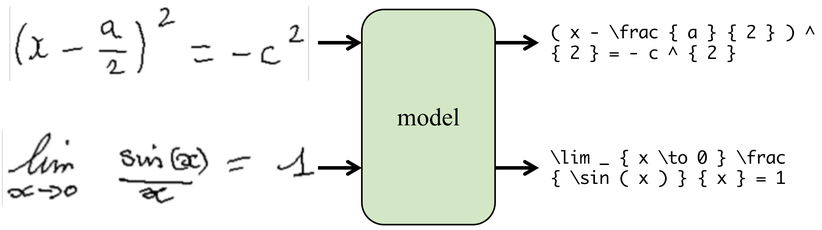
CoMER: Modeling Coverage for Transformer-based Handwritten Mathematical Expression Recognition

Wenqi Zhao, Liangcai Gao
Wangxuan Institute of Computer Technology, Peking University, Beijing, China

ECCV TEL AVIV 2022

1. Introduction

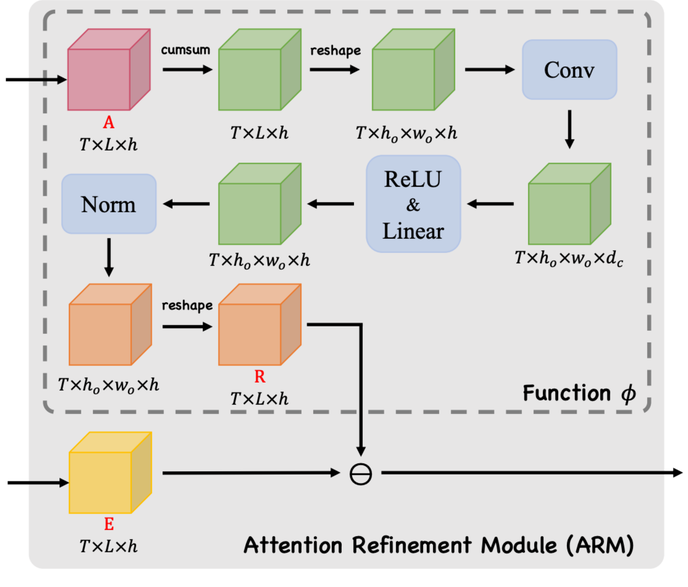
- **Goal:** Generate LaTeX sequence according to the handwritten math expression image.



- **Problem:** Existing handwritten math expression recognition methods suffer from **lack of coverage** problem.
- **Contribution:** We propose a novel Attention Refinement Module (**ARM**), which effectively alleviates the lack of coverage problem without hurting its parallelism.

2. Attention Refinement Module

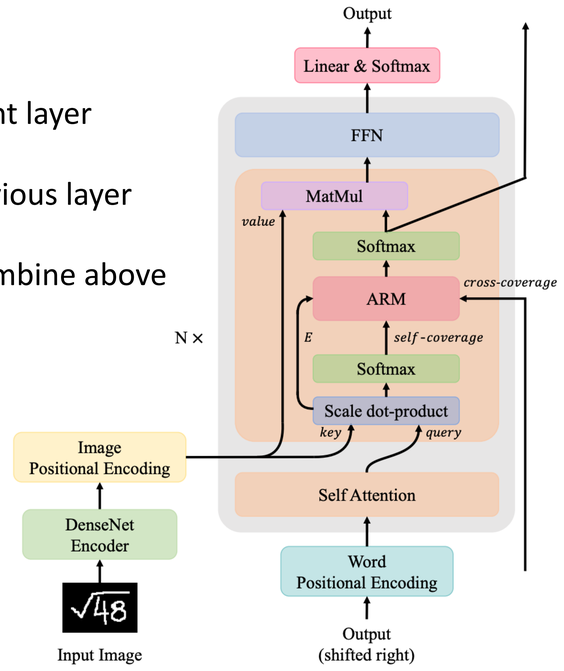
In attention refinement module, We refine the attention term E by the coverage attention weight A.



