Uncertainty Quantification in Table Structure Recognition

Kehinde Ajayi *, Leizhen Zhang *, Yi He, Jian Wu
Department of Computer Science
Old Dominion University
{kajay001, lzhan011} @odu.edu, {yihe, jwu}@cs.odu.edu

***Abstract*—Quantifying uncertainties for machine learning models is a critical step to reduce human verification effort by detecting predictions with low confidence. This paper proposes a method for uncertainty quantification (UQ) of table structure recognition (TSR). The proposed UQ method is built upon a mixture-of-expert approach termed Test-Time Augmentation (TTA). Our key idea is to enrich and diversify the table representations, to spotlight the cells with high recognition uncertainties. To evaluate the effectiveness, we proposed two heuristics to differentiate highly uncertain cells from normal cells, namely, masking and cell complexity quantification. Masking involves varying the pixel intensity to deem the detection uncertainty. Cell complexity quantification gauges the uncertainty of each cell by its topological relation with neighboring cells. The evaluation results based on standard benchmark datasets demonstrate that the proposed method is effective in quantifying uncertainty in TSR models. To our best knowledge, this study is the first of its kind to enable UQ in TSR tasks. Our code and data are available at: https://github.com/lamps-lab/UQTTA.git.**

I. Introduction

Table recognition has been studied in recent years to facilitate document understanding and retrieval tasks [1]. This task can be decomposed into two subtasks: table detection (TD) and table structure recognition (TSR). TD aims to automatically identify tables present in digital documents. TSR aims to identify the rows, columns, and individual text cells in table images. Early works use classical machine learning models such as conditional random fields [2] and hidden markov models [3]. Recently, deep learning approaches e.g., [4]–[6] have been proposed. The output files for TSR models typically contain the coordinates of the identified cells in terms of row and column numbers and coordinates of bounding boxes that enclose the cells, so the cell content can be recognized by subsequent optical character recognition (OCR) software.

The current TSR models can automatically identify cell locations, but the results do not predict uncertainties [6], [7]. This prevents TSR models to be applied in real-world scenarios, such as faithfully extracting domain tabular data for downstream analysis in scientific domains, e.g., materials science. It is costly and sometimes infeasible for domain experts to verify all data extracted by machine learning models. Therefore, automatically quantifying TSR uncertainties is crucial to minimize human effort for data verification.

* Equal contribution

UQ methods have been proposed for deep learning-based solutions of several natural language processing [8] and computer vision tasks [9], but to our best knowledge, it has not been incorporated into TSR. A recent work [4] attempted to incorporate confidence estimation into the cell structures of the tables detected in document images but the confidence scores were represented as binaries indicating whether a cell was detected or not. Our work will abridge the gap by quantifying uncertainties for TSR models as continuous values.

Uncertainties in a machine learning model can arise from two major sources, namely, aleatoric uncertainty (also known as data uncertainty) and epistemic uncertainty (also known as model uncertainty) [10]. Aleatoric uncertainty occurs as a result of measurement noise, data missingness, or outliers [11], while epistemic uncertainty occurs due to the choice of model architecture, hyperparameters, and initialization [12]. Several methods have been proposed to quantify the uncertainties in deep learning-based models, such as Bayesian methods [13], Monte Carlo (MC) dropout [12], and Ensembles [14]. Bayesian neural networks use prior distributions to represent prior beliefs about the parameters of the neural networks, which are updated based on the data during training [13]. Once the model is trained, posterior distributions can be used to estimate uncertainty. MC dropout is based on dropout regularization [12]. Specifically, during prediction, the dropout is applied multiple times to the network, and the variance of the predictions can be used as a measure of uncertainty. An ensemble method involves training multiple models with different architectures and combining their predictions [14]. The variance of the ensemble predictions can be used as a measure of uncertainty. Although Bayesian or MC dropout methods are easier to interpret, they are hard to scale up because they require multiple forward passes through the network for each prediction and require modifications to the neural network architecture to incorporate uncertainty [12]. Ensemble methods are more scalable, robust, and flexible because they are agnostic to neural network architecture [14].

