Multiplicative Interactions and Where To Find Them

2020. 03. 15.

Seongok Ryu, AITRICS

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Inductive bias over function classes

“ Appropriately applied multiplicative interactions can provide a more suitable inductive bias over function classes leading to more data-efficient learning, better generalization, and stronger performance.”

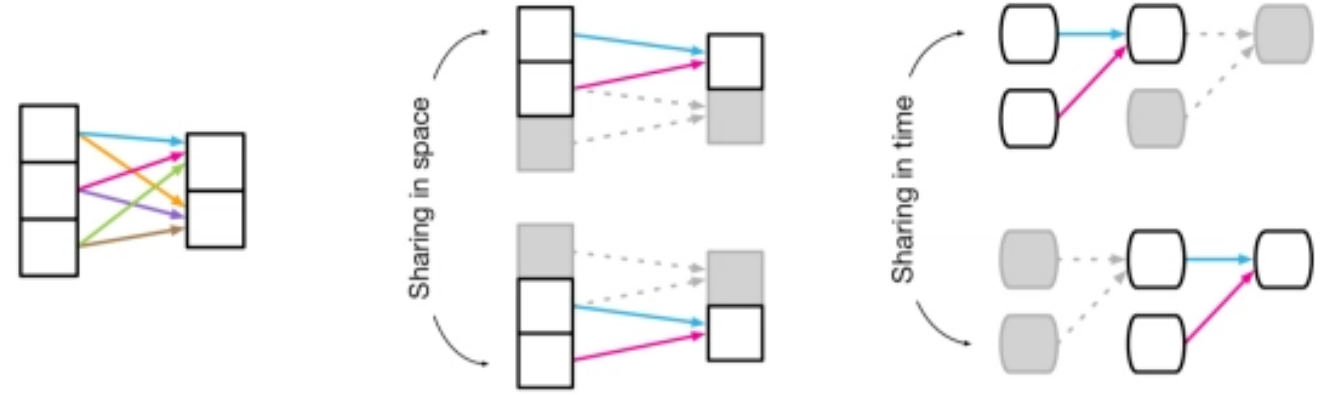


Figure 1: Reuse and sharing in common deep learning building blocks. (a) Fully connected layer, in which all weights are independent, and there is no sharing. (b) Convolutional layer, in which a local kernel function is reused multiple times across the input. Shared weights are indicated by arrows with the same color. (c) Recurrent layer, in which the same function is reused across different processing steps.

Relational inductive biases, deep learning, and graph networks, https://arxiv.org/abs/1806.01261

- Images, waves - CNN
- Sequential data – RNN, Transformer
- Graph data - GNN

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Multiplicative Interactions

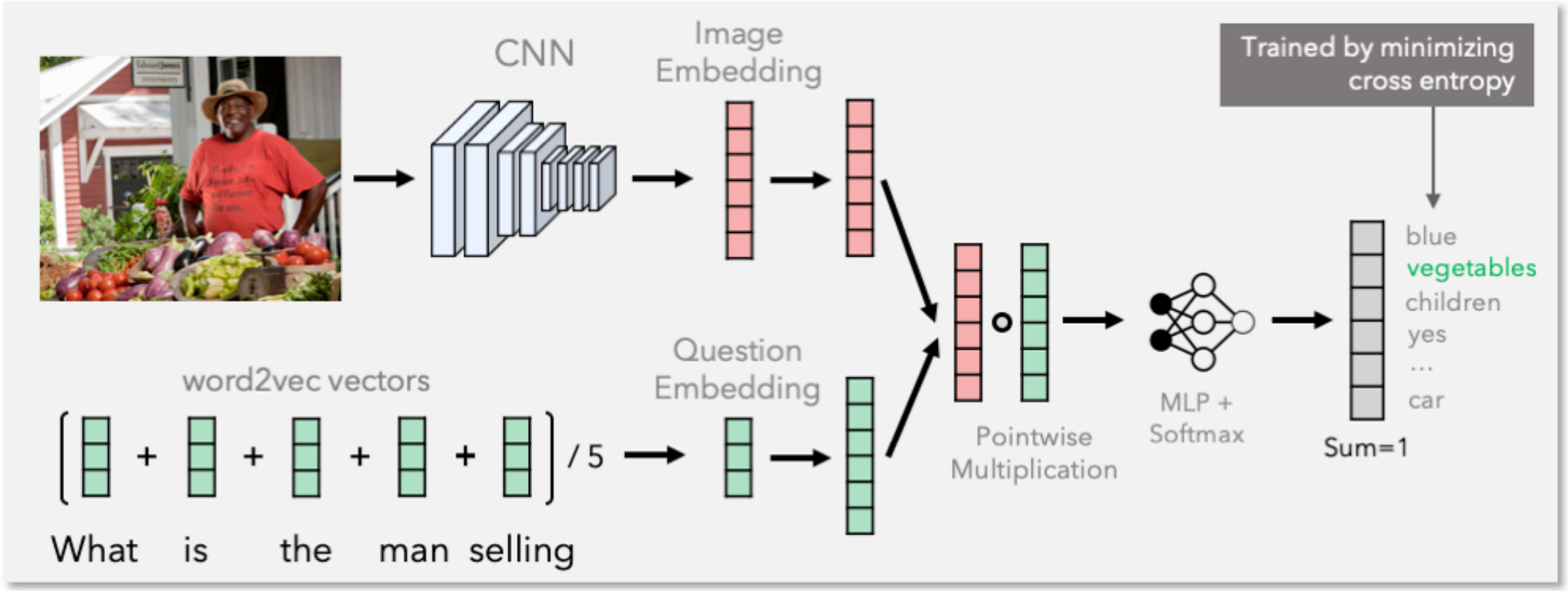
Notation and concept of multiplicative interactions

