DATA COMPRESSION WITH BAYESIAN ATTENTION NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

The lossless data compression algorithm based on Bayesian Attention Networks is derived from first principles.

1 INTRODUCTION: LOSSLESS COMPRESSION AS A PREDICTION PROBLEM

Data compression is a very important area of study for both practical applications as well as for fundamental research in information theory (Salomon, 2006),(MacKay, 2002), (Mahoney, 2013b), (Lossless Compression, 2020) and subject of many competitions, like (Hutter Prize, 2021). See also (Mahoney, 2013a). The neural compression is an active research area (Kingma et al., 2019).

The fundamental approach to compressing a sequence of data elements is to use a probabilistic model that is able to reproducibly compute a probability of the sequence via conditional probabilities of all elements and then encode the sequence probability with an entropy encoder (Shannon, 1948),(Rissanen, 1976),(Pasco, 1976).

The direct consequence of that approach is that the finding of a good predictive model becomes critical for the development of the high performance compression algorithm.

For a sequence of elements

$$s = a_1 a_2 a_3 .. a_n .. a_N \tag{1}$$

the probability of the sequence is a product of conditional probabilities of each element

$$P(s) = P(a_1)P(a_2|a_1)P(a_3|a_1a_2)..P(a_n|s_n)..P(a_N|s_N), \tag{2}$$

where sub-sequences s_n are defined as $s_n = a_1 a_2 .. a_{n-1}, n = 2..N$. To simplify we will consider conditional dependencies for all probabilities as functions of fixed-length sequences $s_{n,n+l}$ of length l with left-side zero padding as needed. Then the probability of the sequence s will be

$$P(s) = \prod_{n=1}^{N} P(a_n|s_{n-l,n}),\ s_{n-l,n} = a_{n-l}a_{n-l+1}..a_{n-1}. \tag{3}$$

By defining a model $P(a|s,w)$ for probability of an element a given a previous sequence s and parametrized by weights w we obtain a parametrized probability of any sequence s

$$P(s|w) = \prod_{n=1}^{N} P(a_n|s_{n-l,n},w),\ s_{n-l,n} = a_{n-l}a_{n-l+1}..a_{n-1}. \tag{4}$$

For a known training dataset of independent sequences $\{s_i, i = 1..M\}$ the probability of the data for a selected model is given by Bayesian integral over weights as it follows from the Bayes theorem (Bayes & Price, 1763)

$$Prob(\{s_i\}|H) = \int_w \prod_{i=1}^{M} P(s_i|w)P_0(w|H)dw, \tag{5}$$

where $P_0(w|H)$ is a prior of weights conditional on hyperparameters H.

The prediction of the probability $P(s|\{s_i\},H)$ of a new sequence s then will be given by an average

$$P(s|\{s_i\},H) = \langle P(s|w)\rangle = \int_w P(s|w) \prod_{i=1}^{M} P(s_i|w)P_0(w|H)dw\ /\ Prob(\{s_i\}|H). \tag{6}$$

We will call the new sequence s a test sequence.

The prediction probability $P(s|\{s_i\}, H)$ is an average over weights with the most significant contribution coming from training sequences that maximize both corresponding probabilities of training data and the test sequence $P(s|w)$ for the same weights in the Eq.(6). The resulting average probability is high when the test sequence s is similar to the most typical training sequences. However, the average will be low when s is very different from the typical training data. The problem is that all training sequences contributing equally to the average and a good prediction is possible only for a test sequence that is very similar to training data.

To improve the prediction of the probability for the given sequence s we will introduce the importance weights for training data samples in the Eq.(6) that will increase contributions of samples with sequences s_i similar to the test sequence s and reduce contributions of other samples.

That approach leads to a definition of Bayesian Attention Networks (BAN) considered next.

The idea to account for long-range correlations between data samples for achieving better compression has a long history. The attention mechanism recently proposed in (Vaswani et al., 2017) was a great success for sequence prediction tasks as well as for other problems including image recognition (Dosovitskiy et al., 2020).

