Confidence Estimation for Error Detection in Text-to-SQL Systems

Oleg Somov[1,2] **and Elena Tutubalina**[1,3,4]

[1]AIRI, Moscow, Russia
[2]MIPT, Dolgoprudny, Russia
[3]Sber AI, Moscow, Russia
[4]ISP RAS Research Center for Trusted Artificial Intelligence, Moscow, Russia
somov@airi.net, tutubalina@airi.net

Abstract

Text-to-SQL enables users to interact with databases through natural language, simplifying the retrieval and synthesis of information. Despite the success of large language models (LLMs) in converting natural language questions into SQL queries, their broader adoption is limited by two main challenges: achieving robust generalization across diverse queries and ensuring interpretative confidence in their predictions. To tackle these issues, our research investigates the integration of selective classifiers into Text-to-SQL systems. We analyse the trade-off between coverage and risk using entropy based confidence estimation with selective classifiers and assess its impact on the overall performance of Text-to-SQL models. Additionally, we explore the models' initial calibration and improve it with calibration techniques for better model alignment between confidence and accuracy. Our experimental results show that encoder-decoder T5 is better calibrated than in-context-learning GPT 4 and decoder-only Llama 3, thus the designated external entropy-based selective classifier has better performance. The study also reveal that, in terms of error detection, selective classifier with a higher probability detects errors associated with irrelevant questions rather than incorrect query generations.

Code — https://github.com/runnerup96/error-detection-in-text2sql

Extended version — https://arxiv.org/abs/2501.09527

1 Introduction

Text-to-SQL parsing (Zelle and Mooney 1996; Zettlemoyer and Collins 2005) aims at converting a natural language (NL) question to its corresponding structured query language (SQL) in the context of a relational database (schema). To effectively utilize Text-to-SQL models, users must clearly understand the model's capabilities, particularly the range of questions it can accurately respond to. In this context, *generalization ability* is crucial for ensuring accurate SQL query generation, while *interpretative trustworthiness* is essential for minimizing false positives—instances where the model generates incorrect SQL queries that might be mistakenly perceived as correct.

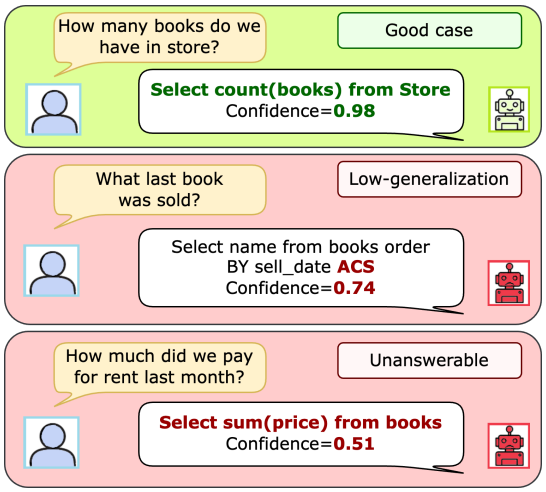

Figure 1: The interaction scenario with Text-to-SQL system. There are three major scenarios where confidence is crucial - good generation detection, error generation, and unanswerable query detection.

Our work explores how model uncertainty estimates can be leveraged to detect erroneous predictions (Fig. 1). We focus on scenarios where models struggle to generalize, such as low compositional generalization, where the model cannot form novel structures not seen in the training set, or low domain generalization, where the model fails to adapt to new schema elements or novel databases. We categorize these errors as *low-generalization*. Additionally, we address cases where questions are unanswerable by the underlying database or require external knowledge, which we refer to as *unanswerable*. Typically, models are trained on question-SQL pairs, so encountering such samples is considered out-of-distribution (OOD), and the model should ideally avoid generating a query. Both types of errors result in incorrect responses, whether as non-executable SQL queries or executable queries that return false responses.

To study compositional and domain generalization in Text-to-SQL, several benchmarks and datasets have been developed over the years to better approximate real-world scenarios, address various aspects of model performance: complex queries involving join statements across multiple

tables (Yu et al. 2018), new and unseen database schemas (Gan, Chen, and Purver 2021; Lee, Polozov, and Richardson 2021), compositional train and test splits (Shaw et al. 2021; Finegan-Dollak et al. 2018), robustness test sets (Bakshandaeva et al. 2022; Chang et al. 2023), dirty schema values and external knowledge requirements (Li et al. 2024; Wretblad et al. 2024), domain-specific datasets that feature unanswerable questions (Lee et al. 2022). To sum up, most prior work evaluates *either* different types of generalization, noisy data *or* model uncertainties.

In this paper, we ask: *can we identify error generations in Text-to-SQL LMs under distribution shift using uncertainty estimation*? Specifically, we examine this from the point of view of *error detection* and *calibration*, examining if the models' probability estimates align accurately with the actual correctness of the answers. We apply T5 (Raffel et al. 2020), GPT 4 (Achiam et al. 2023), and Llama 3 (Meta 2024) with a reject option[1] over popular SPIDER (Yu et al. 2018) and EHRSQL (Lee et al. 2022), covering general-domain and clinical domains. Our findings indicate that in distribution shift settings (cross-database or compositional), the selective classifier achieves high recall but low precision in error detection, leading to loss of total generated queries and deterioration in overall Text-to-SQL system quality. Our analysis also revealed that unanswerable queries are more likely to be detected using our confidence estimates than incorrect queries (Sec. 4). Conversely, in a benchmark with unanswerable queries without a significant compositional or domain shift, the Text-to-SQL system with a selective classifier performs better overall in error detection with a lower rejection loss of correct queries. Furthermore, we examined the calibration characteristics of logit-based confidence estimates (Sec. 5). Under distribution shift, all fine-tuned models lacked proper calibration. Post-hoc calibration methods such as Platt Calibration and Isotonic Regression improved the initial models' calibration, underscoring the importance of calibration techniques in enhancing the reliability of model predictions. As a result, experiments demonstrate that while decoder-only models perform better on certain datasets (compositional or cross-database), encoder-decoder methods exhibit superior calibration for Text-to-SQL after post-hoc calibration. In Sec. 6, we did an in-depth analysis of the relation of selective classifier confidence and generated query complexity.

