An End-to-End Multi-Task Learning Model for Image-based Table Recognition

Nam Tuan Ly and Atsuhiro Takasu
National Institute of Informatics (NII), Tokyo, Japan

Challenges

Image-based table recognition is a challenging task due to the diversity of table styles and the complexity of table structures.
Most of the previous methods focus on a non-end-to-end approach which divides the problem into two separate sub-problems: table structure recognition; and cell-content recognition and then attempts to solve each sub-problem independently using two separate systems.

Proposed Approach

The proposed model consists of one shared encoder, one shared decoder, and three separate decoders which are used for learning three subtasks of table recognition: table structure recognition, cell detection, and cell-content recognition.

The whole system can be easily trained and inferred in an end-to-end approach.

Input-Output of the Approach

The model takes an input table image and produces the table structure information, location of table cells, and contents of table cells, which can be easily transformed into the HTML code (or LaTeX code) representing the table.

Overview of the Proposed Model

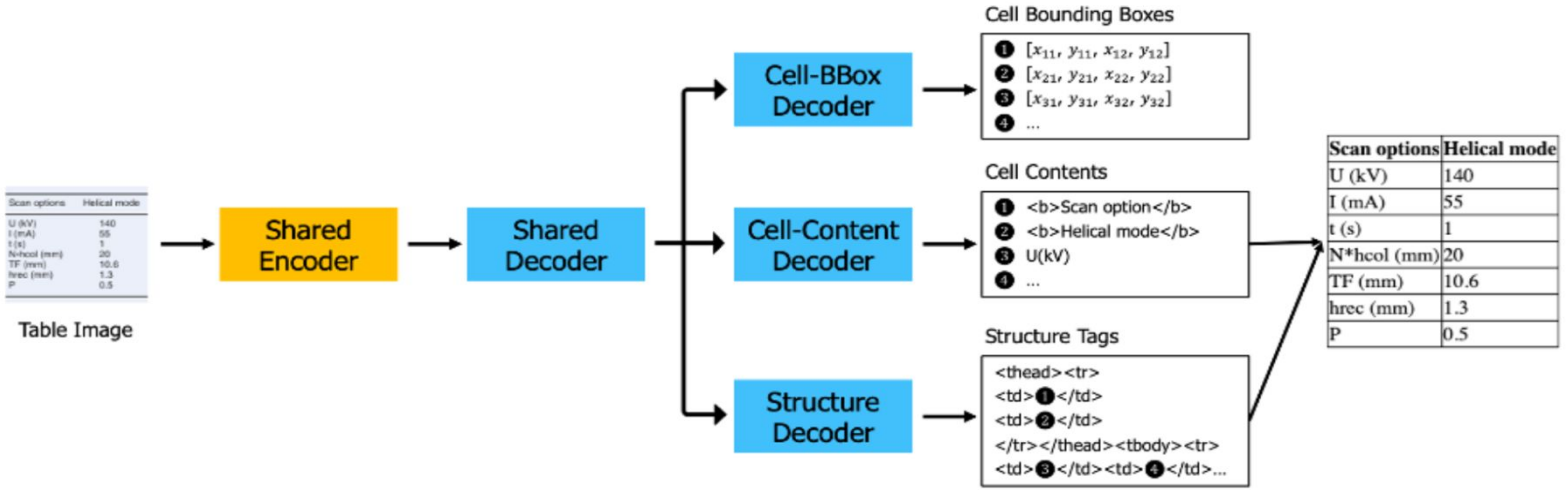

Figure 1. The overview of the proposed model.

Overview of the Approach

The proposed model consists of one shared encoder, one shared decoder, and three separate decoders for three subtasks of the table recognition problem.

