Probabilistic Transformer

Deciphering transformers as a white-box probabilistic model

Kewei Tu (joint work with **Haoyi Wu**)

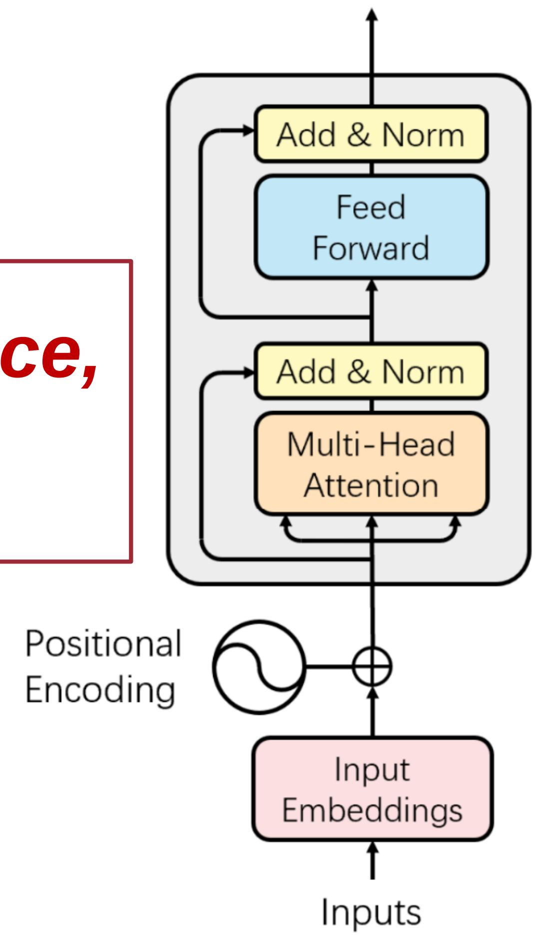
ShanghaiTech University

上海科技大学
ShanghaiTech University

Transformers

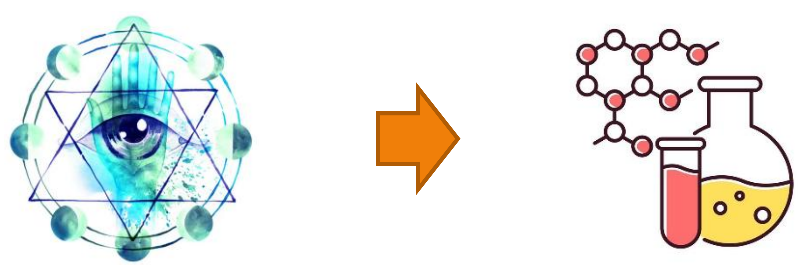
- Transformers are the foundation of LLMs and beyond!

Transformers may work in practice, but do they work in theory??



In a nutshell…

- We propose probabilistic transformers
  - A (non-neural) probabilistic model over natural language syntax
  - Yet, its computation graph is strikingly similar to a transformer encoder!
- Implications?
  - A white-box transformer, which may enable…
  - …interpretation & analyses of existing transformer variants
  - …design of new transformer variants in a principled way



Outline

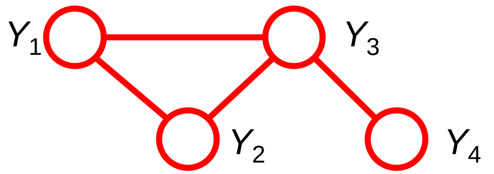
- Preliminary
  - CRF, MFVI, unfolding as GNN
- Probabilistic transformers
  - Model
  - Inference
- Similarities to transformers

Outline

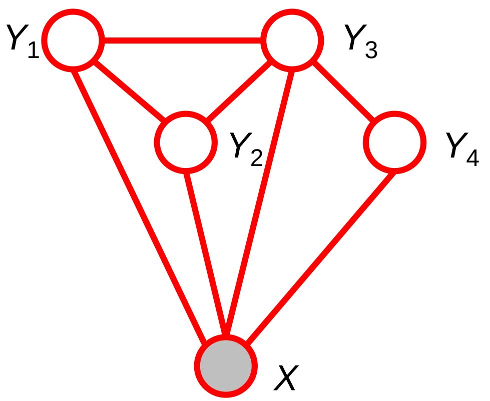
- Preliminary
  - CRF, MFVI, unfolding as GNN
- Probabilistic transformers
  - Model
  - Inference
- Similarities to transformers

Probabilistic Models

- MRF = undirected graph + potential functions
  - For each clique (or max clique), define a potential function
  - A joint probability is proportional to the product of potentials