2 Related Work

Uncertainty estimation and error detection The reliability of Text-to-SQL models or question answering systems, in general, is closely tied to their calibration ability for error detection and result interpretation. Selective prediction, where a model can choose to predict or abstain, has been a longstanding topic in machine learning (Chow 1957; El-Yaniv and Wiener 2010; Dong, Quirk, and Lapata 2018).

The rise of LLMs has highlighted the issue of hallucinations in NLP. Uncertainty estimation is a key research area for developing calibrated Text-to-SQL systems and reliable selective prediction algorithms. Several recent works (Malinin and Gales 2021; van der Poel, Cotterell, and Meister 2022; Ren et al. 2023; Vazhentsev et al. 2023; Fadeeva et al. 2023) have developed methods to estimate uncertainty in language models, aiming to provide better-calibrated uncertainty estimates or to perform error and out-of-domain (OOD) detection. A relevant approach is utilized in (Kadavath et al. 2022), where the authors created a prompt asking the model if the generated prompt is correct. The model's calibration was then measured by the probability of predicting the correct answer when it was correct across the validation set.

One of the most popular directions includes methods that deal with $p(y|x)$ only. These include softmax maximum probability (Hendrycks and Gimpel 2017), temperature scaling (Guo et al. 2017), and ensembles of deep neural networks for uncertainty estimate (Lakshminarayanan, Pritzel, and Blundell 2017) methods. For auto-regressive models, there are several probabilistic approaches, which utilize the softmax distribution, such as normalized sequence probability (Ueffing and Ney 2007) and average token-wise entropy (Malinin and Gales 2021). In our work, we follow the recent approach presented in (Yang et al. 2024), which introduces a maximum entropy estimate for predicting sequence uncertainty. This approach is a better fit for semantic domains where false positive generations must be avoided at all costs. Here, the model's confidence in sequence prediction is determined by its weakest token prediction. These methods aim to provide a calibrated estimate that can be utilized later with a threshold. In contrast to uncertainty estimates and subsequent threshold selection, there are methods that incorporate an OOD-detection component in addition to $p(y|x)$. Chen et al. (2023) developed an additional model to the Text-to-SQL component – a binary classification model with input of question and generated SQL.

Text-to-SQL Text-to-SQL, as a sub-domain of semantic parsing, is deeply influenced by distribution shifts (Suhr et al. 2020; Finegan-Dollak et al. 2018). On one side, there is domain shift, where a model trained on one set of databases must perform well on another set. Multiple datasets like SPIDER (Yu et al. 2018) and BIRD (Li et al. 2024) are designed to evaluate this aspect. On the other side, there is compositional shift, involving novel SQL query structures in the test dataset. (Shaw et al. 2021; Finegan-Dollak et al. 2018) explored the models' ability to generalize to novel SQL templates and noted a significant accuracy drop in results. However, these datasets and splits include only answerable questions for the underlying databases. Recently, a ERHSQL benchmark (Lee et al. 2024b) was presented with a covariate shift, featuring unanswerable queries in the test set or those needing external knowledge.

[1]Selective classification, or classification with a reject option, attempts to abstain on examples that the model is likely to get wrong. This concept was introduced decades ago (Chow 1957, 1970) for decision-making in scenarios where mistakes are costly but abstentions are allowed. In general, by allowing a classifier to abstain, one can improve the performance of a model at the cost of reducing coverage and classifying fewer samples. Further, we use the term "selective Text-to-SQL" or "Text-to-SQL with a reject option".

To sum up, while there has been considerable research on uncertainty estimation, such as calibration in semantic parsing (Stengel-Eskin and Van Durme 2023) and uncertainty constraints (Qin et al. 2022) for better calibration, to our knowledge, there is no evident research on selective prediction for probabilistic uncertainty estimation in Text-to-SQL under distribution shifts. In our work, we explore the calibration characteristics of sequence-to-sequence models under different various distribution shift settings (cross-database shift, compositional shift, and covariate shift). Our goal is to detect incorrect generations or generations involving OOD examples, as seen in ERHSQL.

3 Problem Setup

We study selective prediction in Text-to-SQL systems under distribution shift settings. Specifically, we examine a Text-to-SQL system consisting of two components: the Text-to-SQL model $\mathcal{Y}$, which takes the natural language utterance x and generates an SQL query $\hat{y}$, and a selective classifier $\mathcal{C}$, which decides whether to output the generated query $\hat{y}$ or abstain based on the uncertainty estimate score u. In this section, we formally outline our method for calculating u for generated SQL queries, the selective prediction setup, and the data we evaluated. In three consecutive studies, we investigate the balance between coverage and accuracy of Text-to-SQL models with a reject option (Sec. 4), model calibration, and the relationship between the confidence of the selective classifier and query characteristics (Sec. 5 and 6).

Text-to-SQL Models

In our Text-to-SQL system with reject option, we utilize four models known for their descent ability toward SQL generation. We employ T5-large and T5-3B models from the encoder-decoder family, an in-context-learning GPT-based DAIL-SQL (Gao et al. 2024) and a decoder model Llama 3, which is fine-tuned using both supervised fine-tuning (SFT) and parameter-efficient fine-tuning (PEFT) techniques with LoRa (Hu et al. 2022). We fine-tune both the T5 and Llama models. To form an input x, fine-tuned model receives question q along with database schema S. In in-context-learning with DAIL-SQL, we additionally incorporate relevant question-query pairs as examples for ChatGPT prompt. The model is expected to generate a query $\hat{y}$. The hyperparameters of the fine-tuning are specified in Appendix A.

