A Simple yet Effective Learnable Positional Encoding Method for Improving Document Transformer Model

Guoxin Wang, Yijuan Lu, Lei Cui, Tengchao Lv, Dinei Florencio, Cha Zhang
Microsoft Corporation
{guow,yijlu,lecu,tengchaolv,dinei,chazhang}@microsoft.com

Abstract

Positional encoding plays a key role in Transformer-based architecture, which is to indicate and embed token sequential order information. Understanding documents with unreliable reading order information is a real challenge for document Transformer models. This paper proposes a simple and effective positional encoding method, learnable sinusoidal positional encoding (LSPE), by building a learnable sinusoidal positional encoding feed-forward network. We apply LSPE to document Transformer models and pretrain them on document datasets. Then we finetune and evaluate the model performance on document understanding tasks in form, receipt, and invoice domains. Experimental results show our proposed method not only outperforms other baselines, but also demonstrates its robustness and stability on handling noisy data with incorrect order information.

1 Introduction

Document understanding (in some context known as Document intelligence, Document AI) aims to extract, recognize, and understand information from document images. The performance of document understanding model is largely benefited from recent development of large scale pre-training technique on cross-modality data and effective transformer architectures (Cui et al., 2021). Document Transformer Model, e.g. LayoutLM (Xu et al., 2020b), is pretrained from visually-rich document data which consists of text, layout, and visual information based on Transformer architecture (Shaw et al., 2018). Recently, Xu et al. (2020a); Hong et al. (2021); Appalaraju et al. (2021); Li et al. (2021a) propose various approaches to further improve the performance of Transformer models on more challenging document understanding tasks.

Different from recurrent and convolutional based structures, Transformer based model does not encode relative or absolute position information explicitly since it is solely based on order-invariant attentional mechanism. In the original Transformer architecture (Vaswani et al., 2017), both learnable vector embedding and sinusoidal function are introduced as positional encoding methods for capturing positional information from input tokens. In order to improve positional representation ability, Shaw et al. (2018); Huang et al. (2020); He et al. (2021); Chi et al. (2021) introduce several relative position strategies into attention computation steps in the Transformer. Along with sequential reading order from text, visually-rich documents contain more spatial information of text blocks which poses a greater challenge to understand rich semantic and spatial relationship information at the same time. To obtain text blocks from document images, current off-the-shelf methods are borrowing results from existing Optical Character Recognition (OCR) engine while the reading order of text blocks is mostly arranged by a heuristic manner, top-to-bottom and left-to-right (Clausner et al., 2013; Wang et al., 2021). For documents with complex layout, such as forms, invoices, or receipts, the performance of reading order is not consistent which always leads to irrelevant or embarrassing predictions (Cui et al., 2021). Moreover, existing Document Transformer Models suffer from huge performance degradation on noisy data with unreliable reading order information (Hong et al., 2021). Therefore positional encoding plays an essential role in document Transformer models, which is to encode position embedding from data with inherent reading or spatial information. Thus, it is crucial to improve the robustness and learnability of position encoding methods, and therefore boost the model performance on noisy data with unreliable order and spatial information.

In this paper, we propose a learnable sinusoidal position encoding method, $LSPE$, by building a learnable fully connected feed-forward sinusoidal positional encoding network. And we apply it to

Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022, pages 453–463
November 20–23, 2022.

represent multidimensional position information in the document Transformer model. Compared with current discrete embedding layer in the Transformer model, our method is numeric continuous for position scales which could improve positional representation of relative positions or distances between spatial elements. Our approach keeps the advantage of extrapolability from sinusoidal function which could extend to longer position than training cases. In addition, we build a learnable sinusoidal position network, which helps the pretrained language model to be easily adapted to various downstream tasks effectively.

We pretrain LayoutLM with various positional encoding methods and other baselines. Then we evaluate and compare the models' performance on document understanding downstream tasks. Experimental results show that our $LSPE$ method significantly outperforms other baselines and recent document language models on FUNSD, SROIE and our in-house invoice datasets. In addition, we evaluate the model robustness on noisy data by utilizing global and local shuffling augmentation strategies. Our method shows stable performance than other positional encoding methods with unreliable reading order information. Furthermore, we visualize and analyze similarity of positional representation of each method from 1D to 2D positional embedding of our pretrained models.