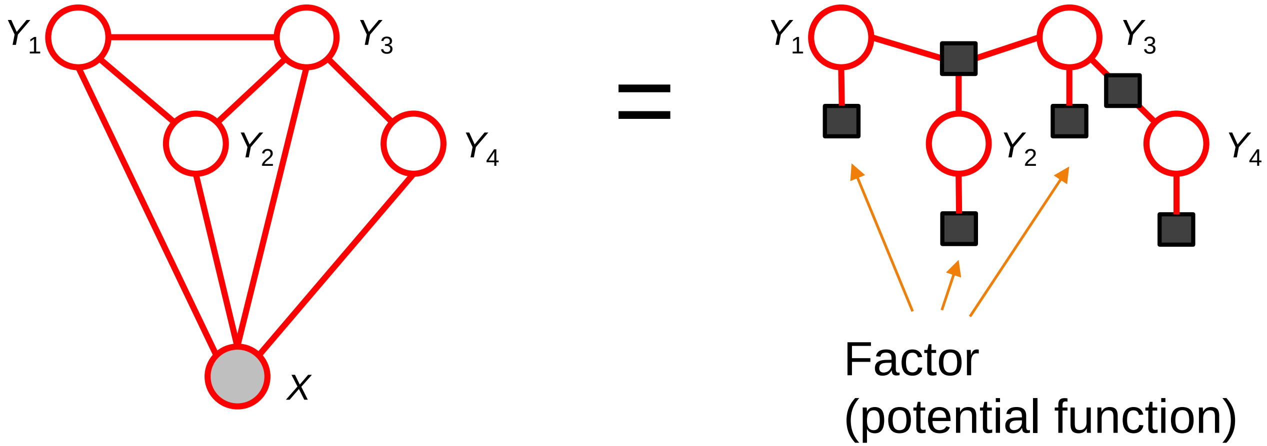


- Conditional Random Fields (CRF)
  - An extension of MRF where everything is conditioned on an input



Factor Graph

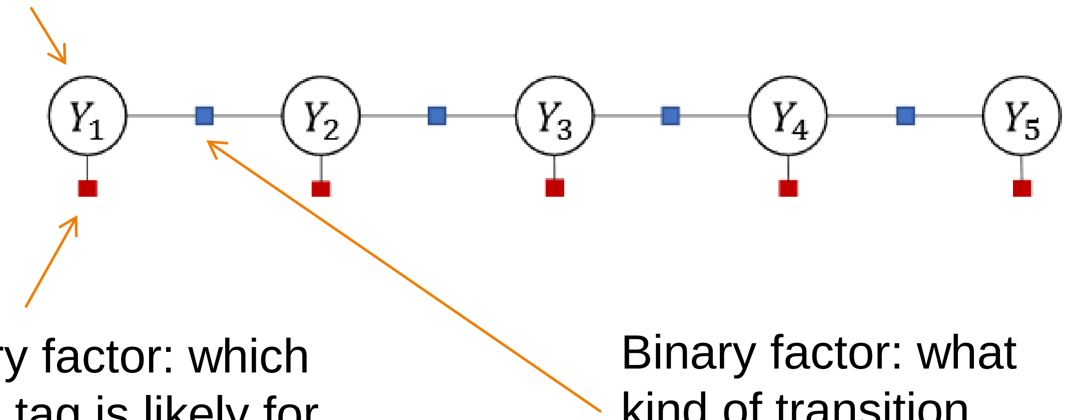
- A factor graph explicitly shows the potential functions (aka factors) in an MRF/CRF



Factor Graph

- Example: HMM for POS tagging

POS tag of a word,
e.g., noun, verb, …



Unary factor: which POS tag is likely for the word

Binary factor: what kind of transition between consecutive POS tags is likely

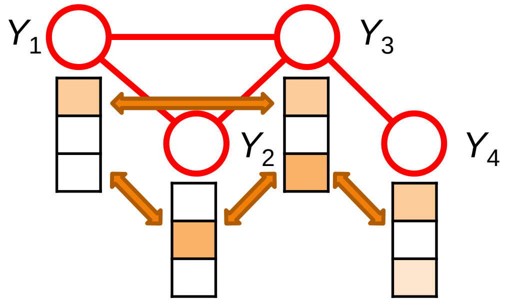
Inference over MRF/CRF

- Inference
  - Some variables are known (evidence)
  - Some variables are latent (we want to marginalize them)
  - Some variables are what we care about (query)

- Exact inference is hard or even intractable in general
- Iterative algorithms for approximate inference
  - Mean-field Variational Inference
  - Loopy Belief Propagation
  - …

Inference over MRF/CRF

- Iterative algorithms for approximate inference
- At each iteration:
  - Compute an intermediate vector (e.g., a discrete distribution) for each random variable…
  - …based on the vectors from the previous iteration
  - …following a fixed graph structure
  - …using fixed model parameters
  - …in a fully differentiable way



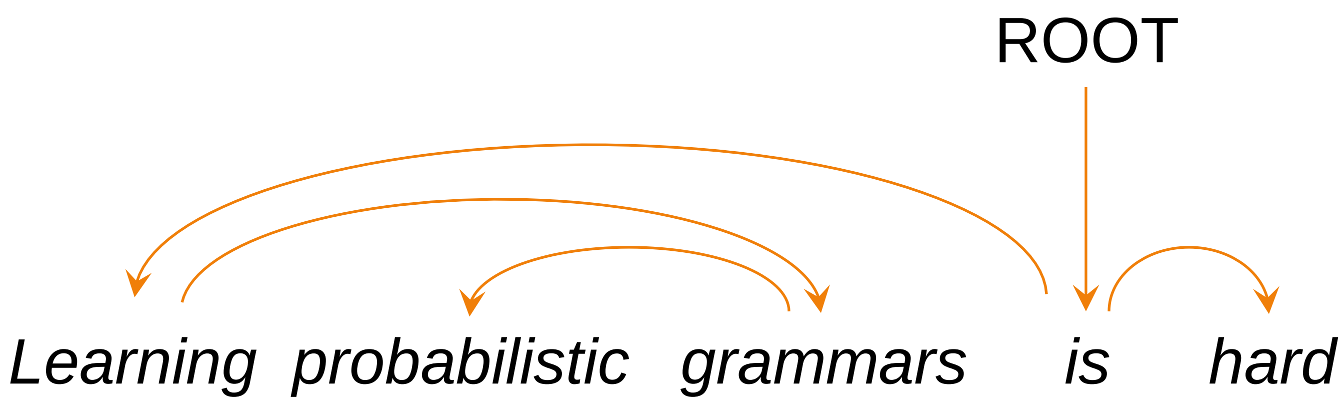
Inference can be unfolded as a Graph Neural Network!

