COLING 2022

NAVER AI LAB

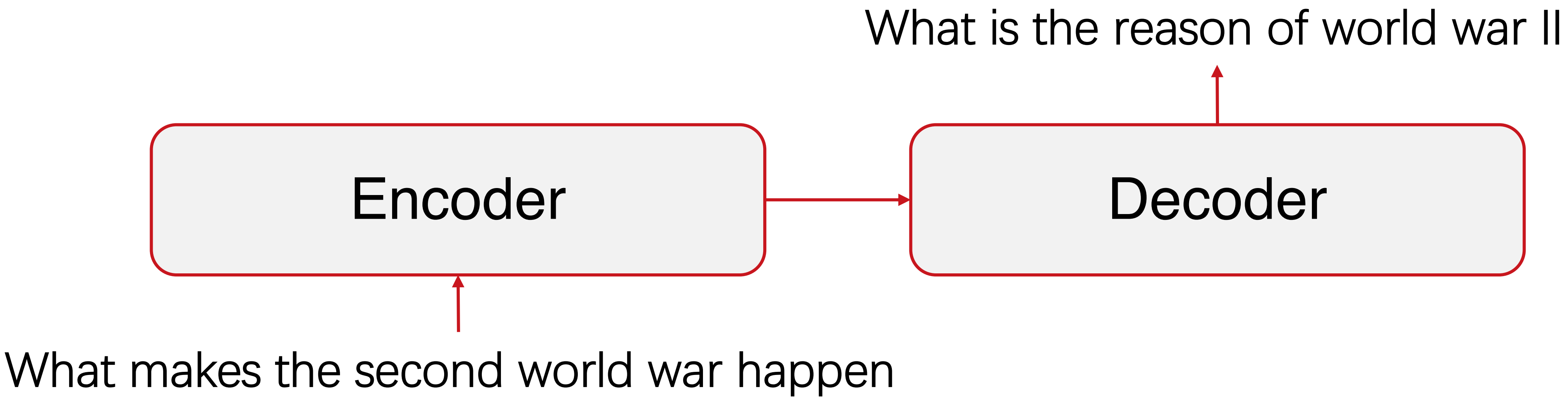
Continuous Decomposition of Granularity for Neural Paraphrase Generation

- Xiaodong Gu[1], Zhaowei Zhang[1], Kang Min Yoo[2], Sang-Woo Lee[2], Jung-Woo Ha[2]

[1]School of Software, Shanghai Jiao Tong University
[2]NAVER AI Lab

Neural Paraphrase Generation

- Given a source sentence x, generate a paraphrase y.
- **Existing approaches**: sequence-to-sequence learning (**Transformers**)



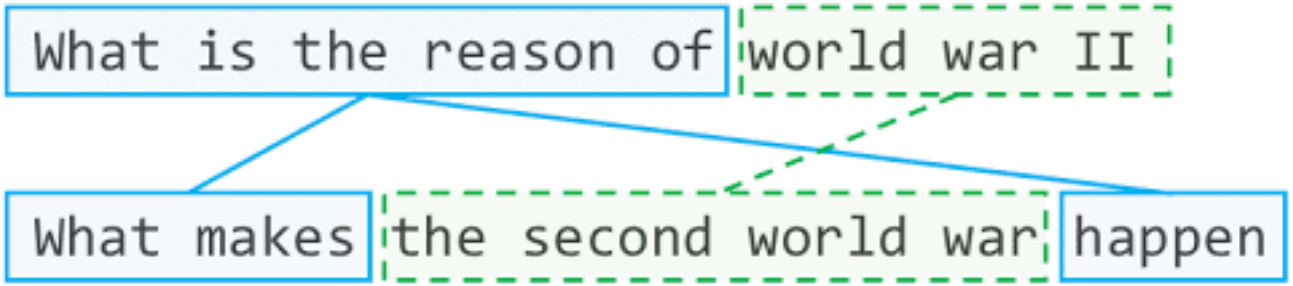
Limitations: **Transformers treat a sentence as a flat sequence of words**

SHANGHAI JIAO TONG UNIVERSITY

NAVER AI LAB

DNPG: Decomposable Neural Paraphrase Generation

- Paraphrase exists in different levels of granularity **[Li et al. ACL' 19]**
  - **Sentential Level**: abstractive, general
  - **Phrasal Level**: diverse, domain-specific



Sentential level: what is the reason of $x → what makes $x happen

Phrasal level: world war II → the second world war

Zichao Li, Xin Jiang, Lifeng Shang, Qun Liu. Decomposable Neural Paraphrase Generation. In ACL 2019.

