Deep Generative Models for NLP

Miguel Rios

April 18, 2019

Content

Why generative models?

Deep Learning

pro Rich non-linear models for classification and sequence prediction.

Probabilistic modelling

Why generative models?

Deep Learning

- pro Rich non-linear models for classification and sequence prediction.
- pro Scalable learning using stochastic approximation and conceptually simple.

Probabilistic modelling

Why generative models?

Deep Learning

- pro Rich non-linear models for classification and sequence prediction.
- pro Scalable learning using stochastic approximation and conceptually simple.
- con Only point estimates.

Probabilistic modelling

Why generative models?

Deep Learning

- pro Rich non-linear models for classification and sequence prediction.
- pro Scalable learning using stochastic approximation and conceptually simple.
- con Only point estimates.
- con Hard to score models, do selection and complexity penalisation.

Probabilistic modelling

Why generative models?

Deep Learning

- pro Rich non-linear models for classification and sequence prediction.
- pro Scalable learning using stochastic approximation and conceptually simple.
- con Only point estimates.
- con Hard to score models, do selection and complexity penalisation.

Probabilistic modelling

- pro Unified framework for model building, inference, prediction and decision making.

Why generative models?

Deep Learning

- pro Rich non-linear models for classification and sequence prediction.
- pro Scalable learning using stochastic approximation and conceptually simple.
- con Only point estimates.
- con Hard to score models, do selection and complexity penalisation.

Probabilistic modelling

- pro Unified framework for model building, inference, prediction and decision making.
- pro Explicit accounting for uncertainty and variability of predictions.

Why generative models?

Deep Learning

- pro Rich non-linear models for classification and sequence prediction.
- pro Scalable learning using stochastic approximation and conceptually simple.
- con Only point estimates.
- con Hard to score models, do selection and complexity penalisation.

Probabilistic modelling

- pro Unified framework for model building, inference, prediction and decision making.
- pro Explicit accounting for uncertainty and variability of predictions.
- pro Robust to over-fitting.

Why generative models?

Deep Learning

- pro Rich non-linear models for classification and sequence prediction.
- pro Scalable learning using stochastic approximation and conceptually simple.
- con Only point estimates.
- con Hard to score models, do selection and complexity penalisation.

Probabilistic modelling

- pro Unified framework for model building, inference, prediction and decision making.
- pro Explicit accounting for uncertainty and variability of predictions.
- pro Robust to over-fitting.
- pro Offers tools for model selection and composition.

Why generative models?

Deep Learning

- pro Rich non-linear models for classification and sequence prediction.
- pro Scalable learning using stochastic approximation and conceptually simple.
- con Only point estimates.
- con Hard to score models, do selection and complexity penalisation.

Probabilistic modelling

- pro Unified framework for model building, inference, prediction and decision making.
- pro Explicit accounting for uncertainty and variability of predictions.
- pro Robust to over-fitting.
- pro Offers tools for model selection and composition.
- con Potentially intractable inference,

Why generative models?

Deep Learning

- pro Rich non-linear models for classification and sequence prediction.
- pro Scalable learning using stochastic approximation and conceptually simple.
- con Only point estimates.
- con Hard to score models, do selection and complexity penalisation.

Probabilistic modelling

- pro Unified framework for model building, inference, prediction and decision making.
- pro Explicit accounting for uncertainty and variability of predictions.
- pro Robust to over-fitting.
- pro Offers tools for model selection and composition.
- con Potentially intractable inference,
- con computationally expensive

Why generative models?

Deep Learning

- pro Rich non-linear models for classification and sequence prediction.
- pro Scalable learning using stochastic approximation and conceptually simple.
- con Only point estimates.
- con Hard to score models, do selection and complexity penalisation.

Probabilistic modelling

- pro Unified framework for model building, inference, prediction and decision making.
- pro Explicit accounting for uncertainty and variability of predictions.
- pro Robust to over-fitting.
- pro Offers tools for model selection and composition.
- con Potentially intractable inference,
- con computationally expensive
- con long simulation time.