Uncertainty Estimate

Given the input sequence x and output sequence y the standard auto-regressive model parameterized by θ is given by:

$$P(y|x,\theta) = \prod_{l=0}^{L} P(y_l|y_{<l}, x, \theta)$$

Where the distribution of each y_l is conditioned on all previous tokens in a sequence $y_{<l} = y_0, ..., y_{l-1}$.

For fine-tuned models, we base our heuristic based on intuition a sequence is only as good as its weakest token prediction $P(y_l|y_{<l}, x, \theta)$ to get the uncertainty estimate u of the whole sequence y. If the model soft-max probabilities p_l are close to uniform, the token prediction is less likely to be correct, in contrast to a peak distribution, where the model is certain about token prediction.

$$\begin{aligned} p_l &= P(y_l|y_{<l}, x, \theta) \\ H(p_l) &= \sum_{v=0}^{|V|} p_v log(p_v) \\ u &= max(H_0, ..., H_L) \end{aligned} \tag{1}$$

For ChatGPT-based DAIL-SQL, we do not have access to the full vocabulary distribution. Therefore, we utilize the Normalized Sequence Probability modification (Ueffing and Ney 2005), which was recently featured in one of the EHRSQL shared task solutions (Kim, Han, and Kim 2024):

$$u = \frac{1}{|L|} \sum_{l=0}^{|L|} log(p_l) \tag{2}$$

Selective Prediction Setting

In the selective prediction task, given a natural language input x, the system outputs $(\hat{y}, u)$ where $\hat{y} \in \mathcal{Y}(x)$ is the SQL query generated by the Text-to-SQL model $\mathcal{Y}$, and $u \in \mathcal{R}$ is the uncertainty estimate. Given a threshold $\gamma \in \mathcal{R}$, the overall Text-to-SQL system predicts the query $\hat{y}$ if $u \geq \gamma$; otherwise, it abstains. The rejection ability is provided by selective classifier $\mathcal{C}$.

Following the experimental setup of El-Yaniv and Wiener (2010), we utilize a testing dataset D_{tst}, considered out-of-distribution (OOD) relative to the training dataset D_{tr}. We split D_{tst} independently and identically into two data samples: a known OOD sample D_{known} and an unknown OOD sample D_{unk}. We use D_{known} to fit our selective classifier or calibrator, and D_{unk} for evaluation.

The main characteristics of the selective classifier $\mathcal{C}$ are its *coverage* and *risk*. Coverage is the fraction of D_{unk} on which the model makes correct predictions, and risk is the error fraction of D_{unk}. As the threshold γ decreases, both risk and coverage increase. We evaluate our experiments in terms of the risk vs. coverage paradigm.

To define the target $\hat{y}$ for selective classifiers in Text-to-SQL, we use the inverted execution match metric (EX) for the gold query g_i and predicted query p_i as defined in Equation 3. This means that we set the positive class as the presence of an error.

$$\hat{y}_i = \begin{cases} 0 & \text{if EX}(g_i) == \text{EX}(p_i) \\ 1 & \text{if EX}(g_i) \neq \text{EX}(p_i) \end{cases} \tag{3}$$

To evaluate results for a particular choice of γ, we utilize recall and precision metrics. Coverage refers to our ability to identify and abstain from wrong SQL query generations, while risk corresponds to the proportion of false positive predictions (incorrect queries deemed correct). Recall measures how effectively the system detects errors, and False Discovery Rate (FDR) $(1 - \text{precision})$ indicates the extent to which we abstain from returning correct SQL queries to the user. For a comprehensive assessment of the selective

performance across different threshold values γ, we employ the Area Under the Curve (AUC) metric.

Distribution Shift in Text-to-SQL

We evaluate the uncertainty estimates of Text-to-SQL models in distribution shift settings, mimicking various types of shifts: domain shift, compositional shift, and covariate shift. Domain and compositional shifts are full shift examples where $p(x_{tst}) \neq p(x_{tr})$ and $p(y_{tst}|x_{tst}) \neq p(y_{tr}|x_{tr})$, while covariate shift involves only a change in $p(x_{tst}) \neq p(x_{tr})$. Our Text-to-SQL pairs D follow $p(x)$ and $p(y|x)$ for training and testing.

To evaluate such distribution shifts, we leverage two Text-to-SQL datasets: SPIDER-based PAUQ (Bakshandaeva et al. 2022) and EHRSQL (Lee et al. 2022). The PAUQ dataset is a refinement of the widely recognized non-synthetic benchmark SPIDER (Yu et al. 2018) for the Text-to-SQL task. We prefer English version of PAUQ over SPIDER because it has 8 times fewer empty outputs ($1665 \rightarrow 231$) and 4 times fewer zero-return queries with aggregations (e.g., *maximal*, *minimal*) ($379 \rightarrow 85$). This improvement is crucial, as zero or empty returns can be considered correct when a model generates an executable yet incorrect SQL query, which is undesirable for our study's focus on execution match in the selective classifier. EHRSQL is a clinical Text-to-SQL dataset that includes pairs of input utterances and expected SQL queries. It covers scenarios where generating an SQL query is not possible for a given question.

We utulize the following splits to represent different aspects of compositionality (Hupkes et al. 2020, 2023):

- **PAUQ in cross-database setting** - This setting uses the original SPIDER dataset split, where the data is divided between training and testing sets with no overlap in database structures. During training on D_{tr}, the model must learn to generalize to novel database structures found in D_{tst}. We refer to this split as **PAUQ XSP**.
- **PAUQ with template shift in single database setting** - This is the compositional PAUQ split based on templates in a single database setting (same set of databases across D_{tr} and D_{tst}), inspired by (Finegan-Dollak et al. 2018). This split forces the model to demonstrate its *systematicity* ability—the ability to recombine known SQL syntax elements from D_{tr} to form novel SQL structures in D_{tst}. We refer to this split as **Template SSP**.
- **PAUQ with target length shift in single database setting** - This is another compositional split of the PAUQ dataset, based on the length of SQL queries, in a single database setting (Somov and Tutubalina 2023). Shorter samples are placed in D_{tr}, and longer samples in D_{tst}, ensuring that all test tokens appear at least once in D_{tr}. We refer to this as **TSL SSP**. It tests the model's *productivity*—its ability to generate SQL queries that are longer than those it was trained on.
- **EHRSQL with unanswerable questions** - This setting uses the original EHRSQL split. Its distinctive feature is the presence of unanswerable questions in D_{tst}. These questions cannot be answered using the underlying database content or require external knowledge, making it impossible to generate a correct SQL query. We refer to this split as **EHRSQL**.