Examples [Li et al. ACL' 19]

Templates (Sentential Level)

Details (Phrasal Level)

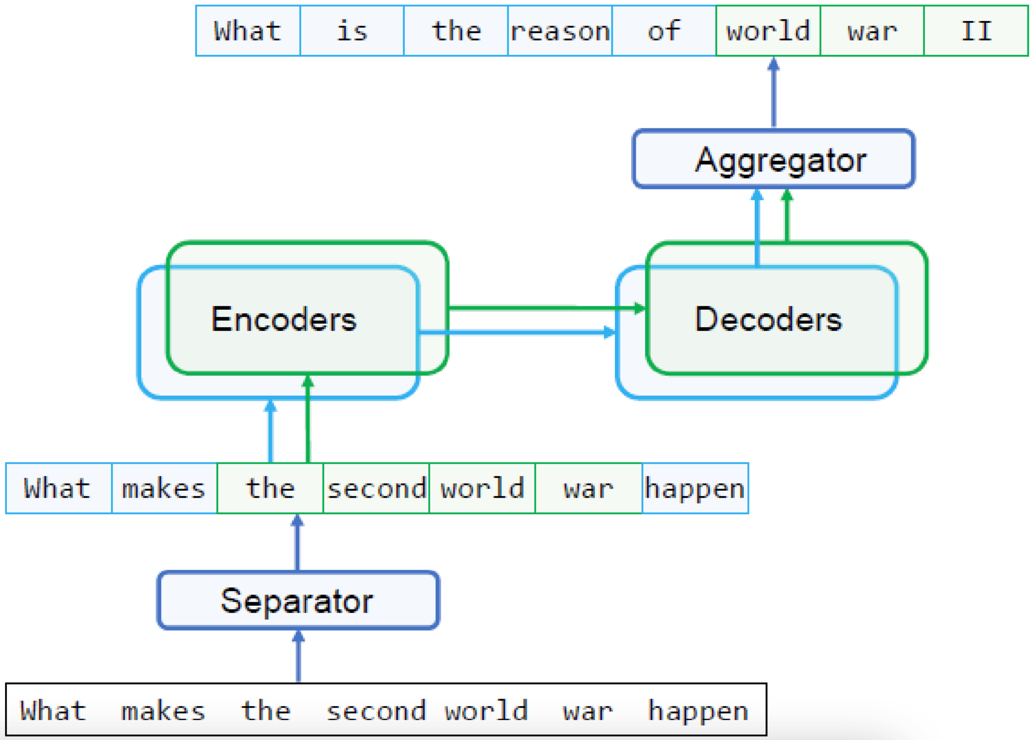
What is the population of New York*?* *How many people is there in* NYC*?*
Who wrote the Winnie the Pooh books*?* *Who is the author of* winnie the pooh*?*
What is the best phone *to buy below* 15k*?* *Which are* best mobile phones *to buy under* 15000*?*
How can I be a good geologist*?* *What should I do to be* a great geologist*?*
How do I reword a sentence *to* avoid plagiarism*?* *How can I* paraphrase my essay *and* avoid plagiarism*?*

Shanghai Jiao Tong University

NAVER AI LAB

DNPG: Decomposable Neural Paraphrase Generation

- **Separator**: classifies each token into templates (z=0) and details (z=1)
- Each class is feed into a **individual encoder** and **decoder**.
- **Aggregator**: the final predictions are aggregated into the final prediction



Problems:

- Binary/discrete granularity
- High computational cost due to multiple encoder-decoder pairs.

[Li et al. ACL’ 19]

On Multiple Levels of Granularity

- There are many ways of decomposing a sentence, corresponding to **multiple levels** of granularity.
- **Numerical** representation of granularity for each token:

Text	What is the reason for World War II?
Decomposition 1	What is the reason for world war II?
Decomposition 2	What is the reason for world war II?
Decomposition 3	What is the reason for world war II?
Decomposition 4	What is the reason for world war II?
Decomposition 5	What is the reason for world war II?

Levels of granularity (marked as superscripts):

What^{1} is^{1} the^{2} reason^{3} of^{2} World^{4} War^{4} II^{5} ?

Continuous granularity?

