On the Calibration and Uncertainty of Neural Learning to Rank Models for Conversational Search

Gustavo Penha
Delft University of Technology
Delft
Netherlands
g.penha-1@tudelft.nl

Claudia Hauff
Delft University of Technology
Delft
Netherlands
c.hauff@tudelft.nl

Abstract

According to the Probability Ranking Principle (PRP), ranking documents in decreasing order of their probability of relevance leads to an optimal document ranking for ad-hoc retrieval. The PRP holds when two conditions are met: **[C1]** the models are *well calibrated*, and, **[C2]** the probabilities of relevance are reported *with certainty*. We know however that deep neural networks (DNNs) are often not well calibrated and have several sources of uncertainty, and thus **[C1]** and **[C2]** might not be satisfied by neural rankers. Given the success of neural Learning to Rank (L2R) approaches—and here, especially BERT-based approaches—we first analyze under which circumstances deterministic neural rankers are calibrated for conversational search problems. Then, motivated by our findings we use two techniques to model the uncertainty of neural rankers leading to the proposed *stochastic rankers*, which output a predictive distribution of relevance as opposed to point estimates. Our experimental results on the ad-hoc retrieval task of conversation response ranking[1] reveal that (i) BERT-based rankers are not robustly calibrated and that stochastic BERT-based rankers yield better calibration; and (ii) uncertainty estimation is beneficial for both risk-aware neural ranking, i.e. taking into account the uncertainty when ranking documents, and for predicting unanswerable conversational contexts.

1 Introduction

According to the Probability Ranking Principle (PRP) (Robertson, 1977), ranking documents in decreasing order of their probability of relevance leads to an optimal document ranking for ad-hoc retrieval[2]. Gordon and Lenk (1991) discussed that

---

[1]The source code and data are available at https://github.com/Guzpenha/transformer_rankers/tree/uncertainty_estimation.

[2]Standard retrieval task where the user specifies his information need through a query which initiates a search by the
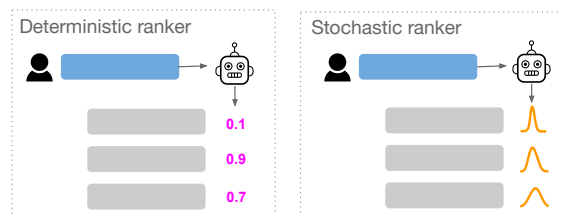

Figure 1: While deterministic neural rankers output a point estimate probability (magenta values) of relevance for a combination of query (blue bars) and document (grey bars), stochastic neural rankers output a predictive distribution (orange curves). The dispersion of the predictive distribution provides an estimation of the model uncertainty.

for the PRP to hold, ranking models must at least meet the following conditions: **[C1]** assign well calibrated probabilities of relevance, i.e. if we gather all documents for which the model predicts relevance with a probability of e.g. 30%, the amount of relevant documents should be 30%, and **[C2]** report certain predictions, i.e. only point estimates such as, for example, 80% probability of relevance.

DNNs have been shown to outperform classic Information Retrieval (IR) ranking models over the past few years in setups where considerable training data is available. It has been shown that DNNs are not well calibrated in the context of computer vision (Guo et al., 2017). If the same is true for neural L2R models for IR, e.g. transformer-based models for ranking (Nogueira and Cho, 2019), **[C1]** is not met. Additionally, there are a number of sources of uncertainty in the training process of neural networks (Gal, 2016) that make it unreasonable to assume that neural ranking models fulfill **[C2]**: *parameter uncertainty* (different combinations of weights that explain the data equally well), *structural uncertainty* (which neural architecture to

---

system for documents that are likely relevant (Baeza-Yates et al., 1999).

use for neural ranking), and *aleatoric uncertainty* (noisy data). Given these sources of uncertainty, using point estimate predictions and ranking according to the PRP might not achieve the optimal ranking for retrieval. While the effectiveness benefits of risk-aware models (Wang, 2009; Wang and Zhu, 2009), which take into account the risk[3], i.e. the uncertainty of the document's prediction scores, have been shown for non-neural IR approaches, this has not yet been explored for neural L2R models.

In this paper we first analyze the calibration of neural rankers, specifically BERT-based rankers for IR tasks to evaluate how calibrated they are. Then, to model the uncertainty of BERT-based rankers, we propose *stochastic* neural ranking models (see Figure 1), by applying different techniques to model the uncertainty of DNNs, namely MC Dropout (Gal and Ghahramani, 2016) and Deep Ensembles (Lakshminarayanan et al., 2017) which are agnostic to the particular DNN.

In our experiments, we test models under *distributional shift*, i.e. the test data distribution is different from the training data, also referred to as out-of-distribution (OOD) examples (Lee et al., 2018). In real-world settings, there are often inputs that are shifted due to factors such as non-stationarity and sample bias. Additionally, this experimental setup provides a way of measuring whether the DNN *"knows what it knows"* (Ovadia et al., 2019), e.g. by outputting high uncertainty for OOD examples.

We find that BERT-based rankers are not robustly calibrated. Stochastic BERT-based rankers have 14% less calibration error on average than BERT-based rankers. Uncertainty estimation from stochastic BERT-based rankers is advantageous for downstream applications as shown by our experiments for risk-aware neural ranking (2% more effective on average relative to a model without risk-awareness) and for predicting unanswerable conversational contexts (improves classification by 33% on average of all conditions).

2 Related Work

Calibration and Uncertainty in IR

Even though to optimally rank documents according to the PRP (Robertson, 1977) requires the model to be calibrated (Gordon and Lenk, 1991) ([**C1**]), the calibration of ranking models has received little attention in IR. In contrast, in the machine learning community there have been a num-

ber of studies about calibration (Ovadia et al., 2019; Maddox et al., 2019), due to the larger decision making pipelines DNNs are often part of and their importance for model interpretability (Thiagarajan et al., 2020). For instance, in the automated medical domain it is important to provide a calibrated confidence measure besides the prediction of a disease diagnosis to provide clinicians with sufficient information (Jiang et al., 2012). Guo et al. (2017) have shown that DNNs are not well calibrated in the context of computer vision, motivating our study of the calibration of neural L2R models.

