Is Attention Better Than Matrix Decomposition?

Zhengyang Geng[1,2], Meng-Hao Guo[3,*],

Hongxu Chen[4], Xia Li[2], Ke Wei[4], Zhouchen Lin[2,5,$]

[1]Zhejiang Lab;

[2]Key Lab. of Machine Perception (MoE), School of EECS, Peking University;

[3]Tsinghua University; [4]School of Data Science, Fudan University; [5]Pazhou Lab;

*Equal First Authorship; [$] Corresponding Author

ICLR 2021

Motivation

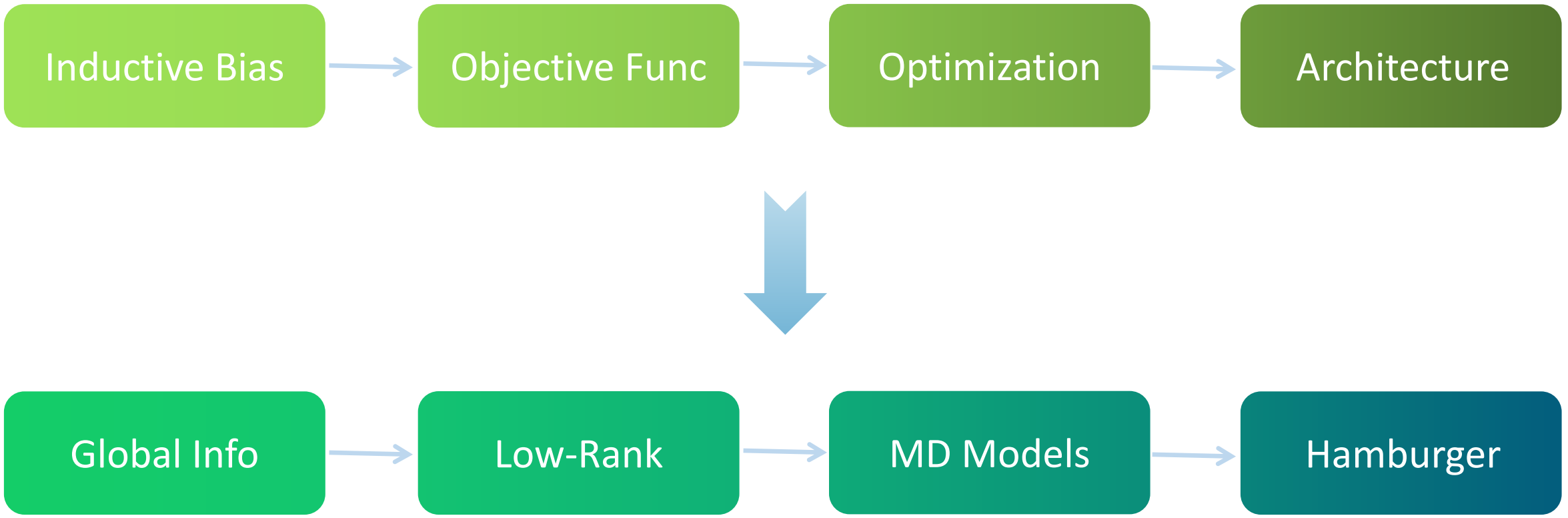
1. When we talk about global information/long-range correlation in attention-related methods, what do we actually mean?

Model the global information in a first-principle way.

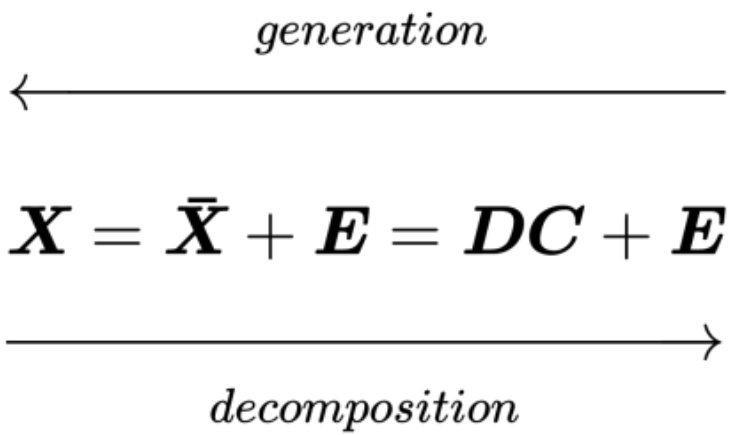
2. Is hand-crafted attention irreplaceable when modeling the global context?

Provide a strong white-box baseline Hamburger for attention mechanism.

