“지휘통제 지능정보 플랫폼” 기반 Vision-LLM을 활용한 병합 셀 테이블의 HTML 변환에 관한 연구

박병훈[1], 민지윤[1], 김예지[1], 황영준[1], 이종호[1], 김기환[1]
[1]티쓰리큐(주)
warmpark@t3q.com, minjyun01@gmail.com, yejik4203@gmail.com,
hyjun0103@t3q.com, leejh@t3q.com, tensor@t3q.com

A Study on HTML Conversion of Merged Cell Tables Using Vision-LLM Based on the “Command and Control Intelligence Information Platform”

Byeong-Hoon Park[1], Ji-Yun Min[1], Ye-Ji Kim[1],
Yeong-Jun Hwang[1], Jong-Ho Lee[1], Ki-Hwan Kim[1]
[1]T3Q(주)

요 약

문서의 디지털화 수요가 급증함에 따라, 정보 추출 및 구조화 연구의 중요성이 커지고 있다. 본 연구는 병합 셀이 포함된 테이블 이미지를 HTML 코드로 변환하기 위해 Vision Language Model의 파인튜닝 학습과 실험을 지휘통제 지능정보 플랫폼 기반에서 진행하였다. 베이스 모델은 MiniCPM-V 2.6을 사용하였으며, 학습 데이터는 TNCR과 PubTables-1M 데이터셋 일부를 수정하여 표 이미지-HTML 코드 쌍으로 구성하였다. 성능 평가는 TEDS 지표를 사용하였으며, 파인튜닝 모델은 100개의 테스트 데이터에 대해 93.15%의 TEDS 점수를 기록하여 베이스 모델(78.63%)보다 향상된 성능을 보였다. 본 연구는 병합 셀이 포함된 테이블 구조 인식 분야 연구에서 파인튜닝을 통해 Vision-LLM의 성능을 향상시킬 수 있음을 보여주는 사례로, 다양한 문서 디지털화 작업에 실제적인 기여를 할 수 있을 것으로 기대된다.

1. 서론

최근 문서의 디지털화에 대한 요구가 급증하면서 문서에서 정보를 추출하고 이를 구조화하는 연구가 중요해지고 있다. 특히 표 데이터(table data)는 중요한 정보를 효율적으로 전달할 수 있는 요소로, 이를 정확하게 인식하고 디지털화하는 과정은 핵심 연구 과제이다.

Vision-LLM(Vision Language Model)은 이미지와 텍스트를 동시에 이해하는 능력을 바탕으로, 표 이미지를 디지털화하는데 유용하다. 특히 병합 셀과 같은 복잡한 표 구조를 인식하려면 단순 OCR 기술보다 Vision-LLM을 활용하는 접근이 필요하다.

문서 내 표를 LLM이 이해할 수 있도록 하려면 구조화된 마크업 언어로 변환하는 과정이 필수적이다. 이에 따라, Vision-LLM을 사용하여 병합된 셀을 포함하는 표 이미지를 HTML 형식으로 변환하는 방법을 제안하고자 한다.

병합 셀(Merged Cells)이란, 표의 여러 행(row) 또는 열(column)에 걸쳐 두 개 이상의 셀이 합쳐져 하나의 셀을 이루는 구조를 의미한다. 병합 셀은 복잡한 데이터를 사람에게 직관적으로 전달하는데 유용하지만, 이를 구조적 형태를 유지하며 디지털 형태로 변환하는 것은 한계가 있다.

(그림 1) 병합 셀을 포함하는 표의 예시

본 논문은 정확한 표 구조 인식과 데이터 추출을 목표로 하며, 특히 병합 셀 구조를 HTML로 온전히 변환하는 데 중점을 둔다.

논문의 구성은 다음과 같다. 2장에서 관련 연구를, 3장에서 모델과 학습 데이터셋을 설명한다. 4장에서 파인튜닝 방법을 소개하고, 5장은 실험 결과를 제시한다. 마지막으로 6장에서 결론을 정리한다.

2. 연구 동향

2.1. 테이블 구조 인식

테이블 구조 인식(Table Structure Recognition, TSR)은 테이블의 구조적 요소인 행, 열, 셀, 헤더, 병합 셀 등을 디지털 형식으로 변환하고, 구조를 복원하는 연구 분야이다. 초기 TSR 연구는 주로 경험적인 규칙 기반 접근법(rule-based approaches)을 사용하였다. 문서의 레이아웃을 분석하고, 특정한 규칙을 발견하면 이를 테이블로 판단하고 구조를 인식하는 방식이었다. 하지만 이 방법은 문서 레이아웃이 비정형적인 경우 정확도가 크게 떨어졌다.[1]

