Principled Uncertainty Estimation for High Dimensional Data

Pascal Notin [1] **José Miguel Hernández-Lobato** [2] **Yarin Gal** [1]

Abstract

The ability to quantify the uncertainty in the prediction of a Bayesian deep learning model has significant practical implications—from more robust machine-learning based systems to more effective expert-in-the loop processes. While several general measures of model uncertainty exist, they are often intractable in practice when dealing with high dimensional data such as long sequences. Instead, researchers often resort to ad hoc approaches or to introducing independence assumptions to make computation tractable. We introduce a principled approach to estimate uncertainty in high dimensions that circumvents these challenges, and demonstrate its benefits in de novo molecular design.

1 Introduction

To understand when one can rely on a model's output, and ultimately build trust in AI systems, we must be able to qualify the model's prediction with a notion of uncertainty. For example, machine-learning based pathology detection systems are progressively being deployed in healthcare. Knowing when the model is uncertain about its prediction allows for practitioners to confidently let the machine handle the easier cases (low model uncertainty), and have more time to dedicate to the more complex cases (high model uncertainty). Such measures of model uncertainty are already used in practice with low dimensional outputs (Filos et al., 2019), but their estimation in high-dimensional domains (e.g., large complex images, long natural language sequences, biological sequences) is often impractical due to the size of the corresponding space. Existing approaches to estimating uncertainty with structured high dimensional data either make use of ad hoc techniques (Xiao et al., 2019), or make simplifying assumptions such as independence of the output dimensions (e.g., elements in a sequence are assumed to be independent of each other, (Malinin and Gales, 2020)).

In this paper we:

- Review existing methods to quantify model uncertainty and illustrate their limitations in the case of high dimensional data (§2);
- Introduce principled estimators to circumvent these limitations and obtain reliable uncertainty estimates in high dimensions (§3);
- Illustrate the benefits from these approaches in a real-world example in de novo molecular design (§4).

2 Background

The overall uncertainty of a model in a given region of the input space can be broken down into two types of uncertainty (Kendall and Gal, 2017):

- **Epistemic uncertainty:** Uncertainty due to lack of knowledge about that region of the input space – the posterior over model parameters is broad in that region due to lack of information about it, and we can reduce that uncertainty by collecting more data;
- **Aleatoric uncertainty:** Uncertainty due to inherent stochasticity in the data in that region – amassing additional data would not further reduce that uncertainty.

Adopting a Bayesian viewpoint, the total uncertainty of a model at an input point z is typically measured by the predictive entropy, ie. the entropy of the predictive posterior distribution P(y|z):

$$PE(z) = H(P(y|z)) = \int_y -\ln P(y|z) * P(y|z) dy \quad (1)$$

If we denote $q(\theta)$ as the distribution over model parameters, we can further decompose the predictive entropy as the sum of two terms:

$$PE(z) = \underbrace{(H(P(y|z)) - \mathbb{E}_{q(\theta)}(H(P(y|z,\theta)))}_{\text{Mutual information}} + \underbrace{\mathbb{E}_{q(\theta)}(H(P(y|z,\theta)))}_{\text{Expected entropy}}$$

[1]Department of Computer Science, University of Oxford, Oxford, UK [2]Department of Engineering, University of Cambridge, Cambridge, UK. Correspondence to: Pascal Notin <pascal.notin@cs.ox.ac.uk>.

Presented at the ICML 2020 Workshop on Uncertainty and Robustness in Deep Learning.

The first term – the Mutual information (MI) between model parameters θ and the prediction y – is a measure of epistemic uncertainty, as it quantifies the magnitude of the change in model parameters that would result from observing y. If the model is quite uncertain about its prediction for y, then the change in model coefficients from observing y should be high. If the model is very certain about its prediction for y, the model parameters will not vary much from observing y:

$$MI(z) = H(P(y|z)) - \mathbb{E}_{q(\theta)}(H(P(y|z,\theta))) \quad (3)$$

The second term – the Expected Entropy (EE) – is a measure of the residual uncertainty, ie. the aleatoric uncertainty:

$$EE(z) = \mathbb{E}_{q(\theta)}(H(P(y|z,\theta))) \quad (4)$$

When focusing on high dimensional problems, an exact estimation of these different measures of uncertainty is impractical. Several approximations and heuristics have been introduced to get around these challenges. The "softmax variance", i.e. the variance of predictions across model parameters, has been shown to work well in practice (Carlini and Wagner, 2016; Feinman et al., 2017). Under certain assumptions and using a Taylor expansion of the logarithm, the Mutual information may be approximated by the softmax variance (Smith and Gal, 2018).

