GMN: Generative Multi-modal Network for Practical Document Information Extraction

Haoyu Cao[1], Jiefeng Ma[2]*, Antai Guo[1], Yiqing Hu[1], Hao Liu[1], Deqiang Jiang[1], Yinsong Liu[1], Bo Ren[1]

[1]Tencent YouTu Lab

[2]University of Science and Technology of China

{rechycao,ankerguo,hooverhu,ivanhliu,dqiangjiang}@tencent.com

jfma@mail.ustc.edu.cn {jasonysliu,timren}@tencent.com

Abstract

Document Information Extraction (DIE) has attracted increasing attention due to its various advanced applications in the real world. Although recent literature has already achieved competitive results, these approaches usually fail when dealing with complex documents with noisy OCR results or mutative layouts. This paper proposes Generative Multi-modal Network (GMN) for real-world scenarios to address these problems, which is a robust multi-modal generation method without predefined label categories. With the carefully designed spatial encoder and modal-aware mask module, GMN can deal with complex documents that are hard to serialized into sequential order. Moreover, GMN tolerates errors in OCR results and requires no character-level annotation, which is vital because fine-grained annotation of numerous documents is laborious and even requires annotators with specialized domain knowledge. Extensive experiments show that GMN achieves new state-of-the-art performance on several public DIE datasets and surpasses other methods by a large margin, especially in realistic scenes.

1 Introduction

Document Information Extraction (DIE) aims to map each document to a structured form consistent with the target ontology (*e.g.*, database schema), which has recently become an increasingly important task. Recent research (Xu et al., 2020, 2021; Wang et al., 2021a; Zhang et al., 2020; Li et al., 2021a) has achieved competitive results for information extraction in the idealized scenario with accurate OCR results, word-level annotations, and serialized document words in reading order. These methods regard DIE as a Sequence Labeling (SL) task. Given OCR results of a document image, the traditional sequence labeling method first serializes words in reading order then classifies each input word into predefined categories.

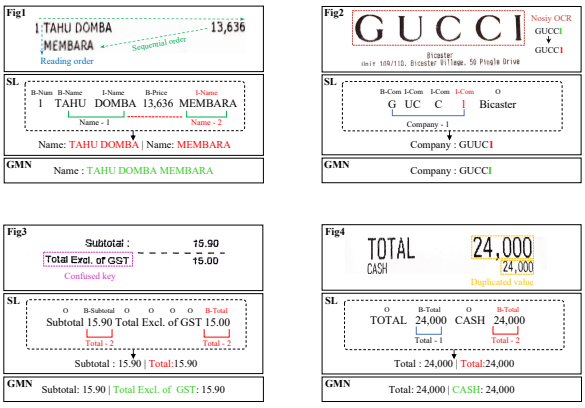


Figure 1: Examples in public DIE benchmarks with practical problems. The three rows from top to bottom in each sub-figure are 1) input images, 2) raw intermediate tags and final results generated by the SL method, and 3) results of our GMN method, respectively. The error parts are marked in red, while the correct parts are in green. Best viewed in color.

As shown in Figure 1, multiple challenging problems for practical document understanding still exist in realistic scenes. 1) Document serialization requires pre-composition processing, which is difficult in real scenarios with ambiguous word orders. One entity may be incorrectly divided into multiple entities when the input sequences are sorted by coordinates. 2) OCR results are usually noisy because of inevitable recognition errors. 3) The volume of keys in practical scenarios is generally substantial and expanded frequently. Existing sequence labeling methods could not identify undefined keys. 4) While facing duplicated values, collecting word-level annotations is necessary for sequence labeling methods. However, this is difficult in practical scenarios since they are costly and labor-intensive.

To address the limitations mentioned above, we propose a robust information extraction method

*Work is done during an internship at Tencent YouTu Lab.

Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 3768 - 3778

named Generative Multi-modal Network (GMN) for practical document understanding. Unlike sequence labeling methods that label each input word with a predefined category, we regard DIE as a translation task that translates source OCR results to a structured format (like key-value pairs in this paper). We use UniLM (Dong et al., 2019) as the basic model structure, which is a transformer-based pre-trained network that can handle both natural language understanding (NLU) and natural language generation (NLG) tasks simultaneously. Conditioned on a sequence of source words, GMN generates one word at each time step to compose a series of key-value pairs in them.

Regarding the sequence serialization problem, a novel two-dimensional position embedding method is proposed while the original one-dimensional positional embedding in the transformer is removed because all information in document understanding can be acquired from 2D layouts. In this manner, GMN bypasses the serialization problem. Furthermore, benefiting from the large-scale self-supervised pre-training processed on a vast document collection, GMN can correct OCR errors commonly encountered in practical scenarios. Moreover, using a weakly supervised training strategy that utilizes only key information sequences as supervision, GMN needs no word-level annotations that are indispensable in traditional sequence labeling methods like LayoutLM (Xu et al., 2020) and StructuralLM (Li et al., 2021a).

Experiments illustrate that the proposed GMN model outperforms several SOTA pre-trained models on benchmark datasets, including SROIE and CORD. The contributions of this paper are summarized as follows:

1) We present GMN tailored for the DIE task, which is more applicable for practical scenarios, including lack of word-level annotations, OCR errors as well as various layouts.

2) We propose a layout embedding method and multi-modal Transformer in a decoupling manner, which jointly models interactions between multiple modalities and avoids the reading order serialization.

3) Experiments on public DIE datasets demonstrate that the proposed method not only achieves a substantial performance improvement but also generalizes well to data under practical scenarios with unseen keys.

