On the *Information* in Deep Networks I

— A Case Study of Information Bottleneck

Zhaopeng Tu, Boyuan Wang, Wenxuan Wang, and Cunxiao Du

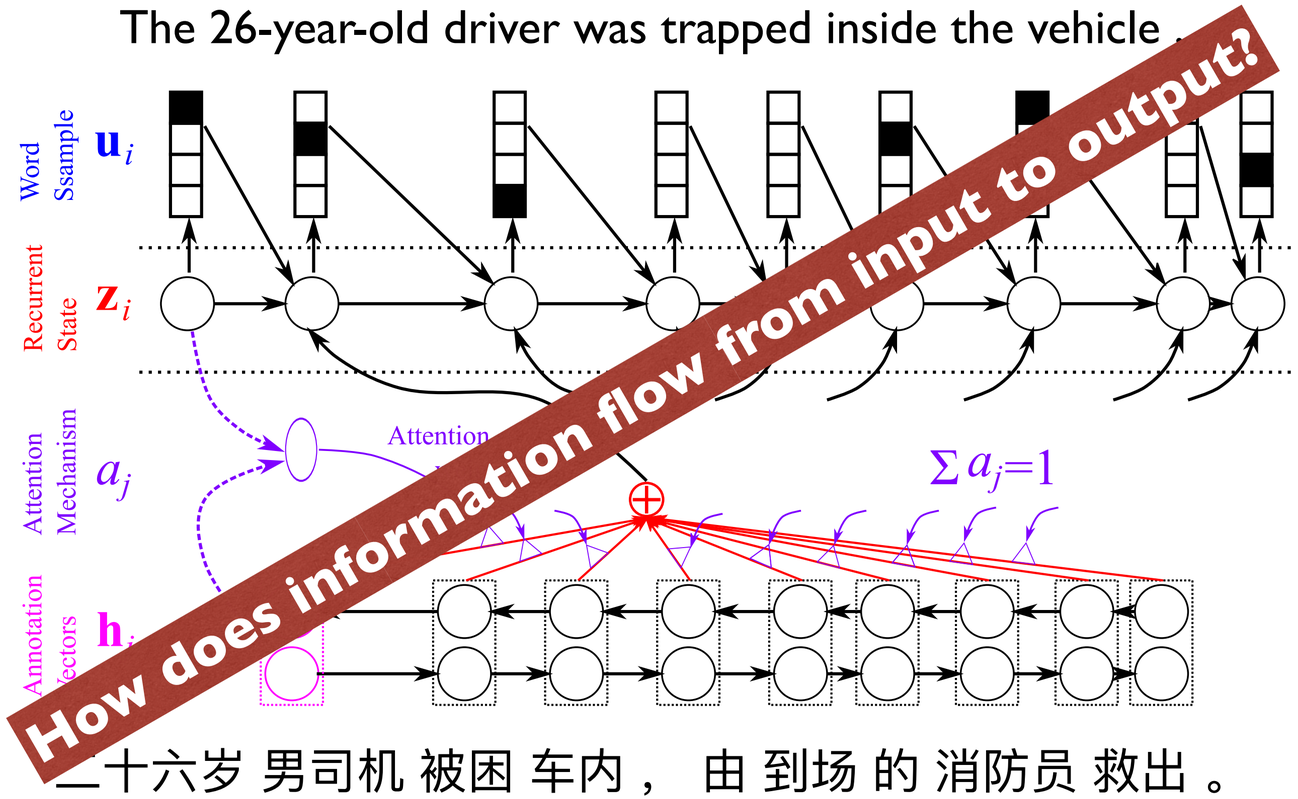
2019-12-25

Why Shall We Care Information?

- Current deep learning models are generally black-boxes. *Information is the key to open the black-boxes!*
  - Trust in model outputs;
  - Design principles of deep neural networks;
  - Error analysis & model refinement.
- Interpretability is the process of giving explanations to humans: *how information flows from source side to target side in deep learning models* (*information flow*)?
- Information flow is *the key problem of NMT*.
  - *Under-translation* => How information is lost?
  - *Mistaken-translation* => How information is perturbed?

Information Flow in NMT

- Information flow is *the key problem of NMT*.
  - *Under-translation* => How information is lost?
  - *Mistaken-translation* => How information is perturbed?



Conditional Generation

Language Understanding

Where is the Information?

- Ba et al. (2016) claim that RNNs typically have two types of memory that have very different time scales and computational roles:
  - **Neural Activities** that represent short-term memory that is updated at every time step;
  - **Weights** that capture long-term memory to connect the hidden layers to the inputs and outputs;

Where is the Information?

- Ba et al. (2016) claim that RNNs typically have two types of memory that have very different time scales and computational roles:
  - **Neural Activities** that represent short-term memory that is updated at every time step;
  - **Weights** that capture long-term memory to connect the hidden layers to the inputs and outputs;
- Similarly, there are two threads of work on defining the information in DNNs:
  - **Information in Activations** that represent the contribution of hidden layers to information flow.
    - Representative Work: *Information Bottleneck* (IB, Tishby et al. 2000; Alemi et al. 2017; Saxe et al. 2018).
  - **Information in Weights** that represent the training data, which can be connected to model generalization and invariance.
    - Representative Work: *Information Lagrangian* (IL, Achille and Soatto 2018; 2020).

- Both IB and IL are special forms of *Rate-Distortion Theory*: the number of bits needed to encode the weights in order to solve the task at some level of precision.
- In this tutorial, we focus on IB, since there has been an increasing amount of work on adapting IB for interpreting the information flow in DNNs — our long-term research goal.

Overview

- Information Bottleneck (IB) theory of Deep Learning
  - *IB expresses the trade-off between the mutual information measures the amount of information that **the hidden layer** contains about the **input** and the **output**.*
    - Deep Learning and the Information Bottleneck Principle (IEEE ITW 2015)
    - Opening the Black Box of Deep Neural Networks via Information (arXiv 2017)
    - On the Information Bottleneck Theory of Deep Learning (ICLR 2018)
- Information Flow: IB for Attribution
  - *IB for attribution to interpret the decision-making of DNNs. Noises are added to intermediate feature maps to **restrict the flow of information**, which quantify (in bits) how much information input regions provide.*
    - Estimating Information Flow in Deep Neural Networks (ICML 2019)
    - Towards a Deep and Unified Understanding of Deep Neural Models in NLP (ICML 2020)
    - Restricting the Flow: Information Bottlenecks for Attribution (ICLR 2020)
- Variational Information Bottleneck (VIB) and its Application
  - *VIB is a variational approximation to IB, which allows to parameterize the information bottleneck model using a neural network and leverage the reparameterization trick for efficient training.*
    - Deep Variational Information Bottleneck (ICLR 2017)
    - Specializing Word Embeddings (for Parsing) by Information Bottleneck (EMNLP 2019, Best Paper)

Outline

- Information Bottleneck (IB) theory of Deep Learning (***Boyuan Wang***)
  - Deep Learning and the Information Bottleneck Principle (IEEE ITW 2015)
  - Opening the Black Box of Deep Neural Networks via Information (arXiv 2017)
  - On the Information Bottleneck Theory of Deep Learning (ICLR 2018)

- Information Flow: IB for Attribution (***Cunxiao Du***)
  - Estimating Information Flow in Deep Neural Networks (ICML 2019)
  - Towards a Deep and Unified Understanding of Deep Neural Models in NLP (ICML 2020)
  - Restricting the Flow: Information Bottlenecks for Attribution (ICLR 2020)

- Variational Information Bottleneck (VIB) and its Application (***Wenxuan Wang***)
  - Deep Variational Information Bottleneck (ICLR 2017)
  - Specializing Word Embeddings (for Parsing) by Information Bottleneck (EMNLP 2019, Best Paper)

Outline

- Information Bottleneck (IB) theory of Deep Learning (***Boyuan Wang***)
  - Deep Learning and the Information Bottleneck Principle (IEEE ITW 2015)
  - Opening the Black Box of Deep Neural Networks via Information (arXiv 2017)
  - On the Information Bottleneck Theory of Deep Learning (ICLR 2018)

- Information Flow: IB for Attribution (*Cunxiao Du*)
  - Estimating Information Flow in Deep Neural Networks (ICML 2019)
  - Towards a Deep and Unified Understanding of Deep Neural Models in NLP (ICML 2020)
  - Restricting the Flow: Information Bottlenecks for Attribution (ICLR 2020)

- Variational Information Bottleneck (VIB) and its Application (*Wenxuan Wang*)
  - Deep Variational Information Bottleneck (ICLR 2017)
  - Specializing Word Embeddings (for Parsing) by Information Bottleneck (EMNLP 2019, Best Paper)

Overview

- Deep Learning and the Information Bottleneck Principle (IEEE ITW 2015)
  - Introducing IB method as a measure of the trade-off for the amount of information that the hidden layer contains about the input and the output. Under this frame work, many works tends to analyze different DNN models, tasks, and objectives.
- Opening the Black Box of Deep Neural Networks via Information (arXiv 2017)
  - Extending the first work and demonstrate the effectiveness of the visualization of DNNs in the information plane for a better understating of the training dynamics, learning processes, and internal representations in Deep Learning (DL).
- On the Information Bottleneck Theory of Deep Learning (ICLR 2018)
  - This work disagrees with the major conclusion from the second paper and put in doubt the generality of the IB theory of deep learning as an explanation of generalization performance in deep architectures.

Overview

- Deep Learning and the Information Bottleneck Principle (IEEE ITW 2015)
  - Introducing IB method as a measure of the trade-off for the amount of information that the hidden layer contains about the input and the output. Under this frame work, many works tends to analyze different DNN models, tasks, and objectives.
- Opening the Black Box of Deep Neural Networks via Information (arXiv 2017)
  - Extending the first work and demonstrate the effectiveness of the visualization of DNNs in the information plane for a better understating of the training dynamics, learning processes, and internal representations in Deep Learning (DL).
- On the Information Bottleneck Theory of Deep Learning (ICLR 2018)
  - This work disagrees with the major conclusion from the second paper and put in doubt the generality of the IB theory of deep learning as an explanation of generalization performance in deep architectures.

Paper1 - Motivation

- DNN and Deep Learning have become the most successful machine learning methods for most supervised learning tasks, but basic questions about the design principles of deep networks, the optimal architecture, the number of required layers, the sample complexity, and the best optimization algorithms, are not well understood.
- This paper aims to propose a theoretical framework for analyzing DNNs through information bottleneck (IB) perspective.

Paper1 - Approach: IB

- IB method is an information theoretic principle to exact the relevant information between input X and output Y. The relevant information is expressed as the mutual information I (X; Y).
- An optimal representation of X, called X_hat, should be able to capture the relevant information (Generalization) while dismissing the irrelevant information (Compression).
- The optimal representation X_hat can be obtained by minimizing the following Lagrangian:

$$\mathcal{L}\left[p\left(\hat{x}|x\right)\right] = I\left(X;\hat{X}\right) - \beta I\left(\hat{X};Y\right)$$

- Alternatively, let distortion D_BI = I(X;Y | X_hat):

$$\tilde{\mathcal{L}}\left[p\left(\hat{x}|x\right)\right] = I\left(X;\hat{X}\right) + \beta I\left(X;Y|\hat{X}\right)$$

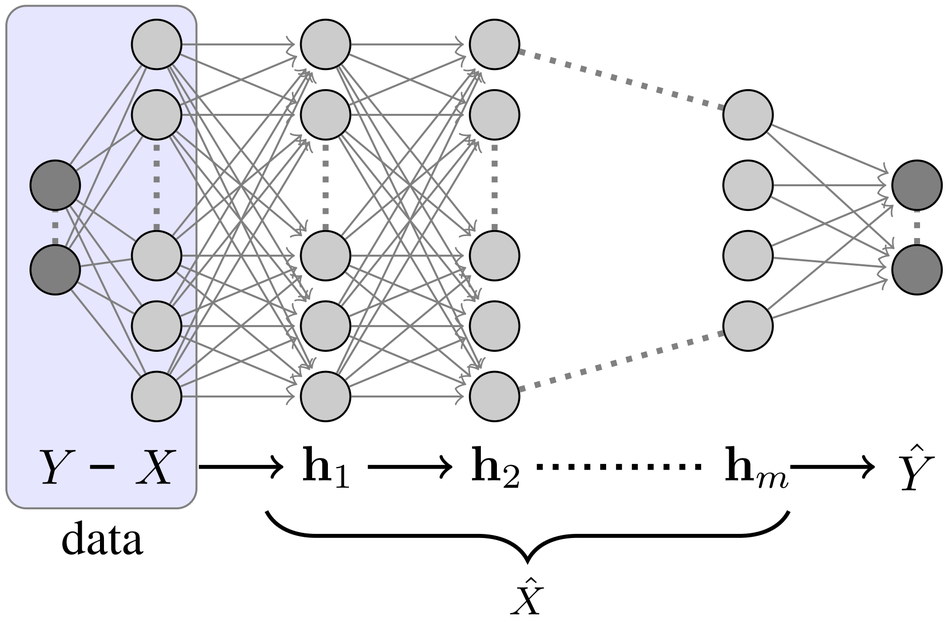
Paper1 - Approach: DNN

- The DNN above forms a layer-wise Markov-Chain because each layer is determined by the previews layer. Thus, we have the following Data Processing Inequality (DPI):

$$I(Y;X) \geq I(Y;\mathbf{h}_j) \geq I(Y;\mathbf{h}_i) \geq I\left(Y;\hat{Y}\right)$$