The second condition ([**C2**]) for optimal retrieval when ranking according to the PRP (Gordon and Lenk, 1991) is that models report predictions with certainty. While the (un)certainty has not been studied in neural L2R models, there are classic approaches in IR that model the uncertainty. Such approaches have been mostly inspired by economics theory, treating variance as a measure of uncertainty (Varian, 1999). Following such ideas, non-neural ranking models that take uncertainty into account (i.e. risk-aware models), and thus do not follow the PRP (Robertson, 1977), have been proposed (Zhu et al., 2009; Wang and Zhu, 2009), showing significant effectiveness improvements compared to the models that do not model uncertainty. Uncertainty estimation is a difficult task that has other applications in IR besides improving the ranking effectiveness: it can be employed to decide between asking clarifying questions and providing a potential answer in conversational search (Aliannejadi et al., 2019); to perform dynamic query reformulation (Lin et al., 2020) for queries where the intent is uncertain; and to predict questions with no correct answers (Feng et al., 2020).

Bayesian Neural Networks

Unlike standard algorithms to train neural networks, e.g. SGD, that fit point estimate weights given the observed data, Bayesian Neural Networks (BNNs) infer a distribution over the weights given the observed data. Denker et al. (1987) contains one of the earliest mentions of choosing probability over weights of a model. An advantage of the Bayesian treatment of neural networks (MacKay, 1992; Neal, 2012; Blundell et al., 2015) is that they are better at representing existing uncertainties in the training procedure. One limitation of BNNs is that they are computationally expensive compared to DNNs. This has lead to the development of techniques that scale well, and do not require mod-

---

[3]In this paper we use risk and uncertainty interchangeably.

ifications of the neural net architecture and training procedure. Gal and Ghahramani (2016) proposed a way to approximate Bayesian inference by relying on dropout (Srivastava et al., 2014). While dropout is a regularization technique that ignores units with probability p during every training iteration and is disabled at test time, Dropout (Gal and Ghahramani, 2016) employs dropout at both train and test time and generates a predictive distribution after a number of forward passes. Lakshminarayanan et al. (2017) proposed an alternative: they employ ensembles of models (Ensemble) to obtain a predictive distribution. Ovadia et al. (2019) showed that Ensemble are able to produce well calibrated uncertainty estimates that are robust to dataset shift.

Conversational Search

Conversational search is concerned with creating agents that fulfill an information need by means of a *mixed-initiative* conversation through natural language interaction. A popular approach to conversational search is its modeling as an ad-hoc retrieval task: given an ongoing conversation and a large corpus of historic conversations, retrieve the response that is best suited from the corpus (this is also known as conversation response ranking (Wu et al., 2017; Yang et al., 2018; Penha and Hauff, 2020; Gu et al., 2020; Lu et al., 2020)). This retrieval-based approach does not require task-specific knowledge provided by domain experts (Henderson et al., 2019), and it avoids the difficult task of dialogue generation, which often suffers from uninformative, generic responses (Li et al., 2016a) or responses that are incoherent given the dialogue context (Li et al., 2016b). One of the challenges of conversational search is identifying unanswerable questions (Feng et al., 2020), which can trigger for instance clarifying questions (Aliannejadi et al., 2019). Identifying unanswerable conversational contexts is one of the applications we employ uncertainty estimation for. Intuitively, if the system has high uncertainty in the available responses, there may be no correct response available. In this paper we focus on pointwise BERT for ranking—a competitive approach for the conversation response ranking task.

3 Method

In this section we introduce the methods used for answering the following research questions: **RQ1** *How calibrated are deterministic and stochastic BERT-based rankers?* **RQ2** *Are the uncertainty estimates from stochastic BERT-based rankers useful for risk-aware ranking?* **RQ3** *Are the uncertainty estimates obtained from stochastic BERT-based rankers useful for identifying unanswerable queries?* We first describe how to measure the calibration of neural rankers (**[C1]**), followed by our approach for modeling and ranking under uncertainty (**[C2]**), and then we describe how we evaluate their robustness to distributional shift.

3.1 Measuring Calibration

To evaluate the *calibration* of neural rankers (**RQ1**) we resort to the Empirical Calibration Error (ECE) (Naeini et al., 2015). ECE is an intuitive way of measuring to what extent the confidence scores from neural networks align with the true correctness likelihood. It measures the difference between the observed reliability curve (DeGroot and Fienberg, 1983) and the ideal one[4]. More formally, we sort the predictions of the model, divide them into c buckets $\{B_0, ..., B_c\}$, and take the weighted average between the average predicted probability of relevance $avg(B_i)$ and the fraction of relevant[5] documents $\frac{rel(B_i)}{|B_i|}$ in the bucket:

$$ECE = \sum_{i=0}^{c} \frac{|B_i|}{n} \left| avg(B_i) - \frac{rel(B_i)}{|B_i|} \right|,$$

where n is the total number of test examples.

3.2 Modeling Uncertainty

First we define the ranking problem we focus on, followed by the BERT-based ranker baseline model (BERT). Having set the foundations, we move to the methods we propose to answer **RQ2** and **RQ3**: a stochastic BERT-based ranker to *model uncertainty* (S-BERT) and a risk-aware BERT-based ranker to *take into account uncertainty provided by S-BERT when ranking* (RA-BERT).