Our Idea

Continuous Decomposition of Granularity

0~1 granularity for each token

Paraphrasing tokens of similar granularity

0.1	0.15	0.2	0.8	0.6	0.5	0.3
What	makes	the	second	world	war	happen

Decoder

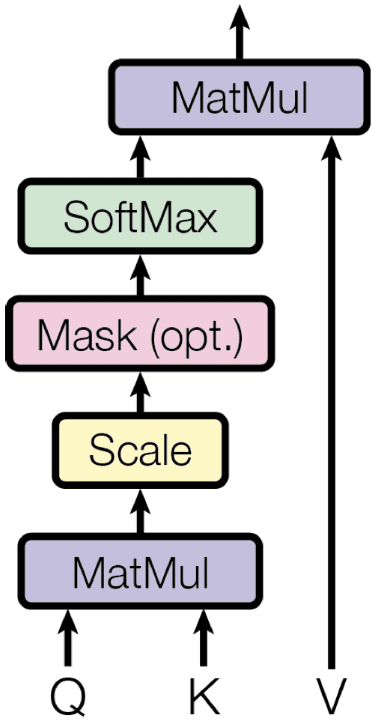
Encoder

What	is	the	reason	of	world	war	II
0	0	0.2	0.3	0.2	0.5	0.5	0.8

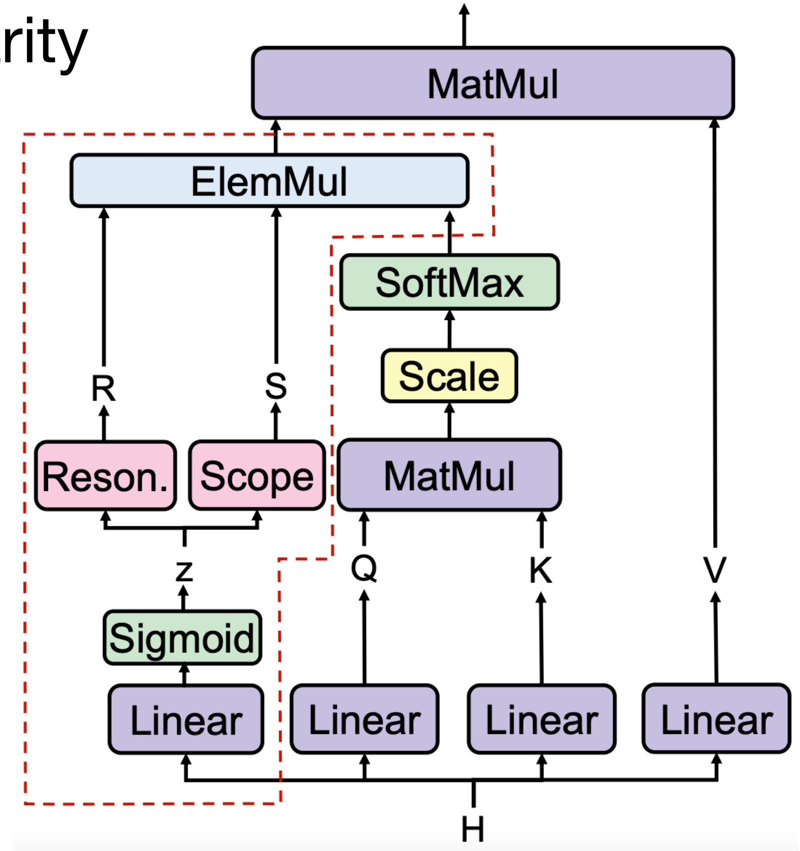
Granularity-Aware Self-Attention

- An attention header that predicts the continuous granularity level [0,1]
- Two attention masks that integrates the granularity