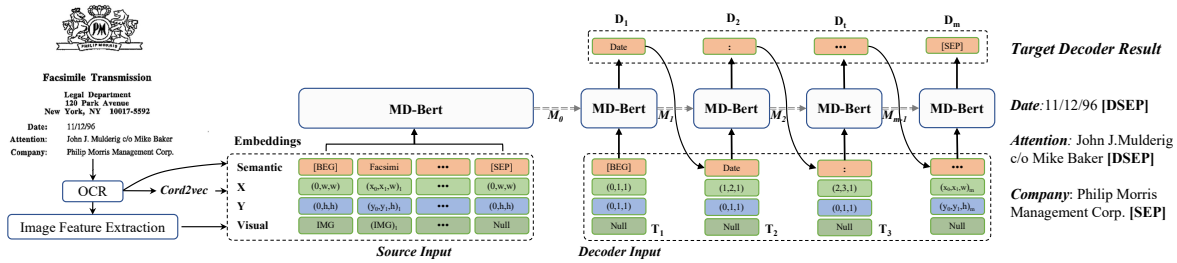
2 RELATED WORKS

Traditional methods (Esser et al., 2012; Schuster et al., 2013; Riloff, 1993) on DIE tasks rely heavily on predefined rules, templates, and handcrafted features, giving rise to difficulty in generalizing to unseen documents. With the development of deep learning technology, document information extraction methods have recently improved substantially in both performance and robustness. These deep learning-based methods can be classified into three categories: textual content-based methods, multi-modal-based methods, and pre-trained Transformer-based methods.

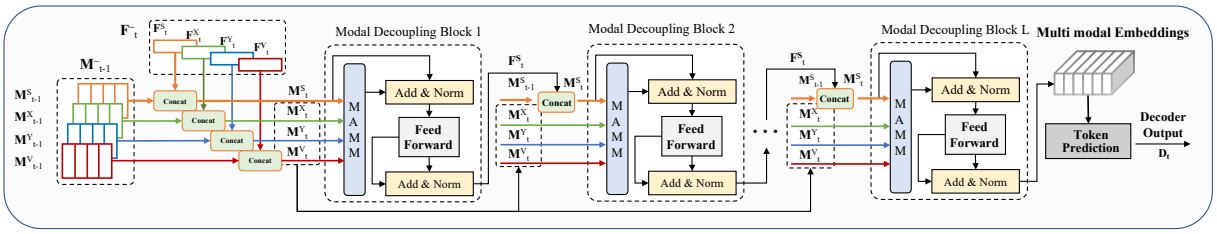
Textual content-based methods. Palm et al. (2017); Sage et al. (2019) adopt the idea from natural language processing and use recurrent neural networks (RNN) to extract entities of interest from documents. However, they discard the layout information during the text serialization, which is crucial for document understanding.

Multi-modal-based methods. Some works (Katti et al., 2018; Hwang et al., 2021) take the layout information into consideration and try to reconstruct character or word segmentation of the document. Katti et al. (2018) encode each document page as a two-dimensional grid of characters that represents text representation with a two-dimensional layout. Yu et al. (2020); Majumder et al. (2020); Zhang et al. (2020); Wang et al. (2021b) further integrate image embeddings for better feature extraction. Yu et al. (2020); Tang et al. (2021) represent documents by graphs, with nodes representing word segments and edges either connecting all the nodes or only spatially near neighbors. Convolutional or recurrent mechanisms are then applied to the graph for predicting the field type of each node. However, due to the lack of large-scale pre-training, the robustness and accuracy of the model are relatively limited.

Pre-trained Transformer-based methods. Recently, pre-trained models (Devlin et al., 2019; Liu et al., 2019) show effective knowledge transferability with large-scale training data and various self-supervised tasks. LayoutLM (Xu et al., 2020) first proposes a document-level pre-training framework that semantic and layout information are jointly learned. LayoutLM V2 (Xu et al., 2021) further improves the LayoutLM model by integrating the image information in the pre-training stage. Li et al. (2021a) propose the StructuralLM pre-training approach to exploit text block information. Methods



(a) The overall architecture of our generative multi-modal network, MD-Bert in encoder and decoder are with parameter sharing.



(b) Detail of MD-Bert module, with internal memory updated recursively.

Figure 2: Architecture of our generative multi-modal network. The MD-Bert module is composed of stacked multi-modal encoders, which fuses multi-modal features and iteratively generates structured results in a fixed order.

mentioned above all use one-dimensional position embeddings to model the word sequence, even with two-dimensional layout embeddings are involved, so that the reading order serialization in the document is required, which is challenging or even impossible due to the complex and diverse layout in the real world. What's more, they are all based on the classification of each input text segment to predefined labels, which means fine-grained annotations are indispensable and lack the ability to correct error OCR results.

On the contrary, the proposed GMN relies on a two-dimensional position embedding to bypass the serialization process and cross-modality encoders in a decoupling manner to model the layout information and the relative position of a word within a document simultaneously.

3 METHODOLOGY

In this section, we first introduce the overall architecture of GMN, followed by illustrating multi-modal feature extraction, generative pre-training model with multi-modal decoupling in detail, respectively.

3.1 Overall Architecture

GMN aims at constructing an enhanced Transformer-based translator architecture for DIE for converting the document to structured, machine-readable data. An overview of the architecture is as shown in Figure 2. It mainly consists of two parts: the multi-modal feature extraction module and stacked cross-modality module named MD-Bert (Modal Decoupling Bert), which simultaneously serves as encoder and decoder following the design of UniLM.

The whole process can be summarized as 1) Multi-modality embeddings of source inputs are extracted through an advanced OCR engine and a small CNN; 2) The extracted features from different modalities are fused as "multi-modal embeddings" through MD-Bert along with memory updating for each layer at each time step; 3) Next, MD-Bert output the encoding results by applying token prediction on multi-modal embeddings; 4) Finally, MD-Bert recursively generates structured results by taking multi-modal embeddings and accumulative memory as inputs until a terminator [SEP] is predicted.

3.2 Multi-Modal Feature Extraction

Based on the multi-modal information, including semantics, layout, and vision, we propose a unified layout embedding method named *Cord2Vec* which simultaneously encodes sequence information and spatial information to avoid complex reading order serialization.