Key Ideas


Inductive Bias
Objective Func
Optimization
Architecture
Global Info
Low-Rank
MD Models
Hamburger

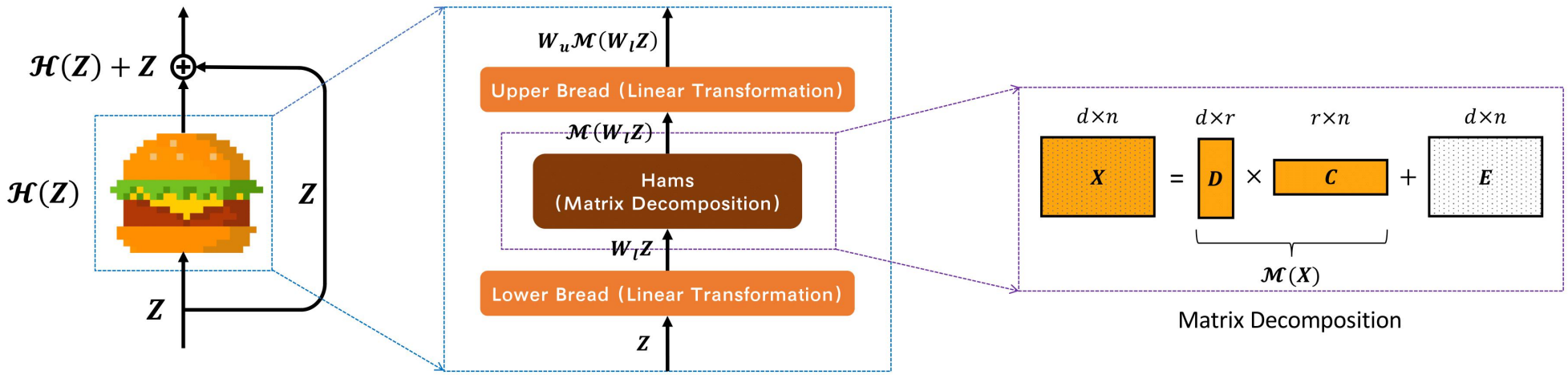
Matrix Decomposition



$$\operatorname{rank}(\bar{\boldsymbol{X}}) \leq \min(\operatorname{rank}(\boldsymbol{D}), \operatorname{rank}(\boldsymbol{C})) \leq r \ll \min(d, n).$$

1. Inverse of Generation
2. Low-Rankness
3. Optimization

Hamburger



$$\min_{D,C} \mathcal{L}(\boldsymbol{X}, \boldsymbol{DC}) + \mathcal{R}_1(\boldsymbol{D}) + \mathcal{R}_2(\boldsymbol{C})$$

- **Learn the global information <=> Solve the the optimization of MD**
- Denote the optimization algorithm to solve the problem as $\mathcal{M}$. $\mathcal{M}$ is the core architection of Hamburger.

Ham

VQ

$$\min_{\boldsymbol{D},\boldsymbol{C}} \|\boldsymbol{X}-\boldsymbol{D}\boldsymbol{C}\|_F \quad \text{s.t. } \mathbf{c}_i \in \{\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_r\}$$

Algorithm 1 Ham: Soft VQ

Input $\boldsymbol{X}$. Initialize $\boldsymbol{D}, \boldsymbol{C}$.
for k from 1 to K **do**
  $\boldsymbol{C} \leftarrow softmax(\frac{1}{T} cosine(\boldsymbol{D}, \boldsymbol{X}))$
  $\boldsymbol{D} \leftarrow \boldsymbol{X}\boldsymbol{C}^\top diag(\boldsymbol{C}\mathbf{1}_n)^{-1}$
end for
Output $\bar{\boldsymbol{X}} = \boldsymbol{D}\boldsymbol{C}$.

NMF

$$\min_{\boldsymbol{D},\boldsymbol{C}} \|\boldsymbol{X}-\boldsymbol{D}\boldsymbol{C}\|_F \quad \text{s.t. } \boldsymbol{D}_{ij} \geq 0, \boldsymbol{C}_{jk} \geq 0$$

Algorithm 2 Ham: NMF with MU

Input $\boldsymbol{X}$. Initialize non-negative $\boldsymbol{D}, \boldsymbol{C}$
for k from 1 to K **do**
  $\boldsymbol{C}_{ij} \leftarrow \boldsymbol{C}_{ij} \frac{(\boldsymbol{D}^\top \boldsymbol{X})_{ij}}{(\boldsymbol{D}^\top \boldsymbol{D}\boldsymbol{C})_{ij}}$
  $\boldsymbol{D}_{ij} \leftarrow \boldsymbol{D}_{ij} \frac{(\boldsymbol{X}\boldsymbol{C}^\top)_{ij}}{(\boldsymbol{D}\boldsymbol{C}\boldsymbol{C}^\top)_{ij}}$
end for
Output $\bar{\boldsymbol{X}} = \boldsymbol{D}\boldsymbol{C}$.