- "How to combine two different streams of information?"
- Specifically, given $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{z} \in \mathbb{R}^m$,
- Our goal is to model an unknown function $f(\mathbf{x}, \mathbf{z}) \in \mathbb{R}^k$ that entails some interaction between the variables.
- In practice, $\mathbf{x}$ and $\mathbf{z}$ might be hidden arbitrary activations, different input modalities (e.g. vision and language), or conditioning information and inputs.

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Multiplicative Interactions

Example) Visual Question Answering (VQA) task



Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Inductive bias over function classes

Contribution of this work

1. To re-explore multiplicative interactions and their design principles

$$f(\mathbf{x}, \mathbf{z}) = \mathbf{x}\mathbb{W}\mathbf{z} + \mathbf{z}^{\boldsymbol{T}}\mathbf{U} + \mathbf{V}\mathbf{x} + \mathbf{b}$$

2. To aid the community's understanding of other models, e.g. hypernetworks, gating, multiplicative RNNs, through them

3. To show their efficacy at representing certain solutions

4. To empirically apply them to large scale sequence modelling and reinforcement learning problems

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Multiplicative Interactions

Notation and concept of multiplicative interactions

- **Standard approach : CONCAT & MLP**

$$f(\mathbf{x}, \mathbf{z}) = \mathbf{W}[\mathbf{x}; \mathbf{z}] + \mathbf{b}$$

where $\mathbf{W} \in \mathbb{R}^{(m+n)\times k}$.

- **Proposed multiplicative interaction**

$$f(\mathbf{x}, \mathbf{z}) = \mathbf{x}\mathbb{W}\mathbf{z} + \mathbf{z}^{\boldsymbol{T}}\mathbf{U} + \mathbf{V}\mathbf{x} + \mathbf{b}$$

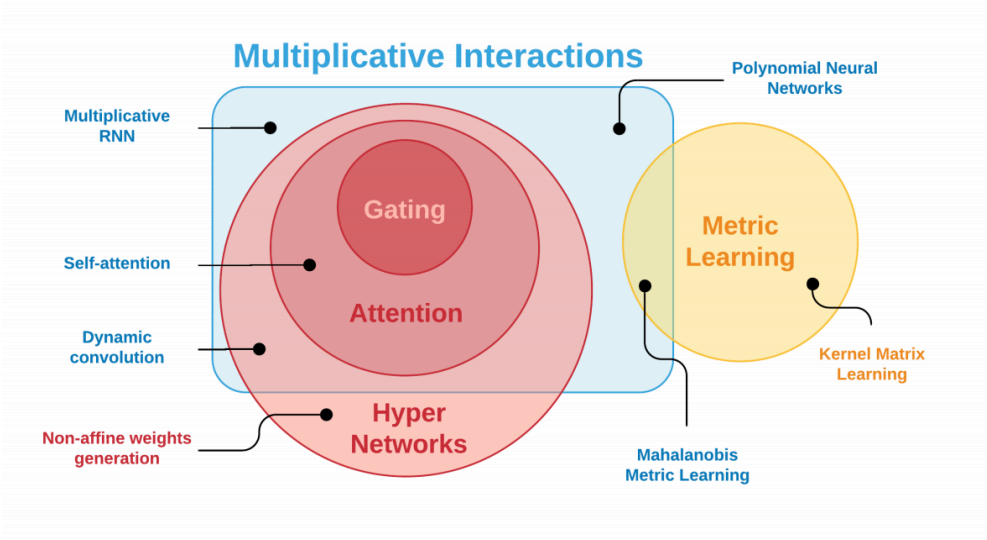
where $\mathbf{W} \in \mathbb{R}^{m\times n\times k}$ and $(\mathbf{x}\mathbb{W}\mathbf{z})$

"This specific form, while more costly, is more flexible, providing the right inductive bias to learn certain families of functions that are of interest in practice."