Why generative models?

- Lack of training data.

Why generative models?

- Lack of training data.
- Partial supervision.

Why generative models?

- Lack of training data.
- Partial supervision.
- Lack of inductive bias.

PGM

- Inference in graphical models is the problem of computing a conditional probability distribution over the values of some of the nodes.

PGM

- Inference in graphical models is the problem of computing a conditional probability distribution over the values of some of the nodes.
- We also want to compute marginal probabilities in graphical models, in particular the probability of the observed evidence.

PGM

- Inference in graphical models is the problem of computing a conditional probability distribution over the values of some of the nodes.
- We also want to compute marginal probabilities in graphical models, in particular the probability of the observed evidence.
- A latent variable model is a probabilistic model over observed and latent random variables.

PGM

- Inference in graphical models is the problem of computing a conditional probability distribution over the values of some of the nodes.
- We also want to compute marginal probabilities in graphical models, in particular the probability of the observed evidence.
- A latent variable model is a probabilistic model over observed and latent random variables.
- For a latent variable we do not have any observations.

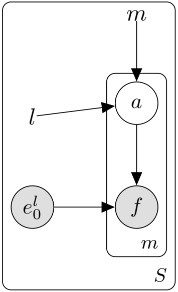
Content

IBM 1-2

Latent alignment

- Count-based models with EM is attempting to find the maximum-likelihood estimates for the data.
- Feature-rich Models (NN to combine features).
- Bayesian parametrisation of IBM.

IBM1: incomplete-data likelihood



Incomplete-data likelihood

$$
\begin{aligned}
p(f_1^m|e_0^l) &= \sum_{a_1=0}^{l} \cdots \sum_{a_m=0}^{l} p(f_1^m, a_1^m | e_{a_j}) && (1) \\
&= \sum_{a_1=0}^{l} \cdots \sum_{a_m=0}^{l} \prod_{j=1}^{n} p(a_j|l,m) p(f_j|e_{a_j}) && (2) \\
&= \prod_{j=1}^{n} \sum_{a_j=0}^{l} p(a_j|l,m) p(f_j|e_{a_j}) && (3)
\end{aligned}
$$

IBM1: posterior

Posterior

$$p(a_1^m | f_1^m, e_0^l) = \frac{p(f_1^m, a_1^m | e_0^l)}{p(f_1^m | e_0^l)} \tag{4}$$

Factorised

$$p(a_j | f_1^m, e_0^l) = \frac{p(a_j | l, m) p(f_j | e_{a_j})}{\sum_{i=0}^{l} p(i | l, m) p(f_j | e_i)} \tag{5}$$

MLE via EM

E-step:

$$\mathbb{E}[n(\mathsf{e} \to \mathsf{f}|a_1^m)] = \sum_{a_1=0}^{l} \cdots \sum_{a_m=0}^{l} p(a_1^m|f_1^m, e_0^l) n(\mathsf{e} \to \mathsf{f}|A_1^m) \tag{6}$$

$$= \sum_{a_1=0}^{l} \cdots \sum_{a_m=0}^{l} \prod_{j=1}^{m} p(a_j|f_1^m, e_0^l) \mathbb{1}_{\mathsf{e}}(e_{a_j}) \mathbb{1}_{\mathsf{f}}(f_j) \tag{7}$$

$$= \prod_{j=1}^{m} \sum_{i=0}^{l} p(a_j = i|f_1^m, e_0^l) \mathbb{1}_{\mathsf{e}}(e_i) \mathbb{1}_{\mathsf{f}}(f_j) \tag{8}$$

M-step:

$$\theta_{e,f} = \frac{\mathbb{E}[n(e \to f|a_1^m)]}{\sum_{f'} \mathbb{E}[n(e \to f'|a_1^m)]} \tag{9}$$

IBM 1-2: strong assumptions

Independence assumptions

- $p(a|m,n)$ does not depend on lexical choices
  a$_1$ cute$_2$ house$_3$ $\leftrightarrow$ una$_1$ casa$_3$ bella$_2$

