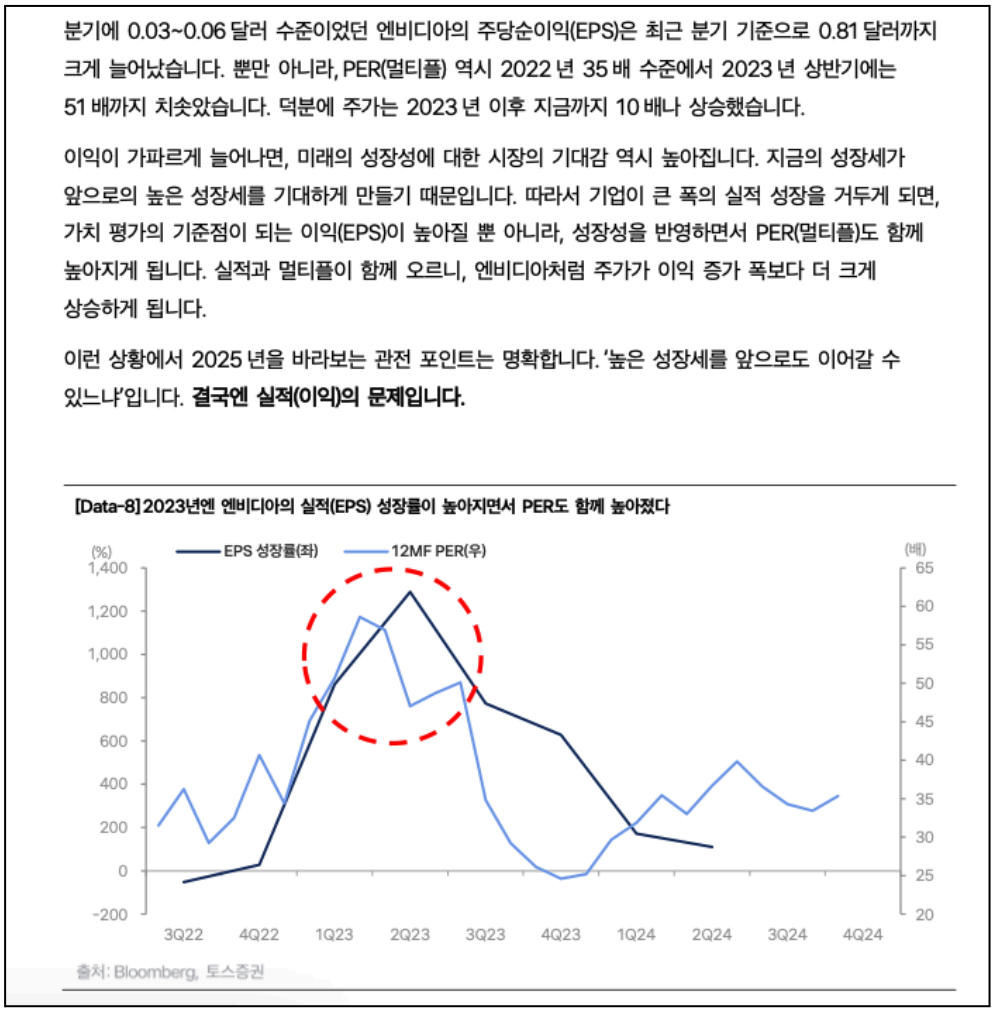
Unifying Vision, Text, and Layout for Universal Document Processing
https://arxiv.org/pdf/2212.02623

CVPR 2023 논문

주제: 이미지, 텍스트, 레이아웃을 동일한 공간에 표현하는 Vision-Text-Layout Transformer 를 제안

선정 이유
- vision transformer에 대한 기본적인 이해가 필요하다고 생각함
- '문서 이해' 관점에서의 VLM 관련 논문이 궁금했음

Introduction
- 일반적인 vision 연구와 달리, '문서 이해' 분야는 텍스트의 기호 추출 (bullet point 등), 스타일 (제목이 두껍다던지…) 그림에 대한 이해와 같이 구조화 된 멀티모달 task -> 2D 공간 레이아웃
- 그리고 콘텐츠와 문서 유형에 따라 서로 다른 구조를 가짐: 명함, 인보이스, 세금 납무 명세서 …
- 과거 연구들은 이런 문서 이해를 위해 제목, 사인 추출, 문서 분류, 테이블 파싱 … 온갖 작업들이 필요했음



- \+ 문서 이해 분야는 텍스트와 이미지가 매우 깊은 상호작용! -> 텍스트가 이미지를 설명
- 문서 이해를 통해 수행하는 task도 매우 다양함 (1) QA, (2) 테이블 설명, (3) 문서 분류, (4) 정보 추출 …
- 이해도 복잡하고, 다운스트림 task도 복잡한데, 이걸 통합할 수 있는 방법이 없을까???

