $w \in \mathbf{x}$

– Draw $w \sim softmax(f_\phi(\theta))$

BoW Decoder

- Neural Keyphrase Generation Model



Input :

$$\mathbf{x}_{seq} = \langle w_1, w_2, ..., w_{|\mathbf{x}|} \rangle$$

- Sequence Encoder

$$\overrightarrow{\mathbf{h}_i} = f_{GRU}(\nu_i, \mathbf{h}_{i-1}),$$

$$\overleftarrow{\mathbf{h}_i} = f_{GRU}(\nu_i, \mathbf{h}_{i+1}).$$

$$[\overrightarrow{\mathbf{h}_i}; \overleftarrow{\mathbf{h}_i}] \longrightarrow \mathbf{h}_i$$

- Topic-Aware Sequence Decoder

$$Pr(\mathbf{y} \,|\, \mathbf{x}) = \prod_{j=1}^{|\mathbf{y}|} Pr(y_j \,|\, \mathbf{y}_{<j}, \mathbf{M}, \theta)$$

$$\mathbf{s}_j = f_{GRU}([\mathbf{u}_j; \theta], \mathbf{s}_{j-1})$$

- Topic-Aware Sequence Decoder

$$\mathbf{s}_j = f_{GRU}([\mathbf{u}_j;\theta], \mathbf{s}_{j-1})$$

attention

$$\alpha_{ij} = \frac{\exp(f_\alpha(\mathbf{h}_i, \mathbf{s}_j, \theta))}{\sum_{i'=1}^{|\mathbf{x}|} \exp(f_\alpha(\mathbf{h}_{i'}, \mathbf{s}_j, \theta))}$$

$$f_\alpha(\mathbf{h}_i, \mathbf{s}_j, \theta) = \mathbf{v}_\alpha^T \, tanh(\mathbf{W}_\alpha[\mathbf{h}_i; \mathbf{s}_j; \theta] + \mathbf{b}_\alpha)$$

$$p_{gen} = softmax(\mathbf{W}_{gen}[\mathbf{s}_j; \mathbf{c}_j] + \mathbf{b}_{gen})$$

- copy mechanism

$$\lambda_j = sigmoid(\mathbf{W}_\lambda[\mathbf{u}_j; \mathbf{s}_j; \mathbf{c}_j; \theta] + \mathbf{b}_\lambda)$$

$$p_j = \lambda_j \cdot p_{gen} + (1 - \lambda_j) \cdot \sum_{i=1}^{|\mathbf{x}|} \alpha_{ij}$$

- Jointly Learning Topics and Keyphrases

NTM: $$\mathcal{L}_{NTM} = D_{KL}(p(\mathbf{z}) \,||\, q(\mathbf{z} \,|\, \mathbf{x})) - \mathbb{E}_{q(\mathbf{z} \,|\, \mathbf{x})}[p(\mathbf{x} \,|\, \mathbf{z})]$$

KG model: $$\mathcal{L}_{KG} = -\sum_{n=1}^{N} \log(Pr(\mathbf{y}_n \,|\, \mathbf{x}_n, \theta_n))$$

$$\mathcal{L} = \mathcal{L}_{NTM} + \gamma \cdot \mathcal{L}_{KG}$$

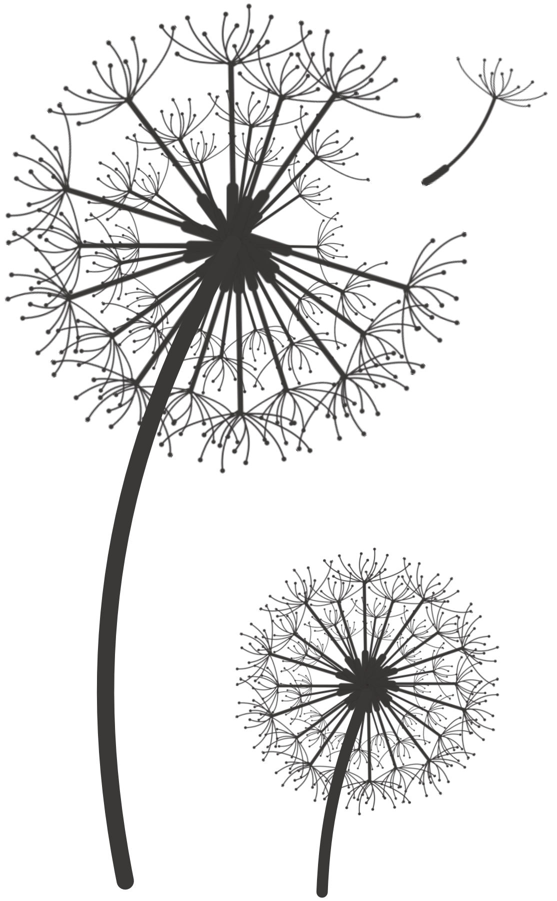
- Evaluation

  - evaluate our performance on keyphrase prediction

Model	Twitter			Weibo			StackExchange		
	F1@1	F1@3	MAP	F1@1	F1@3	MAP	F1@3	F1@5	MAP
<u>Baselines</u>									
MAJORITY	9.36	11.85	15.22	4.16	3.31	5.47	1.79	1.89	1.59
TF-IDF	1.16	1.14	1.89	1.90	1.51	2.46	13.50	12.74	12.61
TEXTRANK	1.73	1.94	1.89	0.18	0.49	0.57	6.03	8.28	4.76
KEA	0.50	0.56	0.50	0.20	0.20	0.20	15.80	15.23	14.25
<u>State of the arts</u>									
SEQ-TAG	22.79 ± 0.3	12.27 ± 0.2	22.44 ± 0.3	16.34 ± 0.2	8.99 ± 0.1	16.53 ± 0.3	17.58 ± 1.6	12.82 ± 1.2	19.03 ± 1.3
SEQ2SEQ	34.10 ± 0.5	26.01 ± 0.3	41.11 ± 0.3	28.17 ± 1.7	20.59 ± 0.9	34.19 ± 1.7	22.99 ± 0.3	20.65 ± 0.2	23.95 ± 0.3
SEQ2SEQ-COPY	<u>36.60</u> ± 1.1	<u>26.79</u> ± 0.5	<u>43.12</u> ± 1.2	<u>32.01</u> ± 0.3	<u>22.69</u> ± 0.2	<u>38.01</u> ± 0.1	31.53 ± 0.1	27.41 ± 0.2	33.45 ± 0.1
SEQ2SEQ-CORR	34.97 ± 0.8	26.13 ± 0.4	41.64 ± 0.5	31.64 ± 0.7	22.24 ± 0.5	37.47 ± 0.8	30.89 ± 0.3	26.97 ± 0.2	32.87 ± 0.6
TG-NET	-	-	-	-	-	-	<u>32.02</u> ± 0.3	<u>27.84</u> ± 0.3	<u>34.05</u> ± 0.4
Our model	**38.49** ± 0.3	**27.84** ± 0.0	**45.12** ± 0.2	**34.99** ± 0.3	**24.42** ± 0.2	**41.29** ± 0.4	**33.41** ± 0.2	**29.16** ± 0.1	**35.52** ± 0.1

Datasets	Twitter	StackExchange
LDA	41.12	35.13
BTM	43.12	43.52
NTM	43.82	43.04
Our model	**46.28**	**45.12**

Table 4: C_V topic coherence score comparison on our two English datasets. Higher scores indicate better coherence. Our model produces the best scores.



THANKS