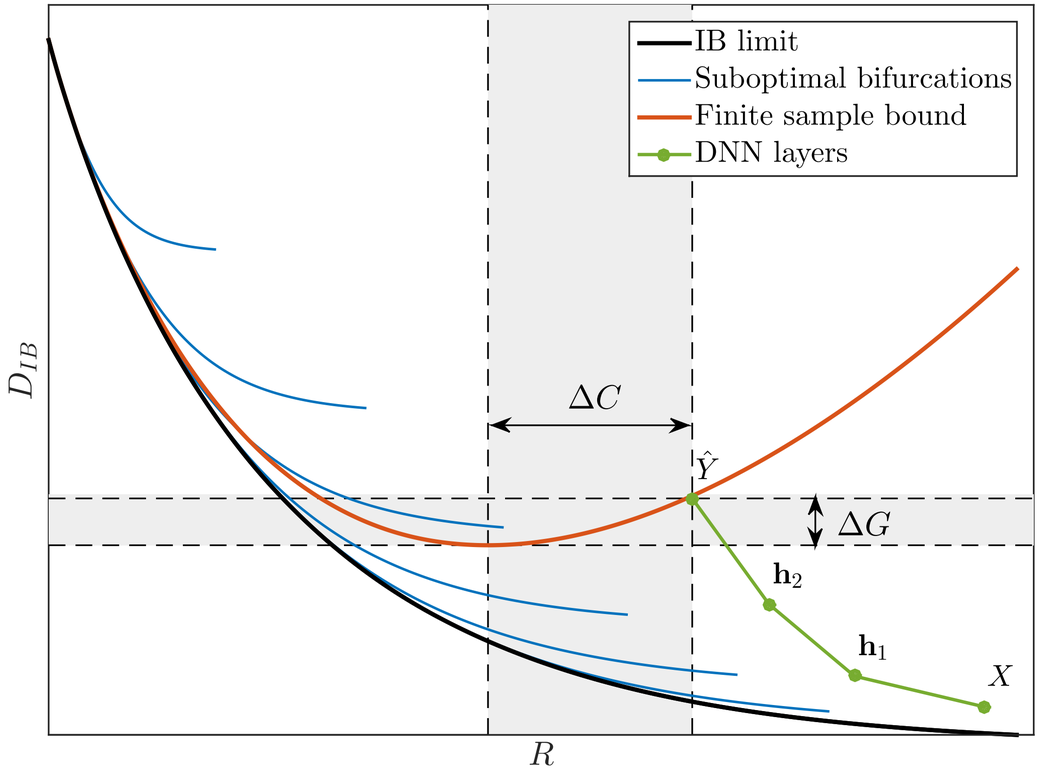
- Based on DPI, we have that the latter layers must have less relevant information of Y.
- Now, IB can be used as a measure of optimality for each layer with the optimal IB limit:

$$I(\mathbf{h}_{i-1};\mathbf{h}_i) + \beta I(Y;\mathbf{h}_{i-1}|\mathbf{h}_i)$$



Paper1 - Qualitative Simulation

- Black curve is the optimal IB limit (trade off).
- Red curve is the worse generalization upper bound for finite sample.
- Green curve is the layer-wise Distortion-Complexity curve, which is bounded by black and red curve on both side.
- deltaC and deltaG are complexity and generalization gap respectively.
- Optimal DNN should close the gap between Y_hat and deltaC as well as deltaG.



Paper1 - Main Conclusions

- This paper provides a new information perspective of DNN training as successive relevant compression of the input variable X given the training data.
- Trained DNN compressed the input to generate predicted Y_hat.
- It is natural that the goal of DNN training is to maximize the mutual information I(Y;Y_hat).
- The networks and all layers can be directly compared to the optimal IB limit, by estimating the mutual information between each layer and the input and output variables, on the information plane.
- New optimization criterion for optimal DNN representations.
- Realistic sample complexity bound on the generalization ability using the IB finite sample bound.

Overview

- Deep Learning and the Information Bottleneck Principle (IEEE ITW 2015)
  - Introducing IB method as a measure of the trade-off for the amount of information that the hidden layer contains about the input and the output. Under this frame work, many works tends to analyze different DNN models, tasks, and objectives.
- Opening the Black Box of Deep Neural Networks via Information (arXiv 2017)
  - Extending the first work and demonstrate the effectiveness of the visualization of DNNs in the information plane for a better understating of the training dynamics, learning processes, and internal representations in Deep Learning (DL).
- On the Information Bottleneck Theory of Deep Learning (ICLR 2018)
  - This work disagrees with the major conclusion from the second paper and put in doubt the generality of the IB theory of deep learning as an explanation of generalization performance in deep architectures.

Paper II - Motivation

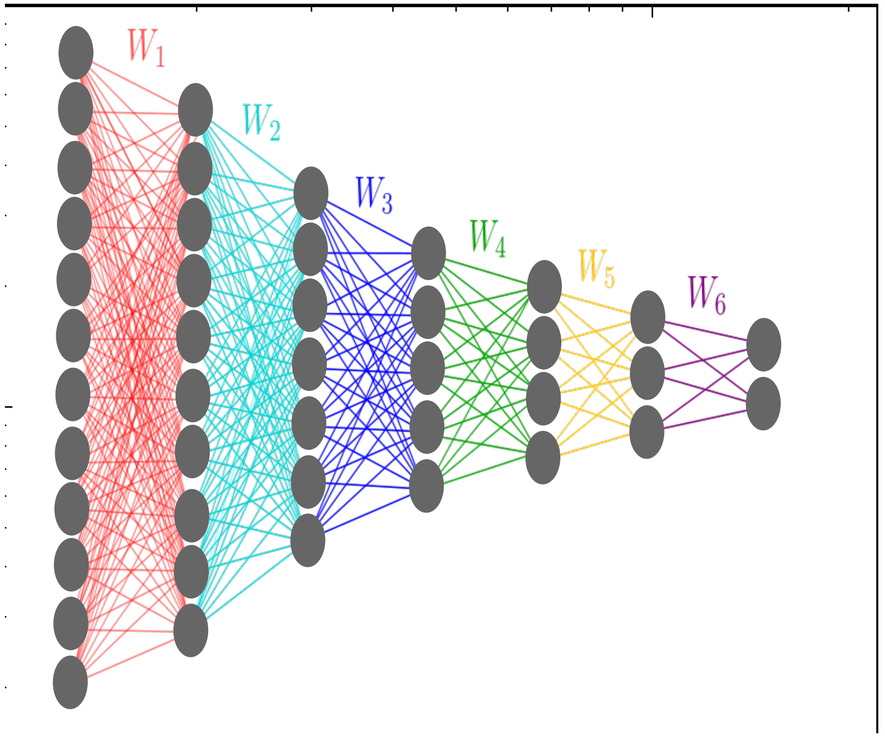
- Previous paper shows that the goal of the network is to optimize the Information Bottleneck (IB) tradeoff between compression and prediction, successively, for each layer.
- In this paper, the authors extend previous IB work and demonstrate the effectiveness of the visualization of DNNs in the information plane for a better understating of the training dynamics, learning processes, and internal representations in Deep Learning (DL).

Paper II - Approach

- Let T denotes any arbitrary layer in DNN. Then we define P(T|X) as encoder and P(Y|T) as decoder.
- Given P (X ; Y), T is uniquely mapped to a point in the information-plane with coordinates (I (X ; T), I (T ; Y)).
- They study the information paths of DNNs in the information plane to study the following issues:
  - The SGD layer dynamics in the *Information plane*.
  - The effect of the training sample size on the layers.
  - What is the benefit of the hidden layers?
  - Do the hidden layers form optimal IB representations?

Paper II - Experiments

- Experiment Setup:
  - Architecture:
    - Fully connected feed-forward
    - 7 hidden layers with widths:
      12, 10, 7, 5, 4, 3, 2
  - Activation Function:
    - Tanh for first 6 layers
    - Sigmoid for the final layer
  - Training:
    - SGD
    - Cross-Entropy loss
  - Data:
    - 4096 synthetic data points with binary (0,1) labels and true distribution P(X,Y) can be computed
  - Models:
    - 50 random initialization + random sampling



Paper II - Experiments

- For all models and all layers, training first fit the data and then *compress* the representation.
- *FYI: the way to estimate the mutual information is suboptimal.*

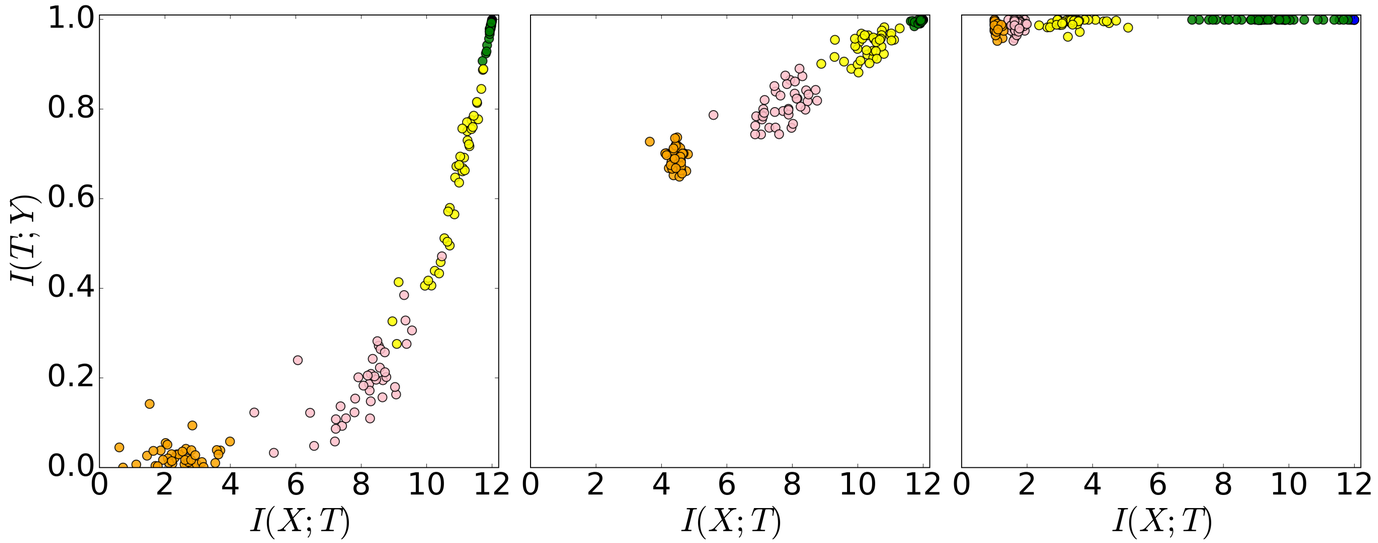


Figure 2: Snapshots of layers (different colors) of 50 randomized networks during the SGD optimization process in the *information plane* (in bits): **left** - with the initial weights; **center** - at 400 epochs; **right** - after 9000 epochs. The reader is encouraged to view the full videos of this optimization process in the *information plane* at *https://goo.gl/rygyIT* and *https://goo.gl/DQWuDD*.

Paper II - Experiments

- Sufficient train sample size pushes up I(T;Y) and gets closer to the optimal IB bound.