In the context of sequential data, the inherent structure in the data generating process often introduces strong dependencies between the output dimensions, e.g. the tokens in a generated sentence. These dependencies can be ignored to approximate the expressions above as the sum over tokens of the token-level equivalent (Malinin and Gales, 2020). For example, if we neglect the dependencies across tokens, the predictive entropy for a sequence $y = (y_1, y_2, ..., y_L)$ may be approximated as the sum of token-level predictive entropies over the L tokens:

$$\begin{aligned} PE(z) = H(P(y|z)) &= \sum_{l=1}^{L} \mathbb{E}_{P(y|z)}[\log P(y_l|z, y_{k<l})] \\ &\approx \sum_{l=1}^{L} \mathbb{E}_{P(y_l|z,y_{k<l})}[\log P(y_l|z, y_{k<l})] \\ &= \sum_{l=1}^{L} H(P(y_l|z, y_{k<l})) \end{aligned} \quad (5)$$

for *some* fixed sequence y used in the conditionals.

While the above has been shown to work well in certain experiments (Malinin and Gales, 2020), valuable information is being discarded when we ignore dependencies across tokens (as we show below). Alternative approaches have been suggested which make use of specialised tools in Natural Language Processing such as the BLEU score (Xiao et al., 2019), but these are difficult to extend to other domains.

3 Uncertainty in deep sequence models

In lieu of the aforementioned heuristics, we set out to estimate the expressions detailed in §2 via Monte Carlo estimations using *importance sampling*. Our scheme avoids the analytically intractable sum over all possible outcomes, which grows exponentially with sequence length. Further, importance sampling with a smartly chosen importance distribution allows us to get a principled and practical approximation to the uncertainty measures. Let C be the number of distinct possible values for y. We will approximate expectations over model parameters by sampling T independent models from $q(\theta)$.

3.1 Mutual information

We start by rewriting the MI as:

$$MI(z) = H(P(y|z)) - \mathbb{E}_{q(\theta)}(H(P(y|z,\theta))) \quad (6a)$$

$$= -\sum_{j=1}^{C} \hat{p}_j * ln(\hat{p}_j)) + \frac{1}{T}\sum_{t=1}^{T}\sum_{j=1}^{C} p_{jt} * ln(p_{jt}) \quad (6b)$$

$$= \sum_{j=1}^{C} (\frac{1}{T}\sum_{t=1}^{T} p_{jt} * ln(p_{jt})) - \hat{p}_j * ln(\hat{p}_j)) \quad (6c)$$

where $\hat{p}_j$ and p_{jt} are shorthands resp. for $P(y = j|z)$ (the posterior predictive distribution) and $P(y = y_j|z, \theta = \theta_t)$ (prediction from with a given model parameter sample θ_t).

In high dimension, we can obtain a tractable approximation via importance sampling:

$$MI(z) = \sum_{j=1}^{C} (\frac{1}{T}\sum_{t=1}^{T} p_{jt} * ln(p_{jt})) - \hat{p}_j * ln(\hat{p}_j)) \quad (7a)$$

$$= \sum_{j=1}^{C} f(y_j)\frac{1}{p(y_j|import.)} * p(y_j|import.) \quad (7b)$$

$$\propto \mathbb{E}_{p_{import.}}(f(y) * \frac{1}{p_{import.}}) \quad (7c)$$

$$\approx \frac{1}{m}\sum_{j=1}^{m} f(\hat{y}_j) * \frac{1}{p_{import.}(\hat{y}_j)} \quad (7d)$$

with $\hat{y}_j \sim p(y_j|import.)$ where $p(y_j|import.)$ is the probability of y_j under the importance distribution and f(.) a shorthand for the summand in the MI expression.