One-step Grad

- We build an abstract model to analyse the grad back through $\mathcal{M}$.

$$\mathbf{h}^{t+1} = \mathcal{F}(\mathbf{h}^t, \mathbf{x})$$

- We find the thorny **scale** and **spectrum** of the Jacobian matrix in the BPTT algorithm.
- Back-propagation through optimization suffers from **vanishing** gradient *w.r.t.* the initialization and **exploding** gradient *w.r.t.* the input!
- We use **the one-step grad** $\widehat{\frac{\partial \mathbf{y}}{\partial \mathbf{x}}}$, i.e., the grad from the last optimization step.

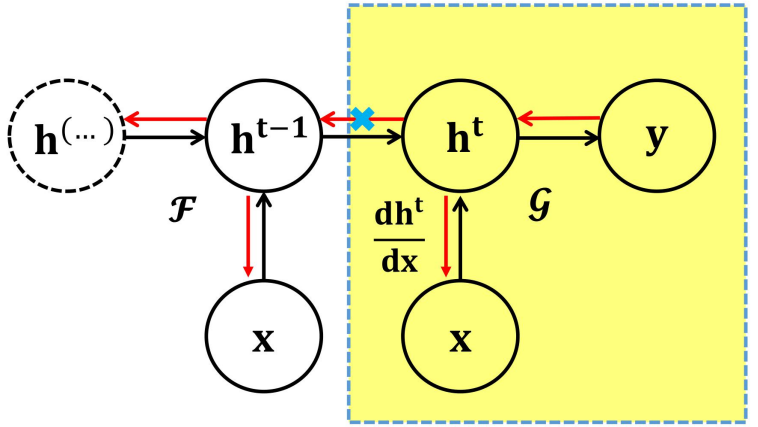
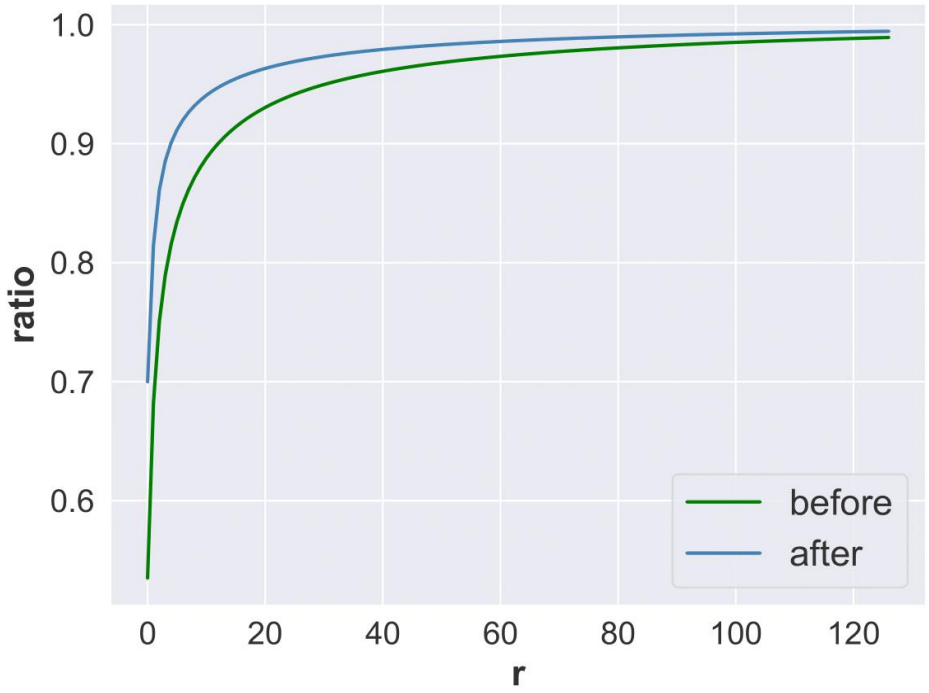