In summary, our contributions could be highlighted as follows: 1) We propose a simple and effective learnable positional encoding method with better learnability and extrapolability. It can be applied to any transformer based models to help them better encode and understand positional information. 2) We pretrain document Transformer models with $LSPE$ and other methods, and evaluate model performance on document understanding tasks. Experimental results show our proposed method outperforms other baselines and recent SOTA approaches on FUNSD, SROIE, and a large-scale invoice dataset.3) By the ablation study of employing global and local block shuffling augmentations, our method demonstrates optimal performance and robustness on noisy data with unreliable reading order information. Finally, our pretrained models with implementation of position encoding code will be publicly available.[1]

[1] https://aka.ms/DocLPE

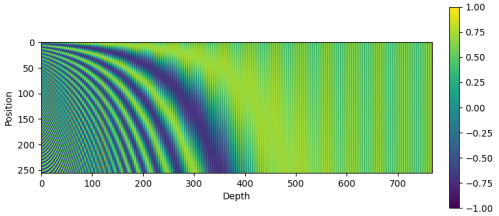


Figure 1: Visualization of 768-dimensional sinusoidal positional encoding for sequence with the maximum length of 256. Each position row p represents the embedding vector $PE_{sine}(p)$ as positional representation.

2 Background

Positional Encoding Methods in Transformer In the original proposal of Transformer architecture (Vaswani et al., 2017), both learnable vector and sinusoidal function are introduced as positional encoding methods and perform nearly identically in their downstream tasks. Although sinusoidal version with predefined wavelength has unique extrapolability which allows to encode longer sequential position than pre-training samples, it does not always perform well on downstream tasks (Shaw et al., 2018), due to the lack of learnability and flexibility. In practical, most pretrained language models, (e.g. (Devlin et al., 2018; Liu et al., 2019)), utilize learnable vector embedding (Gehring et al., 2017) as positional representation. Recently, several approaches are proposed to enhance positional representation by adding relative position information into attention score computation stage to improve performance of Transformer based models (Shaw et al., 2018; Huang et al., 2020; Dai et al., 2019; Dufter et al., 2021). By leveraging relative positional encoding and other advanced pretraining techniques, He et al. (2021) and Chi et al. (2021) achieve state-of-the-art performance on multiple nature language understanding tasks. Li et al. (2021b) explore the position encoding method in vision domain and propose a learnable Fourier feature to enhance positional encoding in Transformer. It outperforms other methods on both accuracy and convergence speed with vision transformer (Dosovitskiy et al., 2020) based model. Since it is non-trivial to modify or replace backbone of model structure during fine-tuning stage, some research works propose auxiliary tasks (Wang et al., 2019; Pham et al., 2021) or data augmentation approaches (Wei and Zou, 2019; Dai and Adel, 2020) to lever-

age absolute or relative position information without modifying model structure.

Document Transformer Models In document understanding area, LayoutLM (Xu et al., 2020b) utilizes the pretrained language model to resolve document understanding tasks, and achieves state-of-the-art performance on multiple document understanding benchmarks. To represent 2D position embedding, it decouples the x- and y- axes of text bounding box and sums up positional representations from each dimension independently. LayoutLMv2(Xu et al., 2020a) introduces spatial-aware self-attention mechanism to enhance the layout representation from both 1d and 2d relative position bias. BROS(Hong et al., 2021) uses relative position information in attentional mechanism along with absolute positional encoding from sinusoidal function, which perceives more spatial layout information. Li et al. (2021a) utilizes shared position information in the text blocks as position representation which further improves entity extraction performance by understanding cell information from layout. Appalaraju et al. (2021) proposes an End-to-End Transformer based model with 1D relative position embedding in attentional mechanism.