Our proposed UQ pipeline adopts the Test-Time Augmentation (TTA), a technique that involves applying data augmentation to samples during inference (or testing) time and then ensembling the predictions [15].

One key component of the vanilla TTA is data augmentation, which is usually task-dependent. The augmentation methods should be controllable and orthogonal. For TSR, we

explore four heuristic methods to augment test table images, which were chosen to probe the differential performance of the pre-trained model using variations of the test data. We combined the original and augmented images to obtain ensemble results, at different confidence levels at a certain Intersection over Union (IoU) threshold, computed by dividing the area of intersection between the predicted bounding boxes (bboxes) and the ground truth bboxes by the area of the union of the two bboxes. If the predicted IoU is greater than the threshold, the two bboxes are deemed to match.

One challenge in evaluating UQ methods is the lack of human-labeled ground truth. Therefore, we proposed two heuristics (1) masking, which involves varying the pixel intensities of table images and then quantifying uncertainties using confidence level estimation, and (2) cell complexity quantification, which models the complexity of cell relations using undirected subgraphs and uses the complexity of subgraphs as a surrogate for recognition uncertainty.

To showcase the efficacy of our UQ model, we apply it to CascadeTabNet [4], a recently proposed TSR model, which was retrainable. However, our UQ framework can be integrated into other retrainable TSR models. The contributions of this paper are below:

1) We proposed a novel ensemble method called `TTA-m` to quantify uncertainties for the results of TSR tasks and showcased its efficacy on a reproducible TSR framework called CascadeTabNet.
2) We proposed two controllable and scalable methods, masking and cell complexity quantification, to build the ground truth uncertainties.
3) We created a new dataset containing table images based on the ICDAR-19 document TSR competition. The new dataset augmented the original data using four heuristics and the ground truth uncertainties based on the two methods above.

II. Related Work

A. Uncertainty Quantification

UQ has been an area of interest in both traditional machine learning and deep learning [10]. Several methods have been proposed to quantify uncertainties in deep learning models, such as Bayesian models, Monte-Carlo Dropout, and Ensembles [12]. One ensemble method is Bayesian model averaging [14], which quantifies uncertainties by averaging predictions from multiple deep learning models training with augmented data. TTA emerged as a straightforward method to enhance ensemble models [16]. TTA is easier to implement and more computationally efficient.

B. Table Structure Recognition

TSR has experienced significant strides recently due to the utilization of deep neural networks. For example, Schreiber et al. [17] devised an innovative strategy amalgamating Faster R-CNN and Fully-Convolutional Network architectures. This convergence facilitated proficient table detection and precise cell position localization. Siddiqui et al. [18] treated table images as comprehensive scenes using deformable convolution operations. This holistic approach offers a fertile ground for imbuing UQ into the fabric of scene-based representation. The work of Xue et al. [7] ventured into the realm of graph-based inference to unravel table syntactic structures using a cell relationship network. Khan et al. [19] harnessed bi-directional Gated Recurrent Units to discern intricate row and column boundaries in tables. The split and merge model proposed by [6] introduces a strategy for addressing TSR through hierarchical decomposition. Lee et al. [5] framed TSR as a challenge of table graph reconstruction. Hashmi et al. [20] implemented Mask R-CNN for anchor estimation in TSR.

However, most existing models are not able to quantify uncertainties of their predictions.

III. TTA-m: Proposed UQ Pipeline

Figure 1 illustrates the architecture of the proposed UQ pipeline. The key modules include (1) Training with data augmentation, (2) Fine-tuning a pre-trained TSR model (using CascadeTabNet as a case study), (3) Inference with fine-tuned models, and (4) Uncertainty estimation. Because the proposed model modified the traditional TTA model, we call it `TTA-m`. For convenience, we define M as the number of augmentation methods applied to the original data.