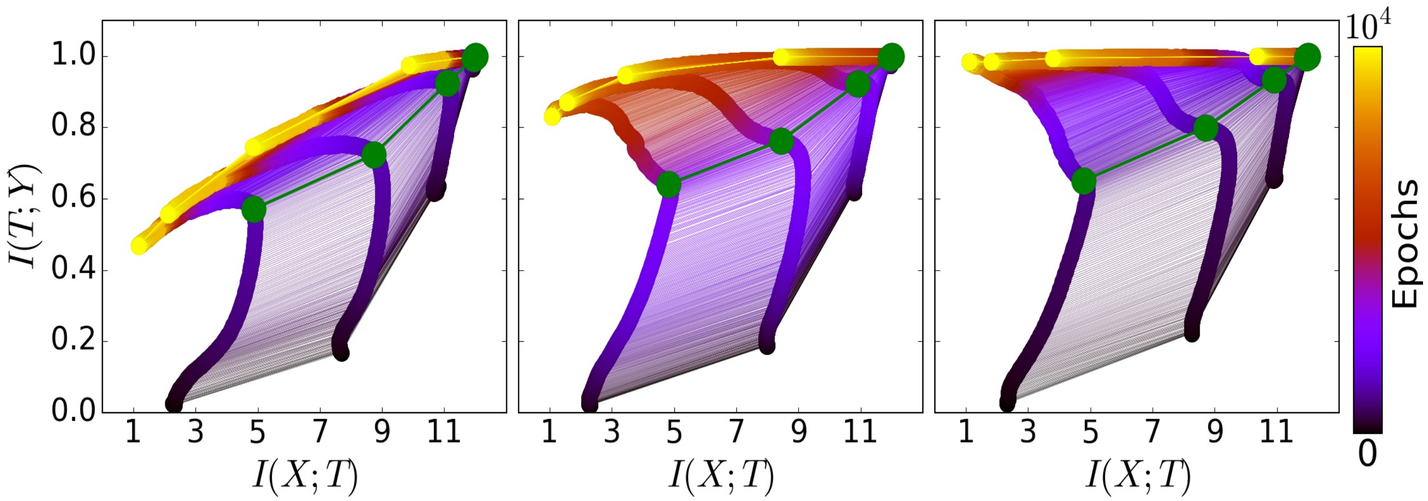


Figure 3: The evolution of the layers with the training epochs in the information plane, for different training samples. On the left - 5% of the data, middle - 45% of the data, and right - 85% of the data. The colors indicate the number of training epochs with Stochastic Gradient Descent from 0 to 10000. The network architecture was fully connected layers, with widths: input=12-10-8-6-4-2-1=output. The examples were generated by the spherical symmetric rule described in the text. The green paths correspond to the SGD drift-diffusion phase transition - grey line on Figure 4

Paper II - Experiments

- Benefit of the hidden layers
  - Adding hidden layers dramatically reduces the number of training epochs for good generalization.
  - The compression phase of each layer is shorter when it starts from a previous compressed layer.
  - The compression is faster for the deeper (narrower and closer to the output) layers.
  - Even wide hidden layers eventually compress in the diffusion phase. Adding extra width does not help.

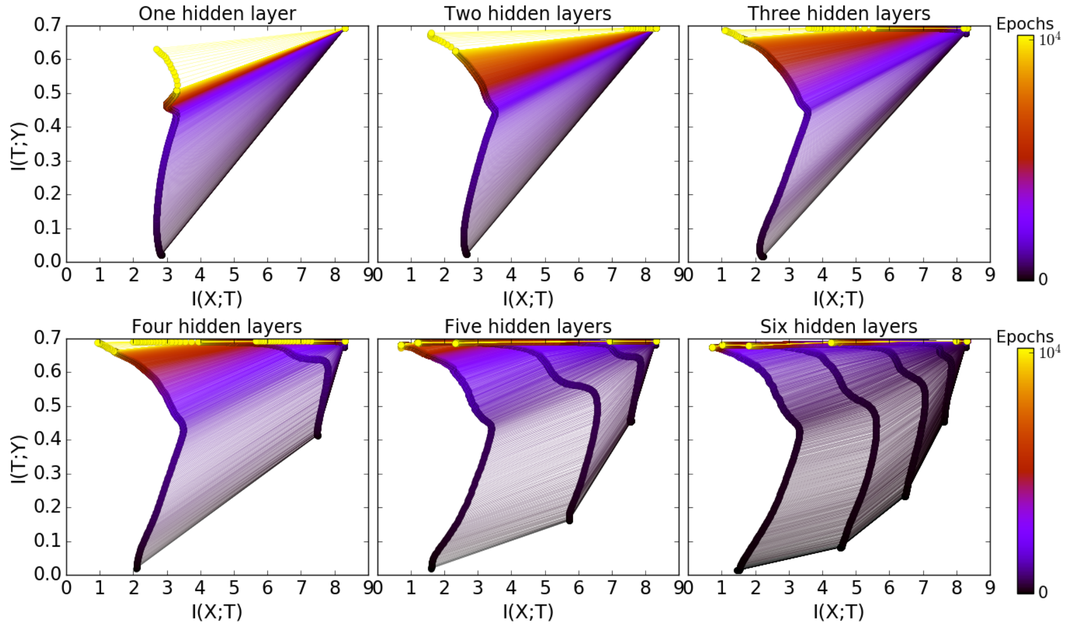


Figure 5: **The layers information paths during the SGD optimization for different architectures.** Each panel is the *information plane* for a network with a different number of hidden layers. The width of the hidden layers start with 12, and each additional layer has 2 fewer neurons. The final layer with 2 neurons is shown in all panels. The line colors correspond to the number of training epochs.

Paper II - Experiments

- Empirical layers trained with SGD lies very close to the optimal IB bound.

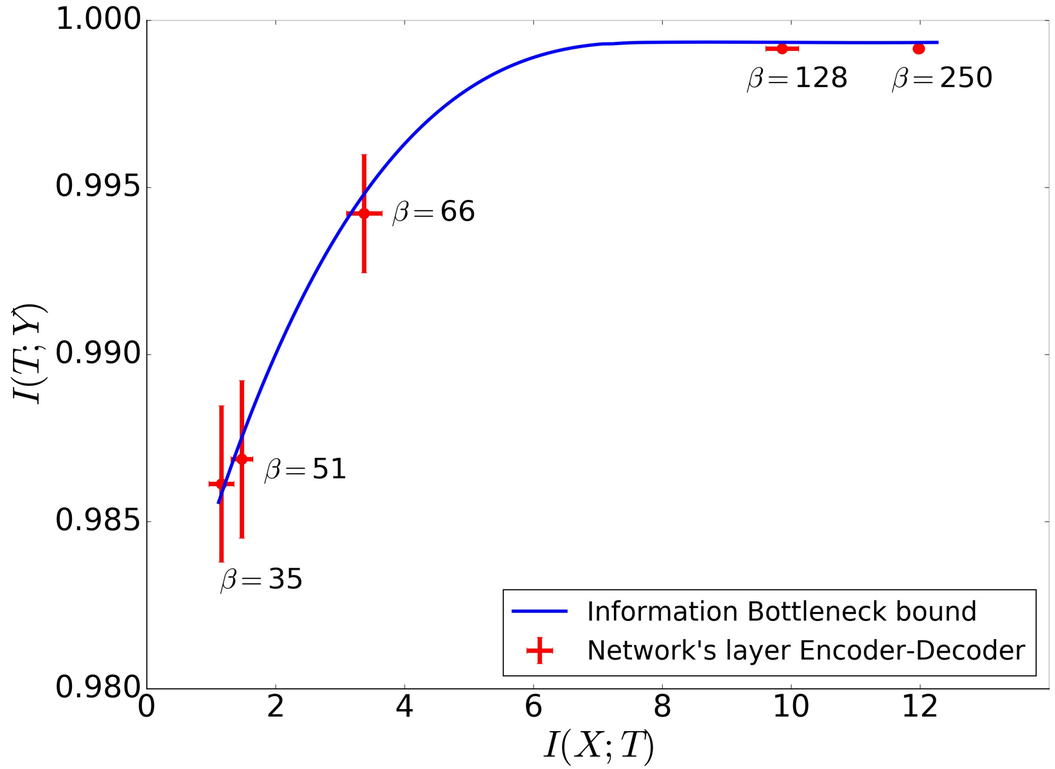


Figure 6: **The DNN layers converge to fixed-points of the IB equations**. The error bars represent standard error measures with N=50. In each line there are 5 points for the different layers. For each point, β is the optimal value that was found for the corresponding layer.

Paper II - Conclusion

- DL training has two phases.
  - The first and shorter phase the layers increase the information on the labels (fitting).
  - The second and much longer phase the layer reduce the information on the input (compression)
- The main advantage of the hidden layers is computational, as they dramatically reduce the stochastic relaxation times.
- The converged layers lie on or very close to the IB theoretical bound, for different values of the tradeoff parameter.

Overview

- Deep Learning and the Information Bottleneck Principle (IEEE ITW 2015)
  - Introducing IB method as a measure of the trade-off for the amount of information that the hidden layer contains about the input and the output. Under this frame work, many works tends to analyze different DNN models, tasks, and objectives.
- Opening the Black Box of Deep Neural Networks via Information (arXiv 2017)
  - Extending the first work and demonstrate the effectiveness of the visualization of DNNs in the information plane for a better understating of the training dynamics, learning processes, and internal representations in Deep Learning (DL).
- On the Information Bottleneck Theory of Deep Learning (ICLR 2018)
  - This work disagrees with the major conclusion from the second paper and put in doubt the generality of the IB theory of deep learning as an explanation of generalization performance in deep architectures.

Paper III - Motivation

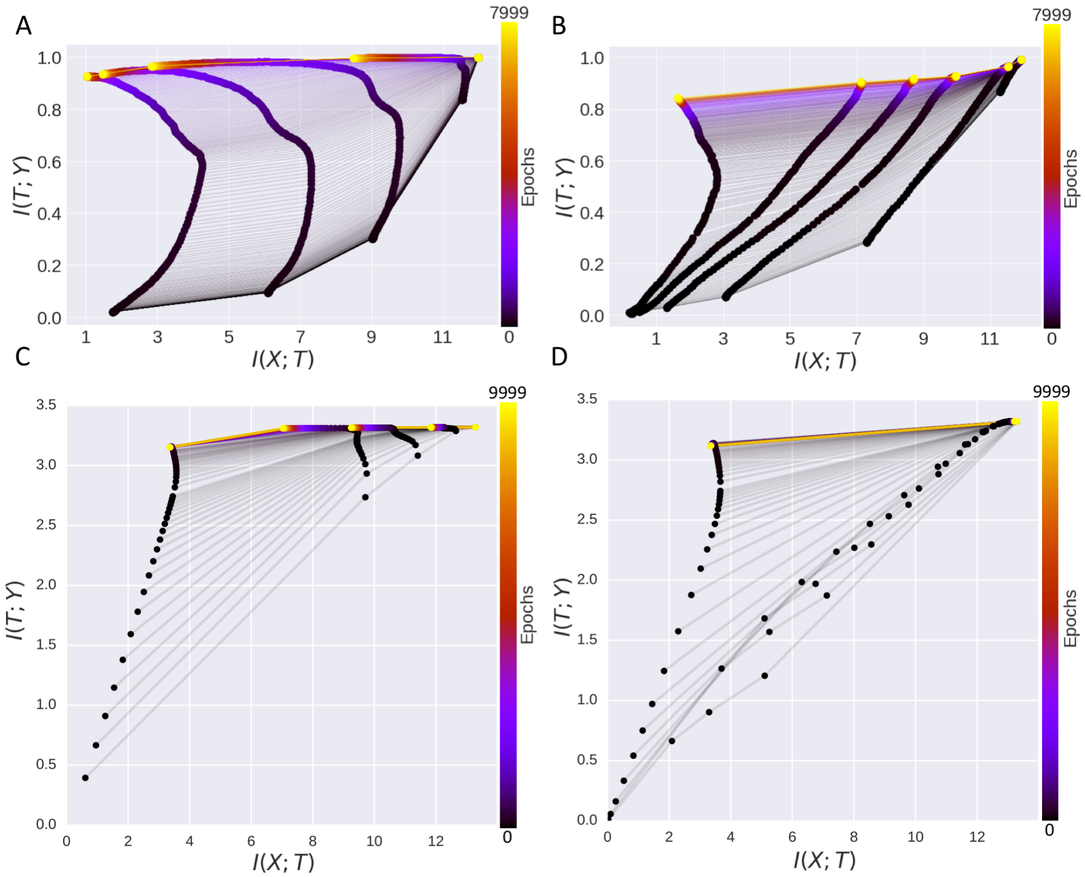
- The information bottleneck (IB) theory of deep learning makes three specific claims:
  - First, that deep networks undergo two distinct phases consisting of an initial fitting phase and a subsequent compression phase;
  - Second, the compression phase is causally related to the excellent generalization performance of deep networks;
  - Third, the compression phase occurs due to the diffusion-like behavior of stochastic gradient descent.
- The paper wants to explore whether this claims hold true for general cases. Through analysis and experiment, they show that strikingly none of the above claims hold true.

Paper III - Approach

- Test different activation for the compression phenomenon
- Through deep linear network to study relationship of compression and generalization
- Change SGD to Batch gradient descent to study the compression phase
- Explicit using irrelevant information to study when compression happens.

Paper III - Experiments

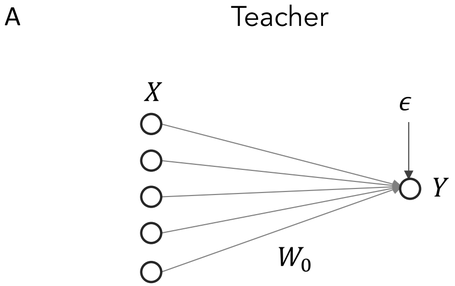
- Experiment Setup:
  - Activations: Tanh (left), ReLU (right)
  - Dataset: synthetic binary data (top)
    MNIST (bottom)
- DNNs with ReLU activation does not express compression for both cases. The compress in the last layer is due to the use of sigmoid function.

