# Uncertainty Modeling for Machine Comprehension Systems using Efficient Bayesian Neural Networks

**Zhengyuan Liu, Pavitra Krishnaswamy, Ai Ti Aw, Nancy F. Chen**
Institute for Infocomm Research, A*STAR, Singapore
{liu_zhengyuan,nfychen}@i2r.a-star.edu.sg

## Abstract

While neural approaches have achieved significant improvement in machine comprehension tasks, models often work as a black-box, resulting in lower interpretability, which requires special attention in domains such as healthcare or education. Quantifying uncertainty helps pave the way towards more interpretable neural networks. In classification and regression tasks, Bayesian neural networks have been effective in estimating model uncertainty. However, inference time increases linearly due to the required sampling process in Bayesian neural networks. Thus speed becomes a bottleneck in tasks with high system complexity such as question-answering or dialogue generation. In this work, we propose a hybrid neural architecture to quantify model uncertainty using Bayesian weight approximation but boosts up the inference speed by 80% relative at test time, and apply it for a clinical dialogue comprehension task. The proposed approach is also used to enable active learning so that an updated model can be trained more optimally with new incoming data by selecting samples that are not well-represented in the current training scheme.

## 1 Introduction

Neural approaches demonstrate strong learning capability, achieve significant improvement in various natural language processing tasks (Devlin et al., 2019), and are increasingly applied in real-world applications (Du et al., 2019; J Kurisinkel and Chen, 2019). However, neural models typically operate as black-box functions and thus lack interpretability. Interpreting (or even if only partially) the output from neural models is important in domains such as healthcare, when a model is prone to make incorrect diagnosis (Settles, 2012). To tackle this issue, one approach is to evaluate the confidence of an output generated by a model regarding an input, which is through quantifying *model uncertainty* or *epistemic uncertainty*. When the uncertainty measure of the model is high, one could be prompted to intervene in the automated decision process by either overriding the system's decision or escalating the situation to a domain expert. This approach would also favor model training with new incoming streams of data that may be ill-represented in the current setting.

Different from the label probability produced by models, epistemic uncertainty is derived from the weight variance under the observation on a certain distribution. Bayesian neural networks (BNNs) (Denker and LeCun, 1990; Buntine and Weigend, 1991), in which prior distributions are applied as additional constraints to weights, have been shown to be effective for quantifying epistemic uncertainty. Instead of obtaining deterministic weights, Bayesian methods update weights via distribution-based estimation (Kendall and Gal, 2017). Therefore, one can sample different possible weights and forward inputs through the network multiple times, then obtain epistemic uncertainty according to the variance of a set of predictions. Moreover, drawing experience from past work, modeling within a Bayesian framework can lead to potentially better representations and predictions in various tasks (Kendall and Gal, 2017; Xiao and Wang, 2019).

Most previous studies applied Bayesian neural approaches on classification or regression tasks. In this paper, we focus on modeling and utilizing epistemic uncertainty for question-answering (QA) systems

*Proceedings of the 28th International Conference on Computational Linguistics: Industry Track*, pages 228–235
Barcelona, Spain (Online), December 12, 2020

in natural language processing, and tackling the aforementioned issues in the healthcare domain. Since the neural network architectures for QA tasks are relatively more complex, there is a need to balance the learning quality and inference speed. To this end, we propose a hybrid neural architecture by integrating Bayesian approximation to a base neural model and optimize its training strategy. We conduct experiments on a clinical conversational scenario in Section 4.1 and a question-answering benchmark dataset (see Appendix B). The result shows that our approach can achieve better performance and is capable of modeling epistemic uncertainty. Furthermore, we analyze the characteristics of the quantified uncertainties and conduct an active learning experiment on the clinical corpus.

## 2 In Relation to Other Work

Neural question-answering approaches, often applied to machine comprehension tasks, have achieved rapid progress lately, benefiting from large-scale corpora (Rajpurkar et al., 2016), semantic vector representations (Pennington et al., 2014), sophisticated neural architectures (Seo et al., 2017), and deep contextual language models (Devlin et al., 2019), pushing the state-of-the-art performance on various benchmarks. However, the extent to which these systems truly understand language remains unclear, and models are vulnerable to adversarial samples (Jia and Liang, 2017).