기존 연구
- (1) Unifying Model Architectures in Multimodal Learning.
    - 텍스트 임베딩과 이미지 패치를 결합해서 입력으로 사용

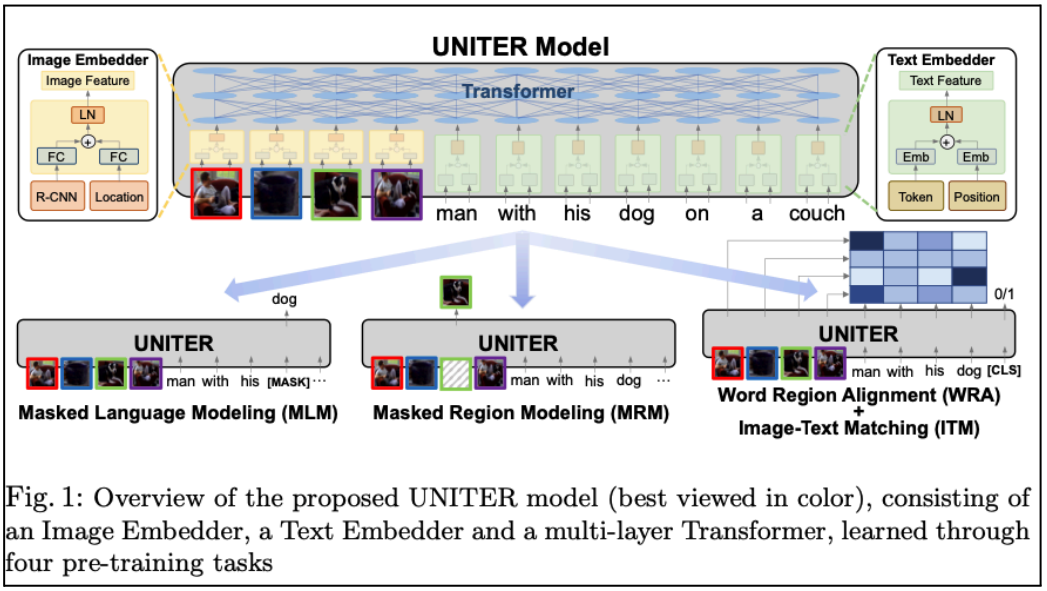


Fig. 1: Overview of the proposed UNITER model (best viewed in color), consisting of an Image Embedder, a Text Embedder and a multi-layer Transformer, learned through four pre-training tasks

- Image 임베딩을 위한 image embedder와 텍스트 임베딩을 위한 text embedder 사용
  - Faster R-CNN으로 이미지 피쳐 추출 (pooled ROI features): https://velog.io/@imfromk/CV-Understanding-RoIsRegion-of-Interest
  - [x1, y1, x2, y2, w, h, w*h] 로 location 임베딩
- 요러면 효율적이긴 하지만, 텍스트와 이미지의 공간적인 이미지 (레이아웃) 정보를 반영할 수 없음
- CLIP 모델처럼 각 모달을 임베딩 한 후 결합하는 방식도 있음 -> 모달 별로 새로운 헤드 적용
- 모달의 특성을 개별적으로 학습할 순 있으나, 모달 간 깊은 관계성을 나타내기는 어려움 (그래프의 특정 부위가 텍스트와 매칭 등..)

- (2) Unifying Tasks with the Generative Framework.
  - 여러 task를 하나의 작업으로 합치는 방법론들
  - 예를 들어, 자연어처리 분야에서 T5 언어 모델은 1800개의 작업을 통합해 학습
  - 하나의 모델로 여러 작업을 처리할 수 있음을 보여줌

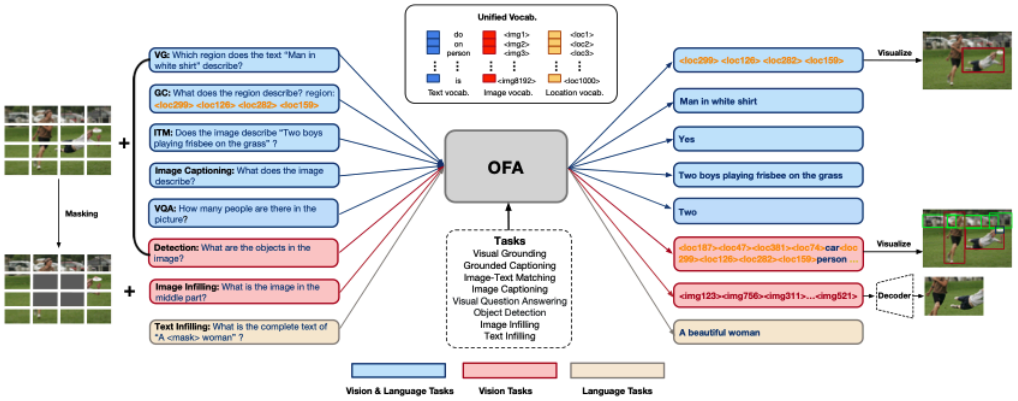


Figure 2: A demonstration of the pretraining tasks, including visual grounding, grounded captioning, image-text matching, image captioning, VQA, object detection, image infilling as well as text infilling.