Document Understanding Tasks RVL-CDIP (Harley et al., 2015) is a document classification dataset with 400K gray-scale English document images in 16 document categories. This dataset is a subset of IIT-CDIP (Lewis et al., 2006) and has been widely used for pre-training language model purpose. Entity extraction is a classic and essential task in nature language understanding. It is to locate the boundary of entities and assign predefined classes to them. There are several popular benchmarks, consisting of multi-modality information with text, layout, and visual, to evaluate the performance of visually-rich document understanding. FUNSD (Guillaume Jaume, 2019) is a form understanding dataset for key-value extraction research [2] with 199 English forms. SROIE (Huang et al., 2019) and CORD (Park et al., 2019) are receipt understanding datasets to extract related entity types in English. XFUND (Xu et al., 2021) is an extended multi-lingual FUNSD dataset, which contains visually-rich documents in seven commonly-used languages.

[2]More license and term of use information at https://guillaumejaume.github.io/FUNSD/work/

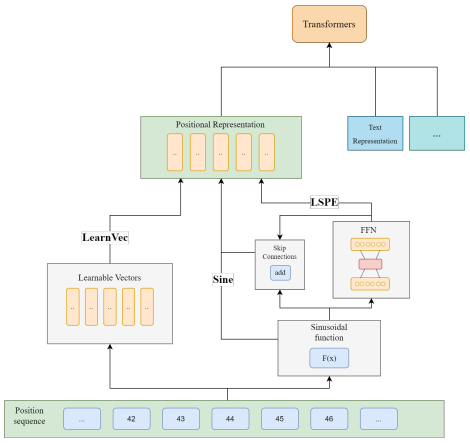


Figure 2: Flow of four positional encoding methods in Transformer based architecture: learnable vector embedding ($LearnVec$), sinusoidal positional encoding ($Sine$), learnable sinusoidal positional encoding ($LSPE$) and $LSPE_{SC}$ with skip connection structure.

3 Methodology

In this section, we formulate our positional encoding method $LSPE$ and introduce its applications on document transformer based language models. In order to evaluate its robustness and stability on noisy data with unreliable order information, we introduce two augmentation strategies: global and local text-block shuffling during fine-tuning stage.

3.1 Learnable Sinusoidal Positional Encoding

Positional representation is utilized as an inductive bias of positional relevance information by positional encoding function (PE) in Transformer model (Vaswani et al., 2017). Sinusoidal positional encoding is originally proposed and employed in attentional mechanism as better extrapolability and spatial correlation from the clean mathematical definition. Figure 1 shows the heatmap of sinusoidal positional encoding method. The hidden representation of position p in a sequence could be computed as Equation 1 for hidden dimension d, where D donates the size of positional representation:

$$
\begin{aligned}
PE_{sine}(p, 2d) &= sin\frac{p}{10000^{2d/D}} \\
PE_{sine}(p, 2d+1) &= cos\frac{p}{10000^{2d/D}}
\end{aligned}
\tag{1}
$$

In practical applications, some pretrained Transformer language models (Gehring et al., 2017; Devlin et al., 2018; Liu et al., 2019; Xu et al., 2020b; Dosovitskiy et al., 2020) treat each position index p as a discrete learnable embedding vector

($LearnVec$) by learning from pre-training and fine-tuning data. This approach is generic and effective to adapt pretrained Transformer models to specific domains and tasks with various behavior of spatial sensitivity. However, for more challenging tasks, such as document understanding tasks, the performance of document Transfomer models with existing positional encoding approach drops significantly on noisy data with unreliable order information (Hong et al., 2021).

We propose a learnable sinusoidal positional encoding ($LSPE$) method by building a fully connected feed-forward sinusoidal position network, which consists of two linear transformations with $GeLU$ (Hendrycks and Gimpel, 2020) as activation function σ in between as:

$$\begin{aligned} FFN(x) &= \sigma(xW_1 + b_1)W_2 + b_2 \\ PE_{LSPE}(p) &= FFN(PE_{sine}(p)) \end{aligned} \quad (2)$$

Skip connection is a generic strategy to sum the input and output representation from a computational unit with a skip edge. In transformer based models, (He et al., 2020) propose a residual attention layer, which has shown some regularization effects that could stabilize training and benefit fine-tuning stages. Inspired by this, we conduct the skip connection strategy in $LSPE$ module as a variant of our method. It could be formulated as eq.3.

$$PE_{LSPEsc}(p) = PE_{sine}(p) + PE_{LSPE}(p) \quad (3)$$

Figure 2 visualizes the flow of our proposed method and baselines in this paper. Compared with discrete embedding, our method extends from sinusoidal function and treats position index as a continuous-valued vector which allows the model to extrapolate to longer length from training cases. Meanwhile, the learnable FFN component boosts the learnability and flexibility of positional representation for multidimensional spatial information.