IBM 1-2: strong assumptions

Independence assumptions

- $p(a|m,n)$ does not depend on lexical choices
  a$_1$ cute$_2$ house$_3$ $\leftrightarrow$ una$_1$ casa$_3$ bella$_2$
  a$_1$ cosy$_2$ house$_3$ $\leftrightarrow$ una$_1$ casa$_3$ confortable$_2$

IBM 1-2: strong assumptions

Independence assumptions

- $p(a|m, n)$ does not depend on lexical choices
  a$_1$ cute$_2$ house$_3$ $\leftrightarrow$ una$_1$ casa$_3$ bella$_2$
  a$_1$ cosy$_2$ house$_3$ $\leftrightarrow$ una$_1$ casa$_3$ confortable$_2$
- $p(f|e)$ can only reasonably explain one-to-one alignments
  I will be leaving soon $\leftrightarrow$ voy a salir pronto

IBM 1-2: strong assumptions

Independence assumptions

- $p(a|m, n)$ does not depend on lexical choices
  a$_1$ cute$_2$ house$_3$ $\leftrightarrow$ una$_1$ casa$_3$ bella$_2$
  a$_1$ cosy$_2$ house$_3$ $\leftrightarrow$ una$_1$ casa$_3$ confortable$_2$
- $p(f|e)$ can only reasonably explain one-to-one alignments
  I will be leaving soon $\leftrightarrow$ voy a salir pronto

Parameterisation

- categorical events are unrelated
  prefixes/suffixes: normal, normally, abnormally, ...
  verb inflections: comer, comi, comia, comio, ...
  gender/number: gato, gatos, gata, gatas, ...

Berg-Kirkpatrick et al. [2010]

Lexical distribution in IBM model 1

$$p(f|e) = \frac{\exp(w_{\mathsf{lex}}^{\top} h_{\mathsf{lex}}(e, f))}{\sum_{f'} \exp(w_{\mathsf{lex}}^{\top} h_{\mathsf{lex}}(e, f'))} \tag{10}$$

Features

- $f \in V_F$ is a French word (decision), $e \in V_E$ is an English word (conditioning context), $w \in R^d$ is the parameter vector, and $h : V_F \times V_E \to R^d$ is a feature vector function.
- prefixes/suffixes
- character n-grams
- POS tags
- Learning using these combination features, e.g. neural networks

Neural IBM

- $f_\theta(e) = \mathrm{softmax}(W_t H_E(e) + b_t)$ note that the softmax is necessary to make t_θ produce valid parameters for the categorical distribution
  $W_t \in \mathbb{R}^{|V_F| \times d_h}$ and $b_t \in \mathbb{R}^{|V_F|}$

MLE

- We still need to be able to express the functional form of the likelihood.

MLE

- We still need to be able to express the functional form of the likelihood.
- Let us then express the log-likelihood (which is the objective we maximise in MLE) of a single sentence pair as a function of our free parameters:

$$\mathcal{L}(\theta|e_0^m, f_1^n) = \log p_\theta(f_1^m|e_0^l) \quad (11)$$

MLE

- We still need to be able to express the functional form of the likelihood.
- Let us then express the log-likelihood (which is the objective we maximise in MLE) of a single sentence pair as a function of our free parameters:

$$\mathcal{L}(\theta|e_0^m, f_1^n) = \log p_\theta(f_1^m|e_0^l) \tag{11}$$

- $p(f|e) = \prod_j p(f_j|e) = \prod_j \sum_{a_j} p(a_j|m,l)p(f_j|e_{a_j})$

MLE

- We still need to be able to express the functional form of the likelihood.
- Let us then express the log-likelihood (which is the objective we maximise in MLE) of a single sentence pair as a function of our free parameters:

$$\mathcal{L}(\theta|e_0^m, f_1^n) = \log p_\theta(f_1^m|e_0^l) \tag{11}$$

- $p(f|e) = \prod_j p(f_j|e) = \prod_j \sum_{a_j} p(a_j|m,l)p(f_j|e_{a_j})$
- Note that in fact our log-likelihood is a sum of independent terms $\mathcal{L}_j(\theta|e_0^m, f_j)$, thus we can characterise the contribution of each French word in each sentence pair

Content

Variational Inference

- We assume that $x = x_1^n$ are observations and $z = z_1^n$ are hidden continuous variables.
  We assume additional parameters θ that are fixed.