- OFA 모델: seq2seq framework, 이미지, 텍스트, 객체를 이상화 해서 하나의 vocabulary에 넣어서 학습함
- 이미지는 ResNet을 사용, 자연어는 BBPE 사용
- 이미지 정보는 256 x 256 이미지를 16 x 16의 코드 시퀀스로 변환시킴: ViT 같은건가..? https://velog.io/@sjinu/%EB%85%BC%EB%AC%B8%EB%A6%AC%EB%B7%B0AN-IMAGE-I

S-WORTH-16X16-WORDS-TRANSFORMERS-FOR-IMAGE-RECOGNITION-AT-SCALE-Vi-T Vision-Transformer

- 코드 시퀀스: 16 x 16 짜리 CNN을 돌리는데, 커널이 16, stride가 16으로 -> 16 x 16 하나의 token으로 여겨서 -> token embedding
- 이것도 문서 내 레이아웃 정보를 다루는데는 한계점이 있다고 함

- (3) Document Artificial Intelligence.
  - 그럼 레이아웃에 집중한 기존 연구는?

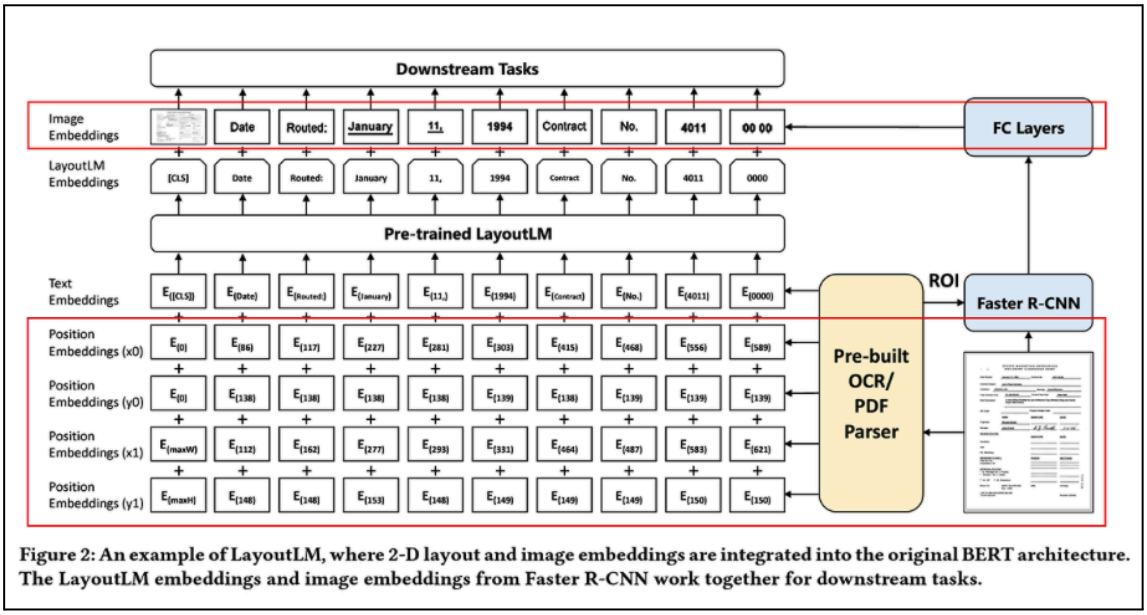


Figure 2: An example of LayoutLM, where 2-D layout and image embeddings are integrated into the original BERT architecture. The LayoutLM embeddings and image embeddings from Faster R-CNN work together for downstream tasks.

  - BERT 구조를 활용해서 텍스트의 2D 위치 정보를 활용
  - 이미지는 Faster R-CNN으로 임베딩해서 추가
  - Downstream task를 위해 head를 task 별로 추가해서 학습해야한다는 단점이 있었음
  - 그외에도 위치 정보를 활용하기 위해 바운딩 박스 (x1, x2 … -> vocab mapping)를 학습하는 pix2seq나, 이미지 위치 정보를 텍스트로 변환해서 자연어로 학습하는 경우 등등…

기존 연구 한계점

- 모달리티 간의 상관 관계는 잘 반영할 수 없었음 (독립 된 대상으로 이해하기 때문에)
- 그리고 모달리티 마다 자꾸 헤드를 넣어주는데, 유동적일 수 없음
- 또한, 문서의 '레이아웃' 정보가 반영되도록 고려되지 않음