1. Granularity head
2. Resonance mask
3. Granularity scope mask



Vanilla Attention



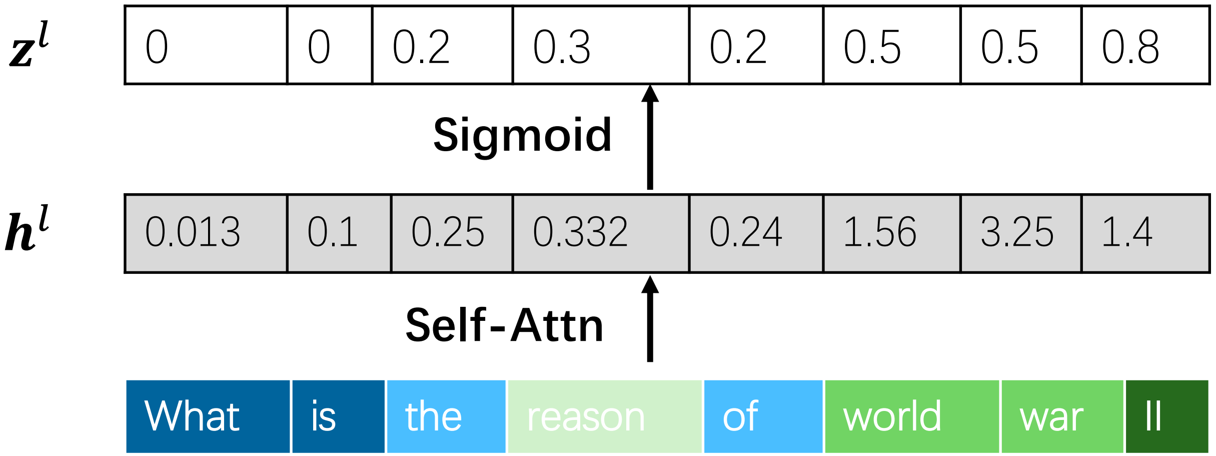
Granularity-Aware Attention

Granularity Head

- An extension of the self-attention head.
- Let $z_i \in [0,1]$ denote the granularity of tokens i,

$$z_i = \text{sigmoid}\,(\mathbf{w}^{\mathrm{T}}\boldsymbol{h}_i)$$

- For layer ℓ, $\mathbf{z}^l = \text{sigmoid}(\mathbf{W}^G\mathbf{H}^{l-1}), l = 2, \ldots, L$



Granularity Resonance Mask

- Tokens of the same level of granularity have the strongest correlation.
- The correlation between token i and j:

$$\mathbf{C}_{ij} = \begin{cases} 1, & \text{if } z_i = z_j \\ 0, & \text{otherwise} \end{cases}$$

In the **binary case** where z_i, $z_j \in \{0, 1\}$

$$\mathbf{C}_{ij} = (1 - z_i) \times \max(0, 1 - (z_i + z_j)) + z_i \times \min(1, 1 - z_i + z_j)$$

Continuous version where z_i, $z_j \in [0, 1]$

Granularity Scope Mask

- Neighboring tokens gain more attention than distant tokens.
- The correlation between tokens i and j :

$$\mathbf{S}_{ij} = \begin{cases} 1 & \text{if } |i-j| < (N-\epsilon)^{(1-z_i)} + \epsilon \\ 0 & \text{otherwise} \end{cases}$$

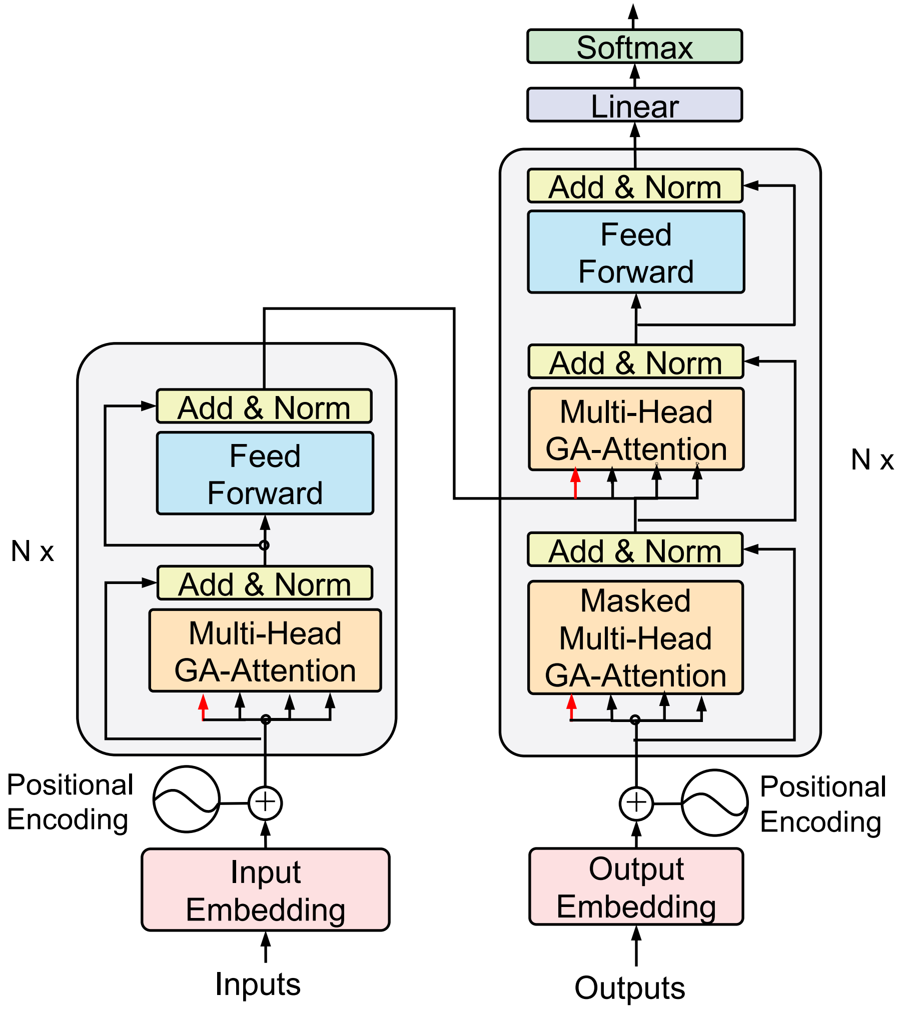
In the **binary case** where z_i, $z_j \in \{0, 1\}$