3.2.1 Semantic Embedding

Intuitively, semantic contents are reliable signals to extract valuable information. The semantic content of each text fragment is acquired from the results of the OCR engine for practical application scenarios. After text fragments are acquired and tokenized, the start indicator tag [BEG] is added in front of the input token sequence, and the end indicator tag [SEP] is also appended to the end. Extra padding tag [PAD] is used to unify the length of sequence with predefined batch length L. In this way, we can get the input token sequence S as

$$S = [[\text{BEG}], t_1, \cdots, t_n, [\text{SEP}], [\text{PAD}], \cdots], |S| = L \quad (1)$$

Here, t_i refers to i-th token in OCR texts. Moreover, though the sequence length of the input token sequence is fixed during training, GMN can handle variable lengths when making the inference due to the novel positional embedding method.

3.2.2 Layout Embedding

DIE task is a typical two-dimensional scene in which relative positions of words are essential evidence. While the reading order serialization is challenging, we propose *Cord2Vec*, a unified embedding method that fully utilizes spatial coordinates rather than one-dimensional sequence order information to bypass this problem.

As for the source input part, we normalize and discretize all coordinates to the integer in the range of $[0, \alpha]$, here α is the max scale which is set to 1000 in our experiment. Then corner coordinates and edge lengths of each text fragment are gained using corresponding bounding boxes. In order to enhance the tokens' interaction in the same box, two tuples $(x_0, x_1, w), (y_0, y_1, h)$ are used to represent the layout information. Here, (x_0, y_0) and (x_1, y_1) are the top-left and bottom-right coordinates of each token, and w is the average width of tokens in the same box while h representing box height. Such embedding represents both the layout and the word order information. As for target tokens generated by GMN which does not have the real coordinate, the *Cord2Vec* assumes each token is tied in the grid of $[W_{grid}, H_{grid}]$ with row-first principle, and each token occupies a pixel with a width and height of 1. After the layout information is acquired, we use two embedding layers to embed x-axis features and y-axis features separately as stated in Equation 2.

$$\begin{aligned} X_i &= \textit{PosEmb2D}_{\text{x}}(x_0, x_1, w), \\ Y_i &= \textit{PosEmb2D}_{\text{y}}(y_0, y_1, h) \end{aligned} \quad (2)$$

Here, $\textit{PosEmb2D}_{\text{x}}$ and $\textit{PosEmb2D}_{\text{y}}$ are the position embedding function which takes coordinate as input. Each input element is embedded separately and then added together with an element-wise function. Note that the placeholder such as [PAD] can be treated as some evenly divided grids, so their bounding box coordinates are easy to calculate. An empty bounding box $X_{\text{PAD}} = (0, 0, 0), Y_{\text{PAD}} = (0, 0, 0)$ is attached to [PAD], and $X_{\text{SEP}} = (0, w, w), Y_{\text{SEP}} = (0, h, h)$ is attached to other special tokens including [BEG] and [SEP].

3.2.3 Visual Embedding

We use ResNet-18 (He et al., 2016) as the backbone of the visual encoder. Given a document page image I, it is first resized to $W * H$ then fed into the visual backbone. After that, the feature map is scaled to a fixed size by average-pooling with the width being W/n and height being H/n, n is the scaling scale. Finally, RoI Align (He et al., 2017) is applied to extract each token's visual embedding with a fixed size. The visual embedding of the i-th token is denoted by $v_i \in (v_1, v_2, v_3, \ldots, v_L)$. For source input, visual embedding can be represented as

$$v_i = \textit{ROIAlign}(\textit{ConvNet}(Image), Pos_i) \quad (3)$$

Here, Pos_i stands for the position of i-th token, and *ConvNet* is a convolutional neural network serving as feature extractor in terms of input image, and then *ROIAlign* takes the image feature and location as input, and extracts the corresponding image features. Note that the [BEG] token represents the full image feature, and the other special tokens, as well as output tokens, are attached to the default null image feature.

3.3 Generative Pre-training Model

3.3.1 Model Structure

In order to learn more general features and make full use of the pre-training data, we propose a unified encoder-decoder module named MD-Bert which is composed of stacked hierarchical multi-modal Transformer encoders. The context of input tokens is from OCR result during the encoding stage, while already decoded tokens are also included in the decoding stage. To solve this problem, inspired by UniLM, we use masking to control which part of context the token should attend to when computing its contextualized representation, as shown in Figure 3. The input features of semantics, layout and computer vision are mapped to hidden states by: $\boldsymbol{F}_0^S = \left\{f_1^S, \ldots, f_N^S\right\}, \boldsymbol{F}_0^X =$

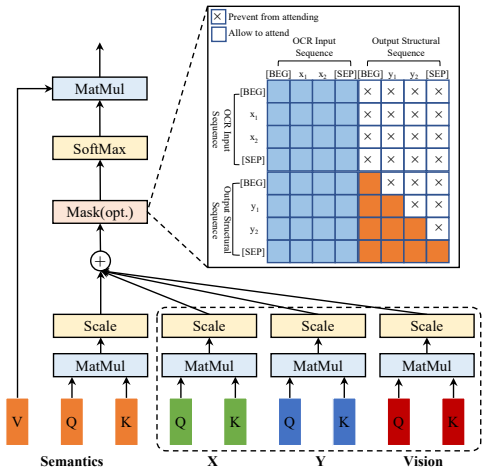


Figure 3: Modal-aware Mask Module: modalities fusion in decoupling manner. The sub-module for processing X are with parameter sharing for each layer, the same for Y and Visual sub-module.