제안 방법

- Seq2Seq 구조를 제안: 모델 구조

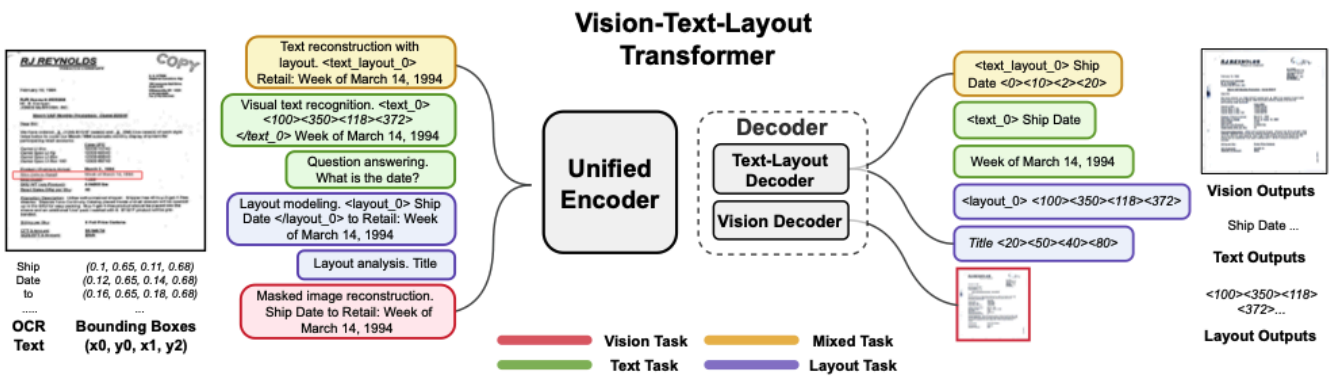


Figure 1. UDOP unifies vision, text, and layout through vision-text-layout Transformer and unified generative pretraining tasks including vision task, text task, layout task, and mixed task. We show the task prompts (left) and task targets (right) for all self-supervised objectives (joint text-layout reconstruction, visual text recognition, layout modeling, and masked autoencoding) and two example supervised objectives (question answering and layout analysis).

- Unified Encoder
  - 인코더의 입력의 원천은 문서 이미지

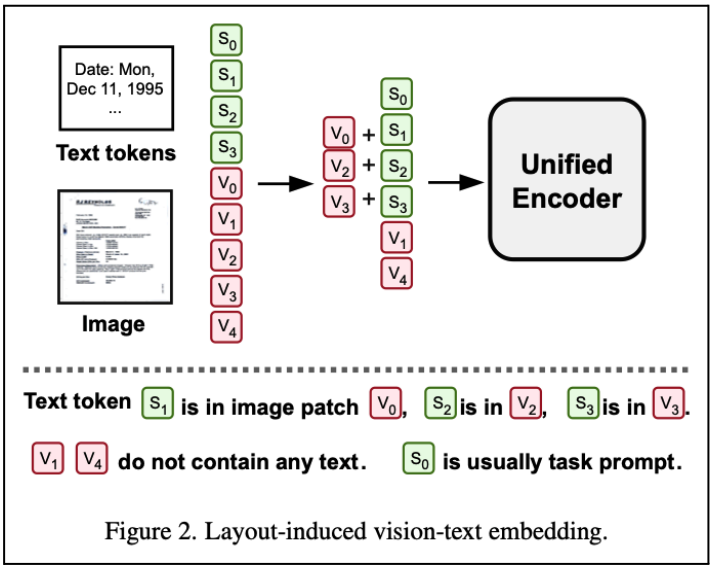


Figure 2. Layout-induced vision-text embedding.

- 문서 이미지를 OCR를 활용해서 텍스트(character)의 바운딩 박스를 획득
- 각 토큰 별로, [이미지, 텍스트, 바운딩박스] 3개의 정보가 paired: Triple 이라고 표현
- 목적: 이미지 픽셀과 텍스트간 1:1 관계성을 저장하고 싶다!
- 이미지 처리
  - 주어진 이미지를 패치로 나누고, 각 패치를 임베딩 벡터로 변환
- 텍스트 처리
  - 텍스트 임베딩 벡터로 변환
- 이미지 & 텍스트 joint representation
  - 텍스트 바운딩 박스가 이미지 패치와 겹친다면 1로 표현

$$\phi(\boldsymbol{s}_i, \boldsymbol{v}_j) = \begin{cases} 1, & \text{if the center of } \boldsymbol{s}_i\text{'s bounding box} \\ & \text{is within the image patch } \boldsymbol{v}_j. \\ 0, & \text{otherwise.} \end{cases}$$

  - 만약 텍스트 si가 겹친다면? vi과 더함

- 레이아웃 통합
    - 이미지 & 텍스트를 연결하는 '레이아웃 임베딩'을 추가 -> 코드에서 모델 아키 구조 확인해보자
    - 텍스트의 바운딩 박스 기준으로 이산화해서 토큰으로 변환 (e.g. (0.1, 0.2, 0.5, 0.6) → <50><100><250><300>) vocab (special tokens): '<50>', '<100>', '<250>', ….
    - 객체가 많아지면 seq len이 엄청 길어짐 …
    - 이 논문의 바운딩 박스 기준은? 음절이면 엄청 많을텐데,,,
- Vision-Text-Layout Decoder
    - Text-Layout decoder
        - 일반적인 seq2seq 처럼 uni-directional transformer decoder 사용
        - text와 layout 토큰을 생성하도록 학습
    - Vision decoder
        - Masked Autoencoder (MAE) - CVPR 2022