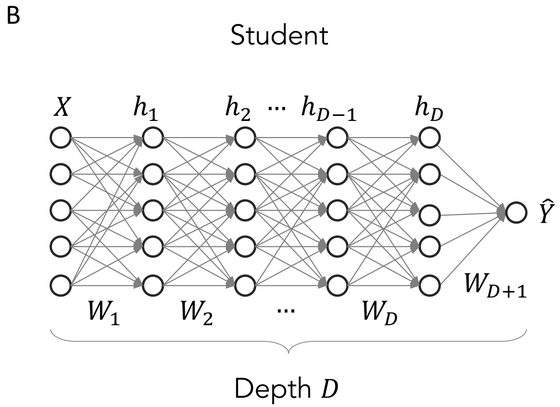


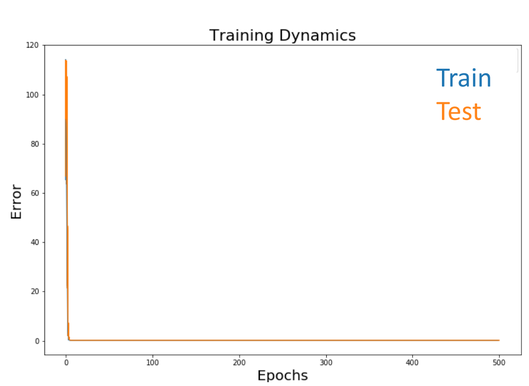
Paper III - Experiments

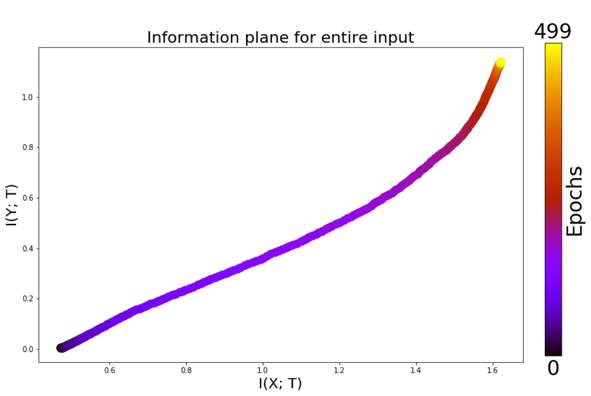
- Experiment Setup:
  - Architecture:
    - Teacher: Linear mapping with Gaussian noise.
    - Student: linear fully connected network
  - Data: generate from Teach net

- Deep linear net generalize very well while show no compression.









Paper III - Experiments

- The plot itself is quite self-explanatory.

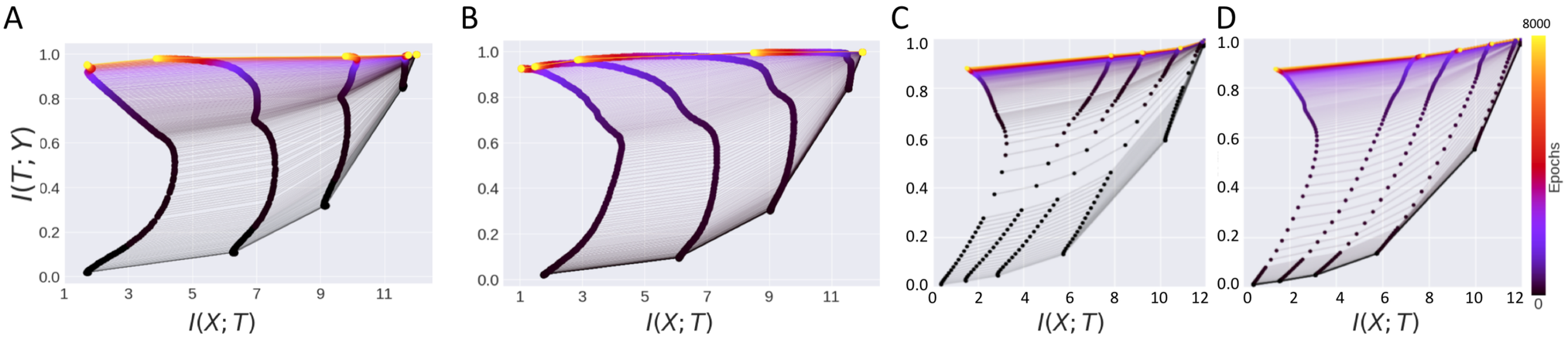


Figure 5: Stochastic training and the information plane. (A) tanh network trained with SGD. (B) tanh network trained with BGD. (C) ReLU network trained with SGD. (D) ReLU network trained with BGD. Both random and non-random training procedures show similar information plane dynamics.

Paper III - Experiments

- Experiment Setup:
  - Architecture:
    - Teacher: Linear mapping with Gaussian noise; Student: linear fully connected network.
  - Data: 30 relevant input + 70 irrelevant input generated by setting W_0 = 0
- Results
  - Combined curve shows no compression as the relevant curve. The irrelevant curve shows compression.
  - Compression happens at the mean time as generalization happens.

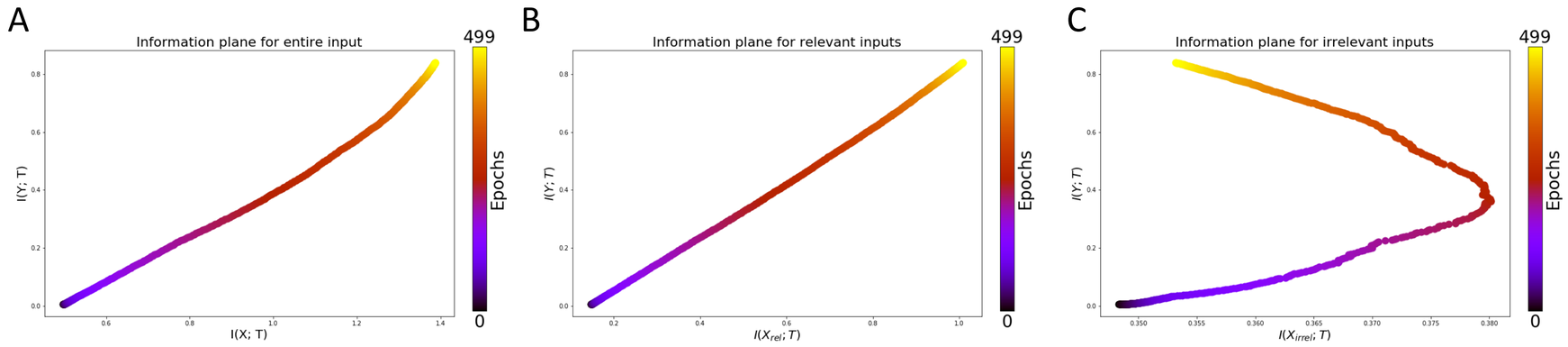


Figure 6: Simultaneous fitting and compression. (A) For a task with a large task-irrelevant subspace in the input, a linear network shows no overall compression of information about the input. (B) The information with the task-relevant subspace increases robustly over training. (C) However, the

Paper III - Conclusion

- Compression phase is a phenomenon for double-sized saturation activation, not a character for DNN in general.
- There is no evident causal connection between compression and generalization: networks that do not compress are still capable of generalization, and vice versa.
- Compression phase, when it exists, does not arise from stochasticity in training.
- Compression happens concurrently with the fitting process rather than during a subsequent compression period.

Review

- Deep Learning and the Information Bottleneck Principle (IEEE ITW 2015)
  - Introducing IB method as a measure of the trade-off for the amount of information that the hidden layer contains about the input and the output. Under this frame work, many works tends to analyze different DNN models, tasks, and objectives.
- Opening the Black Box of Deep Neural Networks via Information (arXiv 2017)
  - Extending the first work and demonstrate the effectiveness of the visualization of DNNs in the information plane for a better understating of the training dynamics, learning processes, and internal representations in Deep Learning (DL).
- On the Information Bottleneck Theory of Deep Learning (ICLR 2018)
  - This work disagrees with the major conclusion from the second paper and put in doubt the generality of the IB theory of deep learning as an explanation of generalization performance in deep architectures.

Summary

- The first paper is the theoretic frame work for analyzing DNN with IB method. While the second work tries to analyze some underlying properties of DNN through the concept of Information plane, the third one raises strong objection for the main results of paper II.
- Personal thoughts: although the third paper proves false many claims from the second paper, it is done through many condition satisfied. I believe NMT contains the problem of compression irrelevant information give the nature of problem, so IB is still a useful frame work in this setting.
- Brain Storming:
  - In needs for a fast and accurate MI estimator for NMT.
  - Does Transformer close the gaps mentioned in paper I.
  - Layer-wise and model with different capacity can be analyzed similarly as in paper II.
  - Markov chain properties may no longer hold. Will IB still hold?

Outline

- Information Bottleneck (IB) theory of Deep Learning (*Boyuan Wang*)
  - Deep Learning and the Information Bottleneck Principle (IEEE ITW 2015)
  - Opening the Black Box of Deep Neural Networks via Information (arXiv 2017)
  - On the Information Bottleneck Theory of Deep Learning (ICLR 2018)

- Information Flow: IB for Attribution (***Cunxiao Du***)
  - Estimating Information Flow in Deep Neural Networks (ICML 2019)
  - Restricting the Flow: Information Bottlenecks for Attribution (ICLR 2020)
  - Towards a Deep and Unified Understanding of Deep Neural Models in NLP (ICML 2019)

- Variational Information Bottleneck (VIB) and its Application (*Wenxuan Wang*)
  - Deep Variational Information Bottleneck (ICLR 2017)
  - Specializing Word Embeddings (for Parsing) by Information Bottleneck (EMNLP 2019, Best Paper)

Overview

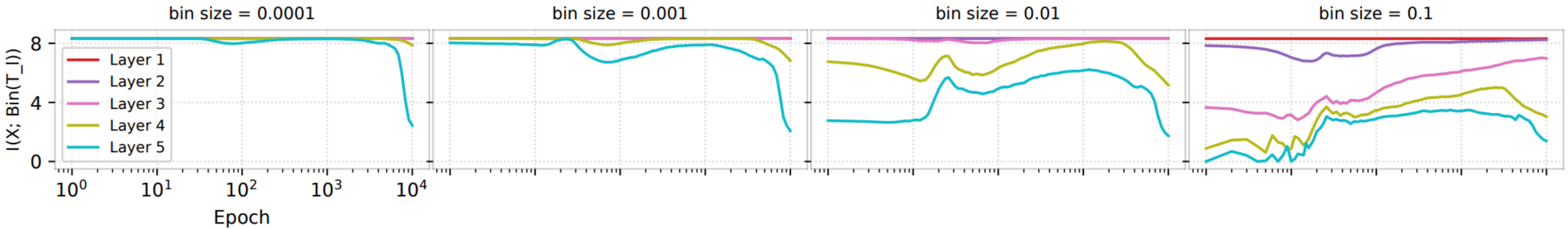
- Estimating Information Flow in Deep Neural Networks (ICML 2019)
  - This work reexamines the "compression" aspect of the IB theory, they identified clustering of the learned features as the geometric underlying compression.
- Restricting the Flow: Information Bottlenecks for Attribution (ICLR 2019)
  - This work proposes a new IB-based method for attribution, based on the the intuition that IB could be used to find which parts of input or feature are redundant.
- Towards a Deep and Unified Understanding of Deep Neural Models in NLP (ICML 2019)
  - This shares a very similar idea to the former one, although they did not refer their method as IB, we still can use a same view to link the two paper. This work main focusing on NLP models and several findings are quite interesting.

Overview

- Estimating Information Flow in Deep Neural Networks (ICML 2019)
  - This work reexamines the "compression" aspect of the IB theory, they identified clustering of the learned features as the geometric underlying compression.
- Restricting the Flow: Information Bottlenecks for Attribution (ICLR 2019)
  - This work proposes a new IB-based method for attribution, based on the the intuition that IB could be used to find which parts of input or feature are redundant.
- Towards a Deep and Unified Understanding of Deep Neural Models in NLP (ICML 2019)
  - This shares a very similar idea to the former one, although they did not refer their method as IB, we still can use a same view to link the two paper. This work main focusing on NLP models and several findings are quite interesting.