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

MI examples

Taxonomy



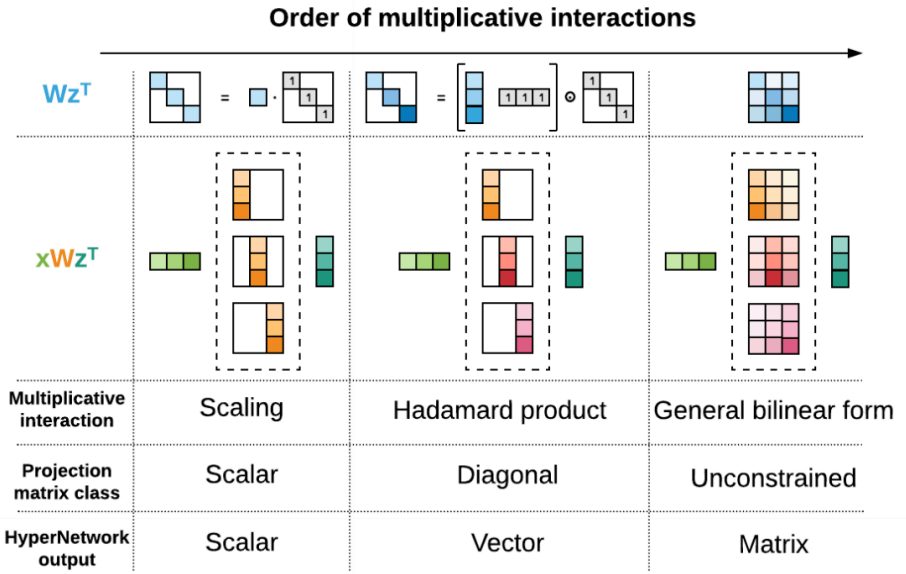


Figure 1: (Left) Venn diagrams of multiplicative interactions with respect to other model classes commonly used in ML. (Right) Comparison of various orders of multiplicative interactions and their relation to other perspectives.

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

MI examples

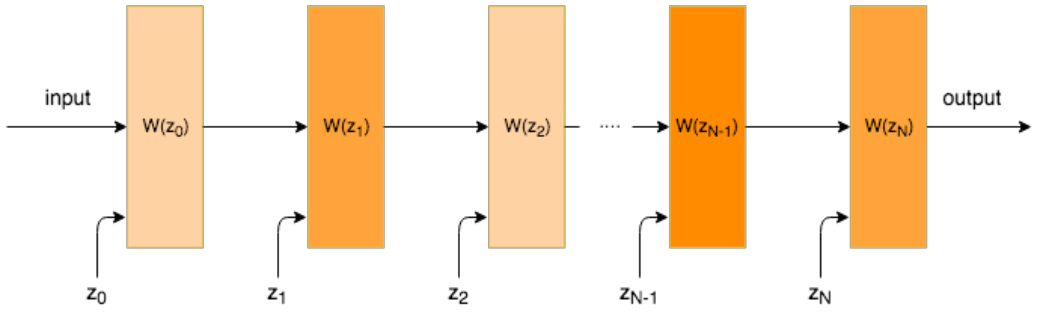
Example 1) Hyper-network. (Ha et. al. ICLR2017)

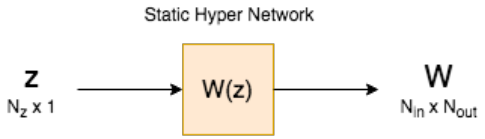
- A hyper-network is is a neural network g that is used to generate the weights of another neural network given some context of input vector $\mathbf{z}$.

$$f(\mathbf{x};\theta) = f\big(\mathbf{x}; g(\mathbf{z};\phi)\big)$$

- The output of entire network is $\mathbf{y}' = \mathbf{W}'\mathbf{x} + \mathbf{b}$, where input-conditional weight matrix and bias vector are given by

$$\mathbf{W}' = \mathbf{z}\mathbb{W} + \mathbf{V} \text{ and } \mathbf{b}' = \mathbf{z}^{\boldsymbol{T}}\mathbf{U} + \mathbf{b}$$





Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

MI examples

Example 2) Gating

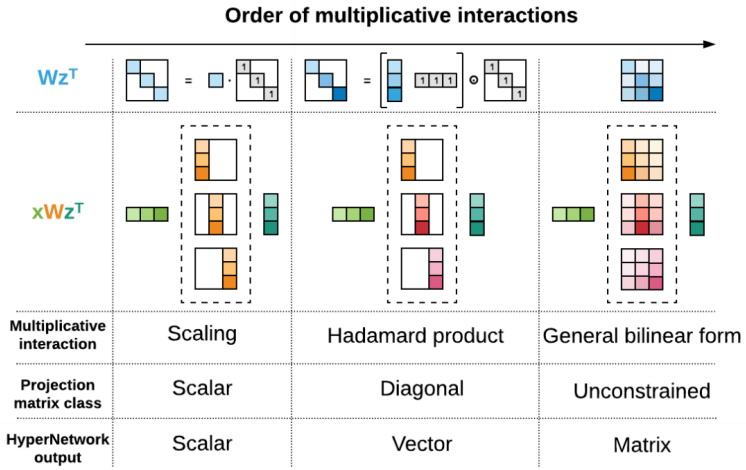
- Gating function $f = \mathbf{a} \odot \mathbf{x}$ is widely used in conv-S2S (Dauphin et. Al. 2017) and WaveNet (v.d.Oord et.al. 2016).