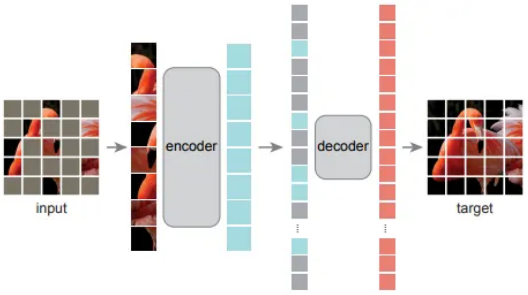


Figure 1. **Our MAE architecture**. During pre-training, a large random subset of image patches (*e.g.*, 75%) is masked out. The encoder is applied to the small subset of *visible patches*. Mask tokens are introduced *after* the encoder, and the full set of encoded patches and mask tokens is processed by a small decoder that reconstructs the original image in pixels. After pre-training, the decoder is discarded and the encoder is applied to uncorrupted images (full sets of patches) for recognition tasks.

        - Image reconstruction을 학습
        - MSE loss를 활용한 pixel 값 (R, G, B: 0-255, 0-255, 0-255) 을 맞추도록 학습
        - + text와 layout information으로부터 생성하도록 만든 것
- Self-Supervised Pretraining Tasks
    - 입력 예시 "**Ship Date to Retail: Week of March 14, 1994**"
    - Step 01: Joint Text-Layout Reconstruction
        - masking 된 텍스트의 위치를 찾아내는 것을 학습
        - 이를 통해 text와 레이아웃 관계를 학습할 수 있음
        - 'Ship Date' 와 'of' 를 마스킹했다고 가정해보자
            - 입력 시퀀스
                - **Joint Text-Layout Reconstruction.** <text_layout_0> to Retail: Week <text_layout_1> March 14, 1994
            - 타겟 시퀀스
                - <text_layout_0> Ship Date <100><350><118><372> <text_layout_1> of <100><370><118><382>
        - 모델은 텍스트와 바운딩 박스 위치를 동시에 복원
    - Step 02: Layout Modeling
        - 주어진 문서에서 텍스트가 위치할 좌표를 예측
        - 문서 구조 (제목이나..) 위치 기반의 정보 추출 (명함에서 이름 등)에 유용
        - 'Ship Date' 와 'of'를 예측한다고 가정해보자
            - 입력 시퀀스
                - **Layout Modeling.** <layout_0> Ship Date </layout_0> to Retail: Week <layout_1> of </layout_1> March 14, 1994
            - 타겟 시퀀스
                - <layout_0> <100><350><118><372> <layout_1> <100><370><118><382>
    - Step 03: Visual Text Recognition
        - 이미지의 특정 위치에 있는 텍스트를 예측

        - 이미지와 텍스트의 상관 관계를 학습
        - 바운딩 박스의 빈칸을 예측한다고 가정해보자
            - 입력 시퀀스
                - **Visual Text Recognition.** <text_0> <100><350><118><372> </text_0> to Retail: Week <text_1> <100><370><118><382> </text_1> March 14, 1994
            - 타겟 시퀀스
                - <text_0> Ship Date <text_1> of
    - Step 04: Masked Image Reconstruction
        - 마스킹 된 이미지의 패치를 복원
        - 이미지의 시각적 구조를 이해하도록 학습
        - 입력 시퀀스
            - Masked Image Reconstruction. Ship Date to Retail: Week of March 14, 1994
        - 타겟
            - 이미지 픽셀

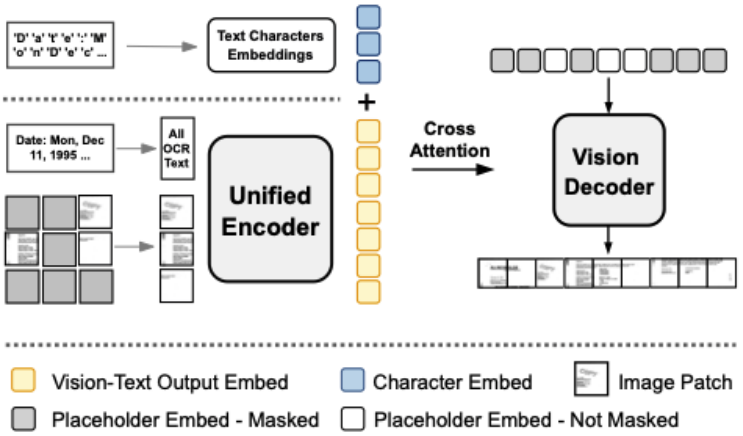


Figure 3. Masked autoencoding with text and layout.