2 Prediction of Sequence Probabilities with Bayesian Attention Networks (BAN)

We will define BAN in a few steps. First, we introduce the importance factors $\rho_i(s)$ for sequences by modifying contribution of each training data sample in the Bayesian integrals

$$P(s|\{s_i\}, H) = \int_w P(s|w) \prod_{i=1}^{M} e^{-\rho_i(s) l(s_i|w)} P_0(w|H) dw \ / \ Prob(\{s_i\}|H), \tag{7}$$

$$Prob(\{s_i\}|H) = \int_w \prod_{i=1}^{M} e^{-\rho_i(s) l(s_i|w)} P_0(w|H) dw. \tag{8}$$

Here $l(s_i|w)$ are loss functions for sequences defined as logs of probabilities $l(s|w) = -\log(P(s|w))$. The importance factors $\rho_i(s)$ are assigned to each training sequence s_i and are functions of predicted sequence s. The importance factors must be constrained by normalization condition $\sum_i \rho_i(s) = M$, so the total contribution of all training sequences is preserved.

The meaning of the importance factors is very simple - it defines the importance of a given training sample s_i to influence a solution for the prediction probability of the test sequence s with higher importance $\rho_i(s)$ increasing contribution of the sample s_i to predicting s. We will clarify the definition of importance weights in the following steps.

Next, we will approximate the average of probability of a sequence as product of averages for predictions of probabilities of elements conditional on corresponding immediate sub-sequences

$$P(s) = P(a_1 a_2 .. a_N) = \langle P(a_1|s_1, w)\rangle \langle P(a_2|s_2, w)\rangle .. \langle P(a_N|s_N, w)\rangle. \tag{9}$$

Then the prediction of the probability of an element is given by

$$\langle P(a|s, w)\rangle = \int_w P(a|s, w) \prod_{i=1}^{K} e^{-\rho_i(s) l(a_i|s_i, w)} P_0(w|H) dw \ / \ Prob(\{a_i\}|H) \tag{10}$$

with the normalization factor $Prob(\{a_i\}|H)$ that keeps $\langle P(a|s, w)\rangle$ normalized defined as follows

$$Prob(\{a_i\}|H) = \int_w \prod_{i=1}^{K} e^{-\rho_i(s) l(a_i|s_i, w)} P_0(w|H) dw. \tag{11}$$

Here the importance factors depend only on input sequences in each sample and not on predicted elements. The index i runs over all K sequence elements a_i in training data in Eqs.(10,11). The losses $l(a_i|s_i, w)$ are per element in training sequences and the importance factors $\rho_i(s)$ control the

contributions of each training sample i and depend on the sub-sequence s of the element a in the prediction probability $P(a|s)$. The importance factors are normalized by condition $\sum_i \rho_i(s) = K$ where K is a total number of training data samples.

The Eqs.(10,11) are defining the Bayesian Attention Network with attention variable $\rho_i(s)$ which is a function of two sequences. The attention or importance factors allow to take into account long range interactions between different data samples.

It is important to note that for any approximation of the integrals in Eqs.(10,11) the solution is dependent on the input sequence s in the data sample we are making prediction for $P(a|s)$, which requires computing the integrals for every test sample.

To resolve this problem and find the practical computing approach we will use a latent variable z representation with the encoder-decoder network that is considered in the next section.

3 Computing Importance via Extended Variational Approximation

Let's define encoder $\rho(z|s)$ and decoder $\rho(s|z)$. To do a prediction for a sample (a,s) the encoder $\rho(z|s)$ gives a probability of z for input s and the decoder $\rho(s_i|z)$ gives a probability of a training sequence s_i for a given z.

The decoder allows to define the normalized importance factor

$$\rho_i(z) = K\rho(s_i|z)/\sum_j \rho(s_j|z). \tag{12}$$

With that we can reformulate Eqs.(10,11) as follows

$$\begin{aligned} \langle P(a|s,w)\rangle &= \int_{w,z} P(a|s,w)\prod_{i=1}^{K} e^{-\rho_i(z)l(a_i|s_i,w)} P_0(w|H)\rho(z|s)dwdz \;/\; Prob(\{a_i\}|H) \\ Prob(\{a_i\}|H) &= \int_{w,z}\prod_{i=1}^{K} e^{-\rho_i(z)l(a_i|s_i,w)} P_0(w|H)\rho(z|s)dwdz \end{aligned} \tag{13}$$