A. Data Augmentation

Data augmentation has become a practice for developing robust and transformation-resistant models [21]. We applied a combination of $M = 4$ distinct data augmentation methods across the training and test stages. These methods encompass the removal of all lines (NLT), the addition of horizontal lines (HLT), the inclusion of vertical lines (VLT), and the incorporation of both horizontal and vertical lines (HLT + VLT). Figure 2 shows augmented table image examples.

B. Fine-tuning A Pre-trained TSR Model

Instead of training the model on all augmented data, we fine-tuned a pre-trained TSR model on each set of augmented table images plus the original table images, resulting in $M+1$ distinct models.

C. Predictions With Fine-tuned Model

In the inference stage, we first applied the same augmentations to the *test* set. Then, instead of evaluating the pre-trained model on the $M+1$ sets of table images, each fine-tuned model was applied to its corresponding test data set. For example, the model fine-tuned on tables with only vertical lines was applied on tables with vertical lines in the test set. Each fine-tuned model can be thus evaluated on standard binary classification metrics (precision, recall, and F1-score). These predictions are intermediate results. The final output is generated by the ensemble module.

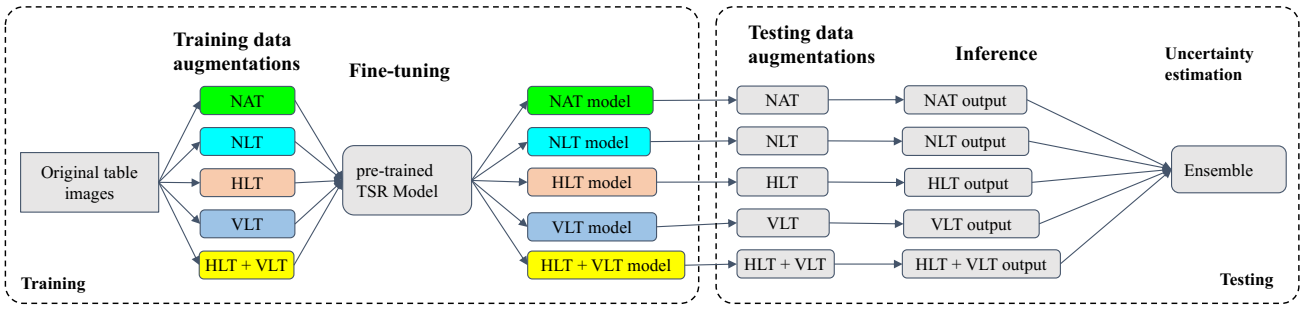

Figure 1. A schematic illustration of the proposed UQ pipeline (TTA–m). In the training phase, we fine-tuned the TSR model on the original tables and augmented tables. In the test phase, each model makes a prediction on table images similar to what it was trained on and then ensembling is applied on the model outputs. NAT: Non-Augmented Tables, NLT: No Lines Tables, HLT: Horizontal Lines Tables, VLT: Vertical Lines Tables.

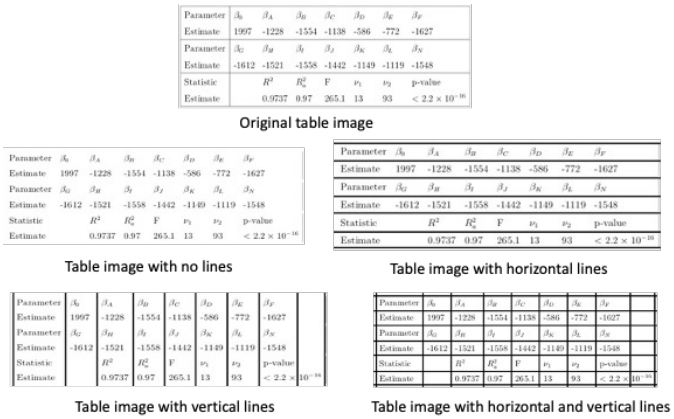

Figure 2. Augmentation examples of a table image.