3. Coverage Mechanism

- **Self-coverage:** current layer
- **Cross-coverage:** previous layer
- **Fusion-coverage:** combine above

The architecture of CoMER model.



4. Comparison

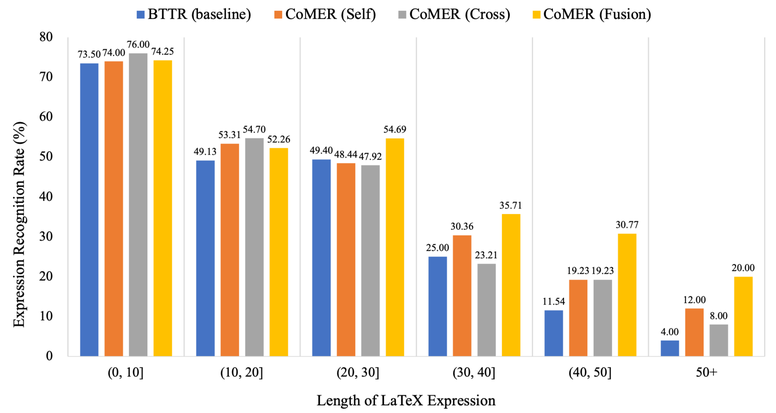
Our CoMER model obtains new SOTA performance on various CROHME datasets.

Performance comparison on the CROHME datasets.

Dataset	Model	ExpRate	≤ 1 error	≤ 2 error	≤ 3 error
CROHME 14	DenseWAP [32]	43.0	57.8	61.9	-
	DenseWAP-TD [33]	49.1	64.2	67.8	-
	WS-WAP [26]	53.65	-	-	-
	Li *et al.* [15]	56.59	69.07	75.25	**78.60**
	Ding *et al.* [9]	58.72	-	-	-
	BTTR [35]	53.96	66.02	70.28	-
	BTTR (baseline)	55.17	67.85	72.11	74.14
	CoMER	**59.33**	**71.70**	**75.66**	77.89
CROHME 16	DenseWAP [32]	40.1	54.3	57.8	-
	DenseWAP-TD [33]	48.5	62.3	65.3	-
	WS-WAP [26]	51.96	64.34	70.10	72.97
	Li *et al.* [15]	54.58	69.31	73.76	76.02
	Ding *et al.* [9]	57.72	70.01	76.37	78.90
	BTTR [35]	52.31	63.90	68.61	-
	BTTR (baseline)	56.58	68.88	74.19	76.90
	CoMER	**59.81**	**74.37**	**80.30**	**82.56**
CROHME 19	DenseWAP [32]	41.7	55.5	59.3	-
	DenseWAP-TD [33]	51.4	66.1	69.1	-
	Ding *et al.* [9]	61.38	75.15	80.23	82.65
	BTTR [35]	52.96	65.97	69.14	-
	BTTR (baseline)	59.55	72.23	76.06	78.40
	CoMER	**62.97**	**77.40**	**81.40**	**83.07**

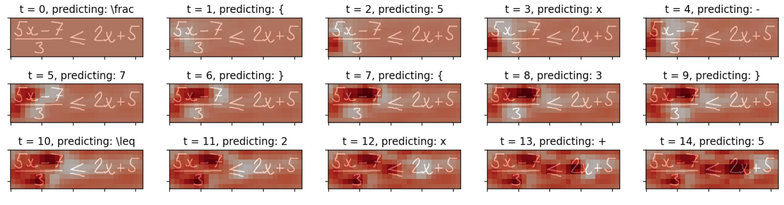
5. Performance at Different Lengths

Our CoMER has better performance when dealing with various lengths of sequences, especially with longer ones.



6. Refinement Term Visualization

Parsed regions are darker, which indicates ARM will suppress the attention weights in these parsed regions.



7. Conclusion

- ARM was proposed to perform attention refinement in the transformer without harming its parallel computing nature.
- We propose self-coverage and cross-coverage to refine the attention weights using the past alignment information from the current and previous layers.

Paper, code, and data are available:
https://github.com/Green-Wood/CoMER