  Residual connection: $\mathbf{h}^{(l+1)} = \mathbf{h}^{(l)} + \mathbf{x}$

  Residual connection w/ gating: $\mathbf{h}^{(l+1)} = \mathbf{z} \odot \mathbf{h}^{(l)} + (\mathbf{1} - \mathbf{z}) \odot \mathbf{x}$, where the coefficient $\mathbf{z} = \mathbf{f}(\mathbf{h}, \mathbf{x}; \phi)$

- Gating can be reformulated with the proposed multiplicative interaction with a particular parameterization

$$\mathbf{W}' = \mathbf{z}\mathbb{W} + \mathbf{V} = \mathrm{diag}(a_1, \ldots, a_n)$$



Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

MI examples

Example 3) Attention

- The output of (self-) attention network is given by

$$y = \sum_{i=1}^{n} m_i \, x_i = \mathbf{m} \odot \mathbf{x}$$

- For the case of the self-attention in Transformer (Vaswani et. al. 2017)

$$\mathbf{m} = \text{softmax}\left(\frac{1}{\sqrt{d}}(\mathbf{x}\mathbf{W}_x)(\mathbf{z}\mathbf{W}_z)^T\right)$$

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

MI examples

Example 4) Metric Learning w/ Mahalanobis distance

- Mahalanobis distance

$$d_{\mathbf{C}}(\mathbf{x}, \mathbf{z}) = \|\mathbf{x} - \mathbf{z}\|_{\mathbf{C}} = (\mathbf{x} - \mathbf{z})^T \mathbf{C}^{-1} (\mathbf{x} - \mathbf{z})$$
$$= \mathbf{x}^T \mathbf{C}^{-1} \mathbf{x} - 2\mathbf{x}\mathbf{C}^{-1}\mathbf{z} + \mathbf{z}^T \mathbf{C}^{-1} \mathbf{z}$$

  which again maps on to multiplcative interaction units.

- In metric learning, however, one usually explicitly defines losses over tuples with direct supervision,

- While here the authors consider building blocks that can learn a metric internally, without direct supervision.

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Expressivity of the model

- Vanilla MLPs are universal approximators.
- Consequently adding new modules/building blocks does not affect the approximation power of NNs, however such modifications can change the hypotheses space - the set of functions that can be represented exactly, and the compactness of a good estimator, as well as learnability.

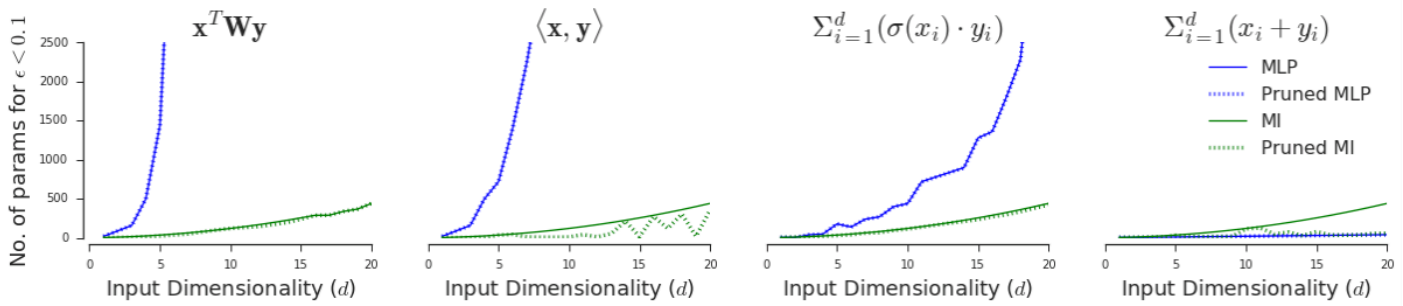


Figure 2: Number of parameters needed for a regular, single layer MLP (blue line) to represent the function up to 0.1 MSE over the domain of a standard d-dimensional Gaussian compared to the same quantity for a multiplicative model (green line). σ denotes sigmoid. Dotted lines represent pruned models where all weights below absolute value of 0.001 were dropped. Note that for MLP all parameters are actually used, while for MI module some of these functions (summation and dot product) can be compactly represented with pruning.

Theorem 1. *Let $\mathcal{H}_{mlp}$ denote the hypotheses space of standard MLPs with ReLU activation function, and let $\mathcal{H}_{mu}$ denote the hypotheses space of analogous networks, but with each linear layer replaced with a multiplicative layer, then we have $\mathcal{H}_{mlp} \subsetneq \mathcal{H}_{mu}$.*

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Experiments

Goals

- Better integrations of

  i) latent variables in decoder models

  ii) task or contextual information in multi-task learning

  iii) recurrent state in sequence models.