3.2.1 Conversation Response Ranking

The task of conversation response ranking (Zhang et al., 2018; Gu et al., 2019; Tao et al., 2019; Henderson et al., 2019; Penha and Hauff, 2020; Yang et al., 2020) (also known as *next utterance selection*), concerns retrieving the best response given the dialogue context. We choose this specific task due to the large-scale training data available, suitable for the training of neural L2R models. Formally, let $\mathcal{D} = \{(\mathcal{U}_i, \mathcal{R}_i, \mathcal{Y}_i)\}_{i=1}^{N}$ be a data set con-

---

[4]See examples of reliability diagrams in Figure 2.
[5]We consider here binary relevance.

sisting of N triplets: dialogue context, response candidates and response relevance labels. The dialogue context $\mathcal{U}_i$ is composed of the previous utterances $\{u^1, u^2, ..., u^\tau\}$ at the turn τ of the dialogue. The candidate responses $\mathcal{R}_i = \{r^1, r^2, ..., r^k\}$ are either ground-truth responses or negative sampled candidates, indicated by the relevance labels $\mathcal{Y}_i = \{y^1, y^2, ..., y^k\}$[6]. The task is then to learn a ranking function $f(.)$ that is able to generate a ranked list for the set of candidate responses $\mathcal{R}_i$ based on their predicted relevance scores $f(\mathcal{U}_i, r)$.

3.2.2 Deterministic `BERT` Ranker

We use BERT for learning the function $f(\mathcal{U}_i, r)$, based on the representation of the `[CLS]` token. The input for BERT is the concatenation of the context $\mathcal{U}_i$ and the response r, separated by SEP tokens. This is the equivalent of early adaptations of BERT for ad-hoc retrieval (Yang et al., 2019) transported to conversation response ranking. Formally the input sentence to BERT is $concat(\mathcal{U}_i, r) = u^1 \mid [U] \mid u^2 \mid [T] \mid ... \mid u^\tau \mid [SEP] \mid r$, where $\mid$ indicates the concatenation operation. The utterances from the context $\mathcal{U}_i$ are concatenated with special separator tokens $[U]$ and $[T]$ indicating end of utterances and turns. The response r is concatenated with the context using BERT's standard sentence separator $[SEP]$. We fine-tune BERT on the target conversational corpus and make predictions as follows: $f(\mathcal{U}_i, r) = \sigma(FFN(BERT_{CLS}(concat(\mathcal{U}_i, r))))$, where $BERT_{CLS}$ is the pooling operation that extracts the representation of the `[CLS]` token from the last layer and FFN is a feed-forward network that outputs logits for two classes (relevant and non-relevant). We pass the logits through a softmax transformation σ that gives us a probability of relevance. We use the cross entropy loss for training. The learned function $f(\mathcal{U}_i, r)$ outputs a point estimate and we refer to it as `BERT`.

3.2.3 Stochastic `S-BERT` Ranker

In order to obtain a predictive distribution, $R_r = \{f(\mathcal{U}_i, r)^0, f(\mathcal{U}_i, r)^1, ..., f(\mathcal{U}_i, r)^n\}$, which allows us to extract uncertainty estimates, we rely on two techniques, namely `Ensemble` (Lakshminarayanan et al., 2017) and `Dropout` (Gal and Ghahramani, 2016). Both techniques scale well

---

[6]Typically, the number of candidates $k \ll K$, where K is the number of available responses and by design the number of ground-truth responses is usually one, the observed response in the conversational data. In our experiments k=10.

and do not require modifications on the architecture or training of BERT.

Using Deep Ensembles (`S-BERT`^{E}) We train M models using different random seeds without changing the training data, each with its own set of parameters $\{\theta_m\}_{m=1}^M$ and make predictions with each one of them to generate M predicted values: $R_r^E = \{f(\mathcal{U}_i, r)^0, f(\mathcal{U}_i, r)^1, ..., f(\mathcal{U}_i, r)^M\}$. The mean of the predicted values is used as the predicted probability of relevance: `S-BERT`$^E(\mathcal{U}_i, r) = E[R_r^E]$, and the variance $var[R_r^E]$ gives us a measure of the uncertainty in the prediction.

Using MC Dropout (`S-BERT`^{D}) We train a single model with parameters θ and employ dropout at test time and generate stochastic predictions of relevance by conducting T forward passes: $R_r^D = \{f(\mathcal{U}_i, r)^0, f(\mathcal{U}_i, r)^1, ..., f(\mathcal{U}_i, r)^T\}$. The mean of the predicted values is used as the predicted probability of relevance: `S-BERT`$^D(\mathcal{U}_i, r) = E[R_r^D]$, and the variance $var[R_r^D]$ gives us a measure of the uncertainty.

3.2.4 Risk-Aware `RA-BERT` Ranker

Given the predictive distribution R_r, obtained either by `Ensemble` or `Dropout`, we use the following function to rank responses with risk-awareness:

$$\text{RA-BERT}(\mathcal{U}_i, r) = E[R_r] - b * var[R_r]$$
$$-2b \sum_i^{n-1} cov[R_r, R_{r_i}],$$

where $E[R_r]$ is the mean of the predictive distribution, and b is a hyperparameter that controls the aversion or predilection towards risk. Unlike (Zuccon et al., 2011), we are not combining different runs that encompass different model architectures. We instead take a Bayesian interpretation of the process of generating a predictive distribution from a single model architecture. We refer to the rankers as `RA-BERT`^{D} and `RA-BERT`^{E}, when using `S-BERT`^{D}'s predictive distribution and `S-BERT`^{E}'s predictive distribution respectively.

3.3 Robustness to Distributional Shift

In order to evaluate whether we can trust the model's calibration and uncertainty estimates, similar to (Ovadia et al., 2019) we evaluate how robust the models are to different types of shift in the

test data. We do so by training the model using one setting and applying it in a different setting. Specifically for all three research questions we test the models under two settings—cross domain and cross negative sampling—which we describe next.