Variational Inference

- We assume that $x = x_1^n$ are observations and $z = z_1^n$ are hidden continuous variables.
  We assume additional parameters θ that are fixed.
- We interested in performing MLE learning of the parameters θ.

Variational Inference

- We assume that $x = x_1^n$ are observations and $z = z_1^n$ are hidden continuous variables.
  We assume additional parameters θ that are fixed.
- We interested in performing MLE learning of the parameters θ.
- This requires marginalization over the unobserved latent variables z.

Variational Inference

- We assume that $x = x_1^n$ are observations and $z = z_1^n$ are hidden continuous variables.
  We assume additional parameters θ that are fixed.
- We interested in performing MLE learning of the parameters θ.
- This requires marginalization over the unobserved latent variables z.
- However this integration is intractable:

$$p_\theta(x) = \int p_\theta(x|z)p_\theta(z)\mathrm{d}z \tag{12}$$

Variational Inference

- We assume that $x = x_1^n$ are observations and $z = z_1^n$ are hidden continuous variables.
  We assume additional parameters θ that are fixed.
- We interested in performing MLE learning of the parameters θ.
- This requires marginalization over the unobserved latent variables z.
- However this integration is intractable:

$$p_\theta(x) = \int p_\theta(x|z)p_\theta(z)\mathrm{d}z \tag{12}$$

- We are also interested on the posterior inference for the latent variable:

$$p(z|x) = \frac{p(x,z)}{p(x)} \tag{13}$$

Variational Inference

- [Jordan et al., 1999] introduce a variational approximation $q(z|x)$ to the true posterior

Variational Inference

- [Jordan et al., 1999] introduce a variational approximation $q(z|x)$ to the true posterior
- The objective is to pick a family of distributions over the latent variables with its own variational parameters, $q_\phi(z)$

Variational Inference

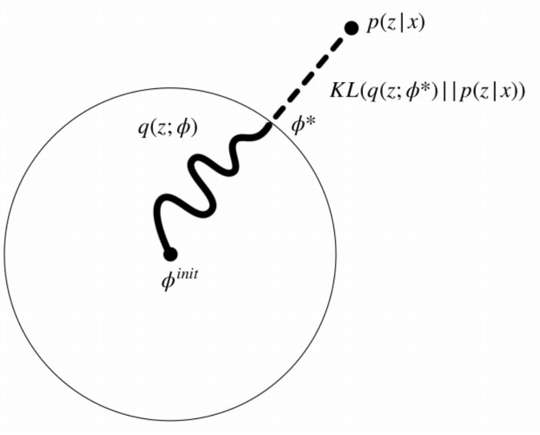
- [Jordan et al., 1999] introduce a variational approximation $q(z|x)$ to the true posterior
- The objective is to pick a family of distributions over the latent variables with its own variational parameters, $q_\phi(z)$
- Then, we find the parameters that makes q close to the true posterior

Variational Inference

- [Jordan et al., 1999] introduce a variational approximation $q(z|x)$ to the true posterior
- The objective is to pick a family of distributions over the latent variables with its own variational parameters, $q_\phi(z)$
- Then, we find the parameters that makes q close to the true posterior
- We use q with the fitted variational parameters as a proxy for the true posterior
  e.g., to form predictions about future data or to investigate the posterior distribution of the latent variables.

Variational Inference

We optimise ϕ_{init} in order to minimize the KL to get $q_\phi(z)$ closer to the true posterior:



KL divergence

- We measure the closeness of two distributions with Kullback-Leibler (KL) divergence.