- Use for neural process regression, multi-task RL and language modelling as exemplar

- Terminology:

$$f(\mathbf{x}, \mathbf{z}) = \mathbf{z}^T \mathbb{W}\mathbf{x} + \mathbf{z}^T \mathbf{U} + \mathbf{B}\mathbf{x} + \mathbf{b}$$

  or referred to as MI $\mathcal{M}(\mathbf{x}, \mathbf{z})$ in the legends

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Experiments

1-1. Toy multi-task learning example

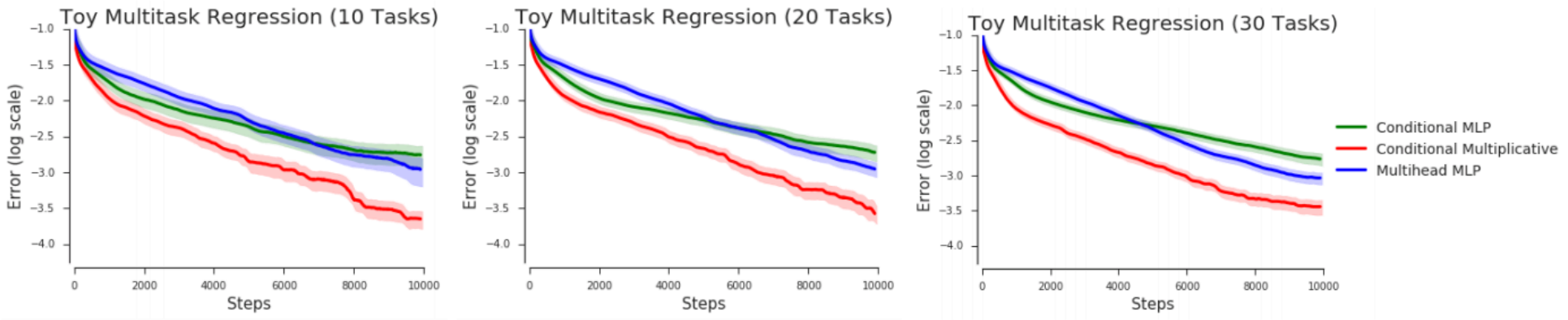


Figure 3: Averaged learning curves for different models while varying the number of tasks in the toy multitask regression domain. Shaded regions represent standard error of mean estimation.

- Conditional MLP: concat(x, linear(20)(z)), linear(30), relu, linear(20), relu, linear(1)
- Conditional MI: MI([linear(30), relu, linear(20), relu](x), linear(20)(z))
- Multiheaded MLP: linear(20), relu, linear(30), relu, linear(1); where the last linear is separate per task

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Experiments

1-1. Toy multi-task learning example

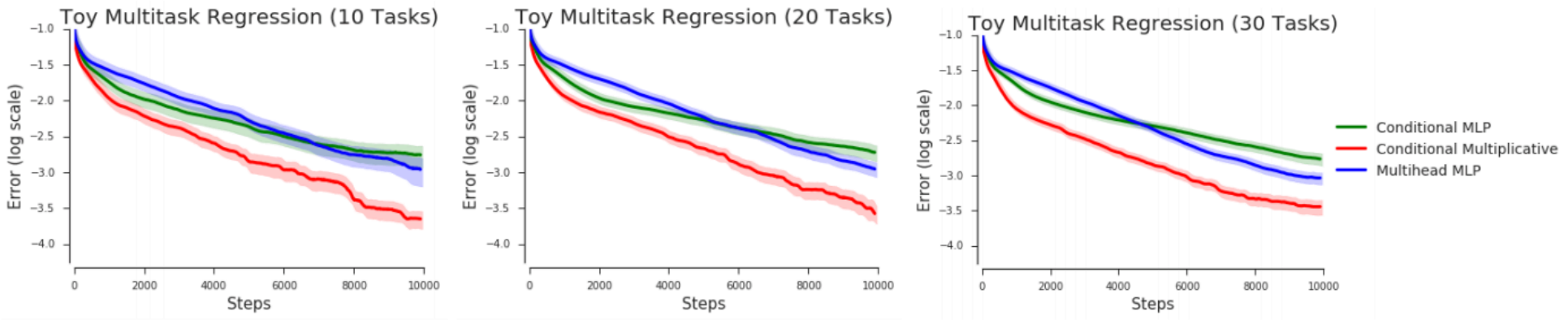


Figure 3: Averaged learning curves for different models while varying the number of tasks in the toy multitask regression domain. Shaded regions represent standard error of mean estimation.

- To regress two different classes of functions, affine and sines -- $y = a_i x + b$ and $y = a_i \sin(10x) + b_i$, where a_i and b_i are sampled per task from a uniform distribution.
- Task-conditioned $\mathcal{M}(x, \mathbf{t})$ layer allows the model to learn both tasks better with less interference and also increased data efficiency.
- See Supplementary Information for more experimental details.