$\{f_1^X, \ldots, f_N^X\}, \boldsymbol{F}_0^Y = \{f_1^Y, \ldots, f_N^Y\}, \boldsymbol{F}_0^V = \{f_1^V, \ldots, f_N^V\}$, a linear mapping is as follows:

$$\begin{aligned} f_i^S &= \boldsymbol{W}_S x_i^S, f_i^X = \boldsymbol{W}_X x_i^X, \\ f_i^Y &= \boldsymbol{W}_Y x_i^Y, f_i^V = \boldsymbol{W}_V x_i^V \end{aligned} \tag{4}$$

where matrices $\boldsymbol{W}_S \in \mathbb{R}^{d_h \times d_S}$, $\boldsymbol{W}_X \in \mathbb{R}^{d_h \times d_X}$, $\boldsymbol{W}_Y \in \mathbb{R}^{d_h \times d_Y}$, $\boldsymbol{W}_V \in \mathbb{R}^{d_h \times d_V}$ are used to project features into hidden-state in d_h dimensions. MD-Bert takes $F^{\sim}$ and memory $M^{\sim}$ of history state as input, generates output and update memory $M^{\sim}$ step by step. where, $F^{\sim} \in \{F^S, F^X, F^Y, F^V\}, M^{\sim} \in \{M^S, M^X, M^Y, M^V\}$, $M^{\sim}$ contains the history state of each layer and previous embeddings of the model. In the first timestep, $M^{\sim}$ is initialized from scratch, input T_0 means the full OCR result and MD-Bert acts as bi-directional encoder. For timestep t, where $t \in [1, m]$, the model takes the output of the previous timestep or $[BEG]$ as input, and outputs the current result, MD-Bert acts as uni-directional decoder.

3.3.2 Cross-Modality Encoder

Traditional multi-modal models usually fuse different modal features by adding or concatenating them together, which inevitably introduces the unwanted correlations between each other during self attention procedure, *e.g.* word-to-position, image-to-word. However, these correlations are harmful to strengthening the model's capability as different data modalities are practically orthogonal, thus we need to design a customized pipeline for each one.

We propose MAMM (modal-aware mask module) encoder, a hierarchical structure multi-modal Transformer model in a decoupling manner that jointly models different modalities. As a consequence, the feature embedding decoupling is decomposed into three embeddings in GMN. The MAMM module follows the design of a basic module in BERT, but replaces the multi-head attention with modal-aware multi-head attention.

It also contains feed-forward (FF) layers, residual connections, and layer normalizations (LN), meanwhile, parameters of modals are not shared. When feeded with different modal content such as semantics, layout and computer vision, MAMM first calculates each modal's attention score separately, then added these attention scores together to get a fusion score, finally use this fusion score to apply masking and following operations on semantic content. As shown in Figure 3, let $\boldsymbol{F}_l^{\sim} = \{f_1^{\sim}, \ldots, f_N^{\sim}\}$ be the encoded feature in the l-th layer. $\boldsymbol{F}_0^{\sim}$ is the vector of the input features as mentioned in Equation 4. Features output by the next layer $\boldsymbol{F}_{l+1}^{\sim}$ can be obtained via:

$$\boldsymbol{F}_{l-att}^{\sim} = LN\left(f_{MAMM}\left(\boldsymbol{F}_l^{\sim}\right) + \boldsymbol{F}_l^{\sim}\right) \tag{5}$$

$$\boldsymbol{F}_{l+1}^{\sim} = LN\left(f_{\mathrm{FF}}\left(\boldsymbol{F}_{l-att}^{\sim}\right) + \boldsymbol{F}_{l-att}^{\sim}\right) \tag{6}$$

where $f_{MAMM}(\cdot)$ is the modal-aware mask function defined as

$$f_{\mathrm{MAMM}}\left(\boldsymbol{F}_l^{\sim}\right) = softmax\left(MaskProd\left(\boldsymbol{F}_l^{\sim}\right)\right) v\left(\boldsymbol{F}_l^{\sim}\right) \tag{7}$$

$$MaskProd\left(\boldsymbol{F}_l^{\sim}\right) = \frac{q\left(\boldsymbol{F}_l^{\sim}\right) k\left(\boldsymbol{F}_l^{\sim}\right)^{\top}}{\sqrt[4]{d_k}} + f_{Maskopt} \tag{8}$$

where $q(\cdot)$, $k(\cdot)$, $v(\cdot)$ are linear transformation layers applied to the proposals' feature, which represent the query, key and value in attention mechanism accordingly. Benefited from the parameters' sharing among layers with regard to X, Y and Vision, GMN has comparable weights as Bert. Symbol d_k is the number of attention headers for normalization, and the $f_{Maskopt}$ is the Mask operation which controls the attention between each token. In GMN, we apply full attention on all OCR input tokens, and input tokens of the model for the output structural sequence can attend to the whole inputs as well as tokens that have been decoded which are like auto-regressive encoder. Finally, $\boldsymbol{F}_{l+1}^{\sim}$ can be obtained by $\boldsymbol{F}_{l-att}^{\sim}$ via a feed-forward sub-layer composed of two fully-att connected layers of function $f_{FF}(\cdot)$. Hierarchically stacked layers form the multi-modal encoder.

3.3.3 Pre-training Method

Similar to UniLM, three cloze tasks including Unidirectional LM, Bidirectional LM and Sequence-to-Sequence LM are used in the GMN. Meanwhile, we propose NER-LM for better entity correlation extraction. The whole loss function is defined as,

$$\mathcal{L} = \mathcal{L}_{\text{uni-LM}} + \mathcal{L}_{\text{bi-LM}} + \mathcal{L}_{\text{s2s-LM}} + \mathcal{L}_{\text{NER-LM}} \quad (9)$$

In a cloze task, we randomly choose some WordPiece (Wu et al., 2016) tokens in the input, and replace them with the special token $[MASK]$. Then, we feed their corresponding output vectors computed by the Transformer network into a softmax classifier to predict the masked token. The parameters of GMN are learned to minimize cross-entropy loss, which is computed using the predicted tokens and the original tokens.

NER-LM is an extension of sequence-to-sequence LM for better integrity constraints on the entity. Given the source segment which includes entity values s_1, s_2 , and the corresponding entity types n_1, n_2 as well as some background sentence *e.g.* b_1, b_2, we form the input format as A "$[BEG]s_1b_1s_2b_2[SEP]$" and B "$[BEG]n_1s_1n_2s_2[SEP]$". Each token in A can access all others of A, while each token in B can access all tokens of A as well as the preceded tokens in B. The target entity in B is masked for prediction during training.