For a comparison of our Template SSP and TSL SSP splits with related work, please see Appendix B.

4 Case Study #1: Selective Text-to-SQL

In this case study, we aim to address the following research questions: **RQ1**: Among selective classifiers, which classifier offers the best trade-off between coverage and risk? **RQ2**: What is the impact on the performance of a Text-to-SQL system when the system is expanded with a selective classifier? **RQ3**: What distribution shifts present the most significant challenge for Text-to-SQL with a reject option? **RQ4:** Given the existence of unanswerable questions in the test set, what types of errors are we more likely to find with a selective classifier?

In our selective classifier methods, we utilize approaches outlined in (El-Yaniv and Wiener 2010; Lee et al. 2022), including the threshold-based approach (Lee et al. 2024a), Logistic Regression, and Gaussian Mixture clustering.

Logistic regression We determine parameters θ using the sigmoid function. During inference, we predict the probability of u_i corresponding to the error prediction based on the probability score of the sigmoid function with fitted parameters.

$$p(y_i|u_i, \theta) = \frac{1}{1 + e^{-(\theta_0 + \theta_1 u_i)}} \qquad \hat{y}_i = [p(y_i|u_i, \theta) > 0.5] \tag{4}$$

Gaussian mixture clustering We consider our uncertainty estimates as a combination of two normal distributions, denoted as $\mathcal{N}_z$. The first distribution, z_0, is associated with the uncertainty scores u_i of correct generations, while the second distribution, z_1, is linked to error generations. We use the expectation-maximization algorithm (EM) to determine the parameters μ_z, σ_z, and the mixture weight π_z for each distribution z. During inference, we predict the most likely distribution for a given uncertainty estimate u_i using:

$$\hat{y}_i = \arg\max_z \left(\pi_z \mathcal{N}(u_i \mid \mu_k, \sigma_k)\right) \tag{5}$$

Results

We evaluated our five models on four distinct datasets, using the F_β score to compare methods: $F_\beta = ((1+\beta^2)\text{tp})/((1+\beta^2)\text{tp}+\text{fp}+\beta^2\text{fn})$, tp and fp stand for false and true positives, respectively, fn for false negatives.

To address **RQ1**, we created a heatmap of $F_{\beta=1}$ scores across all splits and models in Fig. 2. Gaussian Mixture Model demonstrates the best trade-off between precision and recall in a task of error detection. For an in-depth analysis in Appendix C we plotted the F_β scores for other β favoring precision or recall.

Based on selective classification Text-to-SQL tables in Appendix D we built the risk vs coverage comparison in Figure 3 for Gaussian Mixture to answer **RQ2** and **RQ3**. As shown in Figure 3 (left), Gaussian Mixture effectively

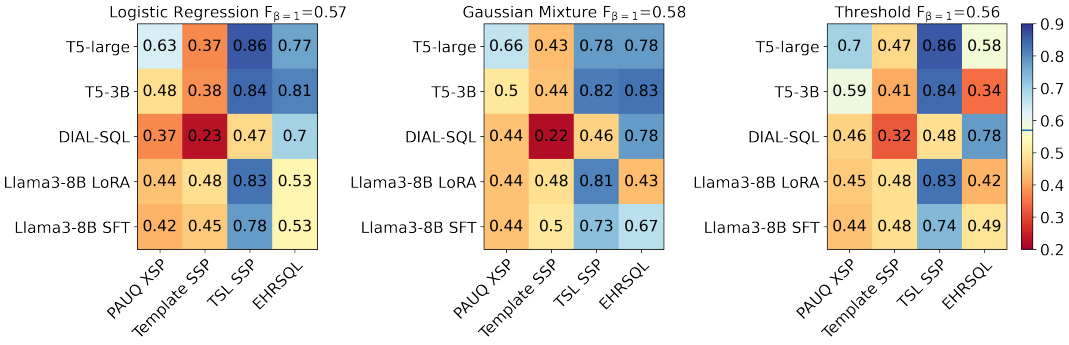

Figure 2: Heatmaps of $F_{\beta=1}$ per split and model for every selective classifier (Logistic Regression, Gaussian Mixture, and Threshold).