$$\mathbf{S}_{ij} = \max(0, \min(1, (N-\epsilon)^{(1-z_i)} + \epsilon - |i-j|))$$

Continuous version where z_i, $z_j \in [0, 1]$

Overall Architecture

Simply replacing the attention module with the proposed GA-Attention module.



Experimental Setup

- Datasets

Language	Train	Valid	Test
Quora Question Pairs	100,000	4,000	20,000
Twitter URLs	110,000	5,000	1,000

- Metrics

iBLEU BLEU-2 BLEU-4 ROUGE-L METEOR

Experimental Setup

- **Baselines**
  - RedidualLSTM (Prakash et al., 2016): an LSTM sequence-to-sequence model using residuals between RNN layers;
  - PointerGenerator (See et al., 2017): RNN seq2seq using copy mechanism;
  - Transformer (Vaswani et al., 2017): the vanilla Transformer model;
  - Transformer+Copy: an enhanced Transformer with copy mechanism (Gu et al., 2016); and
  - DNPG (Li et al., 2019): a popular paraphrase generation model based on Transformer.

Experimental Results

- Automatic Evaluation

Model	Quora					Twitter URL				
	iBLEU	BLEU-2	BLEU-4	ROUGE-L	METEOR	iBLEU	BLEU-2	BLEU-4	ROUGE-L	METEOR
ResidualLSTM	20.45	40.71	26.20	36.19	32.67	20.29	36.75	25.92	32.47	29.44
Pointer-generator	22.65	43.82	28.80	42.36	40.87	25.60	44.50	32.40	38.48	36.48
Transformer	21.14	37.97	26.88	40.14	38.21	24.44	44.45	31.12	31.97	32.49
Transformer+Copy	22.90	44.42	28.94	37.60	38.34	27.07	48.44	34.35	38.37	38.19
DNPG	24.55	47.72	31.01	42.37	42.12	25.92	46.36	32.91	36.77	36.28
FSET	-	**51.03**	33.46	-	38.57	-	46.35	34.62	-	31.67
C-DNPG (R)	**26.94**	47.58	**34.05**	46.17	44.75	27.96	49.98	35.80	38.67	39.39
C-DNPG (S)	26.68	47.48	33.93	**46.22**	**46.66**	28.19	49.10	35.95	38.89	39.06
C-DNPG (R⊙S)	25.96	46.25	33.02	44.64	44.25	**30.25**	49.00	**38.58**	**41.60**	**41.71**
C-DNPG (R+S)	26.66	50.96	33.69	44.45	43.33	28.73	**50.49**	36.61	39.80	40.42

C-DNPG achieves the state-of-the-art results in terms of many metrics.

