Paper Reading

Neural Extractive Summarization with Hierarchical Attentive Heterogeneous Graph Network

hukuda222

December 3, 2021

Introduction

Methodology

Experiment

Conclusion

Introduction

Task Extract summary from documents.

Contribution

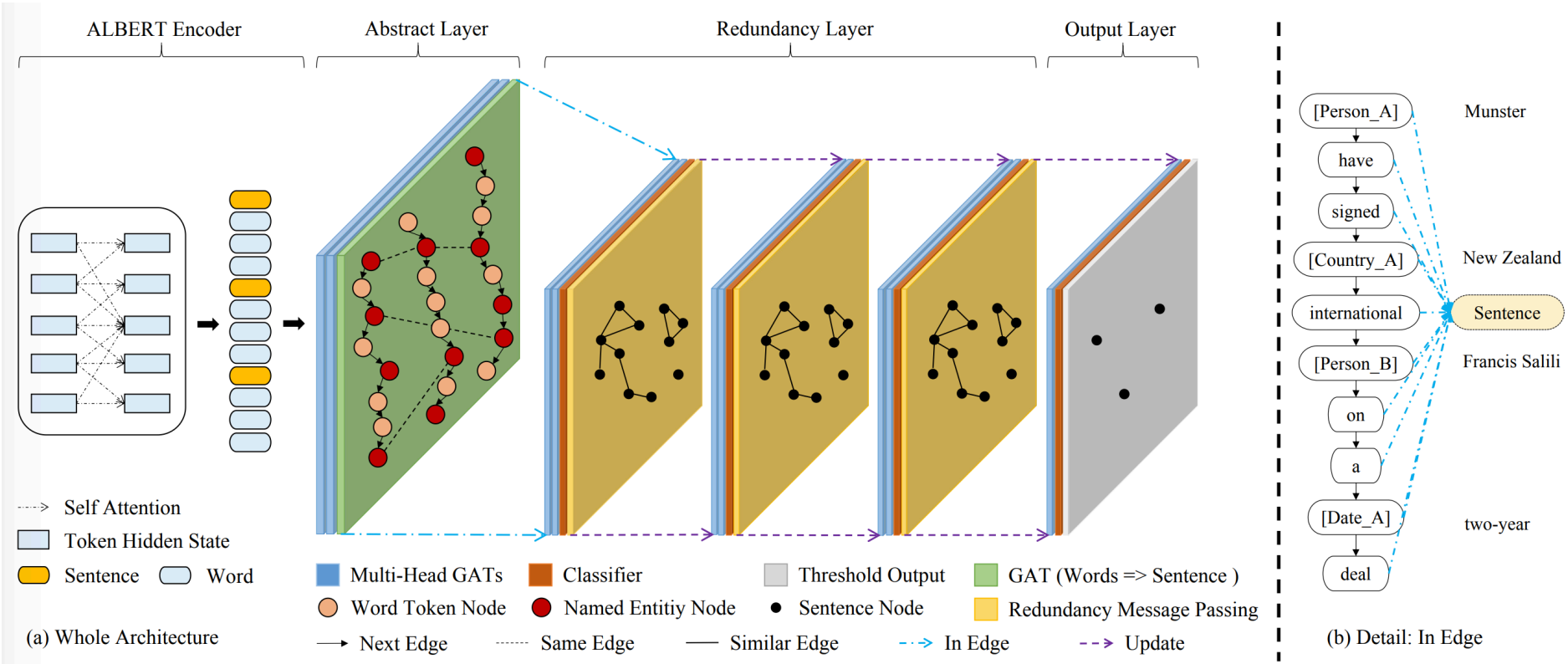
- They propose a hierarchical attentive heterogeneous graph based model(HAHSum) to guide the redundancy information.
- Thier architecture is able to extract flexible quantity of sentences with a threshold

Problem Definition

- $S = \{s_1, s_2, ..., s_N\}$ is a document consisting of N sentences.
- $S* = \{s*_1, s*_2, ..., s*_M \,|\, s_i \in S\}$ is summary including M sentences extracted from a document.
- $Y = \{y_1, y_2, ..., y_N\}$ are derived from S, where $y_i \in \{0, 1\}$ denotes whether sentence s_i should be included in the extracted summary.

Overview

They proposed HAHSum (as shorthand for **H**ierarchical **A**ttentive **H**eterogeneous Graph for Text **Sum**marization).



