1 Introduction

¡++¿

2 Background

¡++¿

2.1 Transformers

¡++¿

The transformer architecture involves several layers of an attention sublayer composed with a fully-connected, feedforward sublayer. Each sublayer is combined with a residual connection. Sublayer outputs are normalized using layer normalization in the original transformer architecture, although (CITE) has found that moving the layer normalization to right before the sublayers to be more effective for training.

The key architectural addition of transformers is the attention sublayer, which allows for the parallel computation of different weighted averages of the values.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\intercal}{\sqrt{d_k}} V\right) \tag{1}$$

¡++¿

The input to the attention sublayer are linearly transformed into the queries, keys, and values by learned matrices, $Q_i = W_q x_i, K = W_k x_i, V = W_k x_i$.

2.2 Kernel interpretation of transformers

¡++¿

Smola related transformers to word2vec and various types of memory networks by generalizing their averaging properties to kernel regression.

2.3 Conditional Bayesian Networks/Conditional Random Fields

¡++¿

Hinton suggested a probabilistic interpretation of neural network layers with sigmoidal activations using conditional random fields, or equivalently conditional Bayesian networks

The logistic sigmoid can be extended to the multivalued case through the softmax function

$$P(Y = y^j | X_1, \ldots, X_k) = \frac{\langle W_j, (X_1, \ldots, X_k) \rangle}{\sum_{j'=1}^{M} \langle W_{j'}, (X_1, \ldots, X_k) \rangle} \tag{2}$$

¡++¿

The undirected equivalent of a conditional Bayesian network is a conditional random field. One especially simple case is when we are defining conditional distributions of binary variables, and we use a restricted boltzmann machine to define the conditional probability of a layer of variables with respect to the previous layer's variables.

$$P(Y|Pa_Y) \propto e^{Y_j \sum_i (w_{j,i} X_i + w_{j,0}} = e^{\sum_i w_{i,j}(\langle Y_j, X_i \rangle + u_i Y_i)} \tag{3}$$

¡++¿

RBMs are parametrized by an $M \times N$ weight matrix, where M, N are the number of ouputs and inputs. This differes from transformer attention in that the variables are binary valued, but in form there is some resemblance to kernel matrices, including Gaussian Process covariance matrices, when $M = N$.

2.4 Contrastive Divergence

¡++¿

3 Theory

¡++¿

3.1 Exponential Family

¡++¿

Attention computes a convex combination of the values, so the weights for each value can be described by a multinomial distribution. When each outcome of the multinomial distribution has positive probabilities, they can be described by an exponential family distribution, where the probability for each outcome can be described by a softmax of the natural parameters.

For classification, we map our input to the natural parameters of the multinoulli distribution.

$$P(y = i|x) = \frac{e^{f_i(x)}}{\sum_i' e^{f_{i'}(x)}} \tag{4}$$

¡++¿

So while we could attempt to model the multinomial probabilities directly by normalizing a positive kernel function, it is also possible to model the exponential family form of a multinomial distribution.

More generally, members of an exponential family of distributions can be written in natural parameter form, through the Pitman-Koopman-Dermois Theorem:

$$p(x|\eta) = h(x)e^{\langle u(x), \eta \rangle - G(\eta)} < ++ > \tag{5}$$

¡++¿

where $h(x)$ is the intrinsic measure, η is the natural parameters, $u(x)$ is the sufficient statistic, and $G(\eta)$ is the log-normalizer. If we were to exclude the log-normalizer, we could treat $\tilde{p}(x|\eta)$ as an unnormalized probability distribution.

$$G(\eta) = \log \int h(x)e^{\langle u(x), \eta \rangle} dx \tag{6}$$

¡++¿

When the sufficient statistic $u(x)$ is simply the random variable x, it is possible to transform any intrinsic measure into an exponential family using a more general form of the Laplace transform of the intrinsic measure:

$$\begin{aligned} G(y) &= \log \mathcal{L}\left\{h(x)\right\}(-y) \\ &= \log \int e^{\langle y,x\rangle} h(x) dx \end{aligned} \tag{7}$$