1. The shared encoder encodes the input table image as a sequence of features.
2. The sequence of features is passed to the shared decoder and then the structure decoder to predict a sequence of HTML tags that represent the structure of the table.
3. When the structure decoder produces the HTML tag representing a new cell, the output of the shared decoder corresponding to that cell and the output of the shared encoder are passed into the cell-bbox decoder and the cell-content decoder to predict the bounding box coordinates and the text content of that cell.
4. Finally, the text contents of cells are inserted into the HTML structure tags corresponding to their cells to produce the final HTML code of the input table image.

Shared Encoder

A CNN backbone network as the feature extractor followed by a positional encoding layer to build the shared encoder. The feature extractor extracts visual features from an input table image before being fed into the positional encoding layer to get the encoded sequence of features. The encoded sequence of features will be fed into the shared decoder and three separate decoders.

Network Architecture

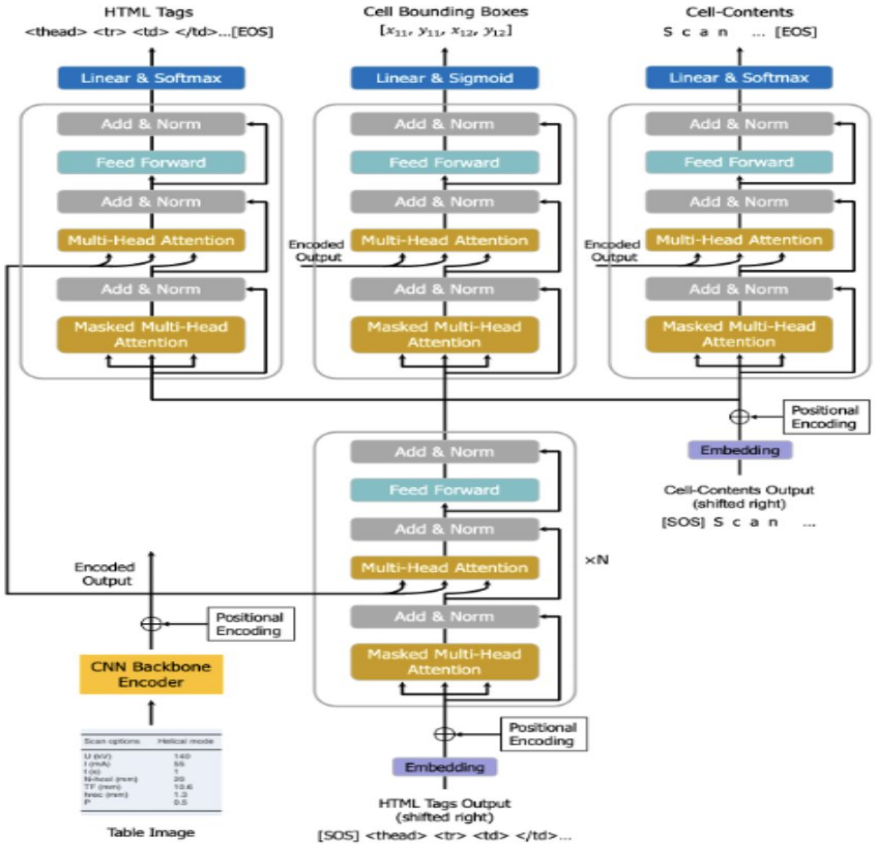

Figure 2. Network architecture of the proposed model.

Shared Decoder

- The architecture of all decoders is inspired by the original Transformer decoder (Vaswani et al., 2017) which is composed of a stack of *N* identical Transformer decoder layers where *N* can be a hyperparameter.

- Each identical Transformer decoder layer (identical layer in short) has three sub- layers: a multi-head self-attention mechanism; a masked multi-head self-attention mechanism; and a position-wise fully connected feed-forward network, and helps the decoders focus on appropriate places in the input table image.

Shared Decoder (Part-2)

- The output of the shared encoder is fed into the multi-head self-attention mechanism of each identical layer as the value and key vectors.

- During training, the right-shifted sequence of target HTML tags (structural tokens) of the table structure (after passing through the embedded layer and the positional encoding layer) is passed into the bottom of the shared decoder as the query vector.