D. Uncertainty Estimation via Ensembles

In this module, we use an ensemble method to aggregate the predictions by fine-tuned models on augmented testing data. Our method is different from the traditional TTA because the training data is also augmented and the TSR model is fine-tuned before it is applied to the corresponding augmented test data. The uncertainty is modeled as the dispersion of the predicted results. The uncertainty estimation process involves progressively combining predictions from $M+1$ models based on the degree of overlap between cell predictions. This aggregation results in a set of merged cells with associated confidence scores, which collectively constitute the output. The steps are detailed below. Here, we use θ_0 to represent the IoU threshold as the criteria to match the bounding boxes of two predicted cells.

1) Obtain all the predicted bounding boxes from a randomly chosen model out of the $M+1$ models (considered as the base model).
2) Obtain predicted bounding boxes from the second model.
3) Calculate the IoU for each predicted cell from the base model against each from the second model. If the IoU $\geq \theta_0$, merge these two cells and remove the second model's cell from its list.
4) Repeat steps 2 and 3 for predictions from the remaining models $i = 3, 4, ..., M+1$.
5) Sequentially use $i = 2, 3, 4, \cdots, M+1$ models as new base models and perform calculations similar to Steps 1 to 4 for the cells that have not been merged.
6) For all cell combinations generated in the above steps:
   a) Count the number of distinct models contributing to each cell combination.
   b) Divide the count by $M+1$ to calculate the confidence score for that combination (Figure 3).

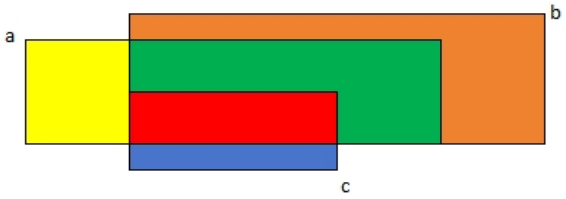

Figure 3. A schematic illustration of how to calculate confidence scores using bounding boxes predicted by three models (a, b, and c). Red color: 3/3 = 100% confidence, Green color: 2/3 = 66.7% confidence, Other colors: 1/3 = 33.3% confidence.

IV. Evaluation

Because the ground truth uncertainties are not available for the annotated cells, we proposed two methods to gauge uncertainties in different scenarios and use them as surrogates to evaluate the UQ pipeline.

a) Masking: The masking technique artificially changes the level of difficulty by varying the intensity of pixels on table images. Firstly, we increased the pixel intensity by a factor of 2 for each cell which makes the pixels appear fainter, and calculated the confidence estimates at the previously defined confidence scores. Next, we multiplied all pixel values by 3. If the pixel value becomes greater than 255, we set it to 255. Then, we estimated the confidence scores of the TSR models at each intensity level. Our results indicate that the intensity of cell pixel values remarkably affects the distribution of confidence scores.

b) Cell complexity quantification.: We observed that TSR models are more likely to make mistakes for tables with complex structures. Specifically, table images may contain cells that span across multiple rows and/or columns, which are challenging for TSR models in general. To quantify the structure complexity, we model a table as a non-directed graph in which the nodes represent table cells, and the edges represent adjacency between cells. We considered four types of adjacency relations: left, top, right, and bottom. We define adjacency degree as the number of adjacency cells of the target cell, where a cell represents a unit within a table that contains meaningful text contents. Intuitively, the higher the degree of a cell is, the more likely the bounding box is incorrectly predicted. Therefore, our evaluation will test whether the fraction of cells detected with low confidence increases with the average degree of a table. Figure 4 illustrates examples of relationships that could exist between the cells of a table image. For evaluation, we manually annotated the relations between cells and constructed a graph for each table in the test set.