As an effective approach to model weight variance and generate predictions, Bayesian neural networks and their variants have been applied in computer vision for image classification (Kendall and Gal, 2017) and autonomous vehicles to better model safety (McAllister et al., 2017). In natural language processing, uncertainty modeling has been adopted in sentiment analysis, named entity recognition and language modeling (Xiao and Wang, 2019). Such approaches have also proved effective in domain-specific active learning such as named entity recognition (Shen et al., 2017). To the best of our knowledge, we take the first stab to introduce neural epistemic uncertainty modeling in question-answering tasks.

Making Bayesian neural networks tractable on large-scale practical problems has been a focus in the research field since the 1990's (Denker and LeCun, 1990; Hinton and Van Camp, 1993; Barber and Bishop, 1998). More recently, several approximation methods have been proposed, including Bayes-by-Backprop (Blundell et al., 2015), which places a prior distribution over model parameters and calculates the Kullback-Leibler (KL) divergence between approximated and expected posterior distribution; and Monte-Carlo Dropout (Gal and Ghahramani, 2016), which applies dropout in both training and inference stages to approximate Bayesian variational inference. Sampling from approximated posterior distribution using gradient uncertainty can also be used to represent uncertainty in predictions (Park et al., 2018).

## 3 Methodology

### 3.1 Modeling Epistemic Uncertainty with Bayesian Neural Networks

A traditional neural model $f^{\mathcal{W}}(.)$ with a specific network architecture $f(.)$ learns and optimizes weights $\mathcal{W}$ by point estimation, therefore the inference process is deterministic. However, in practice, there is a degree of uncertainty associated with the weights (epistemic uncertainty), which can be modeled by representing the weights $\mathcal{W}$ as a distribution. To this end, Bayesian neural networks aim to estimate the posterior distribution of $\mathcal{W}$, based on the observation of data $\mathcal{D}$. Here, the posterior is denoted as $p(\mathcal{W}|\mathcal{D})$ and once it is estimated, the prediction of an input $x$ is generated by marginalizing over the posterior:

$$p(y|x,\mathcal{D}) = \int_{\mathcal{W}} p(y|f^{\mathcal{W}}(x))p(\mathcal{W}|\mathcal{D})d\mathcal{W} \tag{1}$$

However, the exact solution is intractable, thus variational inference (Graves, 2011) is used to estimate the true posterior $p(\mathcal{W}|\mathcal{D})$ with an approximation $q(\mathcal{W})$ parametrized by $\theta$. This approximation is typically obtained by minimizing the KL divergence between the two distributions, and can be performed by Bayes-by-Backprop (Blundell et al., 2015) or Monte-Carlo Dropout (Gal and Ghahramani, 2016). At the inference stage, we can draw weights from the approximated posterior $\widehat{\mathcal{W}} \sim q(\mathcal{W})$.

With the approximated weight distribution, we can employ a weight sampling scheme to represent

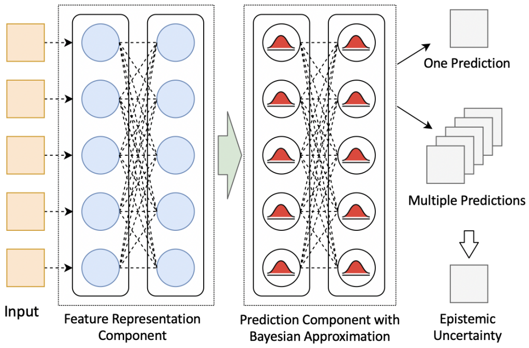
One Prediction
Multiple Predictions
Input
Feature Representation Component
Prediction Component with Bayesian Approximation
Epistemic Uncertainty

Figure 1: Overview of the hybrid framework extended from a traditional neural network. The epistemic uncertainty is calculated upon multiple predictions.

epistemic uncertainty. For the sample $x_i$, the neural model outputs $o_i$ as prediction via softmax operation:

$$o_i = \mathrm{Softmax}(\mathrm{f}^{\widehat{\mathcal{W}}}(\mathrm{x_i})). \tag{2}$$