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Experiments

1-2. Multi-task RL on DMLab-30

- 30 different tasks
- Typical actor-critic RL setup within the Impala framework
- The architecture as in the original works: a stack of conv layers followed by an LSTM, with output $\mathbf{h}_t$ at each time step.
- These are then projected to policies π and value functions V that are shared across all tasks.

$$\pi_t, V_t = \mathcal{M}(\mathbf{h}_t, \mathbf{c})$$

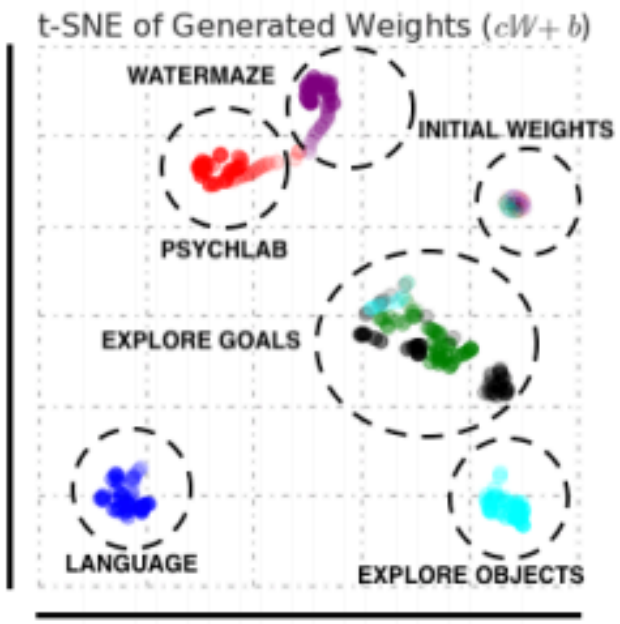
where

i) Task-ID: $\mathbf{c} = \mathrm{ReLU}\big(\mathrm{MLP}(\mathbf{I}_i)\big)$ and $\mathbf{I}_i$ is one-hot task ID.

Ii) Learnt context: $\mathbf{c} = \mathrm{ReLU}\big(\mathrm{MLP}(\mathbf{h}_t)\big)$

Experiments

1-2. Multi-task RL on DMLab-30

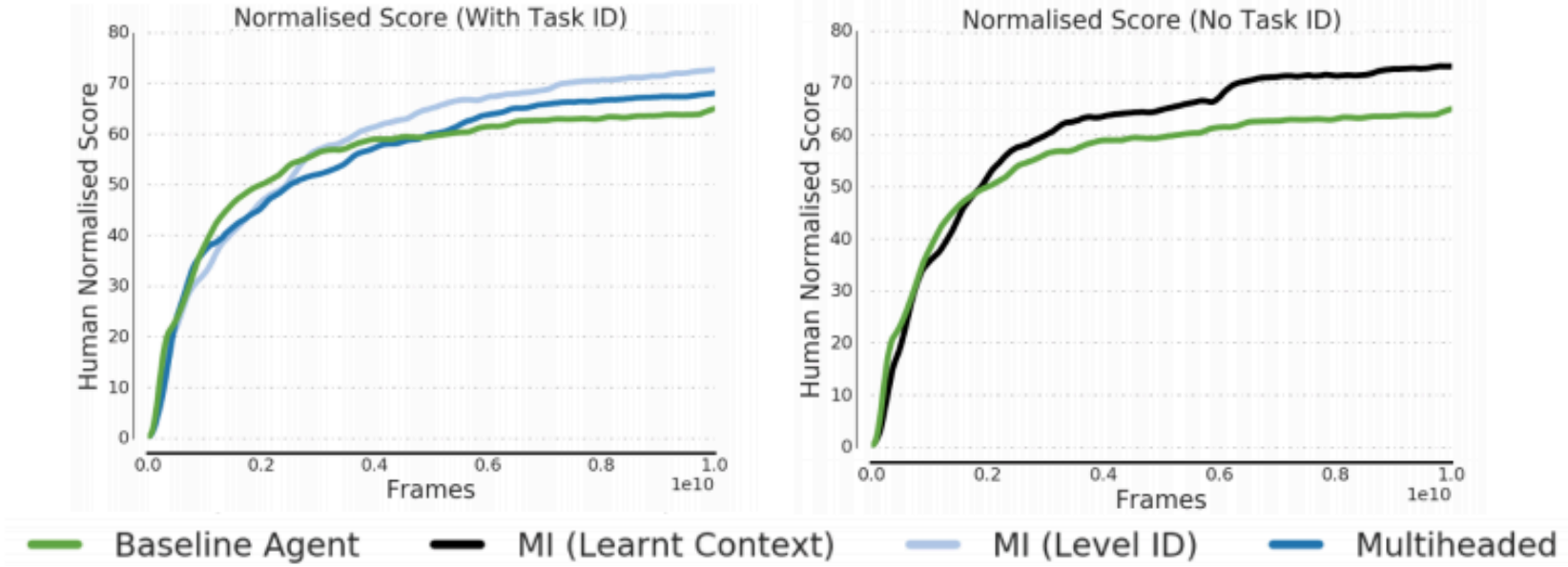


- t-SNE plot of the 2D projection of the state of the LSTM.
- Color: level name, transparency value: time step
- At time step 0: all weights are the same
- As the levels progress, the model can naturally detect or cluster task information

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Experiments

1-2. Multi-task RL on DMLab-30



$$\mathbf{c} = \mathrm{ReLU}\big(\mathrm{MLP}(\mathbf{h}_t)\big) \qquad \mathbf{c} = \mathrm{ReLU}\big(\mathrm{MLP}(\mathbf{I}_i)\big)$$