3.3.1 Cross Domain

We train a model using the training set from one domain known as the source domain $\mathcal{D}_{\mathcal{S}}$ and evaluate it on the test set of a different domain, known as the target domain $\mathcal{D}_{\mathcal{T}}$. This is also known as the problem of domain generalization (Gulrajani and Lopez-Paz, 2020).

3.3.2 Cross Negative Sampling

Pointwise L2R models are trained on pairs of query and relevant document and pairs of query and non relevant document (Lucchese et al., 2017). Selecting the non-relevant documents requires a *negative sampling* (NS) strategy. For the cross-NS condition, we test models on negative documents that were sampled using a different NS strategy than during training, evaluating the generalization of the models on a shifted distribution of candidate documents. We use three NS strategies. In **NS_random** we randomly select candidate responses from the list of all responses. For **NS_classic** we retrieve candidate responses using the conversational context $\mathcal{U}_i$ as query to a conventional retrieval model and all the responses r as documents. In **NS_sentenceEmb** we represent both $\mathcal{U}_i$ and all the responses with a sentence embedding technique and retrieve candidate responses using a similarity measure.

4 Experimental Setup

We consider three large-scale information-seeking conversation datasets[7] that allow the training of neural ranking models for conversation response ranking: **MSDialog** (Qu et al., 2018) contains 246K context-response pairs, built from 35.5K information seeking conversations from the Microsoft Answer community, a QA forum for several Microsoft products; **MANtiS** (Penha et al., 2019) contains 1.3 million context-response pairs built from conversations of 14 Stack Exchange sites, such as *askubuntu* and *travel*; **UDC_DSTC8** (Kummerfeld et al., 2019) contains 184k context-response pairs of disentangled Ubuntu IRC dialogues.

4.1 Implementation Details

We fine-tune BERT (Devlin et al., 2019) (*bert-base-cased*) for conversation response ranking using the *huggingface-transformers* (Wolf et al., 2019). We follow recent research in IR that employed fine-tuned BERT for retrieval tasks (Nogueira and Cho, 2019; Yang et al., 2019), including conversation response ranking (Penha and Hauff, 2020; Vig and Ramea, 2019; Whang et al., 2019). When training BERT we employ a balanced number of relevant and non-relevant—sampled using BM25 (Robertson and Walker, 1994)—context and response pairs. The sentence embeddings we use for cross-NS is sentenceBERT (Reimers and Gurevych, 2019) and we employ dot product calculation from FAISS (Johnson et al., 2017). We consider each dataset as a different domain for cross-NS. We use the Adam optimizer (Kingma and Ba, 2014) with $lr = 5^{-6}$ and $\epsilon = 1^{-8}$, we train with a batch size of 6 and fine-tune the model for 1 epoch. This baseline BERT-based ranker setup yields comparable effectiveness with SOTA methods[8].

4.2 Evaluation

To evaluate the *effectiveness* of the neural rankers we resort to a standard evaluation metric in conversation response ranking (Yuan et al., 2019; Gu et al., 2020; Tao et al., 2019): recall at position K with n candidates[9]: $R_n@K$. To evaluate the *calibration* of the models, we resort to the Empirical Calibration Error (cf. §3.1, using $C = 10$). Throughout, we report the test set results for each dataset. To evaluate the *quality of the uncertainty estimation* we rely on two downstream tasks. The first is to improve conversation response ranking itself via Risk-Aware ranking (cf. §3.2.4). The second, which fits well with conversation response ranking, is to predict unanswerable conversational contexts. Formally the task is to predict whether there is a correct answer in the candidates list $\mathcal{R}$ or not. In our experiments, for half of the instances we remove the relevant response from the list, setting the label as *None Of The Above* (NOTA). The other half of the data has the label *Answerable* (ANSW) indicating that there is a suitable answer
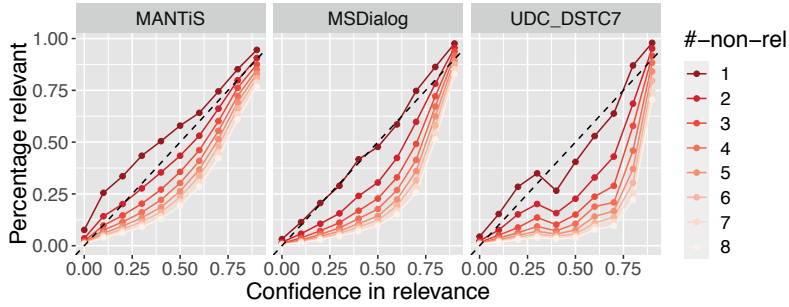

Figure 2: Calibration of `BERT` trained on a balanced set of relevant and non-relevant documents, and tested data with more non-relevant (`#-non-rel`) than relevant (1 per query) documents. A fully calibrated model is represented by the dotted diagonal: for every bucket of confidence in relevance, the % of relevant documents in that bucket is exactly the confidence. The calibration error is the difference between the curves and the diagonal line.

	cross-domain						cross-NS			
Test on →	MANTiS		MSDialog		UDC$_{DSTC8}$		NS$_{random}$		NS$_{sentenceBERT}$	
Train on ↓ (NS$_{BM25}$)	R$_{10}$@1	ECE	R$_{10}$@1	ECE	R$_{10}$@1	ECE	R$_{10}$@1	ECE	R$_{10}$@1	ECE
MANTiS	<u>0.615</u>	<u>0.003</u>	0.653	0.010	0.422	0.028	0.263	0.011	0.310	0.009
MSDialog	0.398	0.009	<u>0.652</u>	<u>0.006</u>	0.495	0.014	0.298	0.029	0.239	0.027
UDC$_{DSTC8}$	0.349	0.016	0.306	0.023	<u>0.834</u>	<u>0.002</u>	0.318	0.050	0.182	0.045