        - 짜잔! 사실 character embedding도 숨어있었습니다..
        - image generation의 퀄리티 향상을 위해 추가된 모듈
        - vision decode에는 입력으로 들어가는 임베딩 벡터가, unified encoder를 통해 나온 text-layout 정보와 character embedding 결과의 cross-attention 결과가 들어감
        - 학습되는 layer고… unified encoder과 독립적으로 이루어짐
- Supervised Pretraining Tasks
    - 라벨 데이터를 이용해서 모델을 추가 학습
    - image + text 관련 task 들
        - 문서 분류
            - 영수증, 계약서, 메모, 보고서 등
        - 레이아웃 분석
            - 문서 내 제목, 단락의 위치 찾기
        - 정보 추출
            - 날짜나 요약 단락 등
        - 질의응답
            - 문서와 관련된 질의응답
        - Document NLI
            - 두 문장의 의미적 관계 (문장 A가 사실일 떄 다른 문장 B의 사실을 보장)
            - Entailment or Not Entailment
- Experiment Setup
    - T5-Large 아키 (input embedding layer 2개: (1) s, v (2) 레이아웃 임베딩) + MAE-large docoder 아키
    - 794M parameters
    - T5 tokenizer 활용 + add special tokens (‘<50>’, ...)
    - IIT-CDIP dataset

- Curriculum learning
  - 1024 resolution 이미지 -> 4096 패치 시퀀스 랭스: 학습에 오래 걸림
  - 224 -> 512 -> 1024로 레졸루션 바꿔가며 학습 (size 별로 얼마나 학습한지는 안나와있었음..)
  - 512 배치, 1 에폭
- 결과 (논문으로 함께 볼게요)

Table 2. Comparison with existing published models on the DUE-Benchmark. Modality T, L, V denote text, layout, or vision.

Model	Modality	Question Answering		Information Extraction			Table QA/NLI		Avg.
		DocVQA	InfoVQA	KLC	PWC	DeepForm	WTQ	TabFact	
Donut [21]	V	72.1	-	-	-	-	-	-	-
$BERT_{large}$ [9]	T	67.5	-	-	-	-	-	-	-
$T5_{large}$ [39]	T	70.4	36.7	74.3	25.3	74.4	33.3	58.9	50.7
$T5_{large}$+U [36]	T	76.3	37.1	76.0	27.6	82.9	38.1	76.0	56.5
$T5_{large}$+2D [36]	T+L	69.8	39.2	72.6	25.7	74.0	30.8	58.0	50.4
$T5_{large}$+2D+U [36]	T+L	81.0	46.1	75.9	26.8	83.3	43.3	78.6	59.8
LAMBERT [10]	T+L	-	-	81.3	-	-	-	-	-
$StructuralLM_{large}$ [26]	T+L	83.9	-	-	-	-	-	-	-
$LayoutLMv2_{large}$ [55]	V+T+L	78.8	-	-	-	-	-	-	-
$LayoutLMv3_{large}$ [16]	V+T+L	83.4	45.1	77.1	26.9	84.0	45.7	78.1	62.9
UDOP	V+T+L	**84.7**	**47.4**	**82.8**	**28.0**	**85.5**	**47.2**	**78.9**	**64.8**