In our question-answering setting, since the final output is generated as an answer span (Wang and Jiang, 2016) and the prediction is a classification task pointing on the input sequence (Vinyals et al., 2015), we quantify the uncertainty at the start and end positions respectively. More specifically, we conduct the Monte-Carlo (MC) integration by repeating the inference process $m$ times, then the answer span is selected based on the mean of all sampled predictions. For any input text $x_i$, we denote the probability of the answer span starting at token $t$ as $p_{i\_t}$. Then, the epistemic uncertainty of answer span starting token is quantified as:

$$U(p_{i\_t}|x_i) = \frac{1}{m}\sum_{j}^{m}(p_{i\_t}^{j})^2 - \mathrm{E}(p_{i\_t})^2 \tag{3}$$

where $x_i$ is the text sequence input, $m$ is the weight sampling time, $p_{i\_t}^{j}$ is the probability produced by the $j$th sampling, and $E$ denotes the expectation of all predictions. The final uncertainty output is the sum of both ends of the predicted answer span.

### 3.2 Hybrid Neural Architecture

Since uncertainty quantification in Bayes-by-Backprop and Monte-Carlo Dropout needs multiple sampling and forward iterations, inference time is linearly increased. In practical scenarios, machine comprehension and dialogue tasks often require deeper and larger neural architectures than classification or regression tasks, thus the inference process becomes more time-consuming. On the other side, in neural language approaches, the implicit linguistic features are modeled hierarchically from token and sentence to document level in a deep contextualized architecture (Clark et al., 2019), and semantic-related features at top neural layers play an important role in the machine comprehension task. Therefore, to speed up the inference process without sacrificing the uncertainty modeling capability, we propose a hybrid neural architecture (see Figure 1). More specifically, we split a neural network for question-answering into two sub-functions: (1) feature representation component, which is a traditional neural network, and (2) prediction component with Bayesian approximation, in which we adopt the Bayesian weight estimation. As the feature representation component produces deterministic outputs, the hybrid model will only conduct weight sampling on the prediction component, thus significantly reducing the inference time. Moreover, by integrating Bayesian weight approximation in Section 3.1 to a base neural network, the hybrid model can still be trained in an end-to-end way,[1] and we can obtain epistemic uncertainty via Equation 3.

[1] There are two training strategies of the hybrid model: joint training from scratch and warm-up training with deterministic weights, and the latter performed slightly better in our experiment.

| Model | Exact Match | F1 Score | Train Time (Iter.) | Test Time (Iter.) |
|---|---|---|---|---|
| Base Model (Bi-DAF) | 77.45 | 79.55 | 0.236 | 0.053 |
| Pure Bayes-by-Backprop | 78.86 | 80.90 | 0.597 | 4.371 |
| FAB Bayes-by-Backprop | 78.57 | 80.41 | 0.318 | 0.828 |
| Pure MCDO | **79.33** | **81.71** | 0.435 | 3.250 |
| FAB MCDO (Our final model) | 79.04 | 81.35 | 0.238 | 0.652 |

Table 1: Left: evaluation scores of various Bayesian models in the clinical scenario. Right: training and inference speed (seconds per iteration) comparison (batch size=128, sampling times=100).

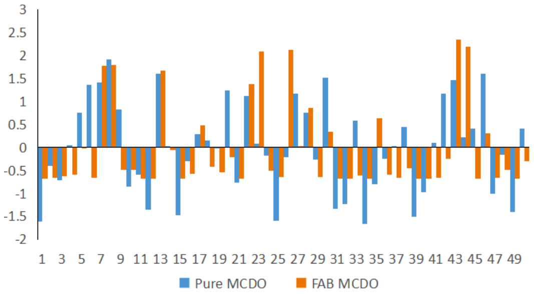

Figure 2: Epistemic uncertainty values of 50 examples. Scores from both models (Pure MCDO and FAB MCDO) have been z-score normalized.

## 4 Experiments and Results

### 4.1 Question-Answering on Clinical Conversations

We evaluated the proposed approach on a spoken dialogue comprehension corpus, consisting of nurse-to-patient symptom monitoring conversations (Liu et al., 2019). This corpus was inspired by real dialogues in the clinical setting where nurses enquire about symptoms of patients. Linguistic structures at the semantic, syntactic, discourse and pragmatic levels were abstracted from these conversations to construct templates for simulating multi-turn dialogues. These conversations cover 9 topics/symptoms (e.g., headache, cough). For each conversation, the average word number[2] is 255 and the average interactive turn number is 15.5. For the comprehension task, questions were raised to query different attributes of a specified symptom; e.g., *How frequently did you experience chest pain?* Answer spans were labeled with start and end indices. The training, validation and test set[3] are 30k, 3k and 1k respectively.