이후 머신러닝 기반 접근법(machine learning approaches)이 도입되어 테이블의 행과 열을 예측하려는 시도가 있었다. 통계적 기계학습 방법은 규칙에 대한 의존도를 줄였으나, 여전히 표의 레이아웃에 대한 가정이 존재하여 성능에 한계가 있었다.[2]

이러한 규칙 기반 또는 통계적 머신러닝 방법은 간단한 표 구조 인식에는 유용했으나, 복잡한 표를 처리하는 것에 어려움이 있었다. 최근에는 딥러닝 기법(Deep Learning approaches)과 대규모 테이블 데이터셋을 활용한 학습 방법이 TSR 연구에 도입되어, 정확도와 처리 능력이 크게 향상되고 있다.[3]

그러나 딥러닝 기반 방법이 성능을 크게 개선했음에도, 병합 셀이 포함된 테이블 구조 인식은 여전히 해결해야 할 과제로 남아있다.[4]

3. 사용한 모델과 데이터셋

3.1. MiniCPM-V 2.6

MiniCPM-V 2.6[5]은 MiniCPM-V 시리즈의 가장 최신 모델로, 멀티 모달 대형 언어 모델이다. 이 모델은 시각 처리 모듈(Visual Processing Module)과 언어 모델(LLM) 부분으로 구성된다. 시각 처리 모듈은 SigLIP-400M을 사용하여 시각 정보를 인코딩하고, 이를 통해 시각 토큰(visual token)을 생성한다. 시각 처리 모듈을 통해 얻은 시각 토큰은 텍스트와 함께 Qwen2-7B를 기반으로 한 언어 모델에 전달되어 함께 처리된다.

3.2. 데이터셋

TNCR(Table Net Detection and Classification Dataset)[6]와 PubTables-1M[7]는 병합 셀을 포함한 표 이미지가 있는 자료로, 이를 기반으로 모델 학습 및 평가를 진행하였다.

3.2.1. TNCR

TNCR[6]은 오픈 액세스 웹에서 수집된 테이블이 포함된 문서 이미지와 바운딩 박스 정보를 제공하는 데이터셋이다. 해당 데이터셋은 테이블 구조에 따라 5가지 유형으로 구분되는데, 이 중 병합 셀이 포함되고 선이 있는 데이터인 Merged cells 유형만을 선택하였다. 이후 바운딩 박스를 기준으로 테이블 이미지만 추출하고, 불필요한 그림이나 유사한 스타일의 표 이미지를 최소화하였다.

3.2.2. PubTables-1M

PubTables-1M[7]은 Microsoft에서 구축한 대규모 데이터셋으로, 표 감지 및 표 구조 인식을 지원한다. 이 중 표 구조 인식을 위한 표 이미지만 선택하였으며, 이후 50여 개의 병합 셀을 포함하는 선이 있는 표 이미지를 선별하였다.

3.2.3. 데이터 변형

수집된 표 이미지 데이터에 대해 GPT-4o mini API를 활용해 HTML 코드를 생성했으나, 이미지의 세부 값을 정확히 반영하지 못하는 한계가 있었다. 이에 따라, 원본 이미지를 사용하지 않고 HTML로 변환한 표를 다시 이미지로 렌더링하여 사용하였다. 이 과정에서 자동 생성된 코드의 표 비율이 맞지 않거나 구조가 어긋난 부분은 검수를 통해 수정하여, 구조와 형식이 균형을 이루도록 조정하였다.

HTML 코드는 <table border="1" cellspacing="0"> 태그로 시작하며, <thead>, <tbody> 태그는 사용하지 않았다. 스타일 코드는 모두 제외했으나, 이미지 생성 과정에서는 표의 비율, 글꼴, 글자 크기 등의 변화를 주었다. 표 내부 값들은 영어 문자 및 일반 유니코드 문자만을 사용하였다. 병합 셀의 표현은 rowspan, colspan을 통해 구현되었다.

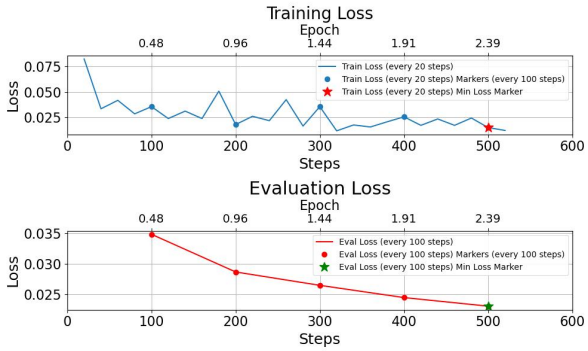
3.2.4. 데이터셋 구성

최종적으로 생성된 데이터셋은 표 이미지-HTML 코드 쌍 1,029개로 이루어졌으며, 학습 및 평가 데이터셋은 약 8:1:1의 비율로 분할하였다.