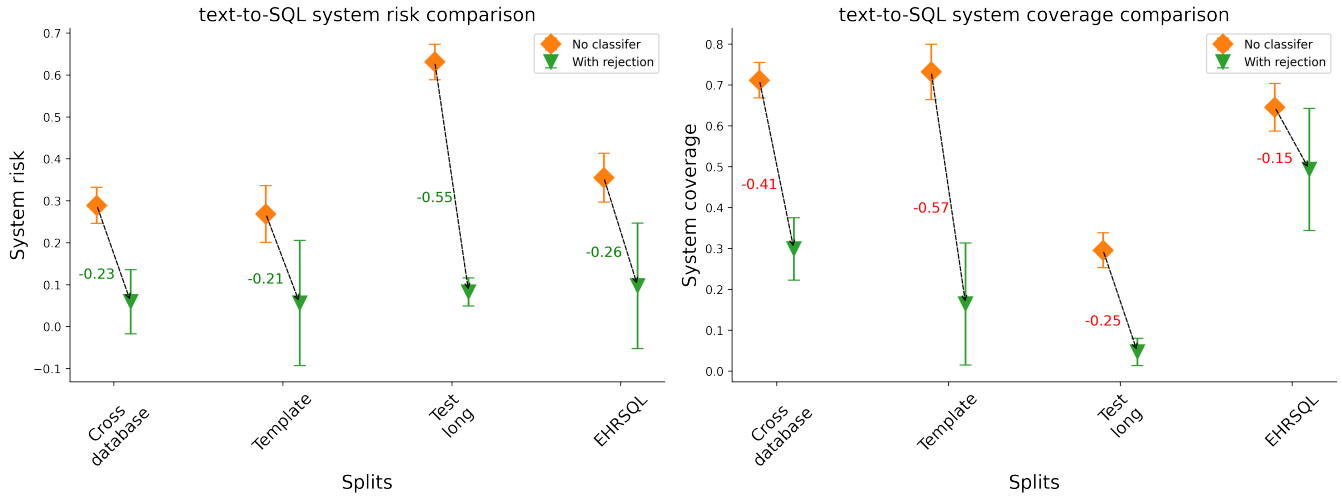

Figure 3: **Left:** The system risk decrease with a Gaussian Mixture for every split averaged between all SQL generation models. **Right**: The system coverage decrease with the presence of an Gaussian Mixture external classifier for every split averaged between all SQL generation models.

identifies error generations, reducing risk by an average of 80% across all splits and models, even under distribution shift. However, this comes at the cost of a high false discovery rate (FDR), as also seen in Figure 3 (right). Under these conditions, system coverage remains low, yielding only 1-2 correct generations per 5 requests. However in settings with minimal full shift, such as EHRSQL, selective Text-to-SQL performs well, achieving an FDR as low as 10% with some models.

Confirming the results of Figure 2 in Table 1 we average the scores of Recall, FDR, and Result EX from Appendix D tables, with Gaussian Mixture having the lowest FDR hence does not worsen the Result EX as other two methods.

For the further analysis of **RQ2** and **RQ3**, we adopt AUC for the selective performance across various threshold values in Appendix E using the probability scores from the Gaussian Mixture classifier. T5-large and T5-3B consistently show superior performance in comparison to other models, especially in the fourth split where they achieve the highest AUC scores (0.93). This suggests that T5-large and T5-3B models are more reliable in terms of Text-to-SQL with a reject option.

To address **RQ4**, we delved into the types of errors most commonly encountered with the Gaussian Mixture selective classifier in the ERHSQL dataset, as shown in Appendix F. Overall, there is a higher chance of encountering a generation of irrelevant questions as opposed to encountering an incorrect generation. This indicates that all models are fairly confident in generating an incorrect query to a relevant question as opposed to generating a query to an irrelevant one. Furthermore, T5-3B, being the most calibrated model as indicated in Table 2, is capable of accurately detecting even incorrect generations.

Takeaway 1 (RQ1, RQ2) The *Gaussian Mixture Model demonstrated the best trade-off between coverage and risk.* The addition of a selective classifier, particularly the *Gaussian Mixture Model, enhances the performance of the Text-to-SQL system by maintaining low False Discovery Rate (FDR) values,* especially on the EHRSQL dataset. This in-

	Recall	FDR	Result EX
Gaussian Mixture	0.798	**0.364**	**0.251**
Logistic Regression	**0.873**	0.469	0.145
Threshold	0.872	0.471	0.143

Table 1: Overall methods comparison averaged across all splits and models.

	MinMax	Platt	Isotonic
T5-large	0.2	0.121	**0.117**
T5-3B	0.17	0.108	**0.106**
Llama3-8B LoRA	0.22	0.216	**0.199**
Llama3-8B SFT	0.21	0.19	**0.175**
DIAL-SQL	0.239	0.16	**0.152**

Table 2: Calibration methods comparison of Brier scores averaged across all splits for each model.

dicates strong error detection capabilities with minimal negative impact on SQL traffic.

Takeaway 2 (RQ3) *The Template SSP split and TSL SSP split were identified as presenting significant challenges for all models.* At the same time, models trained on the EHRSQL dataset under less domain and compositional shifts, show that the error detection method operate much more effectively.

Takeaway 3 (RQ4) *Selective classifier has a higher likelihood of spotting generations to irrelevant questions compared to incorrect generations.* This suggests that the models are generally more confident in generating incorrect queries for relevant questions than in generating queries for irrelevant ones. T5-3B, being the most calibrated, effectively detects incorrect generations.

5 Case Study #2: Calibration Characteristics

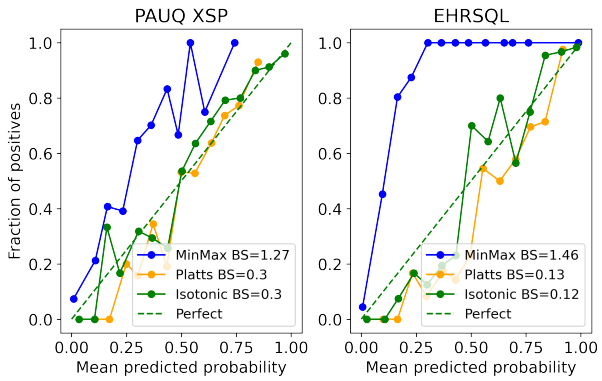

Figure 4: The calibration effect on T5-3B on PAUQ XSP (cross-database setting) and EHRSQL (single clinical database) compared across MinMax, Platts, and Isotonic calibration (BS stands for Brier score).

In this section, we will investigate the following research question (**RQ5**): How do different calibration methods and training datasets influence the calibration of model uncertainty scores, and what trade-offs exist between calibration measures and model execution accuracy? Specifically, the model calibration addresses the question: out of all the instances where we predicted an 80% chance of a query being correct, how often was the query actually correct? A well-calibrated model would have this proportion close to 80%. In this case study we want to measure the calibration of the uncertainty estimates.

In contrast to (Stengel-Eskin and Van Durme 2023) on calibration in semantic parsing, we measure uncertainty estimates at the sequence level, as this is most relevant for system safety. We define the positive class as an execution match result if $EX(g_i) == EX(p_i)$. For calibration of our score u, two calibration methods - Platt calibration and Isotonic calibration - and a naive normalization method (MinMax scaling) were used.