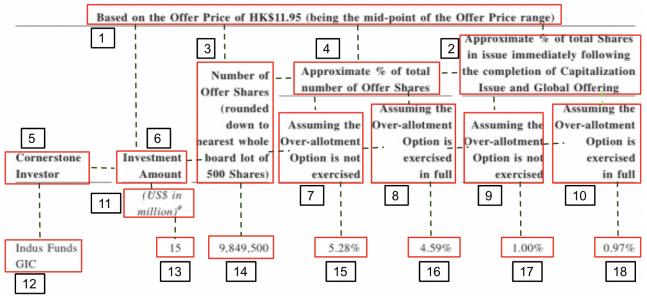

Figure 4. An example of the graph model of a table. Each cell is enclosed by a red box, with an ID labeled next to it. The dashed lines represent the connections of a cell to its adjacency cells and can be used for counting the adjacency degrees of a cell. For instance, cell 5 is connected by 2 green lines, so it has an adjancency degree of 2.

V. Experimental Setup

A. Data

We used the dataset created for the competition at the International Conference on Document Analysis and Recognition (ICDAR) 2019 containing real-world table images. For an even comparison, We used the dataset adopted by Prasad et al. [4] consisting of 543 table images selected from the benchmark dataset created for the competition at the International Conference on Document Analysis and Recognition (ICDAR-19). We randomly selected 443 table images for training and the remaining 100 table images for testing. The ICDAR 2019 dataset was originally used for both TD and TSR. We only used the ground truth labels for TSR.

B. Baseline Methods

We compare `TTA-m` with three baseline methods, including two variants of TTA and an active learning model.

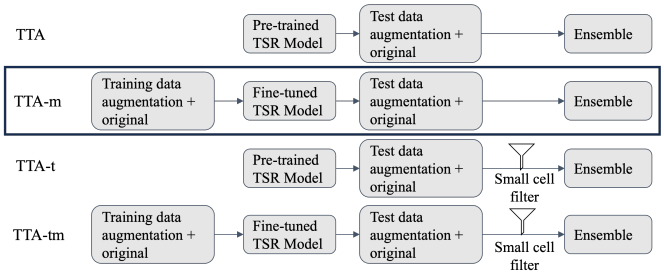

Figure 5. A schematic comparison of TTA variants implemented by this paper. TTA-m is proposed for its highest F1 over the others (Table I).

a) TTA-t: TTA-t adds a small cell filter to the vanilla TTA to exclude small cells predicted by fine-tuned models. In observation, these small cells are usually produced by fine-tuned models using augmented data, and the large cells are produced by the model fine-tuned using the original data. These small cells occupy areas much smaller than the actual cell and thus do not contribute to confidence scores. Therefore, we crafted simple heuristics to remove them before ensembling the predicted results. The small cell filter is applied if its area (S) meets the following conditions. (1) the smaller cell is fully inside a bigger cell, $(S_{\text{small}} \cap S_{\text{large}})/S_{\text{small}} = 1$; (2) the smaller cell is significantly smaller than the bigger cell, or $S_{\text{small}}/S_{\text{large}} \leq 0.5$.

b) TTA-tm: TTA-tm combines the `TTA-t` and `TTA-m` model, which includes both training data augmentation and the small cell filter (Figure 5).

c) Active Learning: We also compare our method against an active learning model proposed by Choi et al. [22]. This method aims to reduce labeling costs by selecting only the most informative samples in a dataset. It uses a mixture density network that estimates a probabilistic distribution for each localization and classification head's output to explicitly estimate the aleatoric and epistemic uncertainty in a single forward pass of a single model. This method uses a scoring function that aggregates these uncertainties for both heads to obtain every image's informativeness score. We fine-tuned the baseline model on the table images in our training set and tested it on the original table images.

C. Experiment Design

We conduct three experiments to evaluate the proposed model. Our goal is to demonstrate that the ensemble results generated by the proposed model can accurately detect cells and that the confidence levels can be reliably used as a measure of uncertainty. To showcase the efficacy, we adopt CascadeTabNet [4] as the TSR model. All experiments are run on a server with 24 Intel Xeon cores, 384GB RAM, and 4x Nvidia 2080 Ti GPUs. The fine-tuning was run 1 time for either the training or the test set.

a) Experiment 1: Cell Detection.: In the first experiment, we compare model performances on cell recognition. The results in Table I show that the `TTA-m` model outperforms

TABLE I: Comparing the models used in our study. The models compared include the original CascadeTabNet, baseline, and the fine-tuned CascadeTabNet on the four augmentation types and original table images.