We choose the importance distribution to be the approximate posterior predictive defined over the output sequences. We generate sequences by sampling a set of parameters from the approximate posterior, and then sampling a sequence from a model defined by that set of parameters. This distribution will sample mostly from regions with high density under the posterior, regions with y which will have non-negligible predictive probability for z to map to. This is in contrast to a naive sum over all possible outcomes y, many of which will have a negligible contribution to the sum.

We obtain the following algorithm for assessing the mutual information MI:

Algorithm 1 Compute Mutual information $MI(z)$

for $j = 1$ **to** m **do**
  $\theta_0 \sim \theta$;
  Sample $y_j \sim$ P(y|z,$\theta = \theta_0$) ;
  for $t = 1$ **to** T **do**
    Sample a model $\theta_t \sim \theta$;
    Compute p_{jt} = $P(y = y_j|z, \theta = \theta_t)$;
  end for
  Compute $\hat{p}_j = \frac{1}{T}\sum_{t=1}^{T} p_{jt}$ (approx. posterior predictive) ;
  Compute $f_j = \frac{1}{T}\sum_{t=1}^{T} p_{jt} * ln(p_{jt}) - \hat{p}_j * ln(\hat{p}_j)$;
end for
Return $\frac{1}{m}\sum_{j=1}^{m} \frac{f_j}{\hat{p}_j}$

Still, in high dimensions, the probability of most elements y_j will be small, and therefore we need to resort in practice to performing operations in the log space (e.g., using the LogSumExp trick) to avoid numerical instability (see Appendix for more details).

3.2 Expected entropy

Similarly, we can derive an algorithm to estimate the expected entropy:

Algorithm 2 Compute Expected entropy $EE(z)$

$EE = 0$
for $t = 1$ **to** T **do**
  Sample a model $\theta_t \sim \theta$
  Sample $y_j \sim$ P(y|z,$\theta = \theta_t$)
  $EE \leftarrow EE + (-log(p_{jt}))$
end for
Return EE/T

3.3 Predictive entropy

And in the same vain, we can derive an algorithm to estimate the predictive entropy:

Algorithm 3 Compute Predictive entropy $PE(z)$

for $j = 1$ **to** m **do**
  $\theta_0 \sim \theta$;
  Sample $y_j \sim$ P(y|z,$\theta = \theta_0$) ;
  for $t = 1$ **to** T **do**
    Sample a model $\theta_t \sim \theta$;
    Compute p_{jt} = $P(y = y_j|z, \theta = \theta_t)$;
  end for
  Compute $\hat{p}_j = \frac{1}{T}\sum_{t=1}^{T} p_{jt}$ (approx. posterior predictive) ;
  Compute $f_j = -\hat{p}_j * ln(\hat{p}_j)$;
end for
Return $\frac{1}{m}\sum_{j=1}^{m} \frac{f_j}{\hat{p}_j} = \frac{1}{m}\sum_{j=1}^{m} -ln(\hat{p}_j)$

4 Experimental results

We illustrate the benefits from our estimator in a real-world example in de novo molecular design.

4.1 Experimental setup

We study the structure of the molecular space where each molecule is represented as a string of characters (its "SMILES" representation). We adopt an approach similar to (Gómez-Bombarelli et al., 2018), and jointly train a variational auto-encoder (*VAE*) along with a network predicting the "QED" (a measure of "drug-likeness") for each molecule (the "Predictive VAE" framework). For the VAE model, the encoder is a 4-layer bidirectional LSTM model and the decoder a 4-layer (unidirectional) GRU model, with dropout in-between each layer. The training test data is comprised of 250k distinct molecules randomly extracted from the ZINC database (Irwin et al., 2012). Each molecule is represented as a (padded) sequence of up to 120 characters, selected from an alphabet of cardinality 35. Consequently, we are in the desired high dimensionality regime (35^{120}). Additional modeling details are provided in Appendix.

In the subsequent experiments, we leverage the measures of uncertainty described above to quantify the uncertainty of the decoder of the VAE model at a given position in latent. These points come from 4 distinct sets:

1. Points from the training data, encoded in the latent space;
2. Points from the test data, encoded in the latent space;
3. Points sampled at random from the VAE prior (standard Gaussian);
4. Points sampled at random far from the VAE prior (Gaussian with standard deviation of 10);

Intuitively we would like our measure of uncertainty to be able to tell apart the points that are within the prior (i.e., any of the first sets 1-3 above) Vs far from the prior.