MinMax normalization can be applied here because the maximum entropy estimate is a monotonic function. This allows us to transform the value range from $[0; +\infty]$ to $[0; 1]$. We refer to the calibrated score as u^c: $u_i^c = \frac{(u_i - min(u_i))}{max(u_i) - min(u_i)}$

Platt calibration (Platt et al. 1999) is represented by a logistic regression function from Eq. 4. The parameters θ_0 and θ_1 are selected on a D_{known} using the maximum likelihood method.

Isotonic regression (Zadrozny and Elkan 2002) involves constructing a piece-wise constant function of the form $g_m(u_i) = \theta_m$ to transform uncertainty estimates by minimizing the quadratic difference. As g_m is a piece-wise constant function, the training of this calibration method involves solving the following optimization problem:

$$\begin{aligned} \min_{M,\theta,a} \quad & \sum_{m=1}^{M}\sum_{i=1}^{N} \mathbb{1}(a_m \le u_i < a_{m+1})(y_i - \theta_m)^2 \\ \text{s.t.} \quad & 0 \le a_1 \le a_2 \le ... \le a_{M+1} = 1, \\ & \theta_1 \le \theta_2 \le ... \le \theta_M \end{aligned} \tag{6}$$

where M is the number of function intervals; $a_1, \ldots, a_{M+1}$ are intervals' boundaries; $\theta_0, \ldots, \theta_M$ are the values of the function g_m. During fitting on D_{known}, Isotonic regression optimizes the heights of the histogram columns for function calibration.

Experimental Results

Evaluation metric While other calibration comparison methods, such as expected calibration error (ECE), exist, the Brier score (Ashukha et al. 2020) offers a more interpretable means of comparing models. If the model is confident in the positive class (indicated by a high estimate of u_i^c), then the difference will be minimal. The Brier scoring method estimate shows the squared difference between the target variable y_i (which can be either 1 or 0) and the predicted probability: Brier Score $= \sum_{i=1}^{N}(y_i - u_i^c)^2$

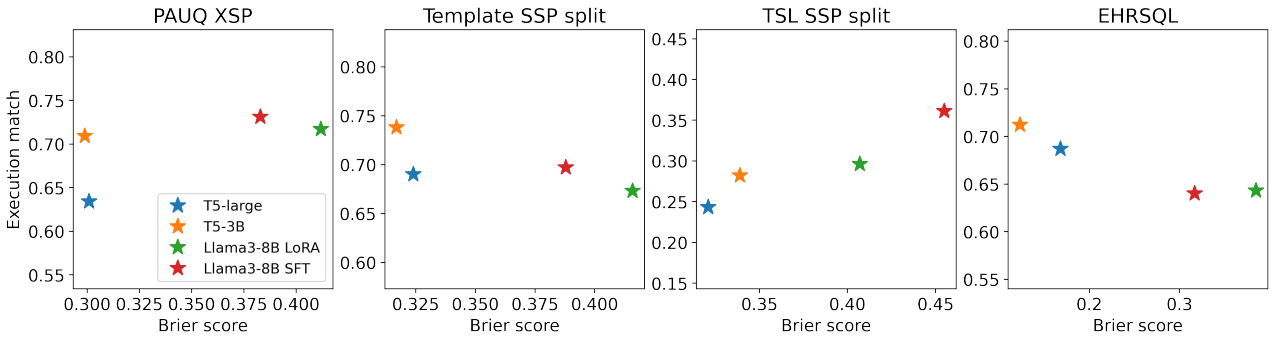

Figure 5: Trade-off plots between execution match and calibration for selected Text-to-SQL models (T5-large, T5-3B, Llama 3 in SFT and LoRa setting).

Results Table 2 presents a comparison of calibration methods using the Brier score, averaged across all data splits. Isotonic regression consistently outperforms both MinMax normalization and Platt calibration across all models, demonstrating its effectiveness in enhancing calibration quality. Notably, as the size of the T5 models increases, the performance of isotonic regression shows improvement.

Fig. 4 illustrates the calibration curves for Platt, Isotonic, and MinMax for T5 across two datasets. As shown in the figure, the original normalized uncertainty estimate is not calibrated, whereas the calibration methods provide significant improvements. In Appendix G we present the isotonic calibrations across all models for TSL SSP and EHRSQL splits, averaged over multiple seeds. It is evident that the shifted datasets (PAUQ XSP, Template SSP, and TSL SSP) do not lead to calibrated models. However, in the EHRSQL split with no complete shift, the models demonstrate effective calibration.

Overall in Figure in 5, we see that our models exist on a trade-off of calibration and generation quality, with some models being of lower generalization quality but a better calibration.

Takeaway 4 (RQ5) The results indicate that *the original entropy estimate of the models' uncertainty is not calibrated. Isotonic regression consistently outperforms other calibration methods like MinMax normalization and Platt calibration* across various models. Additionally, *encoder-decoder architecture models are found to be better calibrated compared to decoder-only models*.

6 Case Study #3: Query Complexity Analysis

In this case study, we investigate the relationship between model confidence in a selective classifier and query complexity, specifically focusing on query length and the number of schema elements in the generated query. Our research question (**RQ6**): Is the probability of rejection by the selective classifier related to query complexity characteristic?

To address this question, we utilized the Gaussian Mixture probabilities (Sec. 4) of the incorrectly generated examples by the T5 model, as T5 models demonstrated the best selective performance across various thresholds (Appendix E). We assessed query complexity using two key indicators: the length of the generated query(in SQL tokens) and the number of unique schema elements in the generated query. Scatter plots in Appendix H were constructed for each data split to analyze the relationship between these query characteristics and the confidence of the selective classifier. Our initial hypothesis was that more complex incorrect queries, characterized by greater length or more schema elements, would correspond to lower probability scores, leading the selective classifier to correctly identify these as incorrect generations. However as the plots show, selective classifier probability does not hold any seeming relation to query complexity.