- In the inference stage, the right-shifted sequence of target HTML tags is replaced by the right-shifted sequence of HTML tags outputted by the structure decoder.

- Finally, the outputs of the shared decoder will be fed into the three separate decoders to predict three sub-tasks of the table recognition problem.

Structure Decoder

- At the top of the shared decoder, the structure decoder uses the outputs of the shared decoder and the outputs of the shared encoder to predict a sequence of HTML tags of the table structure.
- The HTML tags of the table structure are tokenized at the HTML tag level except for the tag of a cell.

- The form of ‘<td></td>’ is treated as one token class and the tag of the spanning cells is broken down into ‘<td’, ‘rowspan=’ or ‘colspan=’, with the number of spanning cells, and ‘>’. Thus, the structural token of ‘<td></td>’ or ‘<td’ represents a new table cell.
- The structure decoder is composed of one identical layer followed by a linear layer and a softmax layer. The identical layer takes the outputs of the shared decoder as the query vector input and the outputs of the shared encoder as the key and value vector inputs.
- The output of the identical layer is fed into the linear layer, and then the softmax layer to generate the sequence of structural tokens.

Cell BBox Decoder

- When the structure decoder generates a structural token representing a new cell (‘<td></td>’ or ‘<td’), the cell-bbox decoder is triggered and uses the output of the shared decoder corresponding to this cell to predict the bounding box coordinates of this cell.

- As shown in Fig. 2, we use one identical layer followed by a linear layer and a sigmoid layer to build the cell-bbox decoder. The identical layer takes the output of the shared decoder and the output of the shared encoder as the input and learns to focus on appropriate places in the input image.

- The output of the identical layer is fed into the linear layer and then the sigmoid layer to predict the four coordinates of the cell bounding box.

Cell-Content Decoder (Part-1)

- Similar to the cell-bbox decoder, the cell-content decoder selects the outputs of the shared decoder referring to the structural tokens representing a new cell (‘<td></td>’ or ‘<td’) and uses them to recognize the text contents of cells.
- The cell-content decoder in the proposed model can be considered a text recognizer and the text output are tokenized at the character level.
- One identical layer followed by a linear layer and a softmax layer to build the cell-content decoder.

Cell-Content Decoder (Part-2)

- The output of the shared encoder are fed into the identical layer as the input value and key vectors.
- During training, the right-shifted target of the cell content (the right-shifted output of the cell-content decoder in the testing phase) is passed through the embedded and the positional encoding layers and then added to the output of the shared decoder before being fed into the identical layer as the query vector.
- Finally, the output of the identical layer is fed into the linear layer and then the softmax layer to generate the cell content.

Network Training

- The shared components are repeatedly trained from the gradients received from three sub- tasks while each of three separate decoders is trained from the gradients obtained from its task. The whole system can be trained end-to-end on pairs of table images and their annotations of the table structure, the text content, and its bounding box per non-empty table cell by stochastic gradient descent algorithms. The overall loss of the proposed model is defined as the following:

$$L = \lambda_1 L_{struc.} + \lambda_2 L_{cont.} + \lambda_3 L_{bbox}$$

where Lstruc. and Lcont. are the table structure recognition loss and the cell-content prediction loss, respectively that are implemented in Cross-Entropy loss, Lbboxis the cell-bbox regression loss which is optimized by L1 loss. $\lambda 1$, $\lambda 2$, and $\lambda 3$ are weight hyperparameters.

Dataset Details

PubTabNet (Zhong et al., 2020) is a large-scale table image dataset that contains over 568k samples with their corresponding annotations of the table structure presented in HTML format, the text content, and its bounding box per non-empty table cell. This dataset is created by collecting scientific articles from PubMed Central Open Access Subset (PMCOA). The dataset is used in the ICDAR2021 competition (Jimeno Yepes et al., 2021) and divided into 500,777 training samples and 9,115 validation samples in the development phase, and 9,064 final evaluation samples in the Final Evaluation Phase.