Sampling a set of decoder parameters is achieved through sampling a dropout mask for the decoder (Kendall and Gal, 2017). We randomly sample 10 distinct decoder masks for each latent position (i.e., T=10) in the following experiments.

While we evaluated Mutual Information, Expected Entropy and Predictive entropy, we focus here on Mutual Information as it delivered the best results to quantify the ability of the decoder to distinguish between points in latent space sampled form the prior Vs far from the prior (results for Expected Entropy and Predictive Entropy are provided in Appendix). We compare results obtained with our proposed importance sampling algorithm with two baselines: Mutual information computed as the sum of token-level mutual

information (Malinin and Gales, 2020), and the softmax variance heuristics.

4.2 Importance sampling

We looked at two different types of visualisations to assess the different approaches to quantify uncertainty:

- **Uncertainty histograms -** representing the distribution of the uncertainty metric for the 4 different set of points discussed above (e.g., training data, samples from the prior);
- **Performance plots -** representing the average validity of the decoded molecules (over 100 decodings for a given point in latent) against the proportion of points retained when rank ordering points by increasing value of decoder uncertainty. The data for this experiment was comprised of 50% of points sampled from the prior, and 50% of points sampled far from the prior. Average validity of decoded molecules near the prior 100%, and less than 1% far from the prior;

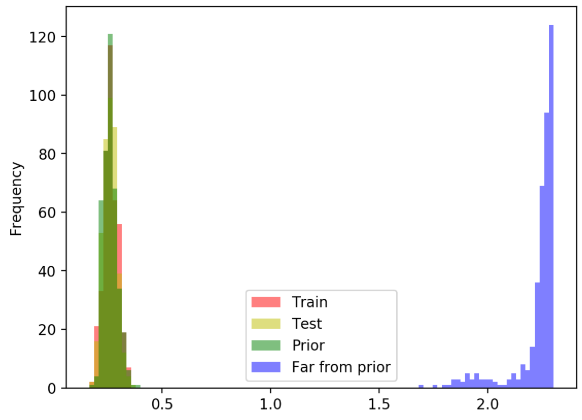

Figure 1. **Uncertainty histogram - Importance sampling.** The distributions of uncertainty estimates for train data, test data and points sampled from the prior all overlap and are disjoint from the distribution of uncertainty for points sampled far from the prior

Our proposed algorithm based in importance sampling delivers the best results to properly estimate the mutual information between prediction and model parameters (Figures 1 and 2). On the uncertainty histogram, the distribution of uncertainty values for the first three sets of points (train data, test data and samples from the prior) all overlap, and they are disjoint from the distribution of uncertainty values for samples far from the prior (all uncertainty values for points far from the prior are strictly above the values near the prior). When rank ordering points based on increasing uncertainty value, the points near the prior get selected before any point far from the prior, therefore the average validity of decoded molecules is near 100% when looking at 50% of the points with lowest uncertainty.

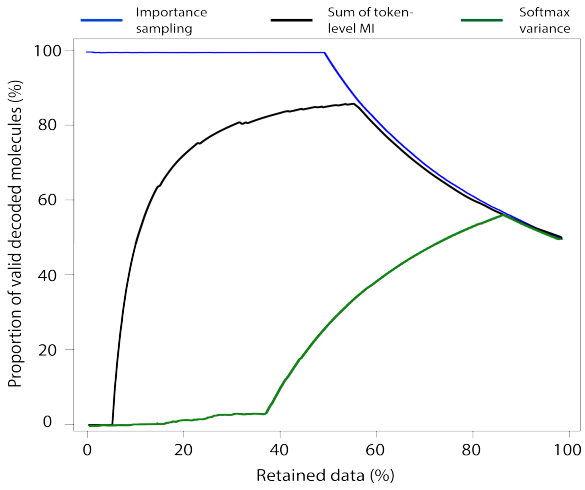

Figure 2. **Performance plots.** The importance sampling approach properly separates points sampled from the prior Vs points sampled far from the prior, hence why we obtain an average decoded validity near 100% when selecting the 50% points with lowest uncertainty. The approach based on the sum of token-level MI assigns very low uncertainty to points far from the prior, leading to low average validity (less than 1%) when selecting the 5% points with lowest uncertainty. The approach based on softmax variance performs even worse, assigning lower uncertainty to a majority of points far from the prior compared to points from the prior.