Edges

In this method, there are three types of nodes: **named entity**, **word**, and **sentence**. They also define four types of edges to represent various structural information:

Next connect sequential named entities and words in one sentence

In connect one named entity node or word node to one sentence node if the named entity or word occurs in this sentence

Same connect the same named entity nodes

Next connect two sentence nodes if they have trigram overlapping

Adjacency matrixs

HAHsum contains multi-granularity levels of information, it can be divided into 3 subgraphs:

A_{word} is used for the word-level graph, constructed by Entity node, Word node, Next edge and Same edge

$A_{word-sent}$ is used for the word-sentence graph, constructed by three types of nodes and In edge

A_{sent} is used for sentence-level graph, constructed by Sentence node and Similar edge

ALBERT Encoder

- almost the same as BERTSUMEXT(Liu and Lapata, 2019)
- in order to accurately use these hidden states to represent each word, we apply an average pooling function to the output for each subwords
- input documents are truncted to 768 BPE tokens
- use pre-trained 'alber-xxlarge-v2'()

Abstract Layer

GAT:Graph Attention Network
W:the hidden state of the nodes S_{abs}:the initial representation of sentence nodes

$$W = GAT(GAT(h^{word}, A_{word}), A_{word})$$
$$[W, S_{abs}] = GAT([W, h^{sent}], A_{word-sent})$$

Redundancy Layer

This layer only deals with sentence-level information $S = h_1, h_2, ..., h_N$ and iteratively updates it L times with classification scores:

$$\tilde{S}_{re}^{l} = GAT(GAT(S_{re}^{l}, A_{sent}), A_{sent})$$

$$P(y_i = 1|\tilde{S}_{re}^{l}) = \sigma(FFN(LN(\tilde{h}_i^{l} + MHAtt(\tilde{h}_i^{l}))))$$

$S_{re}^{0} = S_{abs}$and they get S_{re}^{L} at the end.

Redundancy Layer

They update $\tilde{h}_i^l$ by reducing the redundancy information g_i^l, which is the weighted summation of neighbors information:

$$g_i^l = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} P(y_j = 1 | \tilde{S}_{re}^l) * \tilde{h}_j^l$$

$$h_i^{l+1} = W_c^l * \tilde{h}_i^l - \tilde{h}_i^{lT} W_r^l tanh(g_i^l)$$

$$S_{re}^{l+1} = (h_1^{l+1}, h_2^{l+1}, .., h_{|S|}^{l+1})$$

$\mathcal{N}_i$ is redundancy receptive field for node i, according to A_{sent}.

Redundancy Layer

Specifically, they update $\tilde{h}_i^l$ by reducing the redundancy information g_i^l, which is the weighted summation of neighbors information:

$$\tilde{h}_i^{l'} = W_c^* \tilde{h}_i^l - \tilde{h}_i^{lT} W_r^l tanh(g_i^l)$$
$$p_g^l = \sigma(f_g^l([\tilde{h}_i^{\ l}; \tilde{h}_i^{l'}]))$$
$$h_i^{l+1} = \tilde{h}_i^l \odot p_g^l + \tilde{h}_i^{l'} \odot (1 - p_g^l)$$

$\odot$ denotes element-wise multiplication.

Objective Function

$$P(Y|S) = \Pi_{t=1}^{|S|} P(y_t|S, \mathcal{S}_{abs}, \mathcal{S}_{re})$$

$$\mathcal{L} = -\sum_{t=0}^{L} \frac{L+l}{2L} \sum_{i=1}^{|S|} \{y_i log P(\hat{y}_i|\tilde{S}_{re}^{l}) + (1-y_i) log(1 - P(\hat{y}_i|\tilde{S}_{re}^{l}))\} \quad (1)$$

Experiment

Models	CNN/DM			NYT			Newsroom (Ext)		
	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L
Abstractive									
ABS (2015)	35.46	13.30	32.65	42.78	25.61	35.26	6.1	0.2	5.4
PGC (2017)	39.53	17.28	36.38	43.93	26.85	38.67	39.1	27.9	36.2
TransformerABS (2017)	40.21	17.76	37.09	45.36	27.34	39.53	40.3	28.7	36.5
$\text{MASS}_{\text{Large}}$ (2019)	43.05	20.02	40.08	-	-	-	-	-	-
$\text{UniLM}_{\text{Large}}$ (2019)	43.33	20.21	40.51	-	-	-	-	-	-
$\text{BART}_{\text{Large}}$ (2019)	44.16	21.28	40.90	48.73	29.25	44.48	-	-	-
$\text{PEGASUS}_{\text{Large}}$ (2019a)	44.17	21.47	41.11	-	-	-	-	-	-
$\text{ProphetNet}_{\text{Large}}$ (2020)	44.20	21.17	41.30	-	-	-	-	-	-
Extractive									
Oracle	55.61	32.84	51.88	64.22	44.57	57.27	-	-	-
Lead	40.42	17.62	36.67	41.80	22.60	35.00	53.1	49.0	52.4
SummaRuNNer (2017)	39.60	16.20	35.30	42.37	23.89	38.74	48.96	44.33	49.57
Exconsumm (2019)	41.7	18.6	37.8	43.18	24.43	38.92	68.4	62.9	67.3
$\text{PNBERT}_{\text{Base}}$ (2019a)	42.69	19.60	38.85	-	-	-	-	-	-
$\text{BERTSUMEXT}_{\text{Large}}$ (2019)	43.85	20.34	39.90	48.51	30.27	44.65	70.85	67.03	69.61
$\text{MATCHSUM}_{\text{Base}}$ (2020)	44.41	20.86	40.55	-	-	-	-	-	-
$\text{HAHSum}_{\text{Large}}$(Ours)	**44.68**	**21.30**	**40.75**	**49.36**	**31.41**	**44.97**	**71.31**	**68.75**	**70.83**