Model	Augmentation Method	Precision	Recall	F1
TTA	Vertical lines	0.765	0.713	0.738
	Horizontal lines	0.792	0.707	0.749
	Both lines	0.725	0.677	0.701
	No lines	0.846	0.742	0.793
	Original	0.883	0.767	0.823
TTA ensemble result		0.683	0.824	0.753
TTA-m	Vertical lines	0.854	0.755	0.802
	Horizontal lines	0.841	0.758	0.798
	Both lines	0.844	0.744	0.791
	No lines	0.838	0.738	0.785
	Original	0.883	0.771	0.823
TTA-m ensemble result		0.761	0.835	**0.798**
TTA-tm ensemble result		0.778	0.831	0.806
Baseline	Original	0.899	0.659	0.76

all the baseline models in terms of the F1-scores. Note that we should not compare against F1-scores of models based on individual augmentation methods because they are not the final output of the pipeline. Notably, the employment of the ensemble technique resulted in a reduction in precision but an improvement in recall. However, the poor performance of the fine-tuned model on the augmented table images implies its low generalization capability to such augmented testing data.

b) Experiment 2: Confidence Level as a Measure of Uncertainty.: To demonstrate the confidence level output by the proposed model can be used as a measure of uncertainty, we calculated the percentage of correctly predicted cells at 0.2, 0.4, 0.6, 0.8, and 1.0 confidence levels and an IoU threshold $\theta_0 = 0.5$. The confidence levels were obtained based on the overlap area of the 5 predicted bounding boxes. For example, if only a cell is overlapped by two predicted bounding boxes, the confidence level is 2/5 = 0.4 (as shown in Section III-D). Figure 6 shows that the percentages of the correctly predicted cells increase monotonously with the confidence levels for the `TTA` and `TTA-m` models. However, this trend is not seen for the results produced by the baseline method. Figure 6 also indicates that the `TTA-m` method correctly predicted over 80% of cells with a confidence of 1.0.

c) Experiment 3: Confidence Level as a Measure of Uncertainty Gauged by Pixel Intensity.: The purpose of this experiment is to evaluate proposed models when table image pixel intensity varies. Figure 7 indicates that the whole curve of the fraction of correctly predicted cells is shifted down as the pixel value increases (so the pixels look fainter). The only exception is when the confidence level is 0.8, but the difference is subtle. Intuitively decreasing pixel intensity (increasing pixel values) should increase the difficulty of accurate detection, leading to higher levels of uncertainty. The results in Figure 7

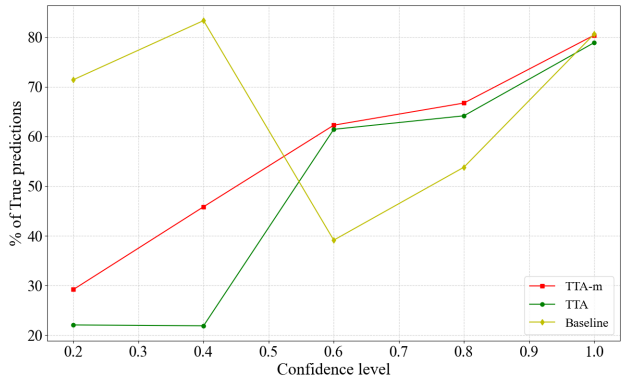

Figure 6. The percentage of true predictions for the TTA (green) and TTA-m (red) and baseline (light-green). TTA-m outperforms both the TTA and baseline models for confidence levels above 0.6.