4 EXPERIMENTS

4.1 Dataset

4.1.1 Pre-training Dataset

Our model is pre-trained on the IIT-CDIP Test Collection 1.0 (Lewis et al., 2006), which contains more than 6 million documents, with more than 11 million scanned document images. Moreover, each document has its corresponding text and metadata stored in XML files which describe the properties of the document, such as the unique identity and document labels. And the NER-LM is pre-trained on the Enron Email Dataset (Klimt and Yang, 2004), which contains 0.5 million emails generated by employees of the Enron Corporation. We follow the organization of the letter and generate the content on the image. The structured information in the letters acts as an entity, such as subject, date, *etc*.

4.1.2 Fine-tuning Datasets

We conduct experiments on three real-world public datasets, FUNSD-R, CORD and SROIE.

The FUNSD-R Dataset. FUNSD (Jaume et al., 2019) is a public dataset of 199 fully annotated forms, which is composed of 4 entity types (*i.e.* Question, Answer, Header and Other). The original dataset has both semantic entity extraction (EE) and semantic entity linking (EL) tasks. It's noteworthy that the linking between different entities are complicated, one header entity may have linking to several question entities with more answer entities linked.

To better evaluate the system performance in the multi-key scenario, we relabel the dataset in key-value pairs format to tackle EE and EL tasks simultaneously. We named the new dataset FUNSD-R, which contains 1,421 keys for training and 397 keys for testing. Meanwhile, there are 267 keys in the test set that have not appeared in the training set. FUNSD-R will be released soon.

The CORD Dataset. The CORD (Park et al., 2019) dataset contains 800 receipts for the training set, 100 for the validation set and 100 for the test set. The dataset defines 30 fields under 4 categories and the task aims to label each word to the right field.

The SROIE Dataset. SROIE (Huang et al., 2019) dataset contains 626 receipts for training and 347 receipts for testing. Each entity of the receipt is annotated with pre-defined categories such as company, date, address, and total.

To further investigate the capacity of our proposed method under more challenging scenarios, we expand "SROIE" and "CORD" datasets to "SROIE-S" and "CORD-S" by shuffling the order of text lines and keep the box coordinates to simulate complex layouts. The evaluation metric is the exact match of the entity recognition results in the F1 score.

4.2 Implementation Details

Model Pre-training. We initialize the weight of GMN model with the pre-trained UniLM base model except for the position embedding layer and visual embedding layer. Specifically, our BASE model has the same architecture: a 12-layer Transformer with 768 hidden sizes, and 12 attention heads. For the LARGE setting, our model has a 24-layer Transformer with 1,024 hidden sizes and 16 attention heads, which is initialized by the pre-

trained UniLM LARGE model. For unidir-LM and bidir-LM methods, we select 15% of the input tokens of sentence A for prediction. We replace these masked tokens with the $[MASK]$ token 80% of the time, a random token 10% of the time, and an unchanged token 10% of the time. For seq-to-seq LM and NER-LM, we select 15% tokens of the sentence B. The target of the token is the next token. Then, the model predicts the corresponding token with the cross-entropy loss.

In addition, we also add the two-dimensional position embedding and visual embedding for pre-training. Considering that the document layout may vary in different page sizes, we scale the actual coordinate to a "virtual" coordinate: the actual coordinate is scaled to have a value from 0 to 1,000, and rescale the images to the size of 512×512.

We train our model on 64 NVIDIA Tesla V100 32GB GPUs with a total batch size of 1,024. The Adam optimizer is used with an initial learning rate of 5e-5 and a linear decay learning rate schedule.

Task-specific Fine-tunings. We evaluate the model following the typical fine-tuning strategy and update all parameters in an end-to-end way on task-specific datasets. We arrange the source OCR result from top to bottom and left to right. In addition, we add the "$[DSEP]$" as the separator between text detection boxes. In SROIE and CORD datasets, we construct the target key-value pairs in a certain order due to the keys being limited. (*i.e.* company, date, address, total). In the FUNSD dataset, we organize the target key-value pairs from top to bottom and left to right. We add the ":" as the separator between key and value and "$[DSEP]$" as the separator between key-value pairs. The max source length parameter is set to 768 in the SROIE and CORD datasets and 1536 in the FUNSD-R datasets, so input sequences below max length will be padding to the same length. The model is trained for 100 epochs with a batch size of 48 and a learning rate of 5e-5. Note that, the annotations of all GMN results are the weakly-supervised label of sentence-level while other methods use word-level annotations.

4.3 Comparison to State-of-the-Arts

We compare our method with several state-of-the-arts on the FUNSD-R, SROIE and CORD benchmarks. We use the publicly available PyTorch models for BERT, UniLM and LayoutLM in all the experiment settings. The results of PICK (Yu et al., 2020), MatchVIE (Tang et al., 2021), BROS (Hong et al., 2021), StrucTexT (Li et al., 2021b), SPADE (Hwang et al., 2021) and DocFormer (Appalaraju et al., 2021) are obtained from the original papers.

Model	Precision	Recall	F1
Bert$_{LARGE}$	0	0	0
LayoutLM$_{LARGE}$	0	0	0
GMN $_{BASE}$	0.5264	0.4866	0.5057
GMN $_{LARGE}$	**0.5568**	**0.5116**	**0.5333**

Table 1: Model results on the FUNSD-R dataset, methods based on sequence labeling yield under scene with larger amount of keys.

Model	Precision	Recall	F1
BERT$_{BASE}$	0.9099	0.9099	0.9099
UniLM$_{BASE}$	0.9459	0.9459	0.9459
BERT$_{LARGE}$	0.92	0.92	0.92
UniLM$_{LARGE}$	0.9488	0.9488	0.9488
LayoutLM$_{LARGE}$	0.9524	0.9524	0.9524
LayoutLMv2$_{LARGE}$	0.9904	0.9661	0.9781
PICK	0.9679	0.9546	0.9612
MatchVIE	-	-	0.9657
BROS	-	-	0.9662
StrucTexT	0.9584	**0.9852**	0.9688
GMN$_{BASE}$	0.9853	0.9633	0.9741
GMN$_{LARGE}$	**0.9956**	0.9690	**0.9821**

Table 2: Model results on the SROIE dataset with Ground Truth Setting.