Outline

- Preliminary
  - CRF, MFVI, unfolding as GNN
- Probabilistic transformers
  - Model
  - Inference
- Similarities to transformers

Dependency parsing

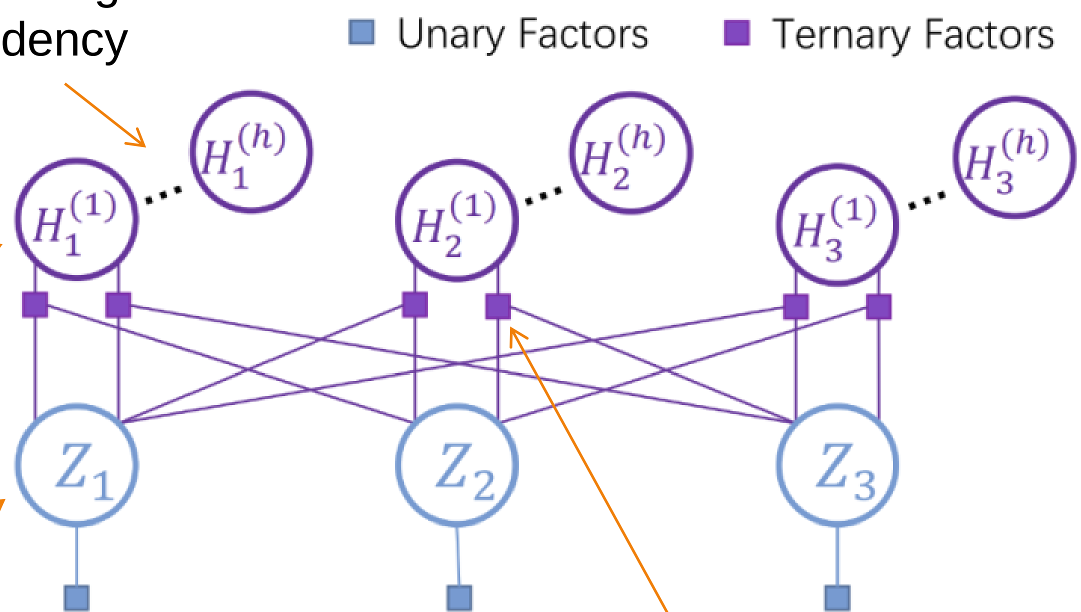
- Identify binary relations (i.e., dependencies) between words that form a tree



- Head-selection: a simplification of dependency parsing
  - Identify the parent word (i.e., dependency head) of each word
  - No tree constraint

Our CRF: head selection over latent word representation

h channels, allowing multiple dependency structures



$H_i \in \{1, \dots, n\}$: index of the dependency head of word i

Z_i: a discrete variable, representing property of word i in the input sentence

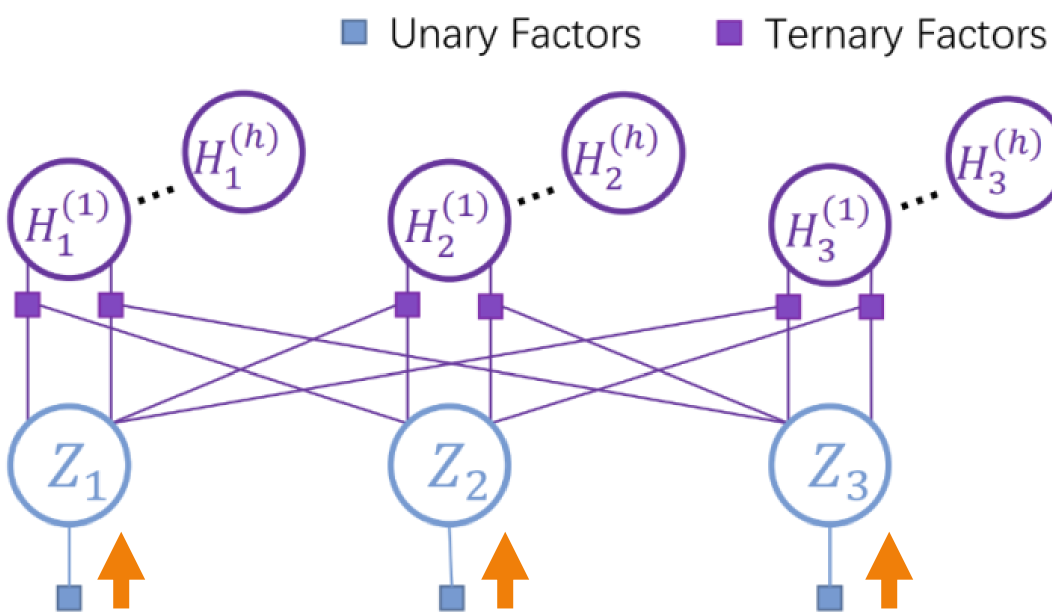
Unary factor: compatibility of Z_i and word i

Ternary factor: compatibility between Z_i and Z_j if word j is the dependency head of word i ($H_i = j$)

Mean Field Variational Inference (MFVI)