We choose Bi-Directional Attention Flow network (Seo et al., 2017) as the base architecture,[4] which fuses question-aware and context-aware attention and performs competitively in various question-answering corpora. Pre-trained word embeddings from Glove (Pennington et al., 2014) were utilized and fixed during training. Out-of-vocabulary words were replaced with the *[unk]* token. The hidden size and embedding dimension were set to 300, and batch size was set to 128. During training, the validation-based early stop strategy was applied. During prediction, we selected answer spans using the maximum product of the start and end position. In our Bayes-by-Backprop (BBB) implementations, weights in the prediction component were sampled from a mixture of two Gaussian distributions with small variances (Blundell et al., 2015) during inference for uncertainty modeling. In our Monte-Carlo Dropout (MCDO) implementations, dropout in the Bayesian approximation component was permanently enabled during training and inference for uncertainty modeling. L2 weight regularization was added to the feature representation component. All models were implemented in Pytorch (Paszke et al., 2019). More details of hyper-parameter configuration are described in Appendix A.

[2] The input sequence length is set of 300, and all text samples are tokenized and padded before feeding to the encoder.

[3] The data used for training and validation are simulated dialogue samples as described in (Liu et al., 2019), and the test set is derived from anonymized samples that acquired as part of a research study approved by the SingHealth Centralised Institutional Review Board (Protocol 1556561515).

[4] We also implemented RNet(Wang et al., 2017) for the question-answering task, in our settings, it performed similar to Bi-DAF. Moreover, we adopted and fine-tuned BERT (Devlin et al., 2019) as a contextual representation backbone; however, the performance did not benefit from it, since the spoken dialogue content is quite different from the pre-trained content.

| Model | Train Size | Exact Match | F1 Score |
|---|---|---|---|
| Training on Set A | 15k | 52.50 | 56.87 |
| Random Selection (+5K from Set B) | 20k | 60.69 | 65.38 |
| FAB MCDO Selection (+5k from Set B) | 20k | 66.80 | 70.64 |
| Training on Full Dataset (Set A + Set B) | 30k | 79.04 | 81.35 |

Table 2: Evaluation results on Bayesian uncertainty based active selection and random collection.

As shown in Table 1, models with uncertainty estimation components achieve higher performance than the base model, and MCDO models perform better. Compared with applied Bayesian estimation on all weights (Pure MCDO), our Feature-and-Bayesian (FAB) model still obtains comparable results. Meanwhile, in the inference stage, the FAB MCDO model is significantly faster than the Pure MCDO model. Then, we quantify the epistemic uncertainty with MCDO models as described in Section 3.1. We set the Monte-Carlo sampling instances to 100, and collect all the predictions. Then we calculated the variance of the softmax probability at the start and end positions respectively. As shown in Figure 2, the epistemic uncertainties of the two models were similar. Moreover, there was a certain overlap (69%) when we ranked the 1k test samples with their uncertainty scores and selected the top-k ones (k=300). This indicates that we can refer to the FAB model's uncertainty output with shorter inference time.

### 4.2 Active Learning on Clinical Conversation QA

Based on the previous result, we explore to apply the proposed hybrid model to active learning. Since it is time-consuming to annotate a large number of clinical data samples from the electronic health records (EHR), we expect to utilize epistemic uncertainty to identify samples that are potentially the most helpful for training (Siddhant and Lipton, 2018). To this end, we split the training set in Section 4.1 to two subsets (set A and set B), and conducted active learning in two steps: (1) We trained the FAB MCDO model on set A (15k samples); (2) We evaluated the epistemic uncertainty on all samples of set B (15k samples); (3) We selected 5k samples in set B with the highest uncertainty scores, added them to the training set, and re-trained the model from scratch. We also randomly selected 5k samples from set B as control. As shown in Table 2, FAB MCDO selection obtains larger performance improvement than the random scheme, achieving 87% performance of full set training with 66.7% samples. Moreover, although adopting various Bayesian active learning methods is beyond the scope of this paper, the proposed model can also be used with other acquisition functions such as BatchBALD (Kirsch et al., 2019).