Due to Bayesian theorem (Bayes & Price, 1763) the encoder probability related to the decoder probability and can be expressed via decoder probability and priors for z and s

$$\rho(z|s) = \frac{\rho(s|z)\rho(z)}{\rho(s)}. \tag{14}$$

By introducing an approximation for the encoder $q(z|s)$ we can use a variational method for simplifying the integral over z following (Kingma & Welling, 2014)

$$\int_z e^{-L(z)}\rho(s|z)\rho(z)dz \geqslant \exp\left(-\int_z q(z|s)L(z,w)dz + \int_z q(z|s)\log\frac{\rho(s|z)\rho(z)}{q(z|s)}dz\right), \tag{15}$$

where total training loss is $L(z,w) = \sum_i \rho_i(z)l(a_i|s_i,w)$. Maximizing the right side of Eq.(15) w.r.t. parameters of $q(z|s)$ allows to find the encoder q.

However, the variational approximation that is based on Jensen's inequality, (Jensen, 1906) $\langle\exp(x)\rangle \geqslant \exp(\langle x\rangle)$, results in the loss of dependency on importance factors in the prediction probability in the Eqs.(13). To retain that dependency we have to use a sharpened Jensen's inequality, (see Appendix A for the proof)

$$\langle e^x\rangle \geqslant e^{\langle x\rangle + \frac{1}{2}\langle(x-\langle x\rangle)^2\rangle}. \tag{16}$$

Then the probability of prediction can be maximized w.r.t. importance weights. Now we can find that the variation of the log of the prediction probability is equal to a correlation function of test and train losses

$$\frac{\delta\log\langle P(a|s,w)\rangle}{\delta\rho_i(z)} = \langle l(a_i|s_i,w)l(a|s,w)\rangle_w - \langle l(a_i|s_i,w)\rangle_w\langle l(a|s,w)\rangle_w \approx \sum_j \sigma_j^2 \frac{\partial l(a_i|s_i,w)}{\partial w_j}\frac{\partial l(a|s,w)}{\partial w_j}, \tag{17}$$

where averages $\langle(..)\rangle_w$ are computed with a posterior distribution of weights

$$Post(w|H_0, z) = \prod_{i=1}^{K} e^{-\rho_i(z) l(a_i|s_i,w)} P_0(w|H_0), \tag{18}$$

by expanding the weights around the means up to the second order and averaging over it.

4 Complete Loss and Recurrent Update Rules for Hyperparameters

Due to the sharpened Jensen's inequality in Eq.(16) the complete loss includes a cross-correlation term between the test sample loss and training sample losses

$$\int_{w,z} P_0(w|H_0) q(z|s) \left(L(z,w) - l(a|s,w) L(z,w)\right) dw dz - \int_z q(z|s) \log \frac{\rho(s|z)\rho(z)}{q(z|s)} dz, \tag{19}$$

where the training loss $L(z,w) = \sum_i \rho_i(z) l(a_i|s_i,w)$.

Finding the importance factors by maximizing the prediction probability for test sample requires to minimize only the cross-correlation loss due to normalization factor in Eq.(13). All other parameters could be found by minimizing the loss in the Eq.(19).

Let's use a posterior distribution for weights that depends on initial hyperparameters H_0 and the latent variable z in the Eq.(18). The method developed in (Tetelman, 2020) allows to compute approximations of the Bayesian integrals by re-parametrizing a prior of weights $P_0(w|H)$ to represent a posterior

$$Post(w|H', z) = \prod_{i=1}^{K} e^{-\rho_i(z) l(a_i|s_i,w)} P_0(w|H_0) \sim P_0(w|H(z, H_0)). \tag{20}$$