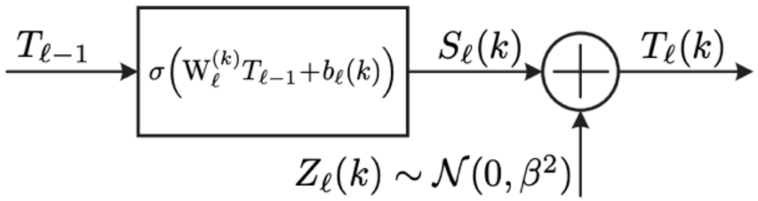
Paper 1 - Motivation

- Binning estimators are inconsistency of binning size.
- Why the model shows “information compression” during training?



Paper 1 - Approach

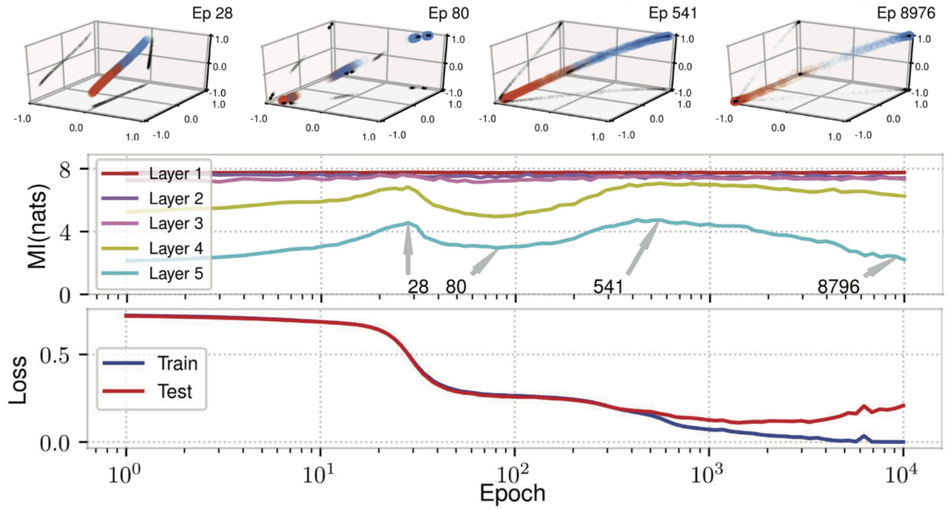
- Noise-based Mutual Information (MI) estimator
  - Noise-based NN



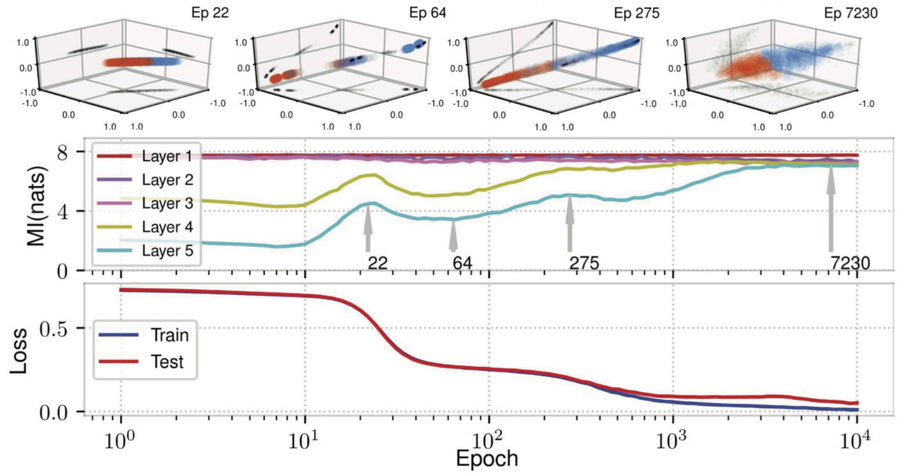
  - $I[X; T_l] = h(p_{T_l}) - \mathbb{E}_x h(p_{T_l|X=x})$
  - $\hat{I}_{\mathrm{SP}} \triangleq h\left(\hat{p}_{\mathcal{S}_\ell} * \varphi_\beta\right) - \frac{1}{n} \sum_{x \in \mathcal{X}} h\left(\hat{p}_{\mathcal{S}_\ell^{(x)}} * \varphi_\beta\right)$
  - $\hat{p}_{s_l}$ estimated by feed **each sample** into DNN previous layer.
  - $\hat{p_{s_l}^{(x)}}$ estimated by feed **one sample N times** into DNN previous layer.

Paper 1 - Experiments

- Noisy-SZT Model



- Orthonormal Regularization N-SZT



Paper 1 - Conclusion

- Compression can occur in noisy network.
- Compression is caused by clustering of internal representations.
- $I[X;T]$ is highly correlated with $H(bin(T))$.
- Deterministic Model has the same conclusion.

Overview

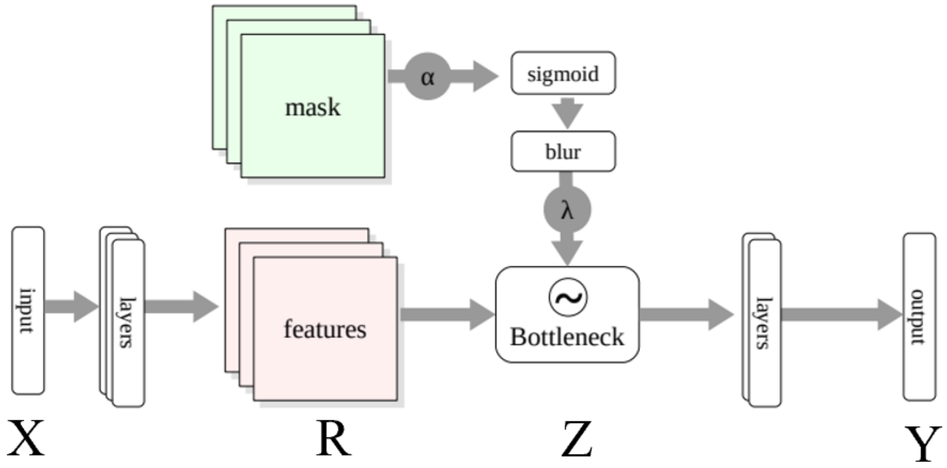
- Estimating Information Flow in Deep Neural Networks (ICML 2019)
  - This work reexamines the "compression" aspect of the IB theory, they identified clustering of the learned features as the geometric underlying compression.
- Restricting the Flow: Information Bottlenecks for Attribution (ICLR 2019)
  - This work proposes a new IB-based method for attribution, based on the the intuition that IB could be used to find which parts of input or feature are redundant.
- Towards a Deep and Unified Understanding of Deep Neural Models in NLP (ICML 2019)
  - This shares a very similar idea to the former one, although they did not refer their method as IB, we still can use a same view to link the two paper. This work main focusing on NLP models and several findings are quite interesting.

Paper 2 - Motivation

- IB could be used to find which parts of input or feature are redundant.
- IB could directly estimate the amount of used information.
- IB does not constrain the internal network structure.

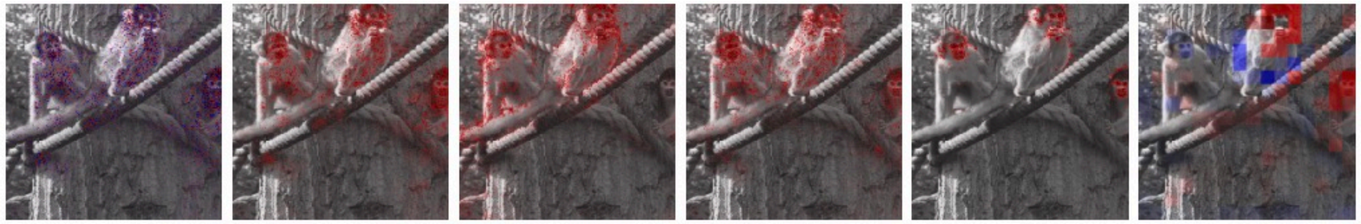
Paper 2 - Approach

- Inject noise to hidden feature Z, use IB to optimize the feature.
- Objective: $maxI[Y;Z] - \beta I[R;Z]$
  - $Z = \lambda(x)R + (1-\lambda(x))\epsilon, \epsilon \sim \mathcal{N}(\mu_R, \sigma_R^2)$
  - $I[Y;Z]$ could be optimize via cross entropy loss.
  - $I[R;Z] \approx \mathbb{E}_R[D_{KL}[P(Z|R)||Q(Z)]], Q(Z) \approx \mathcal{N}(\mu_R, \sigma_R^2)$ (variational inference)

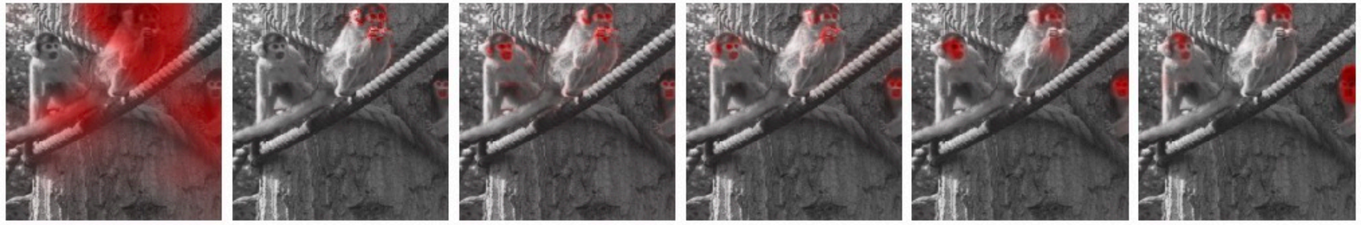


- Attribution Map: $\lambda(x)$

Paper 2 - Experiments



(a) Gradient (b) Saliency (c) SmoothGrad (d) Int. Grad. (e) GuidedBP (f) Occlusion-14



(g) Grad-CAM (h) G.Grad-CAM (i) PatternAttr. (j) LRP (k) *Per-Sample* (l) *Readout*

Model Task	ResNet-50 deg. 8x8	ResNet-50 deg. 14x14	VGG-16 deg. 8x8	VGG-16 deg. 14x14	ResNet-50 bbox	VGG-16 bbox
Random	0.000	0.000	0.000	0.000	0.167	0.167
Occlusion-8x8	0.162	0.130	0.267	0.258	0.296	0.312
Occlusion-14x14	0.228	0.231	0.402	0.404	0.341	0.358
Gradient	0.002	0.005	0.001	0.005	0.259	0.276
Saliency	0.287	0.305	0.326	0.362	0.363	0.393
GuidedBP	0.491	0.515	0.460	0.493	0.388	0.373
PatternAttribution	–	–	0.440	0.457	–	0.404
LRP	–	–	0.462	0.467	–	0.441
Int. Grad. (of Saliency)	0.401	0.424	0.420	0.453	0.372	0.396
SmoothGrad (of Saliency)	0.485	0.502	0.438	0.455	0.439	0.399
Grad-CAM	0.536	0.541	0.510	0.517	0.465	0.399
GuidedGrad-CAM	0.565	**0.577**	0.555	0.576	0.468	0.419
Per-Sample $\beta = 1/k$	**0.573**	0.573	0.581	0.583	0.606	0.566
Per-Sample $\beta = 10/k$	0.572	0.571	**0.582**	**0.585**	**0.620**	**0.593**
Per-Sample $\beta = 100/k$	0.534	0.535	0.542	0.545	0.574	0.568

Paper 2 - Conclusion

- This paper proposed a new novel attribution method based IB.
- This method guarantees that zero-valued attribution is not used for correct classification.
- This method is the only one to provide scores with units (bits).

Overview

- Estimating Information Flow in Deep Neural Networks (ICML 2019)
  - This work reexamines the "compression" aspect of the IB theory, they identified clustering of the learned features as the geometric underlying compression.
- Restricting the Flow: Information Bottlenecks for Attribution (ICLR 2019)
  - This work proposes a new IB-based method for attribution, based on the the intuition that IB could be used to find which parts of input or feature are redundant.
- Towards a Deep and Unified Understanding of Deep Neural Models in NLP (ICML 2019)
  - This shares a very similar idea to the former one, although they did not refer their method as IB, we still can use a same view to link the two paper. This work main focusing on NLP models and several findings are quite interesting.

Paper 3 - Motivation

- How to use mutual information to explain intermediate layer of DNN?
- How can the attribution measure enrich our capability of explaining DNNs and provide insights?