- Iteratively recompute marginal distribution $Q(\cdot)$ of each variable

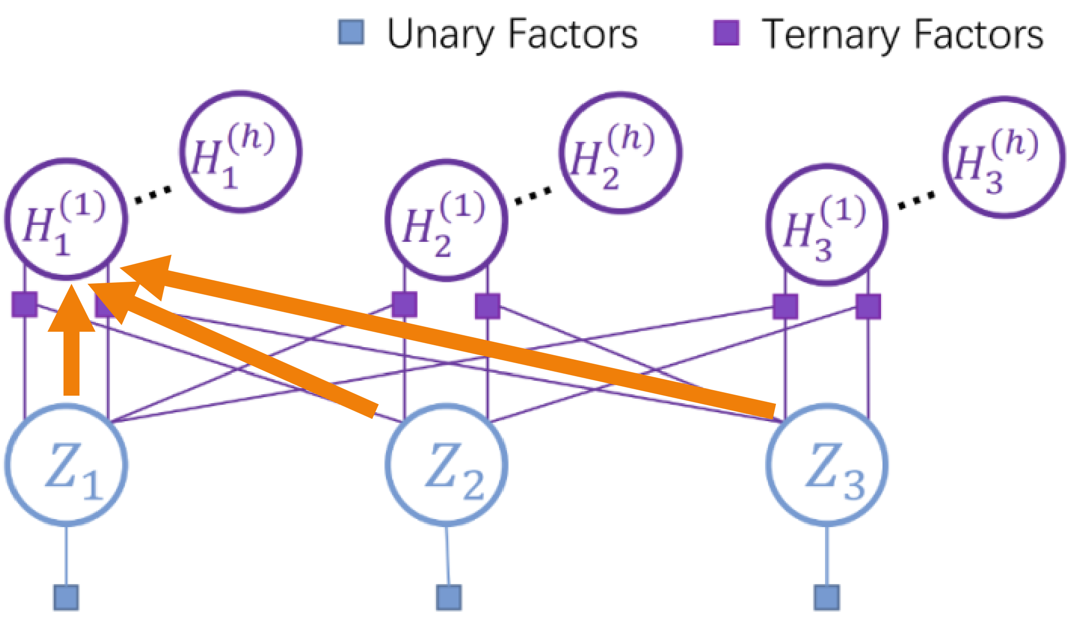
- Initialize $Q(Z_i)$



Mean Field Variational Inference (MFVI)

- Iteratively recompute marginal distribution $Q(\cdot)$ of each variable

- Initialize $Q(Z_i)$
- Repeat
  - Recompute $Q(H_i)$

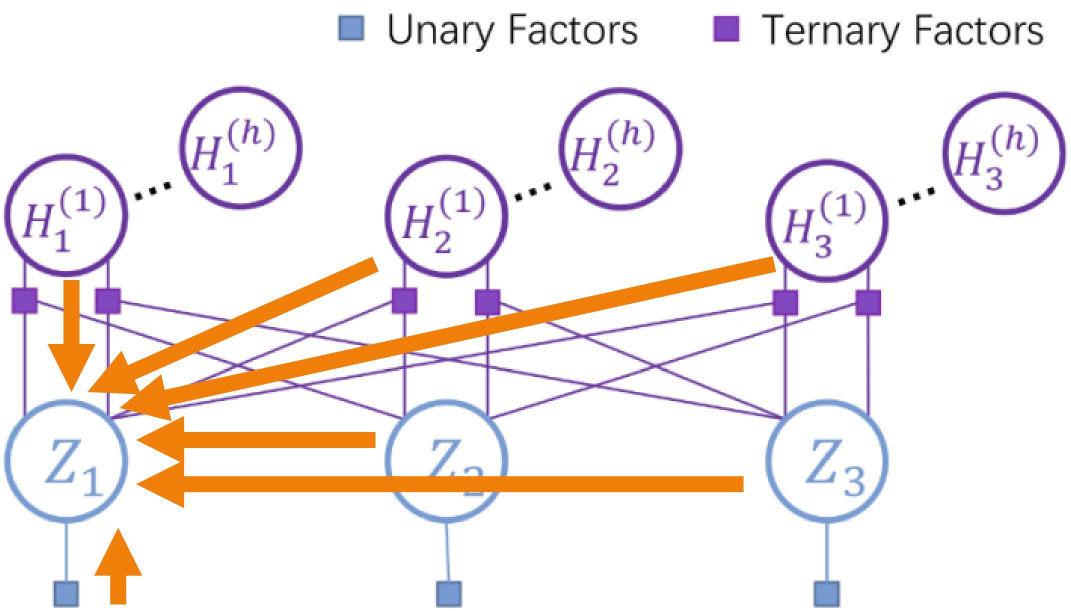


Mean Field Variational Inference (MFVI)