## 5 Conclusion

In this work, we defined how to quantify epistemic uncertainty in question-answering tasks. We further proposed a hybrid neural architecture that achieves performance comparable to regular Bayesian neural networks but offers greater efficiency, speeding up the inference processing time by 80% relative. The proposed approach also enabled active learning for dialogue comprehension tasks so that an updated model was trained more optimally with new incoming data by selecting training samples that may not have been well-represented in the current training dataset.

## Acknowledgements

Research efforts were supported by funding and infrastructure for Digital Health from the Institute for Infocomm Research (I2R), Science and Engineering Research Council, A*STAR, Singapore (Grant Nos SSF A1818g0044, IAF H19/01/a0/023). The clinical data acquisition was funded by the Economic Development Board (EDB), Singapore Living Lab Fund, and Philips Electronics Hospital to Home Pilot Project (EDB grant reference number: S14-1035-RF-LLF H and W). We gratefully acknowledge valuable inputs on the clinical data from the Department of Cardiology and Health Management Unit, Changi General Hospital, Singapore. We thank Weiwen Xu, Yang Guo, and Savitha Ramasamy for insightful discussions and the anonymous reviewers for their precious feedback to help improve and extend this piece of work.

## References

David Barber and Christopher M Bishop. 1998. Ensemble learning in bayesian neural networks. *Nato ASI Series F Computer and Systems Sciences*, 168:215–238.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. 2015. Weight uncertainty in neural network. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1613–1622, Lille, France, 07–09 Jul. PMLR.

Wray L. Buntine and A. Weigend. 1991. Bayesian back-propagation. In *Complex systems*, volume 5(6), pages 603–643.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy, August. Association for Computational Linguistics.

John S. Denker and Yann LeCun. 1990. Transforming neural-net output levels to probability distributions. In *Proceedings of the 1990 Conference on Advances in Neural Information Processing Systems 3*, NIPS-3, pages 853–859, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Nan Du, Kai Chen, Anjuli Kannan, Linh Tran, Yuhui Chen, and Izhak Shafran. 2019. Extracting symptoms and their status from clinical conversations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 915–925, Florence, Italy, July. Association for Computational Linguistics.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059.

Alex Graves. 2011. Practical variational inference for neural networks. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 2348–2356. Curran Associates, Inc.

Geoffrey Hinton and Drew Van Camp. 1993. Keeping neural networks simple by minimizing the description length of the weights. In *in Proc. of the 6th Ann. ACM Conf. on Computational Learning Theory*. Citeseer.

Litton J Kurisinkel and Nancy Chen. 2019. Set to ordered text: Generating discharge instructions from medical billing codes. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6165–6175, Hong Kong, China, November. Association for Computational Linguistics.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark, September. Association for Computational Linguistics.

Alex Kendall and Yarin Gal. 2017. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in neural information processing systems*, pages 5574–5584.

Andreas Kirsch, Joost van Amersfoort, and Yarin Gal. 2019. Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. In *Advances in Neural Information Processing Systems*, pages 7026–7037.

Zhengyuan Liu, Hazel Lim, Nur Farah Ain Suhaimi, Shao Chuen Tong, Sharon Ong, Angela Ng, Sheldon Lee, Michael R. Macdonald, Savitha Ramasamy, Pavitra Krishnaswamy, Wai Leng Chow, and Nancy F. Chen. 2019. Fast prototyping a dialogue comprehension system for nurse-patient conversations on symptom monitoring. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 24–31, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Rowan McAllister, Yarin Gal, Alex Kendall, Mark Van Der Wilk, Amar Shah, Roberto Cipolla, and Adrian Vivian Weller. 2017. Concrete problems for autonomous vehicle safety: Advantages of bayesian deep learning. In *International Joint Conferences on Artificial Intelligence, Inc.*

Chanwoo Park, Jae Myung Kim, Seok Hyeon Ha, and Jungwoo Lee. 2018. Sampling-based bayesian inference with gradient uncertainty. *arXiv preprint arXiv:1812.03285*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8024–8035.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392. Association for Computational Linguistics.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Bidirectional attention flow for machine comprehension. In *Proceedings of the 5th International Conference for Learning Representations*.