테이블의 병합 셀 비율을 계산한 결과, 병합 셀이 전체에서 차지하는 평균 비율은 약 29%, 표준편차는 0.18로 나타났다. 이는 각 테이블을 격자 형태로 나누었을 때, 병합 셀이 얼마의 공간을 차지하는지를 보여준다.

4. 파인튜닝 방법

본 연구는 병합 셀이 포함된 테이블 이미지를 HTML 코드로 변환하는 Vision-LLM의 성능을 향상시키기 위해 파인튜닝을 수행하였다. 베이스 모델의 비전(Vision) 모듈은 학습되도록 설정하고, 언어 모델(LLM) 부분은 고정하였다. 이는 표 이미지의 구조 인식 능력을 강화하는데 집중한 실험이다. 학습 데이터는 병합 셀이 포함된 표 이미지-HTML 코드 쌍이다. 또한, LoRA(Low-Rank Adaptation) 기법[8]을 적용해 효율적인 파인튜닝을 진행하였다.



(그림 2) Training Loss와 Evaluation Loss

다음은 실험 환경과 주요 하이퍼파라미터 설정이다.

- NVIDIA H100(80G) GPU 2장
- 학습률(Learning Rate) : 1e-6
- 초기 15 steps : Warmup 적용
- 배치 크기(Batch Size) : 2

약 2.5 에포크(epoch)까지 학습하여, 가장 손실값이 낮았던 $step = 500$, $loss = 0.0150$을 기준으로 성능을 평가하였다.

5. 실험 결과

베이스 모델과 파인튜닝 모델에 대해 동일한 100개의 테스트 데이터셋에서 추론을 진행하였다. 테스트 데이터는 표 이미지와 함께 영문 텍스트로 입력되었으며, HTML 테이블 형식을 명시하고 'rowspan'과 'colspan' 요소를 언급하였다.

5.1. 평가 지표

본 연구에서는 Tree Edit Distance based Similarity(TEDS)[9]를 사용하여 성능 평가를 실시하였다. TEDS란 트리 편집 거리(Tree Edit Distance)[10]를 기반으로 두 테이블 간의 유사성을 평가하는 지표이다. 이 지표는 0에서 1 사이의 범위로 표현되며, 값이 1에 가까울수록 두 테이블의 유사성이 높다는 것을 의미한다.

편집 거리란 하나의 트리를 다른 트리로 변환하기 위해 필요한 최소 편집 작업(삽입, 삭제, 대체 등)의 비용을 의미한다. 테이블은 HTML에서 트리 구조로 표현되고, 테이블 인식 결과를 전역 트리 구조 수준에서 검사한다. TEDS는 빈 셀, 병합 셀 같은 구조적 오류를 식별할 수 있다. 또한 셀 내 텍스트의 유사성도 포착하여 점수에 반영한다.

두 테이블 A, B에 대한 TEDS는 다음과 같다:

$$TEDS(A,B) = 1 - \frac{EditDist(A,B)}{\max(|A|,|B|)} \quad (1)$$

$EditDist(A,B)$은 두 테이블의 트리 구조 간의 편집 거리이고, $\max(|A|,|B|)$은 두 테이블 중 노드 수가 더 많은 테이블의 노드 수이다. 노드는 HTML 테이블의 구조적 요소 중 <td> 태그에 해당한다.

5.2. 후처리 방법

TEDS[9]의 코드를 바탕으로, 정확한 평가를 위해 추론 결과에 다음의 후처리가 필요하다.

1) <th> 태그를 <td> 태그로 변환한다. <td> 태그만이 트리의 리프노드로 계산되고, <th>는 코드에서 누락되기 때문이다.
2) 셀 안의 내용(content) 앞뒤에 있는 공백을 모두 제거한다. 편집 거리 계산에는 태그 구조뿐만이 아니라, 태그 안의 내용(content)도 함께 계산된다. 이때 공백도 하나의 토큰처럼 처리된다. 의도치 않은 공백으로 인한 오차를 줄이기 위함이다.

테스트 이미지

		Estimate	t	Sig.
Constant		-0.028	-1.598	.003
AR	Lag 1	1.796	16.278	.000
	Lag 2	-1.037	-9.945	.003
	Lag 3	0.232	1.114	.263
	Lag 4	-0.084	-0.773	.440
MA	Lag 1	1.706	23.993	.000
	Lag 2	-0.940	-12.853	.000

Base TEDS : 0.673469

	Estimate	t	Sig.
Constant			
Lag 1	1.796	16.278	.000
Lag 2	-1.037	-9.945	.003
AR			
Lag 3	0.232	1.114	.263
Lag 4	-0.084	-0.773	.440
MA			
Lag 1	1.706	23.993	.000
Lag 2	-0.940	-12.853	.000