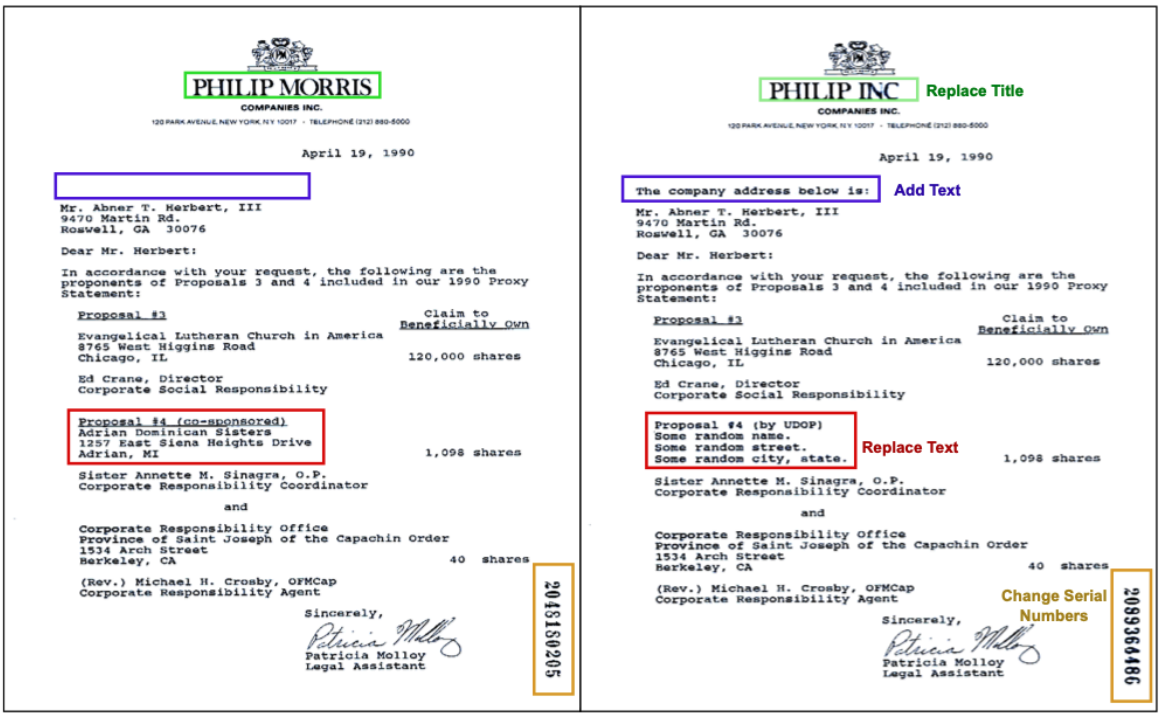


Figure 4. Document generation with customized content (right). Left is the original document. We show four document edits within the same figure including title replacement, text addition, text replacement, and tilted text replacement. All edits are done with one model run.

- 생성도 잘하더라

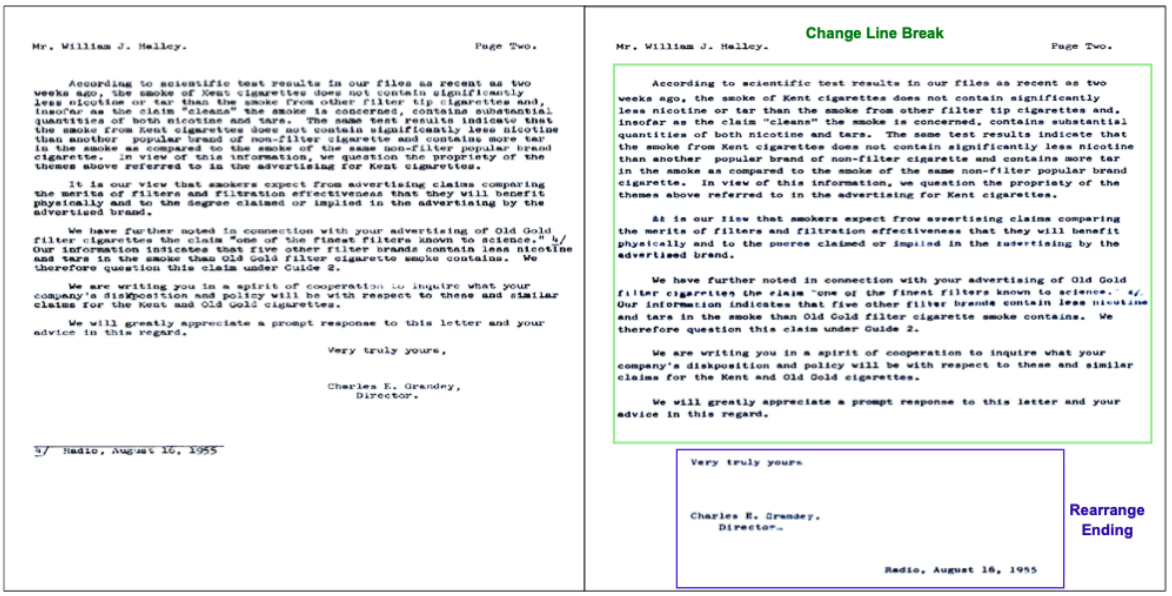


Figure 5. Document generation with customized layout (right). Left is the original document. We change the layout of the document text including line breaks change and text rearrangement. All edits are done with one model run.

-

1. 연구의 핵심 요약

UDOP 모델의 핵심 목표는 **문서 처리(Document Processing)**의 복잡한 요구 사항을 충족할 수 있는 **범용(Universal)** 모델을 구축하는 것이었습니다. 이 모델은 문서에서 흔히 발견되는 다양한 요소(텍스트, 이미지, 레이아웃)를 **통합적**으로 처리할 수 있는 능력을 제공합니다.

1) 기존 모델의 한계 극복
- 이전의 문서 처리 시스템들은 텍스트, 이미지, 레이아웃 정보를 개별적으로 처리하거나 제한된 조합만을 지원했습니다.
- UDOP는 이 세 가지 요소를 **하나의 통합 모델**로 처리하여, 다양한 작업을 동시에 수행할 수 있는 강력한 성능을 제공합니다.

2) 통합 생성 모델
- 기존의 특정 작업에 특화된 모델들과 달리, UDOP는 **생성 기반 접근법(generative approach)**을 사용하여 다중 작업을 처리합니다.
- 이를 통해 문서 편집, 정보 추출, 레이아웃 분석, 질의응답 등 다양한 작업을 **일관된 프레임워크** 내에서 수행할 수 있습니다.

---

2. 주요 기여 (Key Contributions)

1. **비전-텍스트-레이아웃 통합**
   - UDOP는 **VTL Transformer** 구조를 통해 시각, 텍스트, 레이아웃 정보를 자연스럽게 결합합니다.
   - 이를 통해 문서의 의미적 내용과 공간적 구성을 동시에 이해할 수 있습니다.