KL divergence

- We measure the closeness of two distributions with Kullback-Leibler (KL) divergence.
- We focus KL variational inference [Blei et al., 2016], where the KL divergence between $q(z)$ and $p(z|x)$ is optimised.

$$\mathrm{KL}(q\|p) = \mathrm{E}_q\left[\log \frac{q(z)}{p(z|x)}\right] \tag{14}$$

KL divergence

- We measure the closeness of two distributions with Kullback-Leibler (KL) divergence.
- We focus KL variational inference [Blei et al., 2016], where the KL divergence between $q(z)$ and $p(z|x)$ is optimised.

$$\mathrm{KL}(q\|p) = \mathrm{E}_q\left[\log \frac{q(z)}{p(z|x)}\right] \tag{14}$$

- We can not minimize the KL divergence exactly, but we can maximise a lower bound on the marginal likelihood.

Evidence lower bound

- If we use the Jensens inequality applied to probability distributions.
  When f is concave,
  $f(\mathrm{E}[X]) \geq \mathrm{E}[f(X)]$

Evidence lower bound

- If we use the Jensens inequality applied to probability distributions. When f is concave, $f(\mathrm{E}[X]) \geq \mathrm{E}[f(X)]$
- We use Jensens inequality on the log probability of the observations This is the evidence lower bound (ELBO):

$$
\begin{aligned}
\log p_\theta(x) &= \log \int p_\theta(x|z) p_\theta(z) \mathrm{d}z \\
&= \log \int \frac{q_\phi(z)}{q_\phi(z)} p_\theta(x|z) p_\theta(z) \mathrm{d}z \\
&= \log \mathbb{E}_q \left[\frac{p_\theta(x|z) p_\theta(z)}{q_\phi(z)} \right] \\
&\geq \mathbb{E}_q \left[\log \frac{p_\theta(x|z) p_\theta(z)}{q_\phi(z)} \right] \\
&= \mathbb{E}_q \left[\log \frac{p_\theta(z)}{q_\phi(z)} \right] + \mathbb{E}_q \left[\log p_\theta(x|z) \right] \\
&= -\mathrm{KL}\left(q_\phi(z) \| p_\theta(z) \right) + \mathbb{E}_q \left[\log p_\theta(x|z) \right] \\
&= \mathcal{L}(\theta, \phi | x)
\end{aligned}
\tag{15}
$$

ELBO

- The objective is to do optimization of the function $q_\phi(z)$ to maximize the ELBO:

$$
\begin{aligned}
\mathrm{KL}\left(q_\phi(z) \| p_\theta(z|x)\right) &= \mathbb{E}_q\left[\log \frac{q_\phi(z)}{p_\theta(z|x)}\right] \\
&= \mathbb{E}_q\left[\log q_\phi(z) - \log p_\theta(z|x)\right] \\
&= \mathbb{E}_q\left[\log q_\phi(z) - \log \frac{p_\theta(x|z)p_\theta(z)}{p_\theta(x)}\right] \\
&= \mathbb{E}_q\left[\log \frac{q_\phi(z)}{p_\theta(z)}\right] - \mathbb{E}_{qz}\left[\log p_\theta(x|z)\right] + \log p_\theta(x) \\
&= -\mathcal{L}(\theta, \phi|x) + \log p_\theta(x)
\end{aligned}
\tag{16}
$$

Evidence lower bound

- To denote a lower bound on the log marginal likelihood:

$$\begin{aligned} \log p_\theta(\mathbf{x}) &\geq \log p_\theta(\mathbf{x}) - \mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}|\mathbf{x})\right) \\ &= \mathbb{E}_q\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})) \end{aligned} \tag{17}$$

Evidence lower bound

- To denote a lower bound on the log marginal likelihood:

$$\begin{aligned} \log p_\theta(\mathbf{x}) &\geq \log p_\theta(\mathbf{x}) - \mathrm{KL}\left(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}|\mathbf{x})\right) \\ &= \mathbb{E}_q\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x}) \| p(\mathbf{z})) \end{aligned} \tag{17}$$