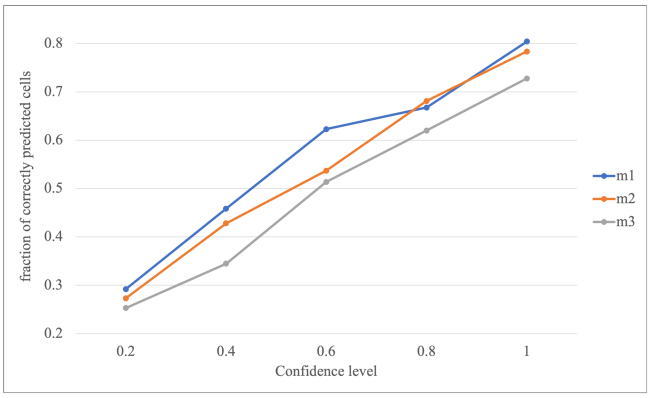

Figure 7. Evaluations of the reliability of the confidence scores as a measure of the uncertainty. m1: no masking applied; m2: pixel values doubled; m3: pixel values tripled.

indicate that this trend is captured by the confidence levels output by the `TTA-m` model.

d) Experiment 4: Confidence Level as a Measure of Uncertainty Gauged by Cell Complexity.: In this experiment, we quantify cell complexity by the adjacency degrees, which is used as a gauge of uncertainty here assuming that cells with more complex structures (and thus adjacency degrees) are likely to be detected with higher uncertainty. Table II shows that approximately 85% of the cells in our test data have between 3 to 4 degrees of relationships with neighboring cells. In general, the confidence level decreases as the degree of relationships between cells increases from 1 to 6 with the exception of degree 5.

VI. Conclusion and Discussion

This study explores UQ in TSR problems by modifying the traditional TTA technique and testing it on a customized CascadeTabNet model [4]. To evaluate the effectiveness of our UQ method, we used masking and cell complexity quantification

TABLE II: Quantifying cell complexity based on the adjacency degree of table cells. The mean confidence level was obtained by taking the average of the confidence scores obtained for all cells for each degree.

Degree	#Cells	Cells%	Mean Confidence Level
1	27	0.5	0.95
2	409	7.61	0.84
3	1878	34.93	0.74
4	2937	54.62	0.71
5	115	2.14	0.77
6	8	0.15	0.65

techniques. These techniques involve adjusting cell pixel intensity and determining cell complexity based on relationships among cells in table images at different confidence levels. The proposed method demonstrated better Experiments indicating the proposed UQ method provides a more reliable uncertainty estimation.

Compared with the vanilla `TTA`, `TTA-m` extends the data augmentation to the training phase, which increases the cost of time to obtain uncertainties of the TSR model. When inferencing the pipeline on a dataset without ground truth labels, one can simply adopt pre-fine-tuned models, so only the test data augmentation is needed.

Our approach to quantifying uncertainty takes into account both data variation and model variation, unlike the vanilla `TTA` method that only considers data variation. This is achieved through fine-tuning the target TSR model using our UQ method, which is not limited to any particular TSR model. Additionally, the data augmentation techniques utilized in our study ensure that the TSR model is invariant to different types of tables.

Our study has the following limitations. First, the lack of ground truth limits the capability of assessing real uncertainties. Although we used the pixel intensity and adjacency degree as proxy gauges of detection uncertainties, the real-world data can be hybrid. Such ground truth data could be built by collecting human corrections of automatic annotations by TSR models. Second, the ways we augmented the table images may not be comprehensive. Specifically, the augmentation techniques we explored might not encompass all possibilities or capture the extensive array of variations present in table images. This limitation can be mitigated by building a library of heuristics to modify table images or by building a corpus of artificially synthesized tables.

ACKNOWLEDGEMENT

This work has been supported in part by the National Science Foundation (NSF) under Grant Nos. IIS-2245946 and IIS-2236578, in part by the Commonwealth Cyber Initiative (CCI), and in part by the Research Institute of Digital Innovation in Learning (RIDIL).

REFERENCES

[1] K. A. Hashmi, K. A. Hashmi, M. Liwicki, M. Liwicki, D. Stricker, D. Stricker, M. A. Afzal, M. A. Afzal, M. Afzal, M. Z. Afzal, and M. Z. Afzal, "Current status and performance analysis of table recognition in document images with deep neural networks," *IEEE Access*, 2021.