Table 1: Calibration (ECE, lower is better) and effectiveness (R$_{10}$@1, higher is better) of `BERT` for conversation response ranking in cross-domain, and cross-NS conditions. All models were trained using NS$_{BM25}$. ECE is calculated using a balanced number of relevant and non relevant documents. <u>Underlined</u> values indicate no distributional shift ($\mathcal{D}_S = \mathcal{D}_T$ and train NS = test NS).

	cross-domain						cross-NS			
Test on →	MANTiS		MSDialog		UDC$_{DSTC8}$		NS$_{random}$		NS$_{sentenceBERT}$	
Train on ↓ (NS$_{BM25}$)	S-BERT^E	S-BERT^D	S-BERT^E	S-BERT^D	S-BERT^E	S-BERT^D	S-BERT^E	S-BERT^D	S-BERT^E	S-BERT^D
MANTiS	-35.13%[†]	-56.14%[†]	-03.42%	-26.89%[†]	-04.94%	-00.83%	-31.35%	-18.65%[†]	-37.65%[†]	-02.79%
MSDialog	+25.05%	+08.27%	-43.11%	-11.54%	+22.77%	+05.85%	-15.91%	-10.58%	-17.17%	-12.93%
UDC$_{DSTC8}$	-54.95%[†]	-09.98%[†]	-25.78%[†]	-09.15%	+24.77%	-01.84%	-08.05%	-01.78%	-04.81%	-01.28%

Table 2: Relative decreases of ECE (lower is better) of S-BERT^E and S-BERT^D over `BERT`. Superscript [†] denote significant improvements (95% confidence interval) using Student's t-tests.

in the candidates list, for which we remove one of the negative samples instead. Similar to Feng et al. (2020), who proposed to use the outputs (logits) of a LSTM-based model in order to predict NOTA, we use the uncertainties as additional features to the classifier for NOTA prediction. The input space with the additional features is fed to a learning algorithm (Random Forest), and we evaluate it with a 5 fold cross-validation procedure using F1-Macro.

5 Results

5.1 Calibration of Neural Rankers (RQ1)

In order to answer our first research question about the calibration of neural rankers, let us first analyze `BERT` under standard settings (no distributional shift). Our results show that `BERT` is both effective and calibrated under no distributional shift condi-

tions. In Table 1 we see that when the target data (*Test on →*) is the same as the source data (*Train on ↓*)—indicated by underlined values—we obtain the highest effectiveness (on average 0.70 R_{10}@1) and the lowest calibration error (on average 0.036 ECE). When plotting the calibration curves of the model in Figure 2, we observe the curves to be almost diagonal (i.e. having near perfect calibration) when there are an equal number of relevant and non-relevant candidates (`#-non-rel = 1`).

However, when we make the conditions more realistic[10] by having multiple non-relevant candidates for each conversational context, we observe in Figure 2 that the calibration errors start to in-

---

[10]In a production system, the retrieval stage would be executed over all candidate responses. As a consequence, the data is highly unbalanced, i.e. only a few relevant responses among potentially millions of non-relevant responses.

	cross-domain						cross-NS			
Test on →	MANTiS		MSDialog		UDC$_{DSTC8}$		NS$_{random}$		NS$_{sentenceBERT}$	
Train on ↓ (NS$_{BM25}$)	RA-BERT^E	RA-BERT^D	RA-BERT^E	RA-BERT^D	RA-BERT^E	RA-BERT^D	RA-B.^{E}	RA-B.^{D}	RA-B.^{E}	RA-B.^{D}
MANTiS	-0.14%	+0.16%†	+0.00%	+0.00%	+0.00%	+0.00%	+4.73%†	+4.58%†	+9.68%†	-2.68%
MSDialog	-2.74%	+0.39%	-1.05%	-0.66%	+5.08%†	-0.10%	-7.61%	+3.29%	-0.61%	+0.63%
UDC$_{DSTC8}$	+0.00%	+0.00%	+0.00%	+0.00%	+0.42%	-0.06%	+6.32%†	+3.83%†	+16.39%†	+17.18%†

Table 3: Relative improvements (higher is better) of $R_{10}@1$ of RA-BERT^E and RA-BERT^D over the mean of stochastic BERT predictions (S-BERT^E and S-BERT^D). Superscript † denote statistically significant improvements over the S-BERT ranker at 95% confidence interval using Student's t-tests.

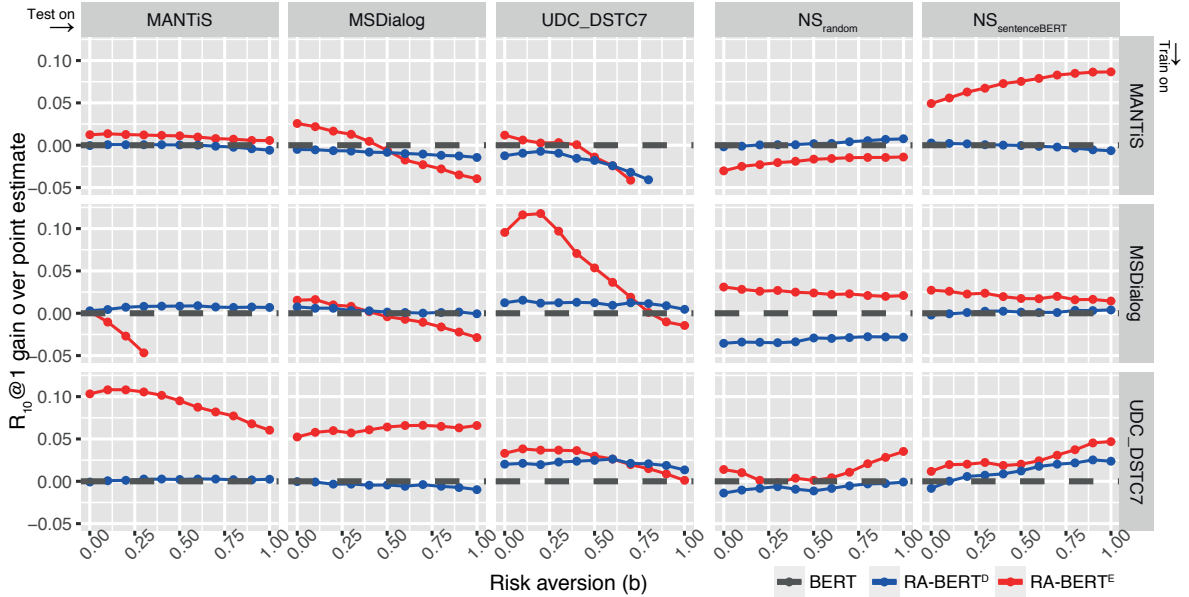
Figure 3: Gains of the Risk-Aware BERT-ranker for different values of risk aversion b.