Takeaway 5 (RQ6) *Contrary to our hypothesis, we did not observe a proportional decline in model confidence for incorrect queries as query complexity increased. Across all splits, even for the most calibrated models, there was no clear relationship between selective classifier confidence and query complexity.*

7 Conclusion

In this paper, we investigated error detection and calibration, utilizing Text-to-SQL LMs with a reject option across general-domain and clinical datasets. We believe our findings could enhance the development of more trustworthy Text-to-SQL models in the future. Future research might concentrate on evaluation across a wider range of datasets and different aspects of compositionality.

Limitations Given that our analysis focused on the SPIDER and EHRSQL datasets, the generalizability of our findings may be limited. We concentrated solely on these domains to validate our results, which may not fully capture the variability of noise distributions across different datasets. However, we consider this a minor limitation, as our goal was to observe the models' behavior under distribution shifts rather than to propose and validate a new model with a reject option.

Ethics Statement The models and datasets used in this work are publicly available for research purposes. All experiments were conducted on four A100 80GB GPUs. Our PyTorch/Hugging Face code will be released with the paper, and we do not anticipate any direct social consequences or ethical issues.

Acknowledgments

We thank Veronica Ganeeva for the figure design. This work was supported by a grant for research centers in the field of artificial intelligence, provided by the Analytical Center for the Government of the Russian Federation in accordance with the subsidy agreement (agreement identifier 000000D730321P5Q0002) and the agreement with the Ivannikov Institute for System Programming of the Russian Academy of Sciences dated November 2, 2021 No. 70-2021-00142.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Ashukha, A.; Lyzhov, A.; Molchanov, D.; and Vetrov, D. 2020. Pitfalls of in-domain uncertainty estimation and ensembling in deep learning. *arXiv preprint arXiv:2002.06470*.

Bakshandaeva, D.; Somov, O.; Dmitrieva, E.; Davydova, V.; and Tutubalina, E. 2022. PAUQ: Text-to-SQL in Russian. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, 2355–2376. Association for Computational Linguistics.

Chang, S.; Wang, J.; Dong, M.; Pan, L.; Zhu, H.; Li, A. H.; Lan, W.; Zhang, S.; Jiang, J.; Lilien, J.; et al. 2023. Dr. Spider: A Diagnostic Evaluation Benchmark towards Text-to-SQL Robustness. In *The Eleventh International Conference on Learning Representations*.

Chen, S.; Chen, Z.; Sun, H.; and Su, Y. 2023. Error detection for text-to-sql semantic parsing. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 11730–11743.

Chow, C. 1970. On optimum recognition error and reject tradeoff. *IEEE Transactions on information theory*, 16(1): 41–46.

Chow, C. K. 1957. An optimum character recognition system using decision functions. *IRE Transactions on Electronic Computers*, EC-6(4): 247–254.

Dong, L.; Quirk, C.; and Lapata, M. 2018. Confidence Modeling for Neural Semantic Parsing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 743–753. Association for Computational Linguistics.

El-Yaniv, R.; and Wiener, Y. 2010. On the Foundations of Noise-free Selective Classification. *Journal of Machine Learning Research*, 11(53): 1605–1641.

Fadeeva, E.; Vashurin, R.; Tsvigun, A.; Vazhentsev, A.; Petrakov, S.; Fedyanin, K.; Vasilev, D.; Goncharova, E.; Panchenko, A.; Panov, M.; Baldwin, T.; and Shelmanov, A. 2023. LM-Polygraph: Uncertainty Estimation for Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 446–461. Association for Computational Linguistics.

Finegan-Dollak, C.; Kummerfeld, J. K.; Zhang, L.; Ramanathan, K.; Sadasivam, S.; Zhang, R.; and Radev, D. 2018. Improving Text-to-SQL Evaluation Methodology. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 351–360. Association for Computational Linguistics.

Gan, Y.; Chen, X.; and Purver, M. 2021. Exploring Underexplored Limitations of Cross-Domain Text-to-SQL Generalization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 8926–8931.

Gao, D.; Wang, H.; Li, Y.; Sun, X.; Qian, Y.; Ding, B.; and Zhou, J. 2024. Text-to-SQL Empowered by Large Language Models: A Benchmark Evaluation. *Proceedings of the VLDB Endowment*, 17(5): 1132–1145.

Guo, C.; Pleiss, G.; Sun, Y.; and Weinberger, K. Q. 2017. On calibration of modern neural networks. In *International conference on machine learning*, 1321–1330. PMLR.

Hendrycks, D.; and Gimpel, K. 2017. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. *Proceedings of International Conference on Learning Representations*.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.

Hupkes, D.; Dankers, V.; Mul, M.; and Bruni, E. 2020. Compositionality Decomposed: How do Neural Networks Generalise? (Extended Abstract). In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, 5065–5069. International Joint Conferences on Artificial Intelligence Organization. Journal track.

Hupkes, D.; Giulianelli, M.; Dankers, V.; Artetxe, M.; Elazar, Y.; Pimentel, T.; Christodoulopoulos, C.; Lasri, K.; Saphra, N.; Sinclair, A.; Ulmer, D.; Schottmann, F.; Batsuren, K.; Sun, K.; Sinha, K.; Khalatbari, L.; Ryskina, M.; Frieske, R.; Cotterell, R.; and Jin, Z. 2023. State-of-the-art generalisation research in NLP: A taxonomy and review. arXiv:2210.03050.

Kadavath, S.; Conerly, T.; Askell, A.; Henighan, T.; Drain, D.; Perez, E.; Schiefer, N.; Hatfield-Dodds, Z.; DasSarma, N.; Tran-Johnson, E.; Johnston, S.; El-Showk, S.; Jones, A.; Elhage, N.; Hume, T.; Chen, A.; Bai, Y.; Bowman, S.; Fort, S.; Ganguli, D.; Hernandez, D.; Jacobson, J.; Kernion, J.; Kravec, S.; Lovitt, L.; Ndousse, K.; Olsson, C.; Ringer, S.; Amodei, D.; Brown, T.; Clark, J.; Joseph, N.; Mann, B.; McCandlish, S.; Olah, C.; and Kaplan, J. 2022. Language Models (Mostly) Know What They Know. arXiv:2207.05221.