- It lower-bounds the marginal distribution of x

Mean Field

- We assume that the variational family factorises:

$$q\left(z_0, \ldots, z_N\right) = \prod_i^N q\left(z_i\right) \tag{18}$$

Mean Field

- We assume that the variational family factorises:

$$q\left(z_0, \ldots, z_N\right) = \prod_i^N q\left(z_i\right) \tag{18}$$

- This simplification make optimisation and inference with VI tractable

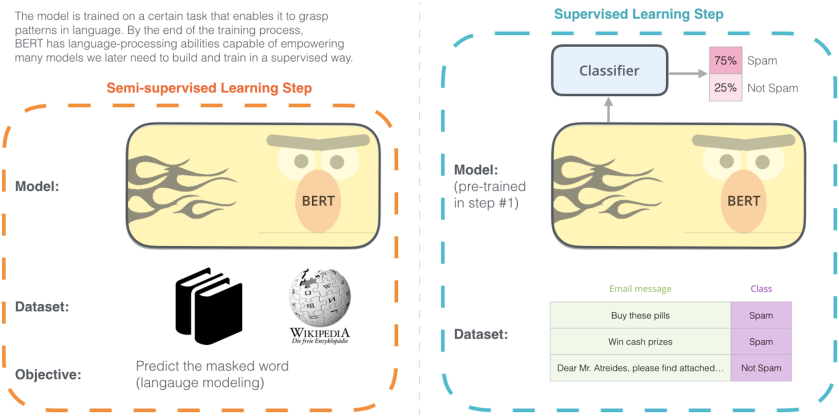
Content

Document modelling

- Know what topics are being discussed on Twitter and by what distribution they occur.

#0 (Obama)	#20 (Musk)	#26 (Tyson)	#35 (Trump)	#43 (Bieber)	#19 (Swift)
president	tesla	earth	will	thanks	tonight
obama	will	moon	great	love	ts1989
america	rocket	just	thank	whatdoyoumean	taylurking
sotu	just	day	trump2016	mean	just
actonclimate	model	one	just	purpose	love
time	launch	time	cruz	thank	thank
work	good	sun	hillary	lol	crowd
economy	dragon	people	new	good	night
americans	falcon	space	people	great	now
change	now	will	makeamericagreatagain	see	show

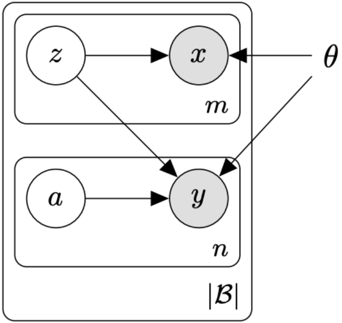
Word representation



Word representation

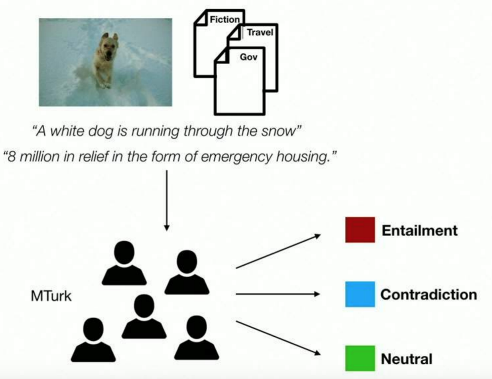
Generative model

- Embed words as probability densities.
- Add extra information about the context.
  e.g. translations as a proxy to sense annotation.



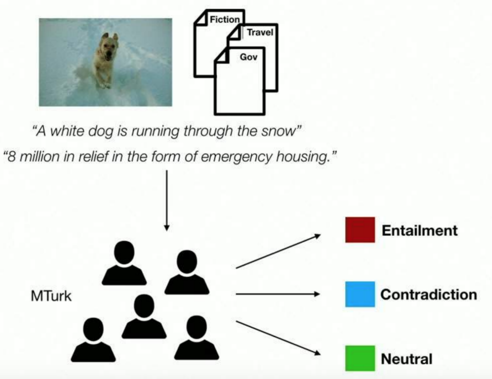
Natural Language Inference

Classification

- 

Natural Language Inference

Classification

- 

- **Generalizations**

  Premise: Some men and boys are playing frisbee in a grassy area.