[2] X. Wei, B. Croft, and A. McCallum, "Table extraction for answer retrieval," *Information retrieval*, vol. 9, no. 5, pp. 589–611, 2006.

[3] F. F. Babatunde, B. A. Ojokoh, S. A. Oluwadare *et al.*, "Automatic table recognition and extraction from heterogeneous documents," *Journal of Computer and Communications*, vol. 3, no. 12, p. 100, 2015.

[4] D. Prasad, A. Gadpal, K. Kapadni, M. Visave, and K. Sultanpure, "Cascadetabnet: An approach for end to end table detection and structure recognition from image-based documents," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, pp. 572–573.

[5] E. Lee, J. Park, H. I. Koo, and N. I. Cho, "Deep-learning and graph-based approach to table structure recognition," *Multimedia Tools and Applications*, vol. 81, no. 4, pp. 5827–5848, 2022.

[6] C. Tensmeyer, V. I. Morariu, B. Price, S. Cohen, and T. Martinez, "Deep splitting and merging for table structure decomposition," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*, 2019, pp. 114–121.

[7] W. Xue, Q. Li, and D. Tao, "Res2tim: Reconstruct syntactic structures from table images," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*, 2019, pp. 749–755.

[8] Y. Xiao and W. Y. Wang, "Quantifying uncertainties in natural language processing tasks," *AAAI Conference on Artificial Intelligence*, 2019.

[9] Y. Shen, Y. Shen, Z. Zhang, M. R. Sabuncu, M. R. Sabuncu, M. R. Sabuncu, and L. Sun, "Real-time uncertainty estimation in computer vision via uncertainty-aware distribution distillation," *arXiv: Computer Vision and Pattern Recognition*, 2020.

[10] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya *et al.*, "A review of uncertainty quantification in deep learning: Techniques, applications and challenges," *Information Fusion*, vol. 76, pp. 243–297, 2021.

[11] J. Chang, Z. Lan, C. Cheng, C. Cheng, C. Cheng, Y. Wei, and Y. Wei, "Data uncertainty learning in face recognition," *arXiv: Computer Vision and Pattern Recognition*, 2020.

[12] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*, 2016, pp. 1050–1059.

[13] V. Mullachery, A. Khera, and A. Husain, "Bayesian neural networks," *arXiv preprint arXiv:1801.07710*, 2018.

[14] R. Rahaman and a. thiery, "Uncertainty quantification and deep ensembles," in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 20 063–20 075.

[15] G. Wang, W. Li, M. Aertsen, J. Deprest, S. Ourselin, and T. Vercauteren, "Test-time augmentation with uncertainty estimation for deep learning-based medical image segmentation," 2018.

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

[17] S. Schreiber, S. Agne, I. Wolf, A. Dengel, and S. Ahmed, "Deepdesrt: Deep learning for detection and structure recognition of tables in document images," in *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, vol. 1, 2017, pp. 1162–1167.

[18] S. A. Siddiqui, I. A. Fateh, S. T. R. Rizvi, A. Dengel, and S. Ahmed, "Deeptabstr: deep learning based table structure recognition," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*, 2019, pp. 1403–1409.

[19] S. A. Khan, S. M. D. Khalid, M. A. Shahzad, and F. Shafait, "Table structure extraction with bi-directional gated recurrent unit networks," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*, 2019, pp. 1366–1371.

[20] K. A. Hashmi, D. Stricker, M. Liwicki, M. N. Afzal, and M. Z. Afzal, "Guided table structure recognition through anchor optimization," *IEEE Access*, vol. 9, pp. 113 521–113 534, 2021.

[21] A. Mikołajczyk and M. Grochowski, "Data augmentation for improving deep learning in image classification problem," in *2018 international interdisciplinary PhD workshop (IIPhDW)*, 2018, pp. 117–122.

[22] J. Choi, I. Elezi, H.-J. Lee, C. Farabet, and J. M. Alvarez, "Active learning for deep object detection via probabilistic modeling," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 264–10 273.