- Iteratively recompute marginal distribution $Q(\cdot)$ of each variable

See paper for all the math

- Initialize $Q(Z_i)$
- Repeat
  - Recompute $Q(H_i)$
  - Recompute $Q(Z_i)$

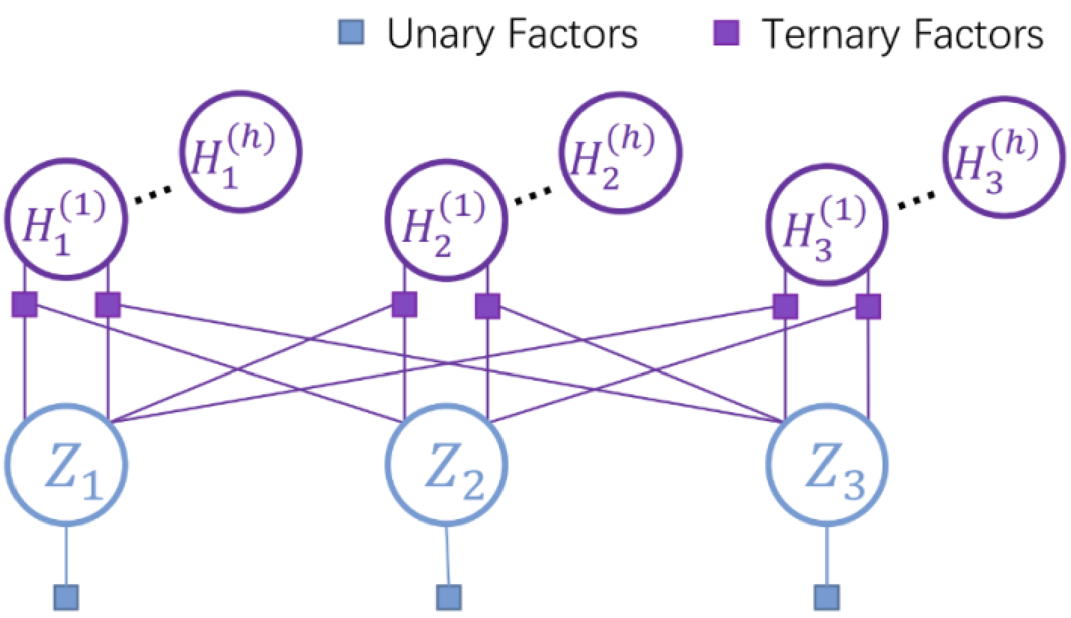


Mean Field Variational Inference (MFVI)

- Iteratively recompute marginal distribution $Q(\cdot)$ of each variable

- Initialize $Q(Z_i)$
- Repeat
  - Recompute $Q(H_i)$
  - Recompute $Q(Z_i)$

- $Q(Z_i)$ can be seen as a contextual representation of word i



Further refinements

- Entropic Frank-Wolfe algorithm
  - Generalization of MFVI
- Rank decomposition of ternary factor
  - $T\left(Z_i, Z_j\right) = \sum_r U(Z_i, r) \times V(Z_j, r)$
- Dependency root
- Incorporating word distance in ternary factors
- …

Learning

- Inference can be unfolded as a Graph Neural Network
- Learning can be done by back-propagation
  - Model parameters: unary & ternary factors
  - Objective function: MLM, downstream tasks, …

Outline

- Preliminary
  - CRF, MFVI, unfolding as GNN
- Probabilistic transformers
  - Model
  - Inference
- Similarities to transformers

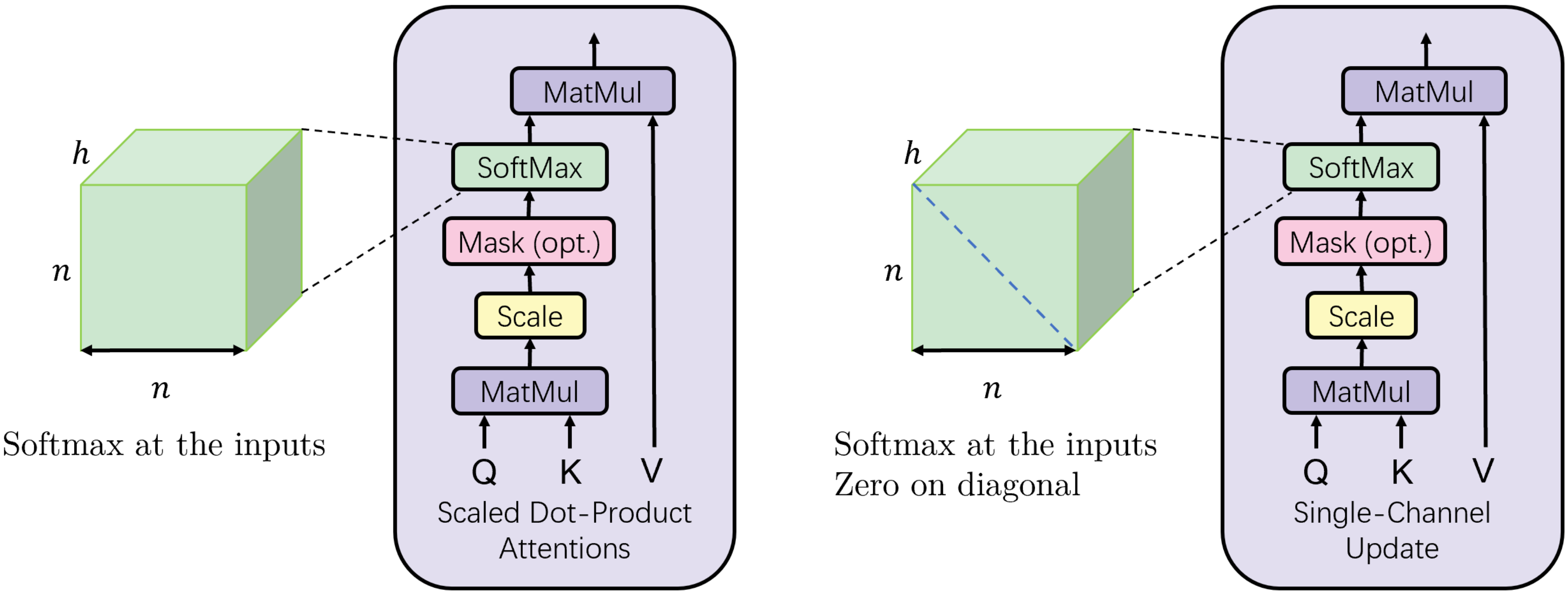
Similarities to transformers

- We compare the computation graph of MFVI on our CRF with transformers
  - Assumption: symmetric ternary factors
- Roughly speaking:

Our intermediate distributions $Q(H_i)$ over dependency heads	$\approx$	Self-attention scores in a transformer
Our intermediate distributions $Q(Z_i)$ over latent word representations	$\approx$	Intermediate word embeddings in a transformer

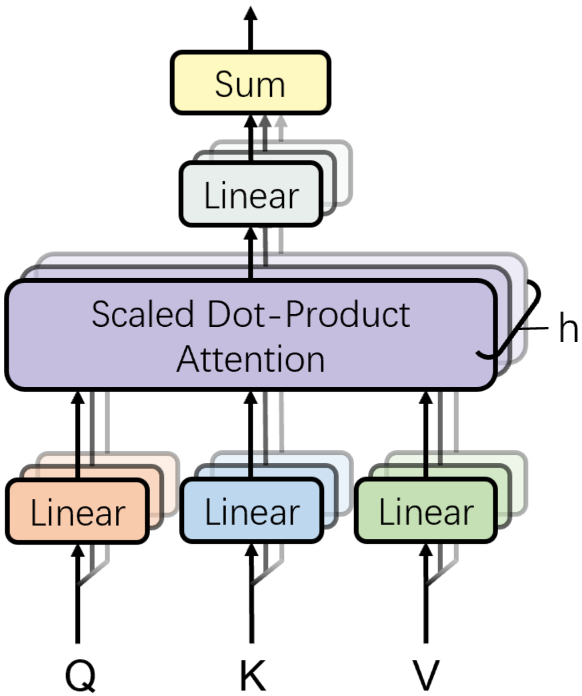
Similarities to transformers

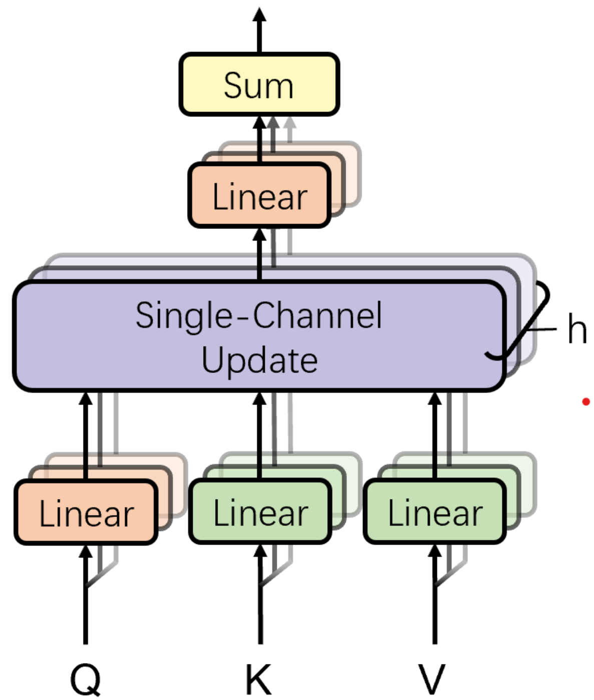
- Single-Channel Update vs. Scaled Dot-Product Attention



Similarities to transformers

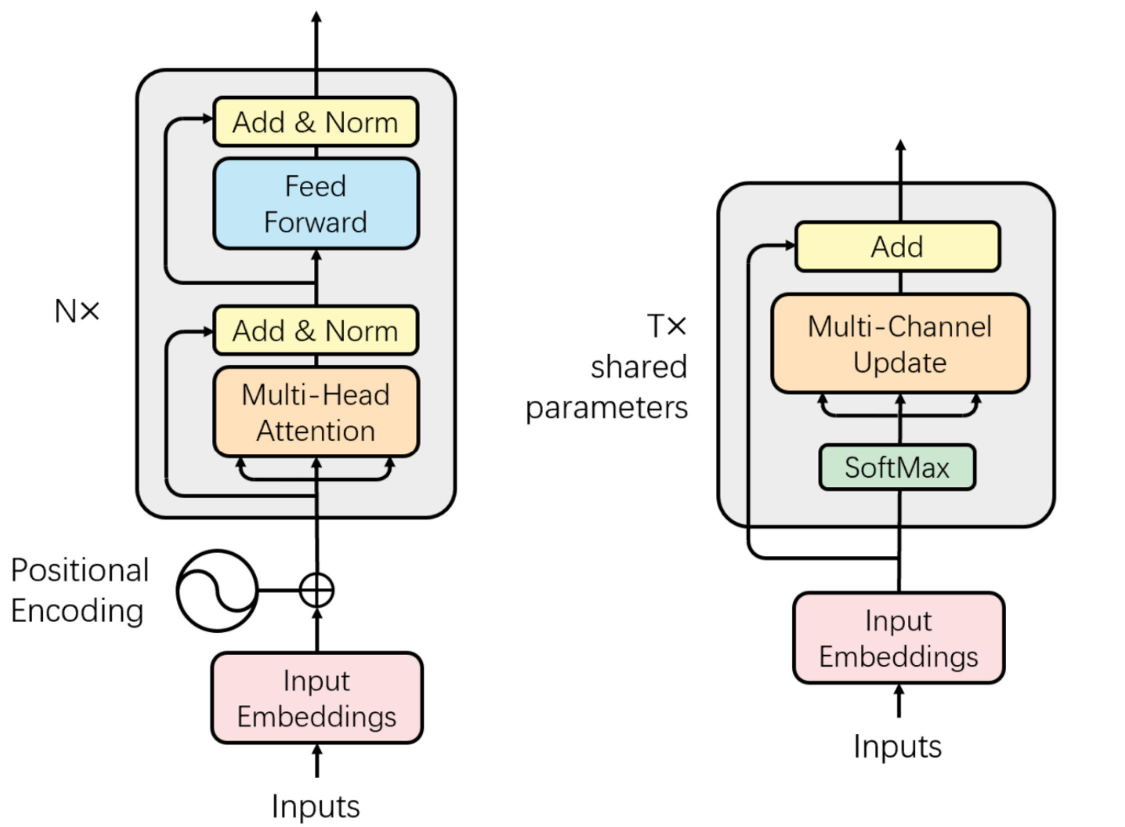
- Multi-Channel Update vs. Multi-Head Attention