Burr. Settles. 2012. Active learning. In *Synthesis Lectures on Artificial Intelligence and Machine Learning*, volume 6.1, pages 1–114.

Yanyao Shen, Hyokun Yun, Zachary Lipton, Yakov Kronrod, and Animashree Anandkumar. 2017. Deep active learning for named entity recognition. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 252–256, Vancouver, Canada, August. Association for Computational Linguistics.

Aditya Siddhant and Zachary C. Lipton. 2018. Deep Bayesian active learning for natural language processing: Results of a large-scale empirical study. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2904–2909, Brussels, Belgium, October-November. Association for Computational Linguistics.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 2692–2700. Curran Associates, Inc.

Shuohang Wang and Jing Jiang. 2016. Machine comprehension using match-lstm and answer pointer. *CoRR*, abs/1608.07905.

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 189–198. Association for Computational Linguistics.

Yijun Xiao and William Yang Wang. 2019. Quantifying uncertainties in natural language processing tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7322–7329.

## A Training Configuration for Clinical QA Scenario

The hyper-parameters of the model adopted on the clinical dialogue comprehension task is shown in Table 3. Moreover, In our hybrid neural architecture, we adopt several strategies which empirically benefit the performance in the training process: (1) In the warm-up training epochs, all weights of the hybrid architecture were updated jointly, with a warm-up learning rate of $2e - 5$. (2) After warm-up training, the prediction component was trained with Bayesian weight estimation by sampling from a mixture of two prior Gaussian distributions, where $\sigma_1 = 0.05$ and $\sigma_2 = 0.1$ (Blundell et al., 2015) or applying Monte Carlo Dropout (Gal and Ghahramani, 2016), and we assigned a learning rate of $1e-3$ to the prediction component while that of the feature representation component was set to $1e-4$; (3) Layer normalization was added in the last layer of the feature representation component, providing feature outputs with lower variance.

| Parameter | Value | Parameter | Value |
|---|---|---|---|
| Feature Layer Dropout Rate | 0.3 | Bayesian Layer Dropout Rate | 0.5 |
| Optimize Algorithm | Adam | Learning Rate | 0.0001 |
| Warm-up Learning Rate | 0.00002 | Warm-up Training Epoch | 2 |
| Hidden Size | 300 | Batch Size | 128 |
| Gradient Norm Clipping | 3.0 | Max Input Length | 300 |

Table 3: Hyper-parameters for the clinical dialogue comprehension task.

## B Evaluation on a Reading Comprehension Benchmark Corpus

In this section, we adapt our approach in a common question-answering benchmark corpus: SQuAD (Rajpurkar et al., 2016). Different from the domain-specific dialogue dataset, models for this benchmark can significantly benefit from utilizing large-scale pre-trained contextual representation. Therefore, following our design in Section 3, here we use a pre-trained language model BERT (Devlin et al., 2019) as the feature representation component, and add two linear layers with Bayesian approximation for the prediction component. We trained the *"bert-base-uncased"* version of BERT along with the prediction component, with a separate optimizer and different learning rate and weight decay configurations. As shown in Table 4, the model can achieve higher performance than the baseline. The uncertainty calculated on all samples of the evaluation set is shown in Figure 3.

| Model | EM Score | F1 Score |
|---|---|---|
| BERT for QA (Traditional) | 81.22 | 88.52 |
| BERT for QA (FAB Bayes-by-Backprop) | 81.30 | 89.81 |
| BERT for QA (FAB MCDO) | 81.75 | 90.05 |

Table 4: Evaluation results on the SQuAD development set.

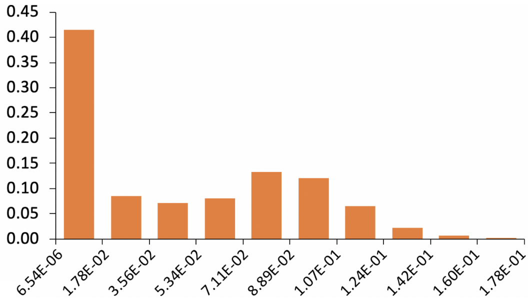

Figure 3: Histogram of epistemic uncertainty values on the SQuaAD evaluation set. X axis is the epistemic uncertainty value. Y axis is the proportion of sample.