crease, moving away from the diagonal. Additionally, when we challenge the model in cross-domain and cross-NS settings, the calibration error increases significantly as evident in Table 1. On average, the ECE is **4.6** times higher for cross-domain and **7.9** times higher for cross-NS. Thus **answering the first part of our first research question about the calibration of deterministic BERT-based rankers, indicating that they do not have robust calibrated predictions**, failing on the scenarios where there is a distributional shift.

In order to answer the remaining part of RQ1, on how calibrated *stochastic* BERT-based rankers are, let us consider Table 2. It displays the improvements (relative drop in ECE) over BERT in terms of calibration. S-BERT^E is on average **14%** better (has less calibration error) than BERT, while S-BERT^D is on average **10%** better than BERT, **answering our first research question: stochastic BERT-based rankers are better calibrated than deterministic BERT-based ranker**. We hypothesize that S-BERT^E leads to less ECE than

S-BERT^D because it better captures the model uncertainty in the training procedure, since it combines different weights that explain equally well the prediction of relevance given the inputs. In the next section we focus on evaluating the effectiveness of such models that are better calibrated and also taking into account uncertainty when ranking.

5.2 Uncertainty Estimates for Risk-Aware Neural Ranking (RQ2)

In order to evaluate the quality of the uncertainty estimations, we first resort to using them as a measure of the risk through risk-aware neural ranking (RA-BERT^D and RA-BERT^E). Figure 3 displays the effectiveness in terms of $R_{10}@1$ gains over BERT for the different settings (cross-domain and cross-NS) when varying the risk aversion b.

We note that when $b = 0$, we are using the mean of the predictive distribution and disregard the risk, which is equivalent to S-BERT^D and S-BERT^E. The ensemble based average S-BERT^E is more effective than the baseline BERT for almost all com-

	cross-domain								
Test on →	MANTiS			MSDialog			UDC$_{DSTC8}$		
Train on ↓ (NS$_{BM25}$)	$E[R^D]$	$+var[R^E]$	$+var[R^D]$	$E[R^D]$	$+var[R^E]$	$+var[R^D]$	$E[R^D]$	$+var[R^E]$	$+var[R^D]$
MANTiS	0.635 (.02)	0.686 (.01)†	**0.792 (.02)**†	0.669 (.03)	0.731 (.04)	**0.855 (.02)**†	0.571 (.04)	0.590 (.08)†	**0.621 (.04)**†
MSDialog	0.561 (.02)	0.598 (.02)†	**0.633 (.02)**†	0.662 (.04)	**0.702 (.01)**†	0.699 (.06)†	0.596 (.04)	0.566 (.06)†	**0.655 (.06)**†
UDC$_{DSTC8}$	0.527 (.04)	0.665 (.02)†	**0.738 (.03)**†	0.523 (.05)	0.691 (.03)†	**0.757 (.04)**†	0.787 (.01)	**0.829 (.03)**†	0.807 (.01)†

Table 4: Results of the *cross-domain* condition for the NOTA prediction task, using a Random Forest classifier and different input spaces. The F1-Macro and standard deviation over the 5 folds of the cross validation are displayed. Superscript † denote statistically significant improvements over $E[R^D]$ at 95% confidence interval using Student's t-tests. Bold indicates the most effective approach.

	cross-NS					
Test on →	NS$_{random}$			NS$_{sentenceBERT}$		
Train on ↓ (NS$_{BM25}$)	$E[R^D]$	$+var[R^E]$	$+var[R^D]$	$E[R^D]$	$+var[R^E]$	$+var[R^D]$
MANTiS	0.557 (.01)	0.604 (.02)†	**0.698 (.02)**†	0.534 (.03)	0.587 (.02)†	**0.647 (.05)**†
MSDialog	0.505 (.02)	0.606 (.02)†	**0.702 (.05)**†	0.522 (.03)	0.611 (.07)†	**0.653 (.04)**†
UDC$_{DSTC8}$	0.565 (.03)	0.800 (.02)†	**0.942 (.04)**†	0.506 (.05)	0.755 (.05)†	**0.821 (.05)**†

Table 5: Results of the *cross-NS* condition for the NOTA prediction task.

binations and S-BERT^D is equivalent to the baseline. When using $b < 0$, we are ranking with risk predilection (the opposite of risk aversion), and in all conditions we found that the effectiveness was significantly worse than when $b = 0$ and thus $b < 0$ is not displayed in Figure 3.

When increasing the risk aversion ($b > 0$), we see that it has different effects depending on the combination of domain and NS. For instance, when training in MSDialog and applying on UDC$_{DSTC8}$, increasing the risk aversion improves effectiveness of RA-BERT^E until b reaches 0.25 and after that the effectiveness drops, meaning that too much risk aversion is not effective. In order to investigate whether ranking with risk aversion is more effective than using the predictive distribution mean, we select b based on the best value observed on the validation set. Table 3 displays the results of this experiment, showing the improvements of RA-BERT^D and RA-BERT^E over S-BERT^D and S-BERT^E respectively. The results show that in a few cases (8 out of 30) the best value of b is 0, for which risk-aversion is not the best option in the development set. We obtain effectiveness improvements primarily on the cross-NS condition (up to 17.2% improvement of $R_{10}@1$), which is the hardest condition (when the models are most ineffective, c.f. Table 1). **This answers our second research question, indicating that the uncertainties obtained from stochastic neural rankers are useful for risk-aware rank-**

ing, specially in the cross-NS setting where the baseline model is quite ineffective. RA-BERT^E is on average 2% more effective than S-BERT^E, while RA-BERT^D is on average 1.7% more effective than S-BERT^D.