4.3 Sum of token-level MI

Repeating the same experiment with the Mutual information computed as the sum of token-level mutual information baseline (Malinin and Gales, 2020), we find that the imposed independence assumption hurts performance considerably (Figure 2 and Appendix Figure 4). This estimator ends up rejecting points the model is correct about, and retaining points for which the model is incorrect.

4.4 Softmax variance

Similarly, the softmax variance, a popular uncertainty measure with categorical data, underperforms with sequence data (Figure 2 and Appendix Figure 6).

5 Conclusion

We developed an importance-sampling based method to estimate various measures of model uncertainty in the high dimensional data setting, and demonstrated the advantages of the method over traditional baselines and heuristics in an experiment in de novo molecular design. In future work, the ability to properly quantify when the decoder is uncertain about its prediction could be used as the basis for a more efficient search of new molecules in latent space under the same “Predictive VAE” framework.

Acknowledgements

This work was supported by GSK and the Engineering and Physical Sciences Research Council (EPSRC ICASE award No. 18000077). This project has received funding from Microsoft Azure and Intel AI Labs.

References

Carlini, N. and Wagner, D. (2016). Towards evaluating the robustness of neural networks.

Feinman, R., Curtin, R. R., Shintre, S., and Gardner, A. B. (2017). Detecting adversarial samples from artifacts.

Filos, A., Farquhar, S., Gomez, A. N., Rudner, T. G. J., Kenton, Z., Smith, L., Alizadeh, M., de Kroon, A., and Gal, Y. (2019). A systematic comparison of bayesian deep learning robustness in diabetic retinopathy tasks.

Gal, Y. and Ghahramani, Z. (2015). Dropout as a bayesian approximation: Representing model uncertainty in deep learning.

Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. (2018). Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 4(2):268–276.

Irwin, J. J., Sterling, T., Mysinger, M. M., Bolstad, E. S., and Coleman, R. G. (2012). Zinc: A free tool to discover chemistry for biology. *Journal of Chemical Information and Modeling*, 52(7):1757–1768. PMID: 22587354.

Kendall, A. and Gal, Y. (2017). What uncertainties do we need in bayesian deep learning for computer vision?

Malinin, A. and Gales, M. (2020). Uncertainty in structured prediction.

Smith, L. and Gal, Y. (2018). Understanding measures of uncertainty for adversarial example detection.

Xiao, T., Gomez, A., and Gal, Y. (2019). Wat heb je gezegd? detecting out-of-distribution translations with variational transformers. *Bayesian Deep Learning Workshop at NIPS 2019*.

Appendix

A LogSumExp trick

In the derivations for the Mutual information discussed in §3, we use the following identities:

$$\ln(\exp(a) - \exp(b)) = a + \ln(1 - \exp(b - a)) \tag{8a}$$

$$\ln(\sum_{t=0}^{T} a_i) = \ln(a_0) + \ln(1 + \sum_{t=1}^{T} \exp(\ln(a_i) - \ln(a_0))) \tag{8b}$$

$$where\ a_0 = \max_i a_i$$

B Modeling details

Encoder:

- LSTM network encoding SMILES representations of molecules into latent space
- Dimensionality of the LSTM hidden state (at each layer): 500
- Number of LSTM layers: 4
- Dimensionality of latent space: 200

Decoder:

- GRU network (4-layer) generating SMILES based on latent representation passed as the first hidden state to the GRU
- Dimensionality of the GRU hidden state (at each layer): 500
- Number of GRU layers: 4
- Dimensionality of latent space: 200

QED network:

- 3-layer DNN to predict QED score of latent state with:
- 2 hidden layer of 1500 units each
- Linear output layer (dimensionality of 1)