Experimental Results

- Qualitative Analysis

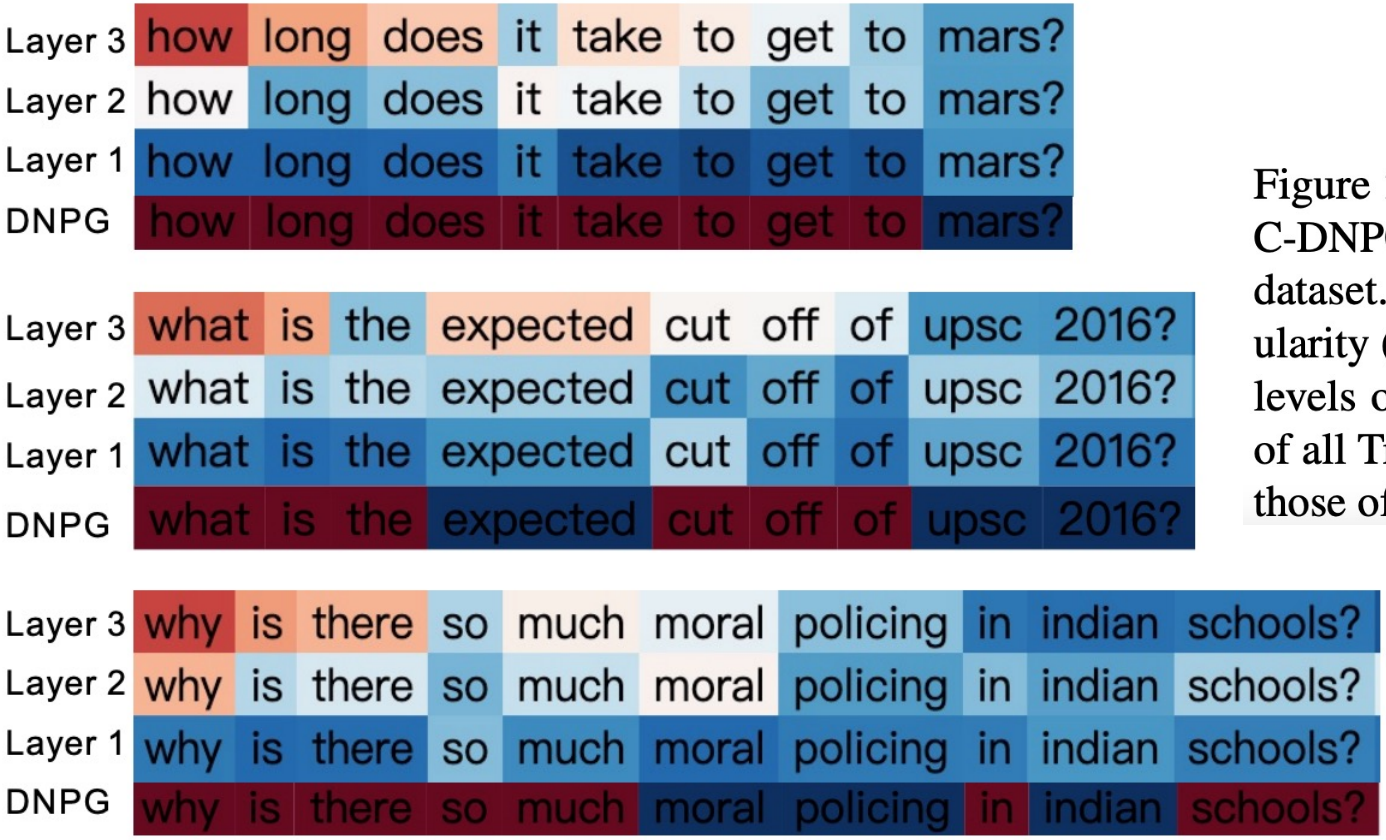


Figure 2: Examples of multi-granularity extracted by C-DNPG (Layer1-3) and DNPG (bottom) on the Quora dataset. Warmer colors represent higher levels of granularity (templates) while colder colors represent lower levels of granularity (details). We present granularity of all Transformer layers and compare the results with those of DNPG.

Experimental Results

- Case Study

Sentence:	What is a good first programming language?
Transformer:	What is good?
DNPG:	What is good for coding?
C-DNPG:	What are the best programming languages for beginners?
Human:	Whats a good and easy programming language to learn?

Sentence:	What will the year 2100 be like?
Transformer:	What is likely to happen in the world?
DNPG:	What are did today. year - year of unique year of country?
C-DNPG:	What will the world look like in 2100?
Human:	What will the year 2099 be like?

Conclusion

C-DNPG – continuous decomposition of granularity for neural paraphrase generation.

- Extending self-attention with a granularity head
- Two novel masks that incorporates granularity into self-attention.

Future Work

- PLMs

Thank You!

Q&A