5.3 Uncertainty Estimates for NOTA prediction (RQ3)

Besides using the uncertainty estimation for risk-aware ranking, we also employ it for the NOTA (None of the Above) prediction task. We compare here different input spaces for the NOTA classifier. $E[R^D]$ stands for the input space that only uses the mean of the predictive distribution for the k candidate responses in $\mathcal{R}$ using S-BERT^D, $+var[R^E]$ uses both $E[R^D]$ and the uncertainties of S-BERT^E for the k candidates and $+var[R^D]$ uses both the scores $E[R^D]$ and the uncertainties of S-BERT^D. Our results show that the uncertainties from S-BERT^D and of S-BERT^E significantly improve the F1 for NOTA prediction for both cross-domain (Table 4, improvement of 24% on average when using S-BERT^D) and cross-NS settings (Table 5, improvement of 46% on average when using S-BERT^D). **We can thus answer our last research question: the uncertainty estimates from stochastic neural rankers do improve the effectiveness of the NOTA prediction task (by an average of 33% across all conditions considered).**

6 Conclusions

In this work we study the calibration and uncertainty estimation of neural rankers, specifically BERT-based rankers. We first show that the deterministic BERT-based ranker is not robustly calibrated for the task of conversation response ranking and we improve its calibration with two techniques to estimate uncertainty through *stochastic neural ranking*. We also show the benefits of estimating uncertainty using risk-aware neural ranking and for predicting unanswerable conversational contexts.

As future work, investigating the use of stochastic rankers in other settings is important, such as other neural L2R architectures, other search and retrieval tasks (Guo et al., 2019; Diaz et al., 2020; Lin et al., 2020), and the ensembling of neural rankers (Zuccon et al., 2011).

Acknowledgements

References

Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W Bruce Croft. 2019. Asking clarifying questions in open-domain information-seeking conversations. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 475–484.

Ricardo Baeza-Yates, Berthier Ribeiro-Neto, et al. 1999. *Modern information retrieval*, volume 463. ACM press New York.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. 2015. Weight uncertainty in neural networks. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning-Volume 37*, pages 1613–1622.

Morris H DeGroot and Stephen E Fienberg. 1983. The comparison and evaluation of forecasters. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 32(1-2):12–22.

John S. Denker, Daniel B. Schwartz, Ben S. Wittner, Sara A. Solla, Richard E. Howard, Lawrence D. Jackel, and John J. Hopfield. 1987. Large automatic learning, rule extraction, and generalization. *Complex Systems*, 1.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*, pages 4171–4186.

Fernando Diaz, Bhaskar Mitra, Michael D. Ekstrand, Asia J. Biega, and Ben Carterette. 2020. Evaluating stochastic rankings with expected exposure.

Yulan Feng, Shikib Mehri, Maxine Eskenazi, and Tiancheng Zhao. 2020. "none of the above": Measure uncertainty in dialog response retrieval. In *ACL*, pages 2013–2020, Online. Association for Computational Linguistics.

Y Gal and Z Ghahramani. 2016. Dropout as a bayesian approximation. In *33rd International Conference on Machine Learning, ICML 2016*, volume 3, pages 1661–1680.

Yarin Gal. 2016. Uncertainty in deep learning. *University of Cambridge*, 1(3).

Michael D Gordon and Peter Lenk. 1991. A utility theoretic examination of the probability ranking principle in information retrieval. *Journal of the American Society for Information Science*, 42(10):703–714.

Jia-Chen Gu, Tianda Li, Quan Liu, Xiaodan Zhu, Zhen-Hua Ling, Zhiming Su, and Si Wei. 2020. Speaker-aware bert for multi-turn response selection in retrieval-based chatbots. *arXiv preprint arXiv:2004.03588*.

Jia-Chen Gu, Zhen-Hua Ling, and Quan Liu. 2019. Utterance-to-utterance interactive matching network for multi-turn response selection in retrieval-based chatbots. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:369–379.

Ishaan Gulrajani and David Lopez-Paz. 2020. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. *arXiv preprint arXiv:1706.04599*.

Jiafeng Guo, Yixing Fan, Liang Pang, Liu Yang, Qingyao Ai, Hamed Zamani, Chen Wu, W Bruce Croft, and Xueqi Cheng. 2019. A deep look into neural ranking models for information retrieval. *Information Processing & Management*, page 102067.

Matthew Henderson, Iñigo Casanueva, Nikola Mrkšić, Pei-Hao Su, Ivan Vulić, et al. 2019. Convert: Efficient and accurate conversational representations from transformers. *arXiv preprint arXiv:1911.03688*.

Xiaoqian Jiang, Melanie Osl, Jihoon Kim, and Lucila Ohno-Machado. 2012. Calibrating predictive model estimates to support personalized medicine. *Journal of the American Medical Informatics Association*, 19(2):263–274.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2017. Billion-scale similarity search with gpus. *arXiv preprint arXiv:1702.08734*.