Ablation Studies

They propose several strategies to improve the performance by relieving the semantic sparsity and redundancy bias, including abstract layer(AL), the iterative redundancy layer(RL), and pre-trained ALBERT.

Models	R-1	R-2	R-L
HAHSum	**44.68**	**21.30**	**40.75**
w/o AL	44.35	20.98	40.49
w/o RL	44.49	21.11	40.58
w/o ALBERT	44.57	21.14	40.53

Human Evaluation

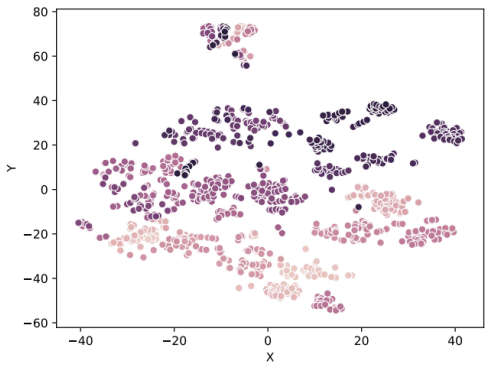
taking informativeness (Can the summary capture the important information from the document) and fluency (Is the summary grammatical) into account.

Models	1st	2nd	3rd	4th	5th	MeanR
SummaRuNNer	0.14	0.27	0.24	0.22	0.13	2.93
BERTSUMEXT	0.20	0.28	0.31	0.16	0.05	2.58
MATCHSUM	0.24	0.36	0.16	0.15	0.09	2.49
HAHSum	0.45	0.34	0.18	0.03	0.00	2.24
Ground-Truth	0.70	0.21	0.05	0.04	0.00	1.43

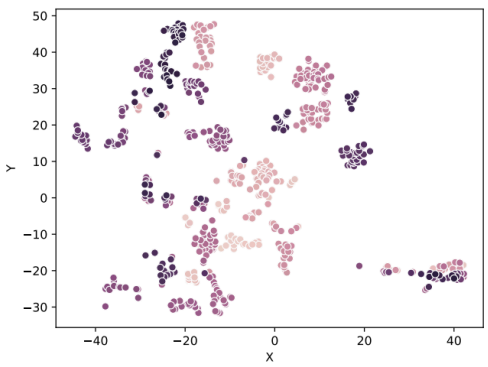
Table 5: Human evaluation on Daily Mail.

Visualization

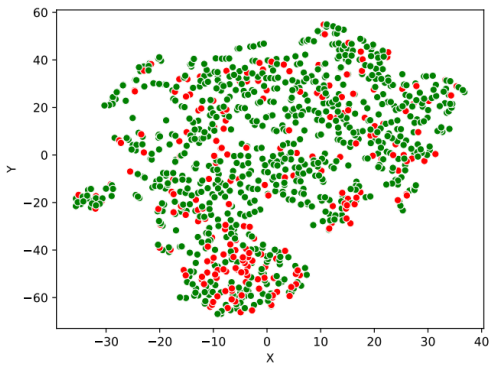
- the darkness determines it's position in one document
- red points are the sentences with label 1, and green points are with label 0



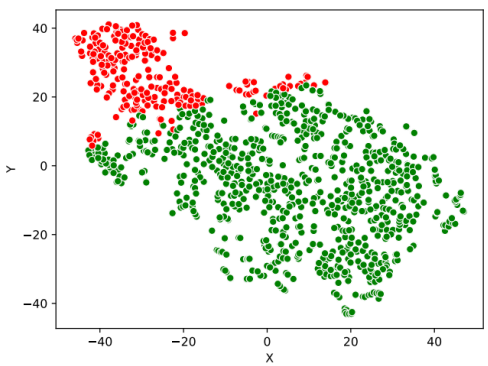
(a) Word Nodes/BERTSUM



(b) Word Nodes/HAHSum



(c) Sent Nodes/BERTSUM



(d) Sent Nodes/HAHSum

Conclusion

- they propose hierarchical attentive heterogeneous graph, aiming to advance text summarization by measuring salience and redundancy simultaneously
- HAHSum produces more focused summaries with fewer superfluous and the performance improvements are more pronounced on more extractive datasets