Fine-tuned TEDS : 0.952381

	Estimate	t	Sig.	
Constant	-0.028	-1.598	.003	
AR	Lag 1	1.796	16.278	.000
	Lag 2	-1.037	-9.945	.003
	Lag 3	0.232	1.114	.263
	Lag 4	-0.084	-0.773	.440
MA	Lag 1	1.706	23.993	.000
	Lag 2	-0.940	-12.853	.000

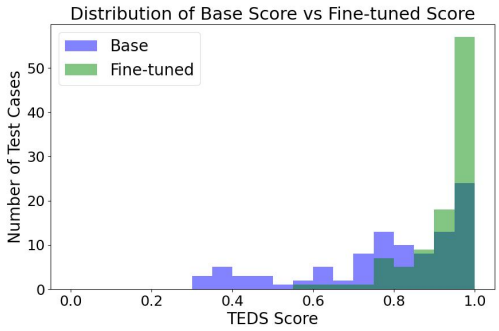
(그림 3) 모델 출력 예시

5.3. 결과

실험 결과, 파인튜닝 모델이 베이스 모델보다 향상된 TEDS 점수를 기록했다. 그림 4는 두 모델의 점수 분포를 나타낸 히스토그램이다. 베이스 모델은 넓은 범위에 걸쳐 분포하는 반면, 파인튜닝 모델은 0.8~1.0 구간에서 높은 빈도를 보인다. 즉, 파인튜닝 모델이 더 많은 데이터에서 높은 점수를 기록하며 전반적으로 더 우수한 성능을 발휘했음을 보여준다.

<표 1> 평균 TEDS 점수 비교

Model	Average TEDS (%)
Base	78.63
Fine-tuned	93.15



(그림 4) TEDS 점수 분포 히스토그램

6. 결론

본 연구는 병합 셀이 포함된 표 이미지를 HTML 코드로 변환하는 과정에서 Vision-LLM의 성능을 향상시키기 위한 파인튜닝 기법을 제안하고 효과를 검증하였다. TEDS 지표로 평가한 결과, 파인튜닝 모델이 베이스 모델보다 전반적으로 더 우수한 점수를 보이며 개선된 성능을 확인할 수 있었다.

본 논문은 Vision-LLM을 통한 병합 셀이 포함된 복잡한 테이블의 HTML 변환에서 파인튜닝의 효과를 검증한 사례로, 향후 더 다양한 테이블 구조의 데이터셋을 확보할 경우 실제적인 성능 확장의 가능성을 제시한다.

사사(Acknowledgement)

본 연구는 국방신속획득기술연구원의 지원으로 수행된 연구임 (No. UC200019D).

참고문헌

[1] Itonori, K, “Table structure recognition based on textblock arrangement and ruled line position,” *Proceedings of 2nd International Conference on Document Analysis and Recognition (ICDAR93), IEEE*, 1993, pp. 765-768.

[2] Kieninger, T., & Dengel, A., "The t-recs table recognition and analysis system." *Document Analysis Systems: Theory and Practice: Third IAPR Workshop, DAS'98 Nagano, Japan, November 4-6, 1998 Selected Papers 3* (pp. 255-270). Springer Berlin Heidelberg, 1999.

[3] Schreiber, S., Agne, S., Wolf, I., Dengel, A., & Ahmed, S., "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, pp. 1162 - 1167, 2017.

[4] Anand, Avinash, et al, "TC-OCR: TableCraft OCR for Efficient Detection & Recognition of Table Structure & Content," *Proceedings of the 1st International Workshop on Deep Multimodal Learning for Information Retrieval.*, 2023, pp.11-18.

[5] Yao, Yuan, et al., "MiniCPM-V: A GPT-4V Level MLLM on Your Phone," *arXiv preprint arXiv:2408.01800*, 2024.

[6] Abdallah, A., Berendeyev, A., Nuradin, I., & Nurseitov, D., "TNCR: Table Net Detection and Classification Dataset," *Neurocomputing*, vol. 473, pp. 79 - 97, 2022.

[7] Smock, B., Pesala, R., & Abraham, R., “PubTables-1M: Towards comprehensive table extraction from unstructured documents,” *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4634 - 4642.

[8] Hu, Edward J., et al., “LoRA: Low-Rank Adaptation of Large Language Models,” *arXiv preprint arXiv:2106.09685*, 2021.

[9] Zhong, X., ShafieiBavani, E., & Jimeno Yepes, A. “Image-based table recognition: data, model, and evaluation,” *European conference on computer vision*, Cham: Springer International Publishing, 2020, pp. 564-580.

[10] Pawlik, M., & Augsten, N., “Tree edit distance: Robust and memory efficient,” *Information Systems*, vol. 56, pp. 157 - 173, 2016.