Results under scene with larger amount of keys. Table 1 shows the model results on the FUNSD-R dataset which is evaluated using entity-level precision, recall and F1 score. In the case of a large number of key categories, especially in the case that some categories have not appeared in the training set, the method based on sequence labeling yield, neither the Bert model, which only contains text modality nor the LayoutLM which also contains layout and visual modalities.

The best performance is achieved by the GMN_{LARGE}, where a significant improvement is observed compared to other methods. Note that, 67.25% of keys have not appeared in the training set, This illustrates that the generative method in GMN is suitable for scenes with a large number of keys.

Results with Ground Truth Setting. Under this setting, the ground truth texts are adopted as model input. As shown in Table 2 and Table 3, even using weakly supervised labels, our approach shows excellent performance on both SROIE and CORD, and yields new SOTA results, which indicates that GMN has a powerful representational capability and can significantly boost the performance on DIE tasks.

Model	Precision	Recall	F1
$BERT_{BASE}$	0.8833	0.9107	0.8968
$UniLM_{BASE}$	0.8987	0.9198	0.9092
$BERT_{LARGE}$	0.8886	0.9168	0.9025
$UniLM_{LARGE}$	0.9123	0.9289	0.9205
$LayoutLM_{LARGE}$	0.9432	0.9554	0.9493
$LayoutLMv2_{LARGE}$	0.9565	0.9637	0.9601
SPADE	-	-	0.925
DocFormer	**0.9725**	0.9674	0.9699
BROS	-	-	0.9728
GMN_{BASE}	0.9547	0.9576	0.9562
GMN_{LARGE}	0.9693	**0.9798**	**0.9745**

Table 3: Model results on the CORD dataset with Ground Truth Setting.

Model	SROIE-E2E		
	Precision	Recall	F1
$BERT_{LARGE}$	0.4066	0.3876	0.3969
$LayoutLM_{LARGE}$	0.4414	0.4236	0.4323
GMN_{BASE}	0.7324	0.7161	0.7242
GMN_{LARGE}	**0.7543**	**0.7334**	**0.7437**
Model	**CORD-E2E**		
	Precision	**Recall**	**F1**
$BERT_{LARGE}$	0.6313	0.6724	0.6512
$LayoutLM_{LARGE}$	0.6684	0.7086	0.6879
GMN_{BASE}	0.7840	0.8133	0.7984
GMN_{LARGE}	**0.8165**	**0.8368**	**0.8265**

Table 4: Model results on the SROIE and CORD datasets with End-to-End Setting.

Results with End-to-End Setting. We adopt Tesseract as OCR engine to get the OCR result of public datasets. It's worth noting that there are exist some OCR errors, the sequence labeling method can not handle, but in our GMN, the matching process between OCR results and ground truth is avoided thanks to the novel layout embedding method in an end-to-end training setting. The performances are shown in Table 4. Our method shows new state-of-the-art performance benefits from the ability to error correction. A detailed analysis of it is introduced in case studies B.

Results with Position Shuffle Setting. In order to verify the robustness of our two-dimensional embedding method, we apply a shuffling operation on boxes of the test dataset. As shown in Table 5, compared with models that have one-dimensional position embeddings, our method is more robust to input disruption with a big gap.

4.4 Ablation Study

An ablation study is conducted to demonstrate the effects of different modules in the proposed model. We remove some components to construct several comparable baselines on the CORD dataset. The statistics are listed in Table 6.

Model	SROIE-S		
	Precision	Recall	F1
$BERT_{BASE}$	0.0702	0.0490	0.0577
$LayoutLM_{BASE}$	0.7169	0.6880	0.7022
GMN_{BASE}	**0.9679**	**0.9424**	**0.9550**
Model	**CORD-S**		
	Precision	**Recall**	**F1**
$BERT_{BASE}$	0.1235	0.1384	0.1305
$LayoutLM_{BASE}$	0.6917	0.7139	0.7026
GMN_{BASE}	**0.9345**	**0.9488**	**0.9416**

Table 5: Model results on the SROIE-S and CORD-S datasets with Position Shuffle Setting.

Model	Precision	Recall	F1
GMN	**0.9693**	**0.9798**	**0.9745**
GMN w/o Image	0.9623	0.9608	0.9616
GMN w/o NER-LM	0.96	0.9585	0.9593
GMN w/o MAMM	0.9576	0.9547	0.9562

Table 6: Model results of different components of our method on the CORD dataset.

The "GMN w/o MAMM" means using the same multi-modal feature as LayoutLM. Compared with LayoutLM, MAMM brings about 1.82% improvement of F1, which verifies the validation of MAMM. The "GMN w/o Image" means removing the image feature extraction. Experiment results show that visual modality can also improve the performance. Moreover, with NER-LM considered, the performance of information extraction increases to 97.45%. Extended experiments including attention visualization analysis and case studies can refer to Appendix A and B.

5 CONCLUSION

In this work, we propose a Generative Multi-modal Network (GMN) for practical document information extraction. Since GMN is a generation method including no pre-defined label category, it supports scenes that contain unknown similar keys and tolerates OCR errors, meanwhile requires no character-level annotation. We conduct extensive experiments on publicly available datasets to validate our method, experimental results demonstrate that GMN achieves state-of-the-art results on several public DIE datasets, especially in the practical scenarios. Though our GMN significantly outperforms other DIE models, there still exists potential to be exploited as regard to practical scenarios. In order to cope with complicated layout information as well as ambiguous semantic representations, we argue that more attention should be paid to the modality embedding and interaction strategy, which has more opportunity to handle such difficult cases.

References

Srikar Appalaraju, Bhavan Jasani, Bhargava Urala Kota, Yusheng Xie, and R. Manmatha. 2021. Docformer: End-to-end transformer for document understanding. *ArXiv preprint*, abs/2106.11539.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 13042–13054.