  Entailment: People play frisbee outdoors.

Natural Language Inference

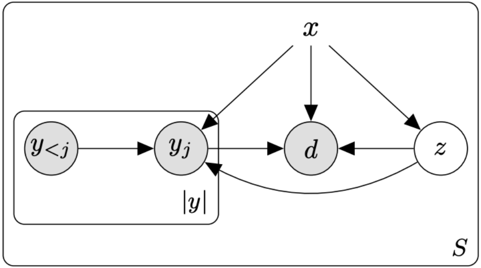
Generative model

- Avoid over-fitting.

Natural Language Inference

Generative model

- Avoid over-fitting.
- Change of prior.



Natural Language Inference

Confidence of classification

- Bayesian NN

Natural Language Inference

Confidence of classification

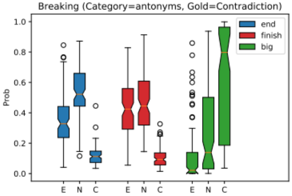
- Bayesian NN
- We place a prior distribution over the model parameters $p(\theta)$

P: group of little kids waiting for the game to start
H: group of little kids waiting for the game to end

P: group of little kids waiting for the game to start
H: group of little kids waiting for the game to finish

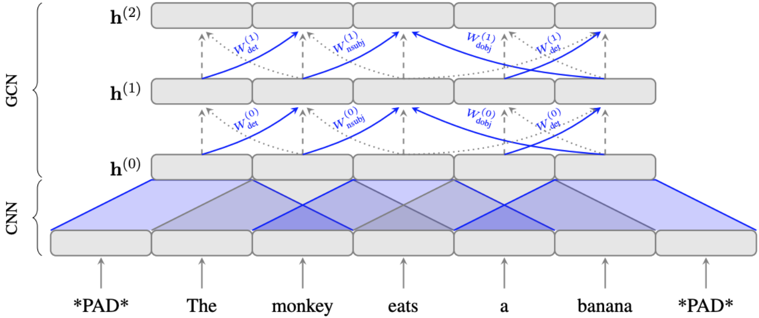
P: group of little kids waiting for the game to start

H: group of big kids waiting for the game to start



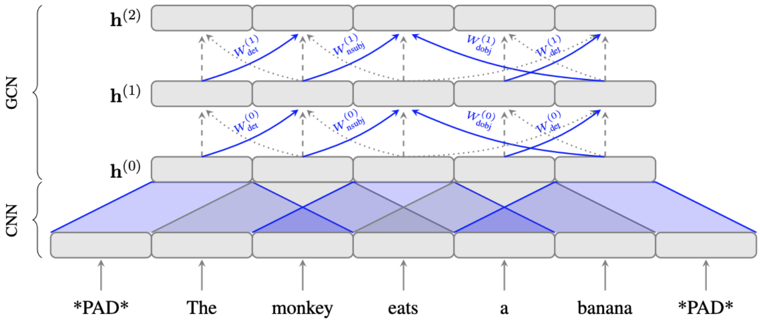
Neural Machine Translation

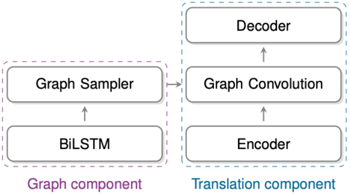
▶



Neural Machine Translation

▶





▶

References I

Taylor Berg-Kirkpatrick, Alexandre Bouchard-Côté, John DeNero, and Dan Klein. Painless unsupervised learning with features. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 582–590, Los Angeles, California, June 2010. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/N10-1083.

D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for statisticians. *ArXiv e-prints*, January 2016.

MichaelI. Jordan, Zoubin Ghahramani, TommiS. Jaakkola, and LawrenceK. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, 1999. ISSN 0885-6125. doi: 10.1023/A:1007665907178. URL http://dx.doi.org/10.1023/A%3A1007665907178.