FinTabNet is another large-scale table image dataset published by X. Zheng et al. (Zheng et al., 2021). The dataset is composed of complex tables from the annual reports of the S&P 500 companies with detailed annotations of the table structure and table cell information like the PubTabNet dataset. This dataset consists of 112k table images which are divided into training, testing, and validation sets with a ratio of 81% : 9.5% : 9.5%.

Table Structure Recognition Result

The cost of insertion and deletion operations is 1. When the edit is substituting a node n_o with n_s, the cost is 1 if either n_o or n_s is not *td*.

When both n_o and n_s are *td*, the substitution cost is 1 if the column span or the row span of n_o and n_s is different. Otherwise, the substitution cost is the normalized Levenshtein similarity between the content of n_o and n_s.

$$TEDS(T_a, T_b) = 1 - EditDist(T_a, T_b) / \max(|T_a|, |T_b|)$$

Table 1: Table structure recognition results on PubTabNet validation set (PTN) and FinTabNet (FTN).

Dataset	Model	TEDS-struc. (%)		
		Sim.	Com.	All
FTN	EDD (Zhong et al., 2020)	88.40	92.08	90.60
	GTE (Zheng et al., 2021)	-	-	87.14
	GTE (FT) (Zheng et al., 2021)	-	-	91.02
	TableFormer (Nassar et al., 2022)	97.50	96.00	96.80
	Our Model	**99.07**	**98.46**	**98.79**
PTN	EDD (Zhong et al., 2020)	91.10	88.70	89.90
	GTE (Zheng et al., 2021)	-	-	93.01
	LGPMA (Qiao et al., 2021)	-	-	96.70
	TableFormer (Nassar et al., 2022)	98.50	95.00	96.75
	Our Model	**99.05**	**96.66**	**97.88**

Sim. (Simple): Tables without multi-column or multi-row cells.
Com. (Complex): Tables with multi-column or multi-row cells.
(FT) Model was trained on PubTabNet and then finetuned.

Cell Detection Performance

Table 2: Cell detection results on PubTabNet validation set. PP: Post-processing.

Model	mAP (%)
EDD + BBox	79.20
TableFormer	82.10
EDD + BBox + PP	82.70
TableFormer + PP	86.80
Our Model	**88.93**

TEDS score on PubTabNet Dataset

Table 3: Table recognition results on PubTabNet validation set.

Model	TEDS (%)		
	Sim.	Com.	All
EDD (Zhong et al., 2020)	91.20	85.40	88.30
TabStruct-Net (Raja et al., 2020)	-	-	90.10
GTE (Zheng et al., 2021)	-	-	93.00
TableFormer (Nassar et al., 2022)	95.40	90.10	93.60
SEM (3) (Zhang et al., 2022)	94.80	92.50	93.70
LGPMA + OCR (1) (Qiao et al., 2021)	-	-	94.60
VCGoup (2) (Ye et al., 2021)	-	-	96.26
Our Model	**97.92**	**95.36**	**96.67**
VCGoup + ME (2) (Ye et al., 2021)	-	-	**96.84**

(1)(2)(3) are 1st, 2nd, and 3rd ranking solutions in ICDAR2021 competition.
ME: Model Ensemble.

Table 4: Table recognition results on PubTabNet final evaluation set.

Team Name	TEDS (%)		
	Simp.	Comp.	All
Davar-Lab-OCR	97.88	94.78	**96.36**
VCGroup	**97.90**	94.68	96.32
XM	97.60	**94.89**	96.27
Our Model	97.60	94.68	96.17
YG	97.38	94.79	96.11
DBJ	97.39	93.87	95.66
TAL	97.30	93.93	95.65
PaodingAI	97.35	93.79	95.61
anyone	96.95	93.43	95.23
LTIAYN	97.18	92.40	94.84