Similarities to transformers

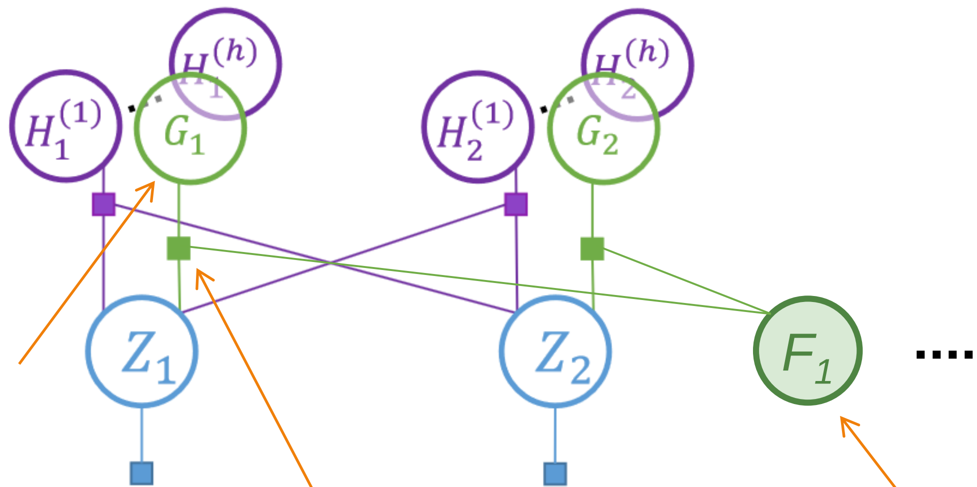
- Full Model Comparison



(a) Transformer (b) Probabilistic Transformer

Adding feed-forward layer

- Adding m global topic variables in our CRF



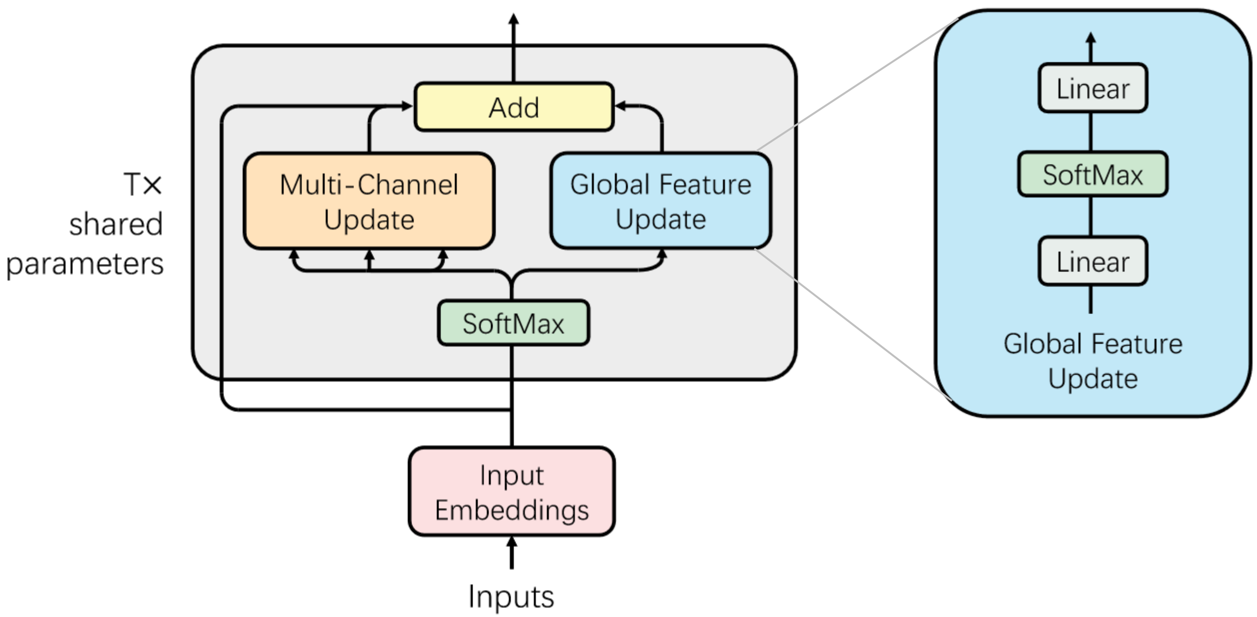
$G_i \in \{1, \dots, m\}$: index of the topic of word i

Ternary factor: compatibility between Z_i and F_j if word i belongs to topic j ($G_i = j$)