Seokhwan Kim, Michel Galley, Chulaka Gunasekara, Sungjin Lee, Adam Atkinson, Baolin Peng, Hannes Schulz, Jianfeng Gao, Jinchao Li, Mahmoud Adada, Minlie Huang, Luis Lastras, Jonathan K. Kummerfeld, Walter S. Lasecki, Chiori Hori, Anoop Cherian, Tim K. Marks, Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, and Raghav Gupta. 2019. The eighth dialog system technology challenge.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Jonathan K. Kummerfeld, Sai R. Gouravajhala, Joseph J. Peper, Vignesh Athreya, Chulaka Gunasekara, Jatin Ganhotra, Siva Sankalp Patel, Lazaros C Polymenakos, and Walter Lasecki. 2019. A large-scale corpus for conversation disentanglement. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in neural information processing systems*, pages 6402–6413.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. 2018. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems*, pages 7167–7177.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. A diversity-promoting objective function for neural conversation models. In *NAACL*, pages 110–119.

Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. 2016b. A persona-based neural conversation model. In *ACL*, pages 994–1003.

Sheng-Chieh Lin, Jheng-Hong Yang, Rodrigo Nogueira, Ming-Feng Tsai, Chuan-Ju Wang, and Jimmy Lin. 2020. Query reformulation using query history for passage retrieval in conversational search.

Junyu Lu, Xiancong Ren, Yazhou Ren, Ao Liu, and Zenglin Xu. 2020. Improving contextual language models for response retrieval in multi-turn conversation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1805–1808.

Claudio Lucchese, Franco Maria Nardini, Raffaele Perego, and Salvatore Trani. 2017. The impact of negative samples on learning to rank. In *LEARNER@ ICTIR*.

David JC MacKay. 1992. A practical bayesian framework for backpropagation networks. *Neural computation*, 4(3):448–472.

Wesley J Maddox, Pavel Izmailov, Timur Garipov, Dmitry P Vetrov, and Andrew Gordon Wilson. 2019. A simple baseline for bayesian uncertainty in deep learning. In *Advances in Neural Information Processing Systems*, pages 13153–13164.

Mahdi Pakdaman Naeini, Gregory F Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the... AAAI Conference on Artificial Intelligence. AAAI Conference on Artificial Intelligence*, volume 2015, page 2901. NIH Public Access.

Radford M Neal. 2012. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. *arXiv preprint arXiv:1901.04085*.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. 2019. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems*, pages 13991–14002.

Gustavo Penha, Alexandru Balan, and Claudia Hauff. 2019. Introducing MANtIS: a novel Multi-Domain Information Seeking Dialogues Dataset. *arXiv preprint arXiv:1912.04639*.

Gustavo Penha and Claudia Hauff. 2020. Curriculum learning strategies for ir. In *European Conference on Information Retrieval*, pages 699–713. Springer.

Chen Qu, Liu Yang, W Bruce Croft, Johanne R Trippas, Yongfeng Zhang, and Minghui Qiu. 2018. Analyzing and characterizing user intent in information-seeking conversations. In *SIGIR*, pages 989–992.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. 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 3973–3983.

Stephen E Robertson. 1977. The probability ranking principle in ir. *Journal of documentation*.

Stephen E Robertson and Steve Walker. 1994. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR'94*, pages 232–241. Springer.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Chongyang Tao, Wei Wu, Can Xu, Wenpeng Hu, Dongyan Zhao, and Rui Yan. 2019. Multi-representation fusion network for multi-turn response selection in retrieval-based chatbots. In *WSDM*, pages 267–275.

Jayaraman J Thiagarajan, Prasanna Sattigeri, Deepta Rajan, and Bindya Venkatesh. 2020. Calibrating healthcare ai: Towards reliable and interpretable deep predictive models. *arXiv preprint arXiv:2004.14480*.

Hal R Varian. 1999. Economics and search. In *ACM SIGIR Forum*, volume 33, pages 1–5. ACM New York, NY, USA.

Jesse Vig and Kalai Ramea. 2019. Comparison of transfer-learning approaches for response selection in multi-turn conversations. In *Workshop on DSTC7*.

Jun Wang. 2009. Mean-variance analysis: A new document ranking theory in information retrieval. In *European Conference on Information Retrieval*, pages 4–16. Springer.

Jun Wang and Jianhan Zhu. 2009. Portfolio theory of information retrieval. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 115–122.

Taesun Whang, Dongyub Lee, Chanhee Lee, Kisu Yang, Dongsuk Oh, and HeuiSeok Lim. 2019. Domain adaptive training bert for response selection. *arXiv preprint arXiv:1908.04812*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, pages arXiv–1910.

Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. 2017. Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. In *ACL*, pages 496–505.

Liu Yang, Minghui Qiu, Chen Qu, Cen Chen, Jiafeng Guo, Yongfeng Zhang, W Bruce Croft, and Haiqing Chen. 2020. Iart: Intent-aware response ranking with transformers in information-seeking conversation systems. *arXiv preprint arXiv:2002.00571*.

Liu Yang, Minghui Qiu, Chen Qu, Jiafeng Guo, Yongfeng Zhang, W Bruce Croft, Jun Huang, and Haiqing Chen. 2018. Response ranking with deep matching networks and external knowledge in information-seeking conversation systems. In *SIGIR*, pages 245–254.

Wei Yang, Haotian Zhang, and Jimmy Lin. 2019. Simple applications of bert for ad hoc document retrieval.

Chunyuan Yuan, Wei Zhou, Mingming Li, Shangwen Lv, Fuqing Zhu, Jizhong Han, and Songlin Hu. 2019. Multi-hop selector network for multi-turn response selection in retrieval-based chatbots. In *EMNLP*, pages 111–120.

Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu. 2018. Modeling multi-turn conversation with deep utterance aggregation. In *ACL*, pages 3740–3752.

Jianhan Zhu, Jun Wang, Ingemar J Cox, and Michael J Taylor. 2009. Risky business: modeling and exploiting uncertainty in information retrieval. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 99–106.

Guido Zuccon, Leif Azzopardi, and Keith van Rijsbergen. 2011. Back to the roots: mean-variance analysis of relevance estimations. In *European Conference on Information Retrieval*, pages 716–720. Springer.