The re-parametrization is computed recursively. With Gaussian priors for all network weights and Gaussian encoder $q(z|s)$ we can find the following update rules for hyperparameters: (mean, variance) pairs (μ, σ^2) for all weights of all networks. The training update steps are as follows

$$i \sim \mathbf{Uniform}[1..K] \tag{21}$$

$$z(s,v) \sim q(z|s,v) \tag{22}$$

$$w(z,v) \sim P_0(w|\mu_w(z|v), \sigma_w(z|v)) \tag{23}$$

$$\rho_i(z,u) = K\rho(s_i|z,u) / \sum_j \rho(s_j|z,u) \tag{24}$$

$$\mathbf{grad_v} = \frac{\partial}{\partial v}\left[\rho_i(z,u) l(a_i|s_i, w(z,v)) - \log \frac{\rho(s|z,u)\rho(z)}{q(z|s,v)}\right] \tag{25}$$

$$\Delta\mu_v += -\varepsilon\sigma_v^2 \mathbf{grad_v} \tag{26}$$

$$\Delta\frac{1}{\sigma_v^2} += \varepsilon(\mathbf{grad_v})^2 \tag{27}$$

$$\mathbf{grad_u} = \sum_j \left(\frac{\sigma_{w,j}^2}{K} \frac{\partial l(a_i|s_i,w)}{\partial w_j} \frac{\partial l(a|s,w)}{\partial w_j}\right) \frac{\partial \rho_i(z,u)}{\partial u} \tag{28}$$

$$\Delta\mu_u += -\varepsilon\sigma_u^2 \mathbf{grad_u} \tag{29}$$

$$\Delta\frac{1}{\sigma_u^2} += \varepsilon(\mathbf{grad_u})^2. \tag{30}$$

Here the learning rate ε is equal to the inverse number of epochs T: $\varepsilon = 1/T$.

The prediction for a sample with input s is done by sampling $z \sim q(z|s,v)$, computing a mean for weight w equal to $\mu_w(z,v)$ and finally computing a prediction probability $P(a|s,\mu_w)$.

REFERENCES

Mr Bayes and Mr Price. An Essay towards Solving a Problem in the Doctrine of Chances. *Philosophical Transactions of the Royal Society of London*, 53:370–418, 1763.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv:2010.11929*, 2020.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. The MIT Press, 2016. ISBN 0262035618.

Hutter Prize. Wikipedia. `https://en.wikipedia.org/wiki/Hutter_Prize`, 2021. Online.

Johan L. Jensen. Sur les fonctions convexes et les ingualits entre les valeurs moyennes. *Acta Math.*, 30:175193, 1906.

Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv:1312.6114*, 2014.

Friso H. Kingma, Pieter Abbeel, and Jonathan Ho. Bit-Swap: Recursive Bits-Back Coding for Lossless Compression with Hierarchical Latent Variables. *arXiv:1905.06845*, 2019.

Lossless Compression. Wikipedia. `https://en.wikipedia.org/wiki/Lossless_compression`, 2020. Online.

David J. C. MacKay. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, USA, 2002. ISBN 0521642981.

Matt Mahoney. Data Compression Programs. `http://mattmahoney.net/dc/`, 2013a. Online.

Matt Mahoney. Data Compression Explained. `http://mattmahoney.net/dc/dce.html`, 2013b. Online.

Richard Pasco. *Source coding algorithms for fast data compression*. PhD thesis, Stanford, Calif., 1976.

Jorma J. Rissanen. Generalized Kraft inequality and arithmetic coding. *IBM 1. Res. Dev.*, pp. 198–203, May 1976.

David Salomon. *Data Compression: The Complete Reference*. Springer-Verlag, Berlin, Heidelberg, 2006. ISBN 1846286026.

Claude Elwood Shannon. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27:379–423,623–656, 1948.

Michael Tetelman. On Compression Principle and Bayesian Optimization for Neural Networks. *arXiv:2006.12714*, 2020.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. *arXiv:1706.03762*, 2017.

A APPENDIX

To prove the sharpened Jensen's inequality we note that for a normal distribution

$$\langle e^x \rangle = e^{\langle x \rangle + \frac{1}{2}\langle (x - \langle x \rangle)^2 \rangle} \tag{31}$$

As Gaussian mixture is an universal approximator for any (non-pathological) distribution, see (Goodfellow et al., 2016) p.65, then it directly proves the sharpened Jensen's inequality in Eq.(16):

$$\int_x dx \sum_k A_k G(x|\mu, \sigma) e^x = \sum_k A_k e^{\mu_k + \frac{1}{2}\sigma_k^2} \geqslant e^{\mu + \frac{1}{2}\sigma^2}. \tag{32}$$

Here $\sum_k A_k = 1$ and $\mu = \sum_k A_k \mu_k, \sigma^2 = \sum_k A_k \sigma_k^2$.