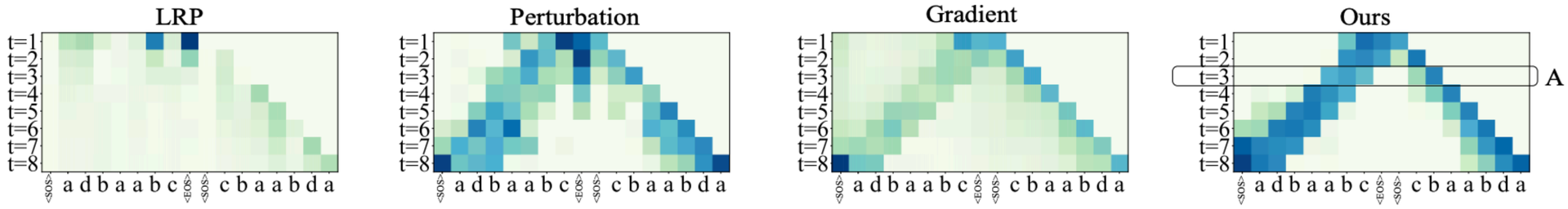
Paper 3 - Approach

- Share same idea with Paper I.
- Inject noise to Embedding feature , use IB to optimize the noise for l-th layer.
- Objective: $maxI[NN_l(X); Z] - \beta I[R; Z]$
  - $z_i = x_i + \epsilon_i, \epsilon_i \sim \mathcal{N}(0, \sigma_i^2)$
  - $I[NN_l(X); Z]$ could be optimize via MSE between $NN_l(X)$ and $NN_l(Z)$.
  - $I[R; Z] \approx \mathbb{E}_R[D_{KL}[P(Z|R)||Q(Z)]], Q(Z) \approx \mathcal{N}(0, \sigma^2)$ *
- Attribution Map: σ_i

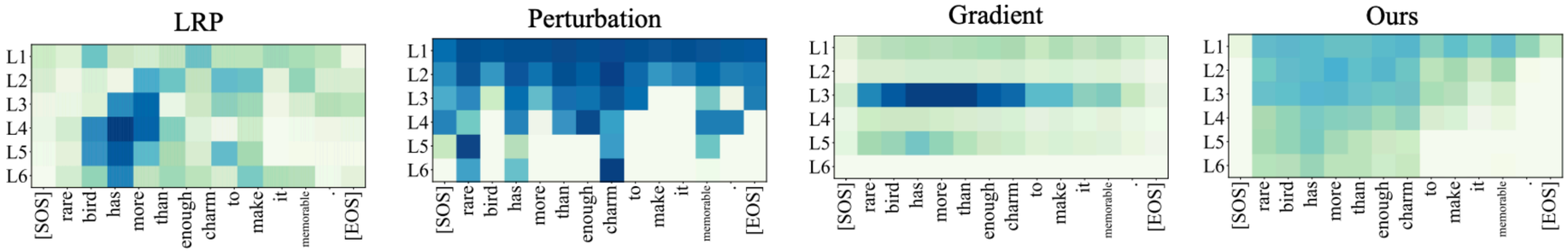
*The explanation is slightly different from the original paper, but we still could get the same loss.

Paper 3 - Experiments

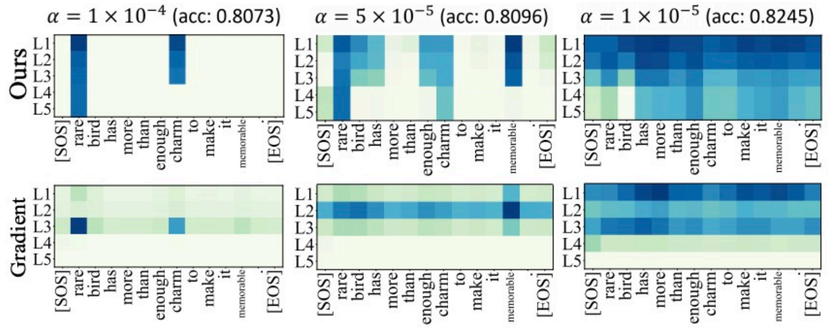
- Time-step



- Layer-wise



- Model-wise



Paper 3 - Experiments

- Important Words

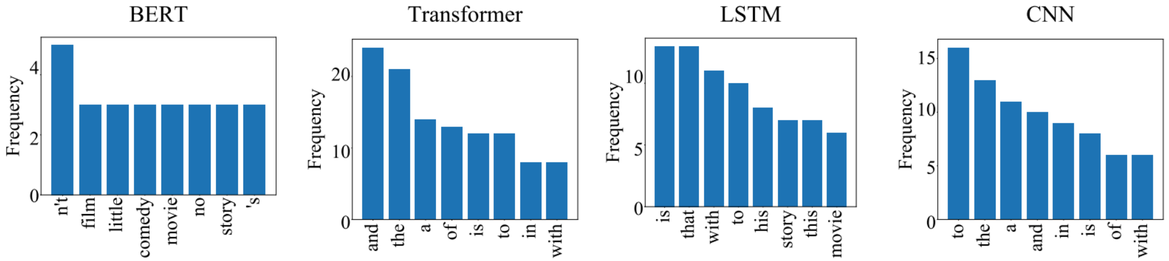
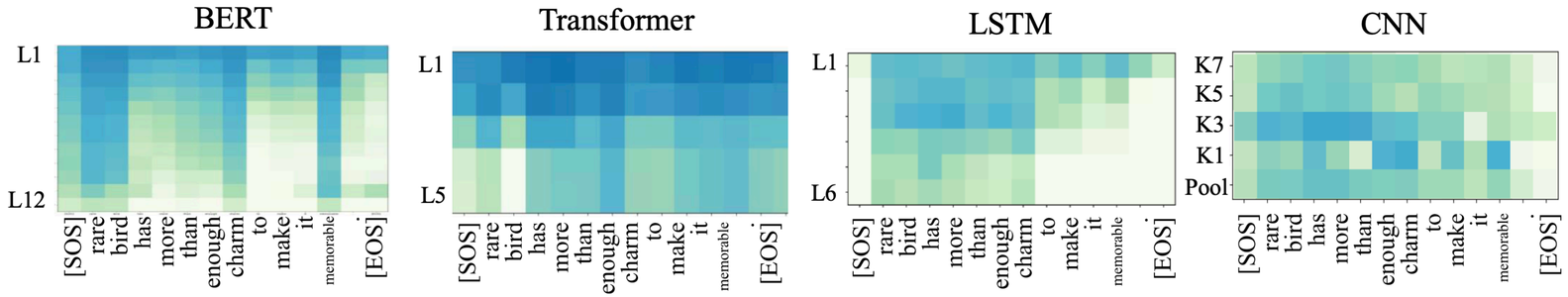
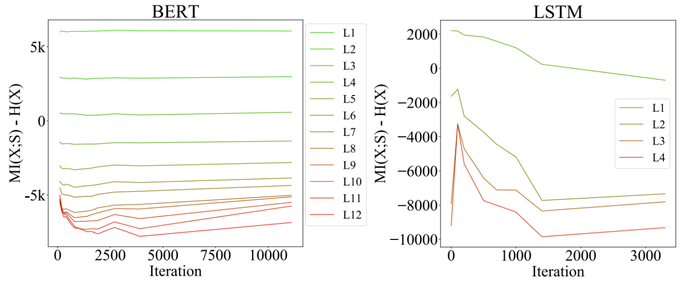


Figure 6. Words leveraged by each model for prediction (100 sampled sentences in SST-2 dataset).

- Layer-wise Information



- During Training



Paper 3 - Conclusion

- This paper proposed a new novel attribution method.
- BERT: Discarding meaningless words at first layers; Fine-tune stably.
- CNN: Focusing on subwords.
- LSTM: Focusing on subwords; Training phrase not stable.
- Transformer: Focusing on individual words.

Review

- Estimating Information Flow in Deep Neural Networks (ICML 2019)
  - This work reexamines the "compression" aspect of the IB theory, they identified clustering of the learned features as the geometric underlying compression.
- Restricting the Flow: Information Bottlenecks for Attribution (ICLR 2019)
  - This work proposes a new IB-based method for attribution, based on the the intuition that IB could be used to find which parts of input or feature are redundant.
- Towards a Deep and Unified Understanding of Deep Neural Models in NLP (ICML 2019)
  - This shares a very similar idea to the former one, although they did not refer their method as IB, we still can use a same view to link the two paper. This work main focusing on NLP models and several findings are quite interesting.

Summary

- The two attribution papers shows clear connections, and the difference are the perturbation methods and $I[Y; Z]$.
- Brain Storms
  - Could we use these attribution methods to explain MT model?
  - The perturbation methods make the after-perturbation distribution far away from the original distribution, that's may hurt attribution. (e.g., EOS problem)
  - Design an effective way to estimate MI is needed.
  - Compared with gradient-based methods, which are the advantages for IB-based?
  - How to evaluate the attribution methods performances?

Outline

- Information Bottleneck (IB) theory of Deep Learning (*Boyuan Wang*)
  - Deep Learning and the Information Bottleneck Principle (IEEE ITW 2015)
  - Opening the Black Box of Deep Neural Networks via Information (arXiv 2017)
  - On the Information Bottleneck Theory of Deep Learning (ICLR 2018)

- Information Flow: IB for Attribution (*Cunxiao Du*)
  - Estimating Information Flow in Deep Neural Networks (ICML 2019)
  - Towards a Deep and Unified Understanding of Deep Neural Models in NLP (ICML 2020)
  - Restricting the Flow: Information Bottlenecks for Attribution (ICLR 2020)

- Variational Information Bottleneck (VIB) and its Application (***Wenxuan Wang***)
  - Deep Variational Information Bottleneck (ICLR 2017)
  - Specializing Word Embeddings (for Parsing) by Information Bottleneck (EMNLP 2019, Best Paper)

Overview

- Information bottleneck objective function is hard to calculate.
- Give information bottleneck objective function a lower bound. The process of optimize the lower bound can be formulated as training a variational neural network.
- Trained on the information bottleneck objective function,
  - Model outperform other regularize term.
  - Model are more robust.
  - Representation are more compact to specific task.

Overview

- Information bottleneck objective function is hard to calculate.
- Give information bottleneck objective function a lower bound. The process of optimize the lower bound can be formulated as training a variational neural network.
- Trained on the information bottleneck objective function,
  - Model outperform other regularize term.
  - Model are more robust.
  - Representation are more compact to specific task.

- Deep Variational Information Bottleneck (ICLR 2017)
  - VIB is a variational approximation to IB, which allows to parameterize the information bottleneck model using a neural network and leverage the reparameterization trick for efficient training.
- Specializing Word Embeddings (for Parsing) by Information Bottleneck (EMNLP 2019)
  - Adopt VIB to nonlinearly compress the pre-trained embeddings, keeping only the information that helps a discriminative parser.

Overview

- Deep Variational Information Bottleneck (ICLR 2017)
  - VIB is a variational approximation to IB, which allows to parameterize the information bottleneck model using a neural network and leverage the reparameterization trick for efficient training.
- Specializing Word Embeddings (for Parsing) by Information Bottleneck (EMNLP 2019)
  - Adopt VIB to nonlinearly compress the pre-trained embeddings, keeping only the information that helps a discriminative parser.

Paper1 - Motivation

- How to approximation to the information bottleneck objective function?
  - The main drawback of the IB principle is that computing mutual information is, in general, computationally challenging.
- How to train a better representation/model under information bottleneck principal?

Paper1 - Review: IB View of DNN

- Given input X and output Y:
  - For any intermediate layer's output representation Z, we want to maximally informative about target Y:

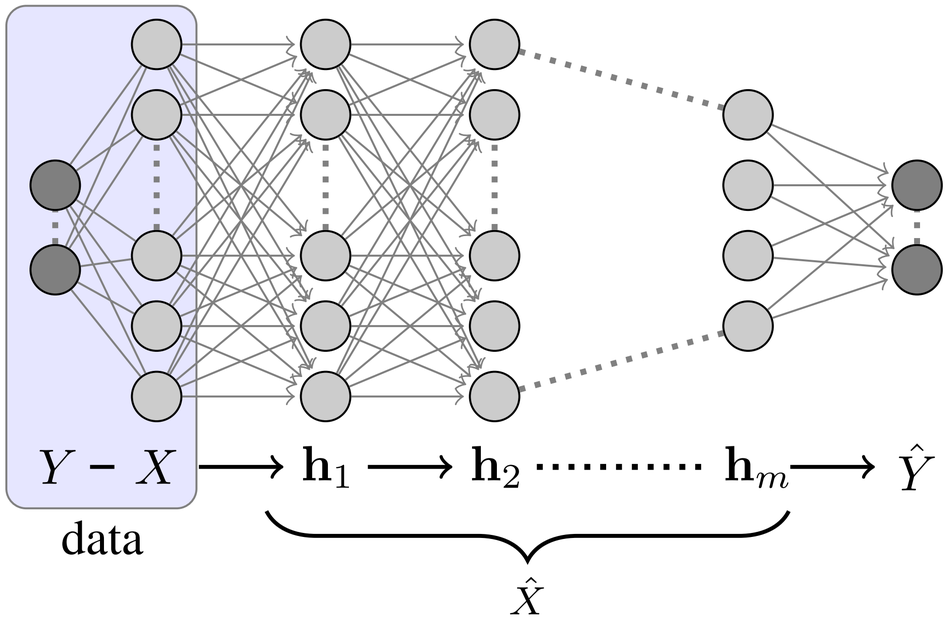
$$I(Z, Y; \boldsymbol{\theta}) = \int dx\ dy\ p(z, y|\boldsymbol{\theta}) \log \frac{p(z, y|\boldsymbol{\theta})}{p(z|\boldsymbol{\theta})p(y|\boldsymbol{\theta})}$$

  - Here our goal is to learn an encoding Z that is maximally expressive about Y while being maximally compressive about X (IB).

$$\max_{\boldsymbol{\theta}} I(Z, Y; \boldsymbol{\theta}) \text{ s.t. } I(X, Z; \boldsymbol{\theta}) \leq I_c$$

  - Using Lagrange method, we can maximize the object function as:

$$R_{IB}(\boldsymbol{\theta}) = I(Z, Y; \boldsymbol{\theta}) - \beta I(Z, X; \boldsymbol{\theta})$$



Paper1 - Approach:VIB

- In this paper, we propose to use variational inference to construct a lower bound on the IB objective, which can be formulated as training a variational neural network.