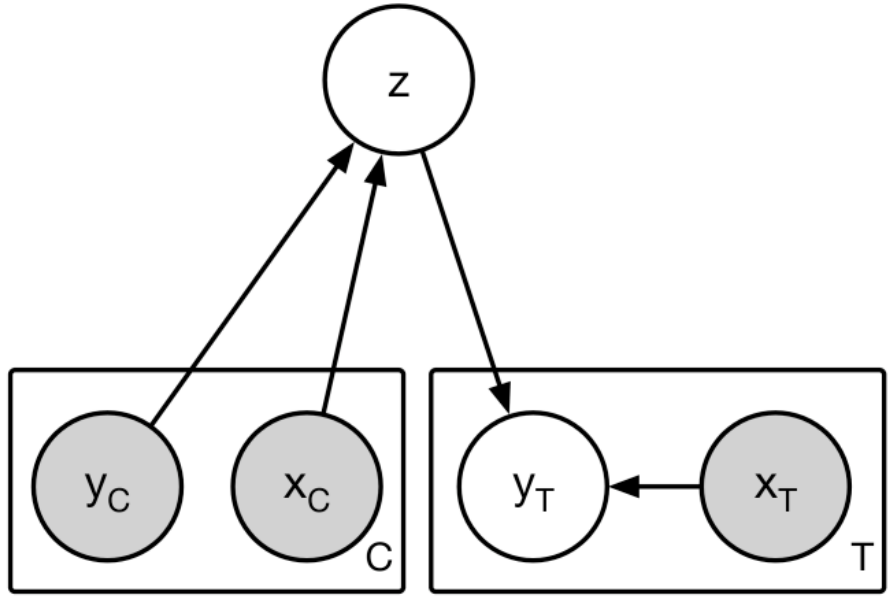
- MI method allows the model to integrate information and provide additional densely-gated conditional-compute paths → effectively learning task-conditioned behaviour.
- Previous SOTA: PopArt Method (Hessel et. al., 2019), ~73% normalised human score.
- The MI method match SOTA performance with a simpler method.

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Experiments

2. Latent variable models with multiplicative decoders in neural processes

- Neural processes learn to infer a distribution over functions that are consistent with the observations collected so far.
- Encoder network gives latent variables $\mathbf{z}$ that are a representation of the function that maps $\mathbf{x}$ to $\mathbf{y}$, i.e. $\mathbf{y} = f(\mathbf{x}, \mathbf{z})$.
- A new data point $\mathbf{x}^*$ is mapped to $\mathbf{y}^*$ by passing $[\mathbf{z}; \mathbf{x}^*]$ through a decoder network.



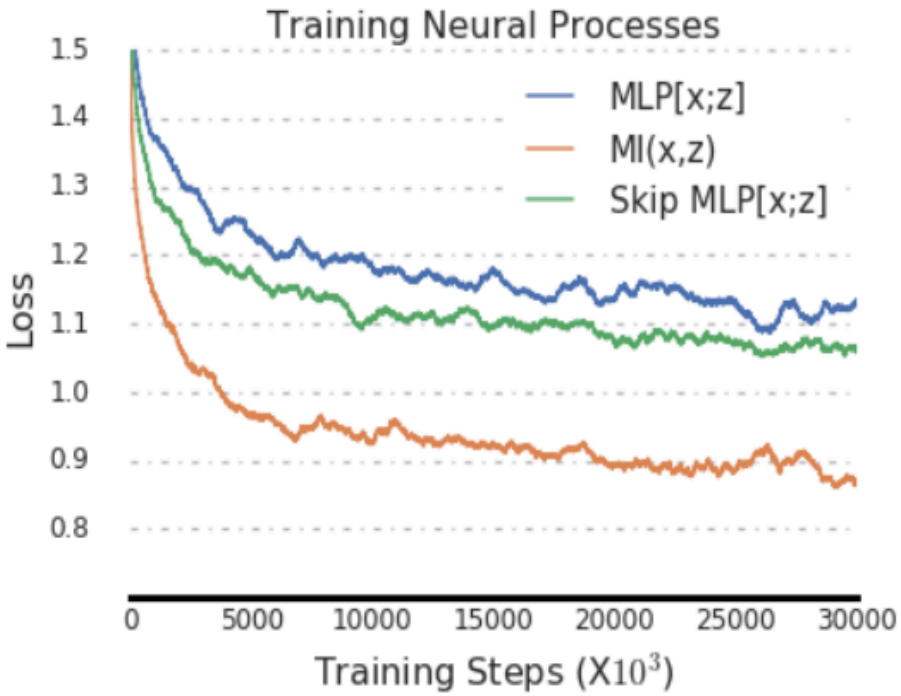
Graphical model of neural processes

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Experiments

2. Latent variable models with multiplicative decoders in neural processes

- The regression task in “Neural processes, Garnelo et. al. 2018”.
- Standard approach : using $\mathrm{MLP}([\mathbf{z}; \mathbf{x}^*])$
- Replacing the MLP decoder with the MI form $\mathcal{M}(\mathbf{z}, \mathbf{x}^*)$
- The MI forms are able to better condition on latent information compared to the baseline MLPs.



Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Experiments

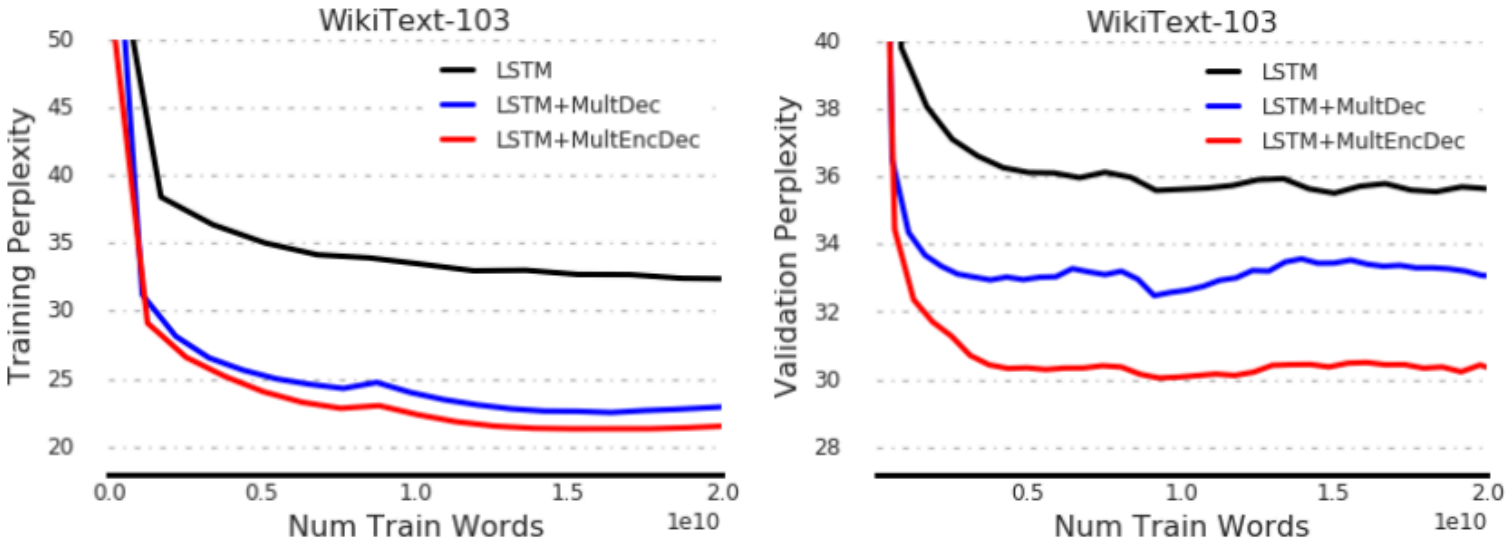
3. Language modelling

- Standard approach:

$$\mathbf{z}_t^i = \mathbf{W}\mathbf{x}_t, \qquad \mathbf{h}_t = \mathrm{LSTM}(\mathbf{z}_t^i), \qquad \mathbf{z}_{t+1}^o = \mathbf{W}_2\mathbf{h}_t\mathbf{x}_t + \mathbf{b}, \qquad \mathbf{y}_{t+1} = \mathrm{softmax}(\mathbf{z}_{t+1}^o\mathbf{W}^T + \mathbf{b})$$

- MI form:

$$\mathbf{c} = \mathrm{ReLU}(\mathbf{W}_3\mathbf{h}_t + \mathbf{b}), \qquad \mathbf{z}_{t+1}^o = \mathcal{M}(\mathbf{c}^T, \mathbf{h}_t)$$



Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Experiments

3. Language modelling

- Leave as future work the integration of the ideas with Transformer based models.

Table 1: Word-level perplexity on WikiText-103

Model	Valid	Test	No. Params
LSTM Rae et al. (2018)	34.1	34.3	88M
Gated CNN Dauphin et al. (2017)	-	37.2	-
RMC Santoro et al. (2018)	30.8	31.6	-
Trellis Networks Bai et al. (2019)	-	30.35	180M
TransformerXL Dai et al. (2018)	**17.7**	**18.3**	**257M**
LSTM (ours)	34.7	36.7	88M
LSTM + MultDec	31.7	33.7	105M
LSTM + MultEncDec	**28.9**	**30.3**	**110M**

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Conclusion & Future direction

3. Language modelling

- Multiplicative interactions and connecting them to a variety of architectures such as hyper-network, gating methods, …

- The ability of such networks to better represent a broader range of algorithmic primitives, e.g. conditional-statements or inner product.

- Future work
  - ✓ Various approximations to MI methods; ways to their implementations more efficient
  - ✓ Their application to newer domains
  - ✓ Attention models use some of these multiplicative interactions – some of the lessons from this work, such as higher order interactions – will allow even greater integration of information in attention systems.

Multiplicative Interactions and Where To Find Them, https://openreview.net/forum?id=rylnK6VtDH

Thank You!