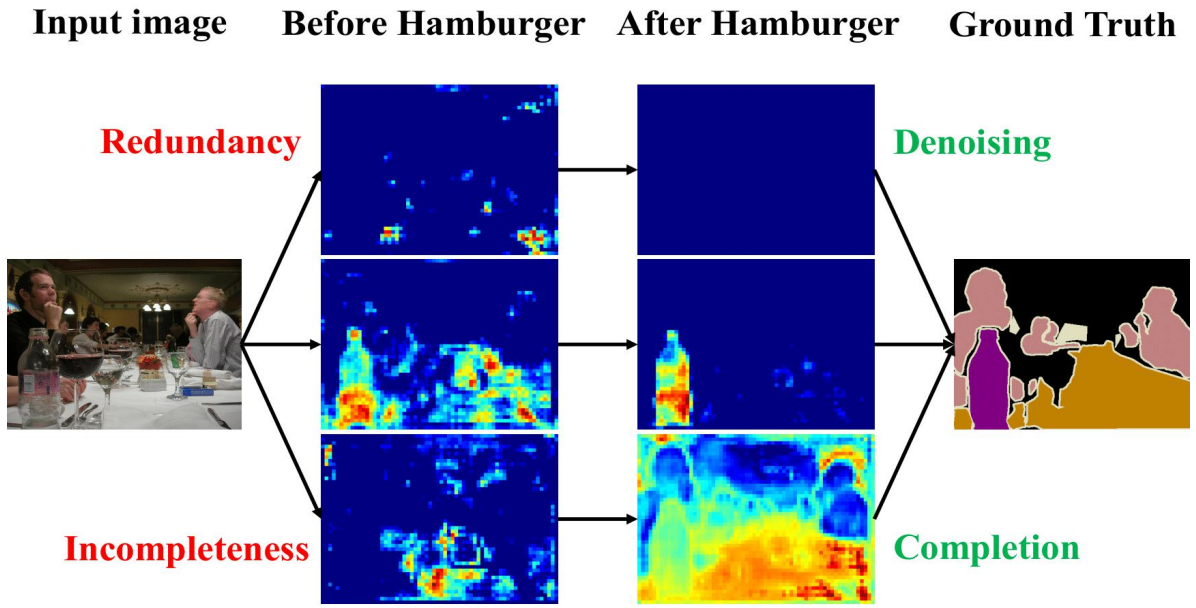
Table 1: One-step Gradient & BPTT

Method	One-step	BPTT
VQ	77.7(77.4)	76.6(76.3)
CD	78.1(77.5)	75.0(74.6)
NMF	78.3(77.8)	77.4(77.0)

A Close Look at Hamburger



Accumulative Ratio



Visualization of Feature Maps

Computation & Memory

Table 3: Comparisons between Hamburger and context modules.

Method	Params	MACs	GPU Load		GPU Time	
			Train	Infer	Train	Infer
SA	1.00M	292G	5253MB	2148MB	242.0ms	82.2ms
DA	4.82M	79.5G	2395MB	2203MB	72.6ms	64.4ms
A^2	1.01M	25.7G	326MB	165MB	22.9ms	8.0ms
APC	2.03M	17.6G	458MB	264MB	26.5ms	11.6ms
DM	3.00M	35.1G	557MB	268MB	65.7ms	23.3ms
ACF	0.75M	79.5G	1380MB	627MB	71.0ms	22.6ms
Ham (CD)	0.50M	**16.2G**	**162MB**	102MB	20.0ms	13.0ms
Ham (NMF)	0.50M	17.6G	202MB	**98MB**	**15.6ms**	**7.7ms**

Performance

SOTA on Semantic Segmentation

Table 4: Comparisons with state-of-the-art on the PASCAL VOC test set w/o COCO pretraining.

Method	mIoU(%)
PSPNet (Zhao et al., 2017)	82.6
DFN* (Yu et al., 2018)	82.7
EncNet (Zhang et al., 2018)	82.9
DANet* (Fu et al., 2019)	82.6
DMNet* (He et al., 2019a)	84.4
APCNet* (He et al., 2019b)	84.2
CFNet* (Zhang et al., 2019b)	84.2
SpyGR* (Li et al., 2020)	84.2
SANet* (Zhong et al., 2020)	83.2
OCR* (Yuan et al., 2020)	84.3
HamNet	**85.9**

Table 5: Results on the PASCAL-Context Val set.

Method	mIoU(%)
PSPNet (Zhao et al., 2017)	47.8
SGR* (Liang et al., 2018)	50.8
EncNet (Zhang et al., 2018)	51.7
DANet* (Fu et al., 2019)	52.6
EMANet* (Li et al., 2019)	53.1
DMNet* (He et al., 2019a)	54.4
APCNet* (He et al., 2019b)	54.7
CFNet* (Zhang et al., 2019b)	54.0
SpyGR* (Li et al., 2020)	52.8
SANet* (Zhong et al., 2020)	53.0
OCR* (Yuan et al., 2020)	54.8
HamNet	**55.2**

Large Scale Image Generation

Table 6: Results on ImageNet 128×128. * are from Tab. 1 and Tab. 2 of Zhang et al. (2019a).

Method	FID↓
SNGAN-projection*	27.62
SAGAN*	18.28
HamGAN-baby	16.05
YLG	15.94
HamGAN-strong	14.77

Take-Home Message

Global Information -> Low-rank Formulation

Optimization as Architecture

The Curse -> Gradient back through MD -> One-step Grad

Enjoy Hamburger!

Thank you!