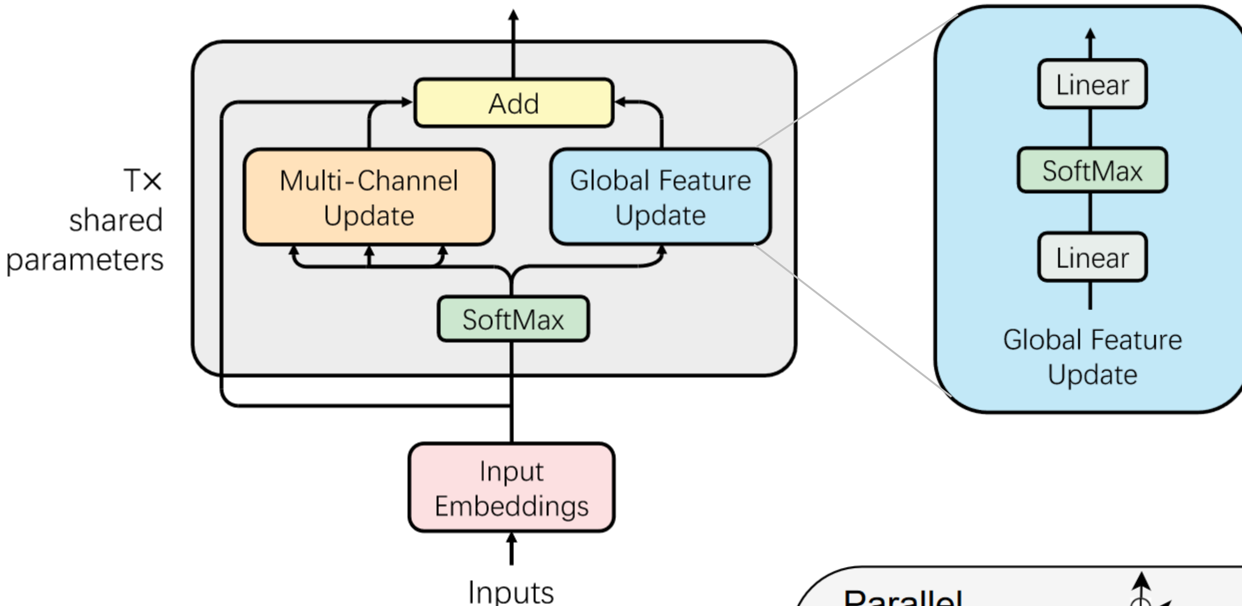
F_i: a discrete global topic variable

Adding feed-forward layer

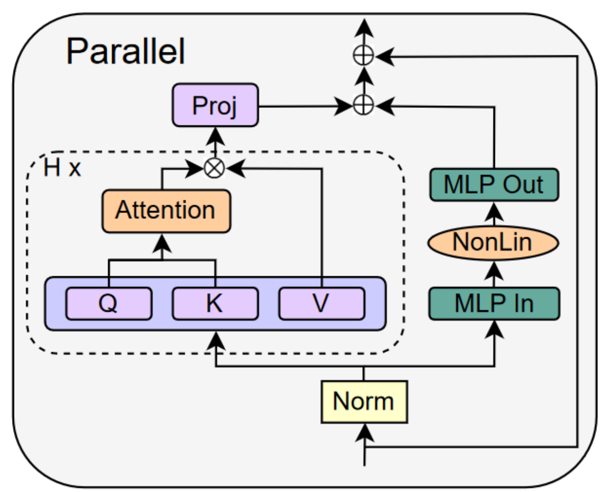
- MFVI computation graph



Adding feed-forward layer



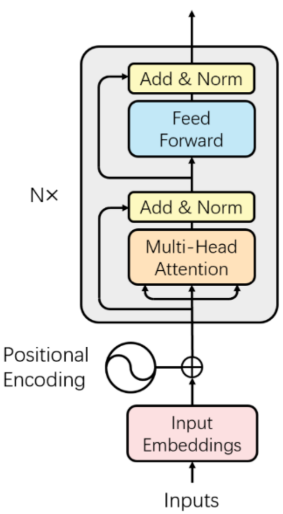
Similar to transformer parallel block (e.g., GPT-J-6B)

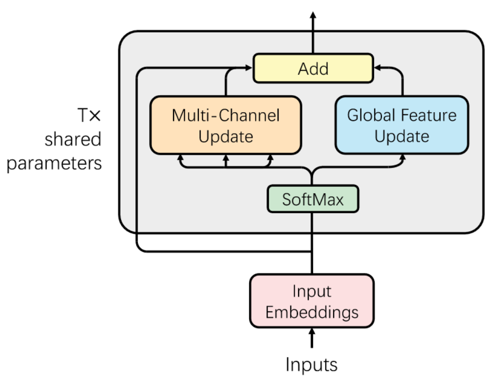


Differences

All the differences make sense!

- Feed-forward vs. Parallel feed-forward
- Residual connection vs. Adding input
  - Input injection
- Post layer norm vs. Softmax before each layer
  - Similar to pre-RMSNorm
- No parameter sharing vs. Layer-wise parameter sharing
  - Similar to Universal Transformer, ALBERT, …





Outline

- Preliminary
  - CRF, MFVI, unfolding as GNN
- Probabilistic transformers
  - Model
  - Inference
- Similarities to transformers
- Summary

Summary

- Probabilistic transformers: a white-box transformer
  - A purely probabilistic syntactic model
  - Approximate inference using mean field variational inference
  - Its computation graph is very similar to a transformer!

- Implications
  - …interpretation & analyses of existing transformer variants
  - …design of new transformer variants in a principled way

…new transformer variants

- Haoyi Wu and Kewei Tu, "Layer-Condensed KV Cache for Efficient Inference of Large Language Models". ACL 2024.
  - Inspired by autoregressive decoding with probabilistic transformers
  - Condense KV cache into a few layers
  - Achieving up to 26× higher throughput with competitive performance

Summary

- Paper
  - https://aclanthology.org/2023.findings-acl.482/
- Code
  - https://github.com/whyNLP/Probabilistic-Transformer

Thank you!

Q&A