C Additional experimental results

C.1 MI - Importance sampling

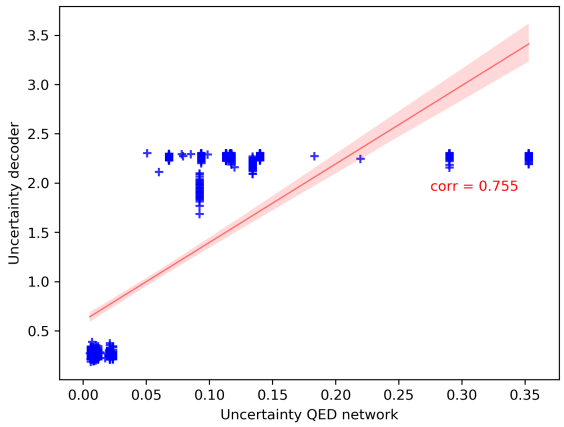

Figure 3. **Correlation QED uncertainty Vs decoder uncertainty - Importance sampling**. Data comprised of 50% values in prior and 50% far from prior. QED uncertainty is measured via MC dropout (Gal and Ghahramani, 2015). We observe that the Decoder uncertainty and QED uncertainty are highly correlated, and that both assign high values (resp. low values) to points far from the prior (resp. from the prior)

C.2 MI - Sum of token-level MI

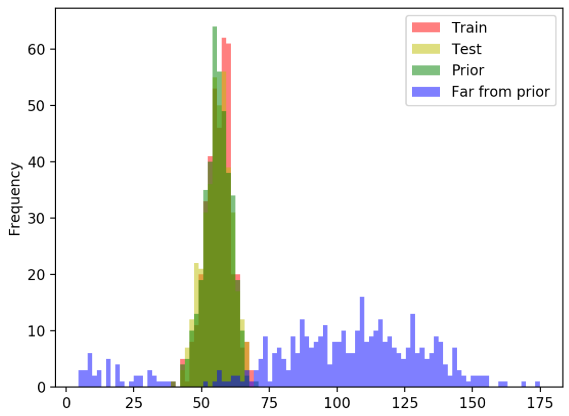

Figure 4. **Uncertainty histogram - MI as the sum of token-level MI.** About 10% of the points sampled far from the prior get assigned an uncertainty estimate that is lower than that of any point in the training/testing data and points sampled from the prior.

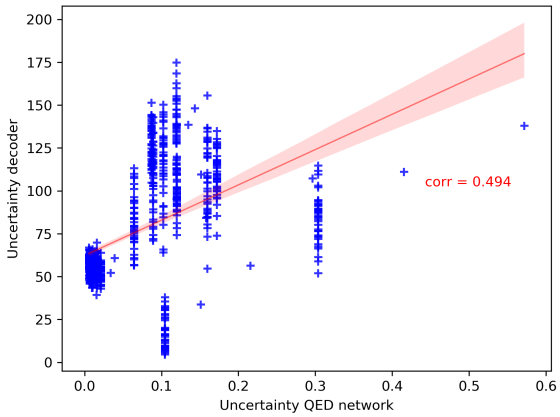

Figure 5. **Correlation QED uncertainty Vs decoder uncertainty - Sum of token-level MI**. Data comprised of 50% values in prior and 50% far from prior. QED uncertainty is measured via MC dropout (Gal and Ghahramani, 2015). We observe a relatively high correlation between the Decoder and QED uncertainty estimates, although not as strong as via the approach based on importance sampling.

C.3 Mutual Information (MI) - Softmax Variance

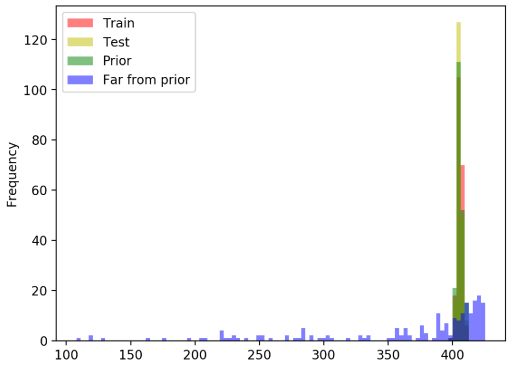

Figure 6. **Uncertainty histogram - MI with Softmax Variance approximation.** A large proportion of points sampled far from the prior (65%) get assigned an uncertainty value that is lower than that of any point in the training/testing data and points sampled from the prior.