¡++¿

In particular, if we choose our intrinsic measure to be a distribution over a discrete set of points, such as from an empirical distribution or the possible outcomes of a multinoulli distribution, the induced exponential family distribution is given by a dot-product similarity softmax:

$$\begin{aligned} h(x) &= \frac{1}{N}\sum_{n=1}^{N} \delta_{x_n}(x) \\ p(x|y) &= h(x)\frac{e^{\langle y,x\rangle}}{\frac{1}{N}\sum_{n=1}^{N} e^{\langle ymx_n\rangle}} \\ &= \frac{e^{\langle y,x\rangle}}{\sum_{j=1}^{N} e^{\langle y,x_i\rangle}}, \text{if} \quad x \in \{x_1, \ldots, x_N\} \end{aligned} \tag{8}$$

¡++¿

Thus, the exponential family induced from a set of discrete outcomes has the dot-product similarity form of contrastive divergence, which is used in transformers. In particular, if we choose a query point x_i as our natural parameter, our distribution provides us similarity probability to the points of the empirical distribution:

$$p(x_k|x_i) = \frac{e^{\langle x_k,x_i\rangle}}{\sum_{j=1}^{N} e^{\langle x_j,x_i\rangle}} \tag{9}$$

¡++¿

Therefore, dot-product attention can be interpreted as an exponential family where the sample space is restricted to key variables, and for self-attention and the natural parameters to the queries. In the specific case of self-attention, the queries the same points as the keys and hence are taken from the same sample space.

We can interpret the weighted averaging of the attention sublayer through a particularly useful property of exponential families, relating the expected sufficient statistic to the gradient of the log partition function.

$$\nabla_\eta G(\eta) = E_{x\sim p(x|\eta)}\left[u(x)\right] < ++ > \tag{10}$$

¡++¿

In the particular case of the exponential family induced from an empirical distribution, the expectation is exactly the output of the attention layer, which can also be shown to be Nadaraya-Watson kernel regression:

$$\nabla_\eta G(x_q) = \sum_{i=1}^{N} \frac{e^{\langle x_q, x_i \rangle}}{\sum_{i'=1}^{N} e^{\langle x_q, x_{i'} \rangle}} x_i \tag{11}$$

¡++¿

Ignoring the weight matrices, this is ith output of an attention head, so we can also view it as a gradient ascent step of the log normalizer, $G(\eta)$. In effect, we are taking parallel gradient step of log normalizer function, at each query point.

The DEQ interpretation of applying the same attention layer and adding it to a residual connectio is that we are doing a discrete update, possibly towards a fixed point. Iterating our gradient ascent for our log normalizer function would ultimately cause divergence to infinity. If we were to hope for convergence, we would need to compensate for the expansion of the gradient step with a contraction step.

In practice, it turns out models do include a contractive step. For contrastive divergence, they do not use dot-product similarity; rather, they use cosine similarity. This is effectively projecting the points on to a unit sphere, so it no longer matters if the points are being pushed towards infinity. Transformers use layer normalization, so the set of input hidden states become renormalized into a Gaussian distribution with a learned mean and (diagonal) covariance.

One interpretation is that there is a true probability distribution of the hidden states, and we have collected samples from that probability distribution as an empirical data distribution of the true probability distribution. Then we apply the Laplace transformation upon the discrete distribution to induce an exponential family distribution, and we are now applying gradient ascent upon the the log normalizer to generate a new empirical data distribution.

Intuitively, this would push the points outwards towards infinity. Contrastive divergence resolves this by using cosine similarity, which is a projection of the points back to the unit sphere. Transformers uses layer normalization, which renormalizes the values back to a learned mean and variance.

One question of interest is what is the fixed point of this attention layer? If we were to not use an empirical distribution, the fixed point should be a gaussian distribution, so we would assume that the attention layer spreads apart the points while the normalization shrinks it back down towards a Gaussian distribution.

Hence, we would expect

Another question is that if we replaced an empirical distribution of points with an non-discrete distribution, would we eventually converge to Gaussian fixed point distribution?

3.2 Laplace Transformation

¡++¿