Kim, S.; Han, D.; and Kim, S. 2024. ProbGate at EHRSQL 2024: Enhancing SQL Query Generation Accuracy through Probabilistic Threshold Filtering and Error Handling. In *Proceedings of the 6th Clinical Natural Language Processing Workshop*, 687–696. Association for Computational Linguistics.

Lakshminarayanan, B.; Pritzel, A.; and Blundell, C. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30.

Lee, C.-H.; Polozov, O.; and Richardson, M. 2021. KaggleDBQA: Realistic Evaluation of Text-to-SQL Parsers. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2261–2273.

Lee, G.; Chay, W.; Cho, S.; and Choi, E. 2024a. TrustSQL: Benchmarking Text-to-SQL Reliability with Penalty-Based Scoring. arXiv:2403.15879.

Lee, G.; Hwang, H.; Bae, S.; Kwon, Y.; Shin, W.; Yang, S.; Seo, M.; Kim, J.-Y.; and Choi, E. 2022. Ehrsql: A practical text-to-sql benchmark for electronic health records. *Advances in Neural Information Processing Systems*, 35: 15589–15601.

Lee, G.; Kweon, S.; Bae, S.; and Choi, E. 2024b. Overview of the EHRSQL 2024 Shared Task on Reliable Text-to-SQL Modeling on Electronic Health Records. In *Proceedings of the 6th Clinical Natural Language Processing Workshop*, 644–654. Association for Computational Linguistics.

Li, J.; Hui, B.; Qu, G.; Yang, J.; Li, B.; Li, B.; Wang, B.; Qin, B.; Geng, R.; Huo, N.; et al. 2024. Can llm already serve as a database interface? a big bench for large-scale database grounded text-to-sqls. *Advances in Neural Information Processing Systems*, 36.

Malinin, A.; and Gales, M. 2021. Uncertainty Estimation in Autoregressive Structured Prediction. arXiv:2002.07650.

Meta, A. 2024. Introducing meta llama 3: The most capable openly available llm to date. *Meta AI*.

Platt, J.; et al. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3): 61–74.

Qin, B.; Wang, L.; Hui, B.; Li, B.; Wei, X.; Li, B.; Huang, F.; Si, L.; Yang, M.; and Li, Y. 2022. SUN: Exploring Intrinsic Uncertainties in Text-to-SQL Parsers. arXiv:2209.06442.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21(140): 1–67.

Ren, J.; Luo, J.; Zhao, Y.; Krishna, K.; Saleh, M.; Lakshminarayanan, B.; and Liu, P. J. 2023. Out-of-Distribution Detection and Selective Generation for Conditional Language Models. arXiv:2209.15558.

Shaw, P.; Chang, M.-W.; Pasupat, P.; and Toutanova, K. 2021. Compositional Generalization and Natural Language Variation: Can a Semantic Parsing Approach Handle Both? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 922–938. Association for Computational Linguistics.

Somov, O.; and Tutubalina, E. 2023. Shifted PAUQ: Distribution shift in text-to-SQL. In *Proceedings of the 1st GenBench Workshop on (Benchmarking) Generalisation in NLP*, 214–220. Association for Computational Linguistics.

Stengel-Eskin, E.; and Van Durme, B. 2023. Calibrated Interpretation: Confidence Estimation in Semantic Parsing. *Transactions of the Association for Computational Linguistics*, 11: 1213–1231.

Suhr, A.; Chang, M.-W.; Shaw, P.; and Lee, K. 2020. Exploring Unexplored Generalization Challenges for Cross-Database Semantic Parsing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8372–8388. Online: Association for Computational Linguistics.

Ueffing, N.; and Ney, H. 2005. Word-Level Confidence Estimation for Machine Translation using Phrase-Based Translation Models. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, 763–770. Association for Computational Linguistics.

Ueffing, N.; and Ney, H. 2007. Word-level confidence estimation for machine translation. *Computational Linguistics*, 33(1): 9–40.

van der Poel, L.; Cotterell, R.; and Meister, C. 2022. Mutual Information Alleviates Hallucinations in Abstractive Summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 5956–5965. Association for Computational Linguistics.

Vazhentsev, A.; Tsvigun, A.; Vashurin, R.; Petrakov, S.; Vasilev, D.; Panov, M.; Panchenko, A.; and Shelmanov, A. 2023. Efficient Out-of-Domain Detection for Sequence to Sequence Models. In *Findings of the Association for Computational Linguistics: ACL 2023*, 1430–1454. Association for Computational Linguistics.

Wretblad, N.; Riseby, F. G.; Biswas, R.; Ahmadi, A.; and Holmström, O. 2024. Understanding the Effects of Noise in Text-to-SQL: An Examination of the BIRD-Bench Benchmark. *arXiv preprint arXiv:2402.12243*.

Yang, Y.; Kim, S.; Kim, S.; Lee, G.; Yun, S.-Y.; and Choi, E. 2024. Towards Unbiased Evaluation of Detecting Unanswerable Questions in EHRSQL. *arXiv preprint arXiv:2405.01588*.

Yu, T.; Zhang, R.; Yang, K.; Yasunaga, M.; Wang, D.; Li, Z.; Ma, J.; Li, I.; Yao, Q.; Roman, S.; et al. 2018. Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 3911–3921.

Zadrozny, B.; and Elkan, C. 2002. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 694–699.

Zelle, J. M.; and Mooney, R. J. 1996. Learning to parse database queries using inductive logic programming. In *Proceedings of the national conference on artificial intelligence*, 1050–1055.

Zettlemoyer, L. S.; and Collins, M. 2005. Learning to map sentences to logical form: structured classification with probabilistic categorial grammars. In *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*, UAI'05, 658–666. AUAI Press. ISBN 0974903914.