- **Variational Inference**
  - p(x) is hard to compute. We can use a more simple distribution q(x), like a Gaussian distribution, to approximate p(x).
  - Calculating distribution -> Minimizing KL(p(x), q(x)).
  - Variational Inference Model.

Paper1 - Derivation of Lower Bound

- The IB objective is $I(Z,Y) - \beta I(Z,X)$

$$I(Z,Y) = \int dy\, dz\, p(y,z) \log \frac{p(y,z)}{p(y)p(z)} = \int dy\, dz\, p(y,z) \log \frac{p(y|z)}{p(y)}$$

let $q(y|z)$ be a variational approximation to $p(y|z)$

$$\mathrm{KL}[p(Y|Z), q(Y|Z)] \geq 0 \implies \int dy\, p(y|z) \log p(y|z) \geq \int dy\, p(y|z) \log q(y|z)$$

- Hence:

$$\begin{aligned} I(Z,Y) &\geq \int dy\, dz\, p(y,z) \log \frac{q(y|z)}{p(y)} \\ &= \int dy\, dz\, p(y,z) \log q(y|z) - \int dy\, p(y) \log p(y) \\ &= \int dy\, dz\, p(y,z) \log q(y|z) + H(Y)\,. \end{aligned}$$

Paper1 - Derivation of Lower Bound

- Since

$$p(y,z) = \int dx\, p(x,y,z) = \int dx\, p(x)p(y|x)p(z|x)$$

- We have a lower bound on the first term:

$$I(Z,Y) \geq \int dx\, dy\, dz\, p(x)p(y|x)p(z|x) \log q(y|z)$$

- Requiring
  - samples from both our joint data distribution
  - samples from stochastic encoder
  - a tractable variational approximation in q(y|z).

Paper1 - Derivation of Lower Bound

- Now let's consider the I(Z,X) term in $R_{IB}(\boldsymbol{\theta}) = I(Z, Y; \boldsymbol{\theta}) - \beta I(Z, X; \boldsymbol{\theta})$

$$I(Z, X) = \int dz\, dx\, p(x, z) \log \frac{p(z|x)}{p(z)} = \int dz\, dx\, p(x, z) \log p(z|x) - \int dz\, p(z) \log p(z)$$

- Let r(z) be a variational approximation to p(z).

$$\mathrm{KL}[p(Z), r(Z)] \geq 0 \implies \int dz\, p(z) \log p(z) \geq \int dz\, p(z) \log r(z)$$

$$I(Z, X) \leq \int dx\, dz\, p(x) p(z|x) \log \frac{p(z|x)}{r(z)}$$

Paper1 - Computing IB Objective in Practice

- So we get our IB objective lower bound:

$$I(Z,Y) - \beta I(Z,X) \geq \int dx\, dy\, dz\, p(x)p(y|x)p(z|x) \log q(y|z)$$
$$- \beta \int dx\, dz\, p(x)p(z|x) \log \frac{p(z|x)}{r(z)} = L$$

- In practice, we can approximate p(x, y) using the empirical data distribution

$$p(x,y) = \frac{1}{N} \sum_{n=1}^{N} \delta_{x_n}(x)\delta_{y_n}(y)$$

$$L \approx \frac{1}{N} \sum_{n=1}^{N} \left[\int dz\, p(z|x_n) \log q(y_n|z) - \beta\, p(z|x_n) \log \frac{p(z|x_n)}{r(z)} \right]$$

Paper1 - Computing IB Objective in Practice

- Suppose we use an encoder of the form:

$$p(z|x) = \mathcal{N}(z|f_e^{\mu}(x), f_e^{\Sigma}(x))$$

where f is an MLP which outputs both the K-dimensional mean μ of z as well as the K × K covariance matrix Σ.

- Using the reparameterization trick

$$p(z|x)dz = p(\epsilon)d\epsilon$$

where z = f(x, $\boldsymbol{\varepsilon}$) is a deterministic function of x and the Gaussian random variable $\boldsymbol{\varepsilon}$

- We finally get

$$J_{IB} = \frac{1}{N}\sum_{n=1}^{N} \mathbb{E}_{\epsilon\sim p(\epsilon)}\left[-\log q(y_n|f(x_n,\epsilon))\right] + \beta\,\mathrm{KL}\left[p(Z|x_n), r(Z)\right]$$

Paper1 - Implementation Details

- **Encoder**:

$$p(z|x) = \mathcal{N}(z|f_e^{\mu}(x), f_e^{\Sigma}(x))$$

where f is an 2 layers MLP of the form $784 - 1024 - 1024 - 2K$, where K is the size of the bottleneck. The first K outputs from f encode μ, the remaining K outputs encode σ.

- **Decoder**: Simple logistic regression model of the form

$$q(y|z) = \mathcal{S}(y|f_d(z)),$$

where S is softmax function and

$$f_d(z) = Wz + b$$

maps K dimensional latent code to 10 classes:

$$r(z) = \mathcal{N}(z|0, I)$$

Paper1 - Experiments on MNIST

- VIB Outperforms other regularization.
- When trade-off term $\beta > 0.01$ (bigger compression rate) , error rate shoot up.
- Different bottleneck dimension K follow the same property.
- The VIB objective makes models significantly more robust to such adversarial examples.

Model	error
Baseline	1.38%
Dropout	1.34%
Dropout (Pereyra et al., 2017)	1.40%
Confidence Penalty	1.36%
Confidence Penalty (Pereyra et al., 2017)	1.17%
Label Smoothing	1.40%
Label Smoothing (Pereyra et al., 2017)	1.23%
VIB ($\beta = 10^{-3}$)	**1.13%**

Paper1 - Experiments on ImageNet

- Setup
  - Pre-trained Inception ResnetV2 as feature extractor to extract representation at the penultimate layer.
  - The experiment regime is identical to the MNIST task.
- Results
  - In all cases the accuracy is lower than the original 80.4% accuracy.
  - But VIB network is more robust for adversarial attacking.

Paper1 - Conclusion

- Present a variational approximation to the information bottleneck.
- Using neural network and SGD to optimize the IB objective function.
- Models trained with the VIB objective outperform those trained with other regularization.
- Models trained with the VIB objective are more robust to adversarial attacking.

Overview

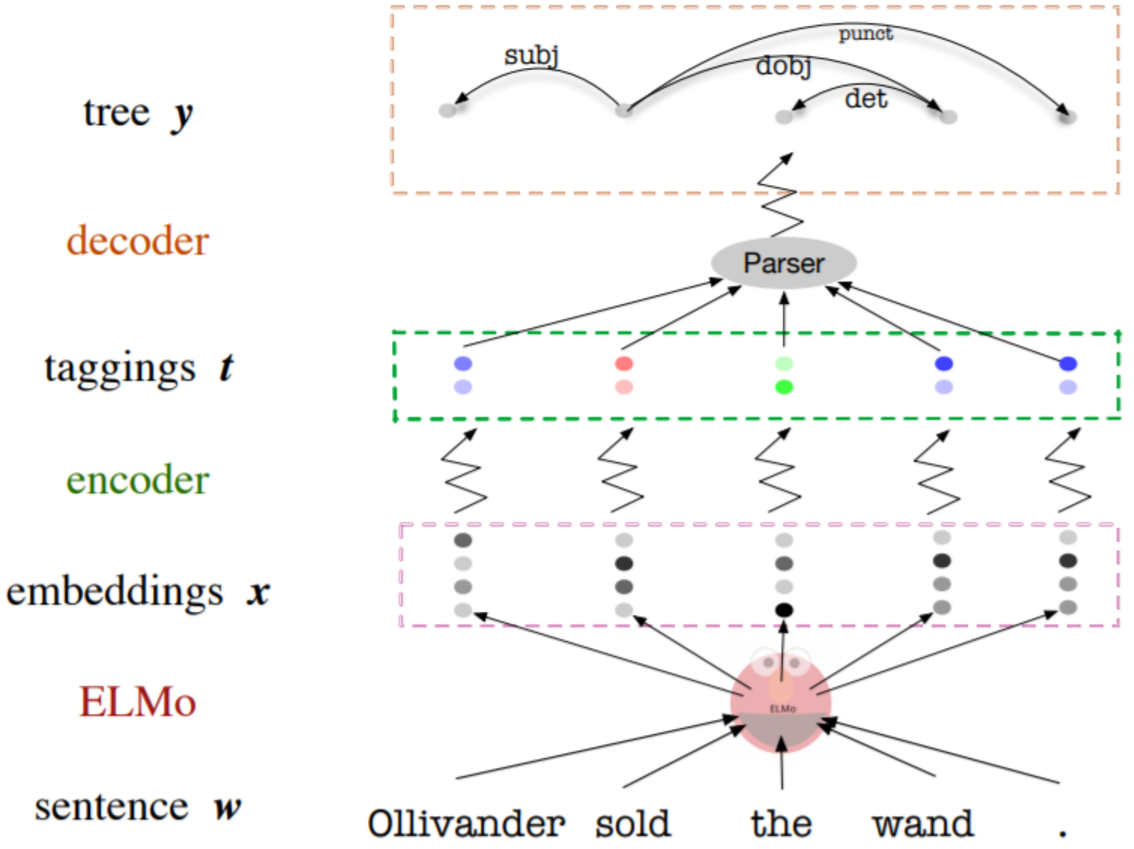
- Deep Variational Information Bottleneck (ICLR 2017)
  - VIB is a variational approximation to IB, which allows to parameterize the information bottleneck model using a neural network and leverage the reparameterization trick for efficient training.
- Specializing Word Embeddings (for Parsing) by Information Bottleneck (EMNLP 2019)
  - Adopt VIB to nonlinearly compress the pre-trained embeddings, keeping only the information that helps a discriminative parser.

Paper II - Motivation

- Pre-trained embedding, like ELMo and BERT, contain rich syntactic and semantic information but are too big.
- How to compress word/token embedding with useful information?

Paper II - Approach

- Compress the embeddings by extracting the information needed to reconstruct parse trees (can be other tasks).
- The compression function is trained by VIB.



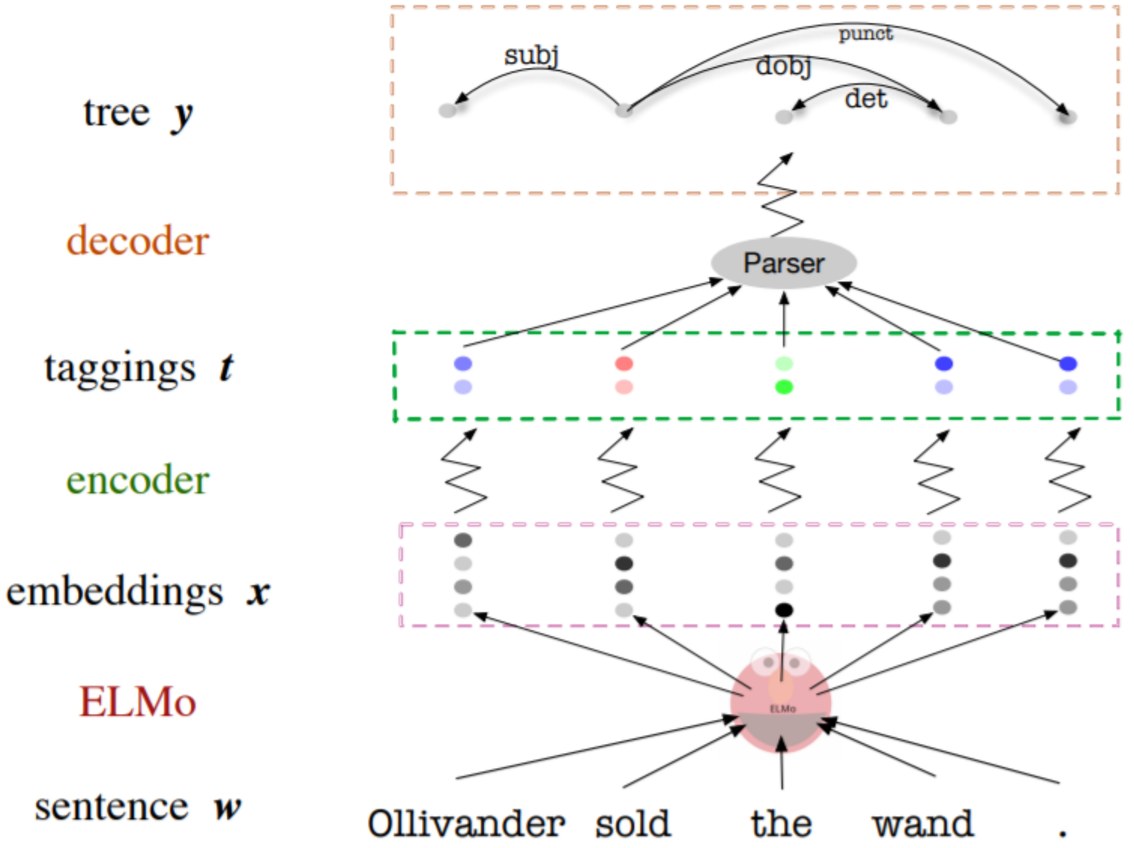
Paper II - Approach

- Add terms to control the context sensitivity of the extracted tags:

$$\mathcal{L}_{IB} = -\mathrm{I}(Y;T) + \beta\,\mathrm{I}(X;T) + \gamma \sum_{i=1}^{n} \mathrm{I}(T_i; X|\hat{X}_i)$$

where

- T_i is the tag associated with the i_th word.
- X_i is the ELMo token embedding of the ith word.
- \hat{X_i} is the same word's ELMo type embedding (before context is incorporated).