2. **범용 사전학습 전략**
   - 자가지도학습(Self-Supervised Learning)과 지도학습(Supervised Learning)을 결합하여 강력한 성능을 달성했습니다.
   - 다양한 사전학습 목표(텍스트 복원, 레이아웃 예측, 이미지 복원 등)를 통해 문서 작업의 범용성을 높였습니다.

3. **고품질 생성 및 편집 기능**
   - UDOP는 문서의 내용을 생성하고 수정할 수 있는 기능을 지원합니다.
   - 예를 들어, 제목 변경, 강조 표시 추가, 새로운 표 삽입 등 **문서 편집 및 생성** 기능을 제공합니다.

4. **다양한 작업에서 최신 성능 달성**
   - 질의응답(QA), 정보 추출(IE), 레이아웃 분석(Layout Analysis) 등의 작업에서 **최신 성능(State-of-the-Art)**을 달성했습니다.
   - 문서 처리 분야에서 여러 작업을 하나의 모델로 해결할 수 있는 가능성을 열었습니다.

---

3. 모델의 유연성과 확장성

1. **다중모달 학습의 확장성**
   - UDOP는 다양한 문서 형식(예: PDF, 영수증, 계약서 등)을 처리할 수 있습니다.
   - 다른 언어 및 도메인에서도 학습과 적용이 가능하여, **다국어 및 다분야 확장성**을 제공합니다.

2. **일관된 학습 및 추론 방식**
   - UDOP는 모든 작업을 **시퀀스 생성 방식**으로 처리합니다.
   - 이로 인해 모델 학습과 추론 과정이 단순화되었고, 새로운 작업에 쉽게 확장할 수 있습니다.

3. **프롬프트 기반 학습**
   - 다양한 프롬프트(prompt)를 입력하여 모델의 출력을 제어할 수 있습니다.
   - 이를 통해 모델의 사용성과 활용성을 더욱 향상시켰습니다.

---

4. 사회적 영향과 윤리적 고려사항

UDOP 모델의 강력한 기능은 다양한 문서 처리 작업에서 효율성을 높일 수 있지만, 이러한 기술의 **악용 가능성**에 대한 윤리적 고려도 필요합니다.

1. **문서 위조 및 조작 위험**
   - UDOP는 고품질 문서 생성 및 편집 기능을 제공하므로, 문서 위조나 조작에 악용될 가능성이 있습니다.
   - 따라서 보안 및 인증 기능을 추가로 강화해야 할 필요성이 언급됩니다.

2. **데이터 편향(Bias) 문제**
   - 모델이 학습한 데이터에 포함된 편향(bias)이 결과에 영향을 미칠 수 있습니다.
   - 공정성과 신뢰성을 보장하기 위해 편향을 최소화하는 추가 연구가 필요합니다.

3. **프라이버시 보호**
   - 문서 내 민감한 정보를 보호하고, 개인정보 유출 위험을 줄이기 위한 안전장치가 필요합니다.

---

5. 향후 연구 방향 (Future Work)

논문은 UDOP의 성능을 더욱 향상시키고, 적용 범위를 넓히기 위한 몇 가지 연구 방향을 제안합니다.

1. **모델 크기와 효율성 개선**
   - 더 크고 복잡한 데이터셋에 대해 학습하면서도 효율성을 유지하는 모델 구조를 연구.
   - 경량화된 모델 버전 개발을 통해 실시간 적용 가능성 증대.

2. **다국어 지원**
   - 다국어 문서 처리 능력을 강화하여 글로벌 적용성을 높이기 위한 추가 연구.

3. **사용자 맞춤형 기능 개발**
   - 특정 사용자 요구사항(서식 변경, 서명 삽입 등)에 대응할 수 있는 기능 확장.

4. **모달 확장**
   - 이미지 및 텍스트뿐만 아니라, 오디오와 동영상 같은 **다른 모달 데이터**와의 통합 연구.

5. **윤리적 대응 강화**
   - 모델의 악용 가능성을 방지하고, 신뢰할 수 있는 문서 생성 시스템을 구축하기 위한 윤리적 가이드라인 강화.

---

6. 결론 요약

UDOP는 **문서 처리의 패러다임을 전환**한 모델로, 텍스트, 이미지, 레이아웃 정보를 통합하여 다양한 작업을 수행할 수 있는 **범용적이고 강력한 모델**입니다.

1. **다중모달 학습**을 통해 기존 모델들의 한계를 극복하고,
2. **생성 기반 접근법**으로 다양한 작업을 처리할 수 있는 확장성과 유연성을 제공하며,
3. **최신 성능(State-of-the-Art)**을 달성하여 실질적인 응용 가능성을 입증했습니다.

하지만 윤리적 문제(문서 위조, 개인정보 보호 등)에 대한 고려와 함께, 다국어 지원, 경량화, 모달 확장과 같은 과제들이 앞으로 해결해야 할 중요한 연구 주제로 남아 있습니다.