Daniel Esser, Daniel Schuster, Klemens Muthmann, Michael Berger, and Alexander Schill. 2012. Automatic indexing of scanned documents: a layout-based approach. In *Document Recognition and Retrieval XIX, part of the IS&T-SPIE Electronic Imaging Symposium, Burlingame, California, USA, January 25-26, 2012, Proceedings*, volume 8297 of *SPIE Proceedings*, page 82970H. SPIE.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. 2017. Mask R-CNN. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2980–2988. IEEE Computer Society.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society.

Teakgyu Hong, Donghyun Kim, Mingi Ji, Wonseok Hwang, Daehyun Nam, and Sungrae Park. 2021. BROS: A Pre-trained Language Model Focusing on Text and Layout for Better Key Information Extraction from Documents. *ArXiv preprint*, abs/2108.04539.

Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and C. V. Jawahar. 2019. ICDAR2019 competition on scanned receipt OCR and information extraction. In *2019 International Conference on Document Analysis and Recognition, ICDAR 2019, Sydney, Australia, September 20-25, 2019*, pages 1516–1520. IEEE.

Wonseok Hwang, Jinyeong Yim, Seunghyun Park, Sohee Yang, and Minjoon Seo. 2021. Spatial dependency parsing for semi-structured document information extraction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 330–343, Online. Association for Computational Linguistics.

Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. 2019. FUNSD: A dataset for form understanding in noisy scanned documents. In *2nd International Workshop on Open Services and Tools for Document Analysis, OST@ICDAR 2019, Sydney, Australia, September 22-25, 2019*, pages 1–6. IEEE.

Anoop R Katti, Christian Reisswig, Cordula Guder, Sebastian Brarda, Steffen Bickel, Johannes Höhne, and Jean Baptiste Faddoul. 2018. Chargrid: Towards understanding 2D documents. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4459–4469, Brussels, Belgium. Association for Computational Linguistics.

Bryan Klimt and Yiming Yang. 2004. The enron corpus: A new dataset for email classification research. In *Machine Learning: ECML 2004, 15th European Conference on Machine Learning, Pisa, Italy, September 20-24, 2004, Proceedings*, volume 3201 of *Lecture Notes in Computer Science*, pages 217–226. Springer.

David D. Lewis, Gady Agam, Shlomo Argamon, Ophir Frieder, David A. Grossman, and Jefferson Heard. 2006. Building a test collection for complex document information processing. In *SIGIR 2006: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, Washington, USA, August 6-11, 2006*, pages 665–666. ACM.

Chenliang Li, Bin Bi, Ming Yan, Wei Wang, Songfang Huang, Fei Huang, and Luo Si. 2021a. StructuralLM: Structural pre-training for form understanding. 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)*, pages 6309–6318, Online. Association for Computational Linguistics.

Yulin Li, Yuxi Qian, Yuechen Yu, Xiameng Qin, Chengquan Zhang, Yan Liu, Kun Yao, Junyu Han, Jingtuo Liu, and Errui Ding. 2021b. Structext: Structured text understanding with multi-modal transformers. In *MM '21: ACM Multimedia Conference, Virtual Event, China, October 20 - 24, 2021*, pages 1912–1920. ACM.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *ArXiv preprint*, abs/1907.11692.

Bodhisattwa Prasad Majumder, Navneet Potti, Sandeep Tata, James Bradley Wendt, Qi Zhao, and Marc Najork. 2020. Representation learning for information

extraction from form-like documents. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6495–6504, Online. Association for Computational Linguistics.

Rasmus Berg Palm, Ole Winther, and Florian Laws. 2017. Cloudscan - A configuration-free invoice analysis system using recurrent neural networks. In *14th IAPR International Conference on Document Analysis and Recognition, ICDAR 2017, Kyoto, Japan, November 9-15, 2017*, pages 406–413. IEEE.

Seunghyun Park, Seung Shin, Bado Lee, Junyeop Lee, Jaeheung Surh, Minjoon Seo, and Hwalsuk Lee. 2019. Cord: A consolidated receipt dataset for post-ocr parsing.

Ellen Riloff. 1993. Automatically constructing a dictionary for information extraction tasks. In *Proceedings of the 11th National Conference on Artificial Intelligence. Washington, DC, USA, July 11-15, 1993*, pages 811–816. AAAI Press / The MIT Press.

Clément Sage, Alexandre Aussem, Haytham Elghazel, Véronique Eglin, and Jérémy Espinas. 2019. Recurrent neural network approach for table field extraction in business documents. In *2019 International Conference on Document Analysis and Recognition, ICDAR 2019, Sydney, Australia, September 20-25, 2019*, pages 1308–1313. IEEE.

Daniel Schuster, Klemens Muthmann, Daniel Esser, Alexander Schill, Michael Berger, Christoph Weidling, Kamil Aliyev, and Andreas Hofmeier. 2013. Intellix - end-user trained information extraction for document archiving. In *12th International Conference on Document Analysis and Recognition, ICDAR 2013, Washington, DC, USA, August 25-28, 2013*, pages 101–105. IEEE Computer Society.

Guozhi Tang, Lele Xie, Lianwen Jin, Jiapeng Wang, Jingdong Chen, Zhen Xu, Qianying Wang, Yaqiang Wu, and Hui Li. 2021. Matchvie: Exploiting match relevancy between entities for visual information extraction. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pages 1039–1045. ijcai.org.

Jiapeng Wang, Chongyu Liu, Lianwen Jin, Guozhi Tang, Jiaxin Zhang, Shuaitao Zhang, Qianying Wang, Yaqiang Wu, and Mingxiang Cai. 2021a. Towards robust visual information extraction in real world: New dataset and novel solution. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 2738–2745. AAAI Press.

Jiapeng Wang, Tianwei Wang, Guozhi Tang, Lianwen Jin, Weihong Ma, Kai Ding, and Yichao Huang. 2021b. Tag, copy or predict: A unified weakly-supervised learning framework for visual information extraction using sequences. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pages 1082–1090. ijcai.org.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *ArXiv preprint*, abs/1609.08144.

Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, Min Zhang, and Lidong Zhou. 2021. LayoutLMv2: Multi-modal pre-training for visually-rich document understanding. 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)*, pages 2579–2591, Online. Association for Computational Linguistics.

Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2020. Layoutlm: Pre-training of text and layout for document image understanding. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 1192–1200. ACM.

Wenwen Yu, Ning Lu, Xianbiao Qi, Ping Gong, and Rong Xiao. 2020. PICK: processing key information extraction from documents using improved graph learning-convolutional networks. In *25th International Conference on Pattern Recognition, ICPR 2020, Virtual Event / Milan, Italy, January 10-15, 2021*, pages 4363–4370. IEEE.

Peng Zhang, Yunlu Xu, Zhanzhan Cheng, Shiliang Pu, Jing Lu, Liang Qiao, Yi Niu, and Fei Wu. 2020. TRIE: end-to-end text reading and information extraction for document understanding. In *MM '20: The 28th ACM International Conference on Multimedia, Virtual Event / Seattle, WA, USA, October 12-16, 2020*, pages 1413–1422.

Appendix

A Attention Visualization

To further explore what context information is focused by our GMN, we visualize the attention map of the multi-head Transformer, as shown in Figure 4. The input tokens of the model are marked in black and the decoding results are marked in orange, while X-axis represents the attended tokens.

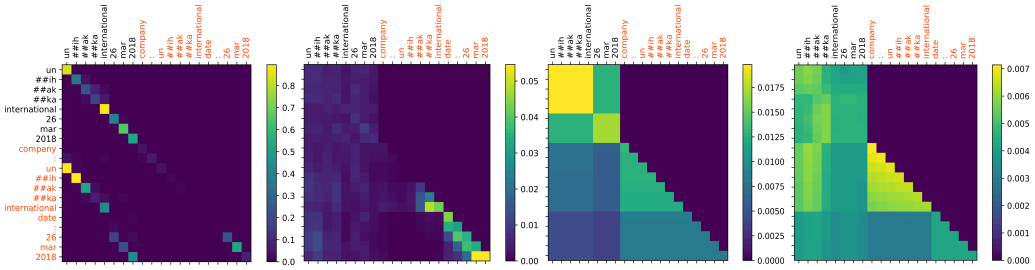


Figure 4: Visualization of the attention map of the multi-head Transformer, representing semantic/X-coord/Y-coord/visual attention results in a row respectively. The decoding result is marked in orange. Best viewed in color.

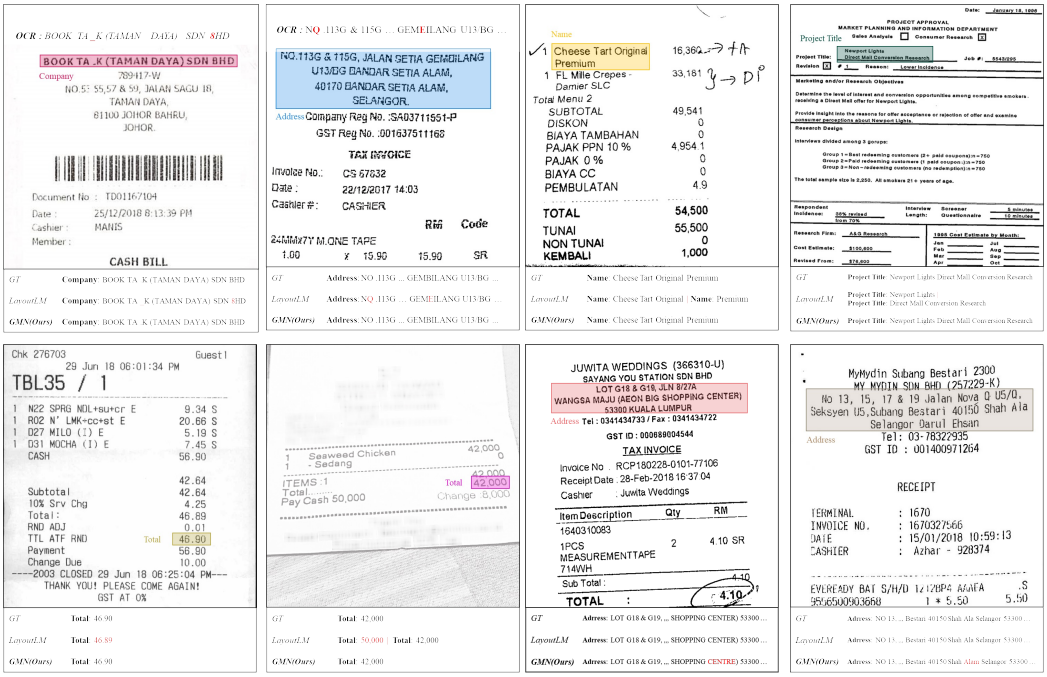


Figure 5: Samples from SROIE/CORD/FUNSD datasets, key examples are highlighted by color boxes. Best viewed in color.

As shown in Figure 3, we use the *Mask* operator to control the attention between each token. The input OCR tokens can attend to each other but the output tokens can only be attended to the already decoded tokens. Consequently, the upper right area of the attention map has no active response, and the area in the lower right corner shows a stepped pattern.

We can observe that the semantic attention mechanism plays an important role in modeling local dependence. In semantic attention, the input OCR tokens mainly focus on themselves and their nearby semantically relevant parts. In contrast, decoded tokens mostly focus on the counterparts in the original tokens, showing a reasonable alignment. Meanwhile, layout and visual attention mechanisms focus on more global information, complementing the semantic attention mechanism.

B Case studies

The motivation behind GMN is to tackle the practical DIE tasks. To verify this, we show some examples of the output of LayoutLM and GMN, as shown in Figure 5. In the sub-figure A and B, GMN successfully corrects the recognition error of OCR results thanks to semantic learning on a large-scale corpus. In the sub-figure C~F, GMN accurately generates the key-value pairs with complex layouts and ambiguous contexts thanks to the novel position embedding method, in comparison LayoutLM is unable to merge the value entities correctly. It's noteworthy that the sub-figures G and H are failed cases, which are caused by semantic obfuscation and reasonable complement to missing character. These examples show that GMN is capable of correcting OCR errors and predicting more accurately in practical scenarios.