C.4 Expected Entropy (EE) - Importance sampling

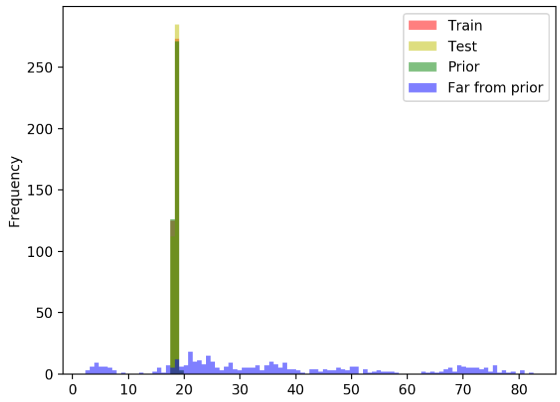

Figure 7. **Uncertainty histogram - Expected Entropy (EE) via importance sampling.** This uncertainty estimate does not allow to discriminate between points sampled from the prior (or training/testing data points) and points sampled far from the prior.

C.5 Expected Entropy (EE) - Sum of token-level EE

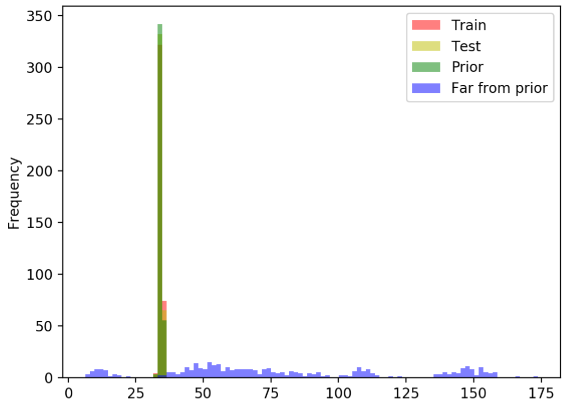

Figure 8. **Uncertainty histogram - Expected Entropy (EE) as sum of token-level EE**. This uncertainty estimate does not allow to discriminate between points sampled from the prior (or training/testing data points) and points sampled far from the prior.

C.6 Predictive Entropy (PE) - Importance sampling

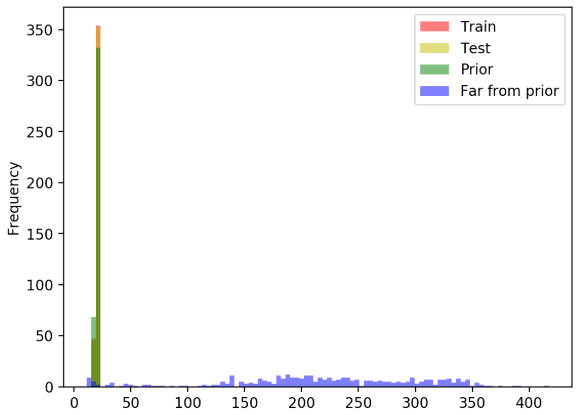

Figure 9. **Uncertainty histogram - Predictive Entropy (PE) via importance sampling**. This uncertainty estimate provides good separation between points sampled from the prior (or training/testing data points) and points sampled far from the prior, although not as clear cut as the Mutual Information equivalent (distributions have a relatively small overlap).

C.7 Predictive Entropy (PE) - Sum of token-level EE

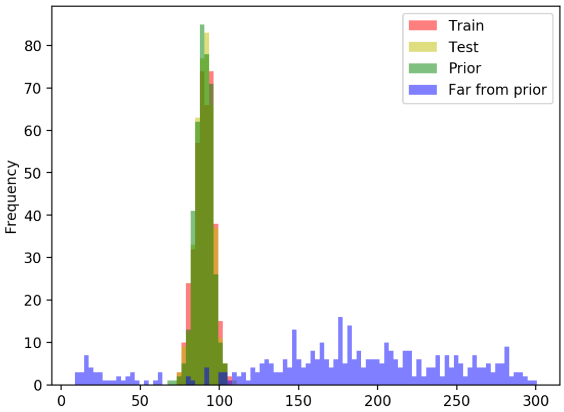

Figure 10. **Uncertainty histogram - Predictive Entropy (PE) as sum of token-level EE**. This uncertainty estimate does not allow to discriminate between points sampled from the prior (or training/testing data points) and points sampled far from the prior.