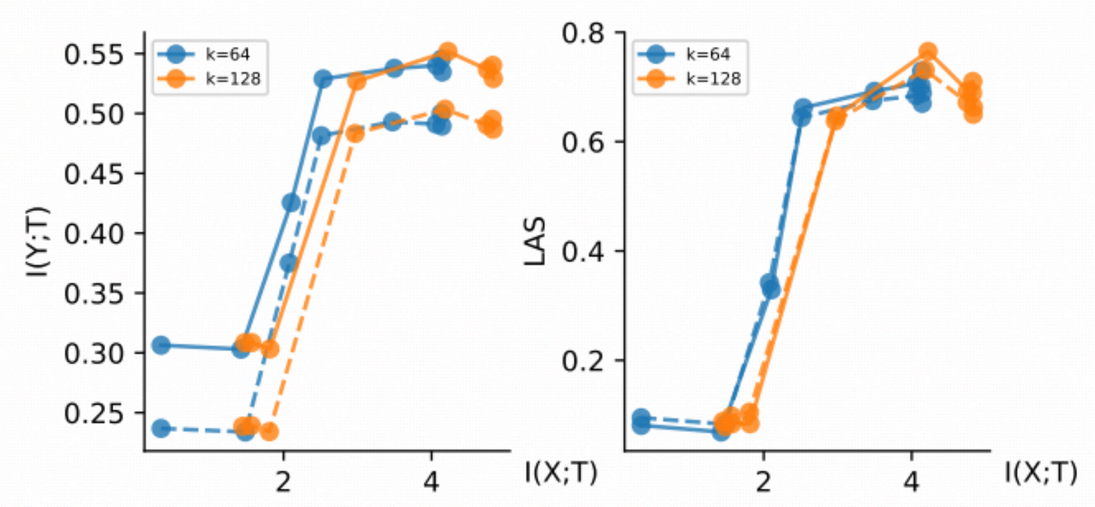


Paper II - Approach

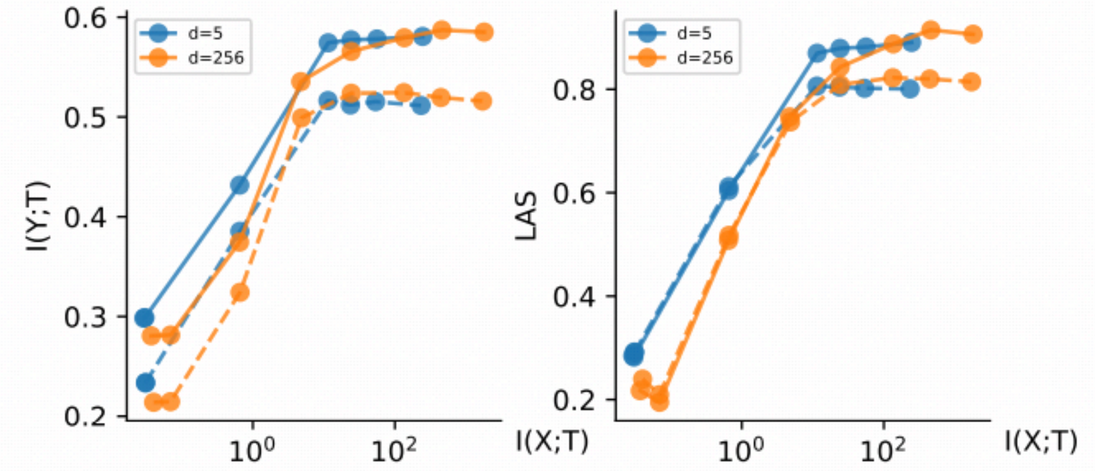
- **Encoder**
  - To obtain continuous tags, define p (t_i | x_i) such that $t_i \in R$ is Gaussian-distributed with mean vector and diagonal covariance matrix computed from the ELMo word vector x_i via a feedforward neural network with 2d outputs.
  - To obtain discrete tags, define p (t_i | x_i) such that $t_i \in \{1, \ldots, k\}$ follows a softmax distribution, where the k softmax parameters are similarly computed by a feedforward network with k outputs.
- **Decoder**
  - We use the deep biaffine dependency parser as our variational distribution q(y | t), which functions as the decoder.
  - This parser uses a Bi-LSTM to extract features from compressed tags or vectors and assign scores to each tree edge

Paper II - Experiments

- After a critical point, the additional information retained in T does not contribute much to predicting Y, *which means it is fine to compress.*



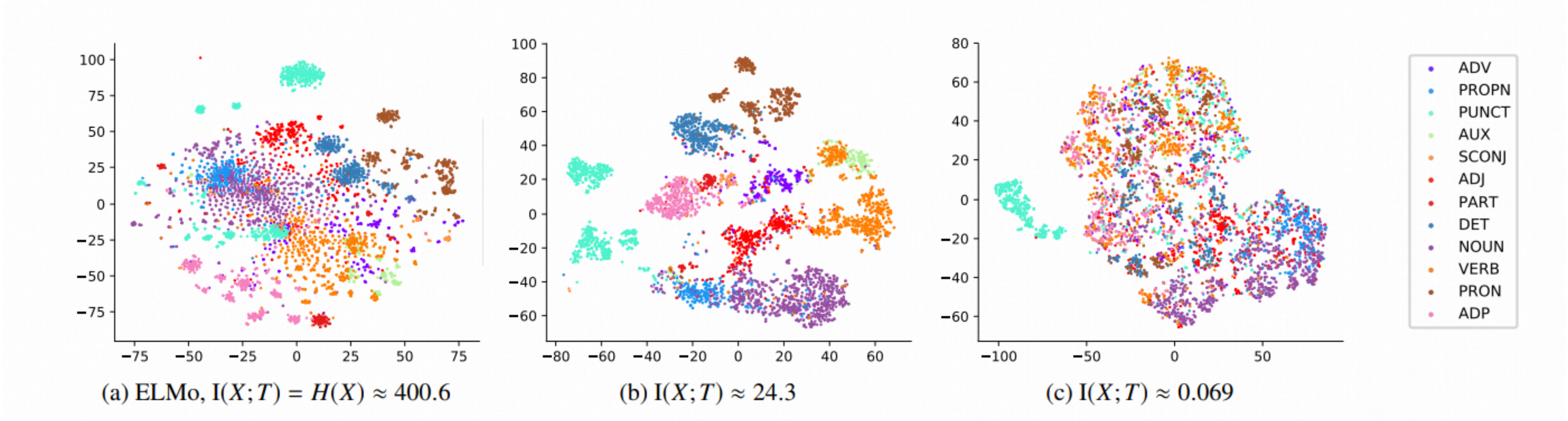
(a) Discrete Version



(b) Continuous Version

Paper II - Experiments

- Visualization of VIB model on the projected space of the tag.
- With moderate compression rate, our automatically induce stochastic tags that correlate with gold POS tags.



(a) ELMo, $\mathrm{I}(X;T) = H(X) \approx 400.6$

(b) $\mathrm{I}(X;T) \approx 24.3$

(c) $\mathrm{I}(X;T) \approx 0.069$

Paper II - Experiments

- The VIB method outperforms other baselines for parsing.

Models	Arabic	Hindi	English	French	Spanish	Portuguese	Russian	Chinese	Italian
Iden	0.751	**0.870**	0.824	0.784	0.808	0.813	0.783	0.709	**0.863**
PCA	0.743	**0.866**	0.823	0.749	0.802	0.808	0.777	0.697	0.857
MLP	0.759	**0.871**	0.839	0.816	**0.835**	0.821	0.800	0.734	**0.867**
VIBc	**0.779**	**0.866**	**0.851**	**0.828**	**0.837**	**0.836**	**0.814**	**0.754**	**0.867**
POS	0.652	0.713	0.712	0.718	**0.739**	**0.743**	**0.662**	0.510	0.779
VIBd	**0.672**	**0.736**	**0.742**	**0.723**	**0.725**	0.710	**0.651**	**0.591**	**0.781**

Paper II - Conclusion

- Propose a variational information bottleneck (VIB) method to nonlinearly compress the pre-trained embeddings, keeping only the information that helps a discriminative parser.
- Compress each word embedding to
  - A discrete tag > capture most of the information in POS tag annotations.
  - A continue vector > yields a more accurate parser on 8 of 9 languages

Review

- Information bottleneck objective function is hard to calculate.
- Give information bottleneck objective function a lower bound. The process of optimize the lower bound can be formulated as training a variational neural network.
- Trained on the information bottleneck objective function,
  - Model outperform other regularize term.
  - Model are more robust.
  - Representation are more compact to specific task.

- Deep Variational Information Bottleneck (ICLR 2017)
  - VIB is a variational approximation to IB, which allows to parameterize the information bottleneck model using a neural network and leverage the reparameterization trick for efficient training.
- Specializing Word Embeddings (for Parsing) by Information Bottleneck (EMNLP 2019)
  - Adopt VIB to nonlinearly compress the pre-trained embeddings, keeping only the information that helps a discriminative parser.

Summary

- Give information bottleneck objective function a lower bound. The process of optimize the lower bound can be formulated as training a variational neural network.
- Trained on the information bottleneck objective function
  - Model outperform than using other regularize term.
  - Model are more robust.
  - Representation are more compact to specific task.

Overview

- Information Bottleneck (IB) theory of Deep Learning
  - *IB expresses the trade-off between the mutual information measures the amount of information that **the hidden layer** contains about the **input** and the **output**.*
    - Deep Learning and the Information Bottleneck Principle (IEEE ITW 2015)
    - Opening the Black Box of Deep Neural Networks via Information (arXiv 2017)
    - On the Information Bottleneck Theory of Deep Learning (ICLR 2018)
- Information Flow: IB for Attribution
  - *IB for attribution to interpret the decision-making of DNNs. Noises are added to intermediate feature maps to **restrict the flow of information**, which quantify (in bits) how much information input regions provide.*
    - Estimating Information Flow in Deep Neural Networks (ICML 2019)
    - Towards a Deep and Unified Understanding of Deep Neural Models in NLP (ICML 2020)
    - Restricting the Flow: Information Bottlenecks for Attribution (ICLR 2020)
- Variational Information Bottleneck (VIB) and its Application
  - *VIB is a variational approximation to IB, which allows to parameterize the information bottleneck model using a neural network and leverage the reparameterization trick for efficient training.*
    - Deep Variational Information Bottleneck (ICLR 2017)
    - Specializing Word Embeddings (for Parsing) by Information Bottleneck (EMNLP 2019, Best Paper)

Conclusions

- We believe that the direction of *information flow in deep learning* will be a long and fruitful journey. Machine translation is well suited for validating approaches on this topic.
- How is information correctly flowed from the source to the target?
  - The theoretical understanding of DL remains unsatisfactory: *trainability*, *generalizability*, and *expressivity*.
  - Defining information is crucial: *information in the activations* (Tishby et al. 2000; Alemi et al. 2017; Saxe et al. 2018) and *information in the weights* (Achille and Soatto 2018; Achille and Soatto 2020).
  - How to measure information flow beyond adding noises?