3.2 Positional Representation in Document Transformer Language Model

Distinct from nature language data which only consist of 1D order information, visually-rich document data require more model capacity to represent both 1D and 2D positional information from individual element. Given token x_i series from a document D, let p_i donate 1D position index and b_i as $((x_0, y_0), (x_1, y_1))$ present the bounding box in normalized 2D coordinate system.

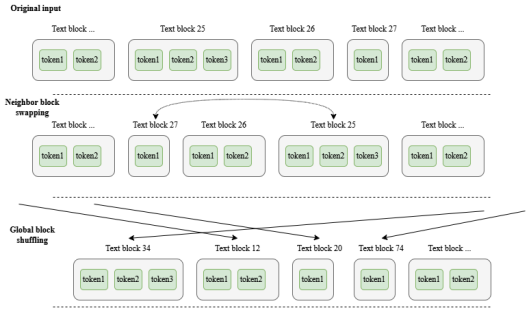


Figure 3: An example of text block shuffling augmentation methods, Neighbor Block Swapping and Global Block Shuffling.

.

As a general and commonly used pre-trained model for Document AI, LayoutLM (Xu et al., 2020b) utilizes independent 2D spatial embedding layers along with 1D position embedding initialized from pretrained BERT (Devlin et al., 2018) to represent positional information. Its composed positional representation R_i is computed via:

$$\begin{aligned} \mathcal{R}_i^{2D} &= \sum_{j=0}^{k} (PE_x(x_j) + PE_y(y_j)) \\ \mathcal{R}_i &= PE_{1d}(p_i) + \mathcal{R}_i^{2D} \end{aligned} \quad (4)$$

Where k donates the count of points in the bounding box, and PE_{1d}, PE_x, PE_y are the positional encoding methods for 1D order and 2D spatial information separately. The original positional encoding of LayoutLM is a learnable embedding which is identical to $PE_{LearnVec}$ at section 3.1 in this paper. The composed positional representation will be summed up with text embedding and token type embedding vectors as input of Transformer.

3.3 Text Block Shuffling Augmentations

In practical, understanding documents with incorrect reading order is a real challenge for document Transformer model which always leads to irrelevant or embarrassing error results. We introduce two text block shuffling augmentation methods: **Global Block Shuffling** and **Neighbor Block Swapping**, to simulate the noisy reading order scenario as shown in Figure 3. We apply these shuffling methods on text block level to a document, and keep the relative word order in the same text block. Text block is defined as a group of continual words in a spatial region (or a line of words).

In the **Global Block Shuffling**, we obtain the block information for each token, and shuffle the order of block index but keep the relative token

order of internal OCR line. In the **Neighbor Block Swapping**, each text block is swapped to its neighbor block randomly, and the distance d of swapped block pairs follows a normal distribution function $\mathcal{N}(0, \sigma^2)$.

The intuition of applying augmentation methods on text block level is to generate samples which are closed to error cases in real-world document understanding applications, and the text block information could be obtained from OCR engines.

4 Experiments

We apply four positional encoding methods ($LearnVec$, $Sine$, $LSPEsc$, $LSPE$) to a representative transformer based model: LayoutLM without visual feature. We conduct pretraining and finetuning on these models to identify the affect of different positional encodings to the performance of transformers on document understanding tasks.

4.1 Pretraining

We pretrain LayoutLM with four positional encoding method as well as baseline methods on a 1M random subset of IIT-CDIP (Lewis et al., 2006) pretraining data set. The name of positional encoding method is used to indicate the pretrained model in the result table.

All pretraining jobs run on 8 NVIDIA Tesla V100 32GB GPUs with approximately 150 hours for each job. The pretraining hyper-parameters are shown in Table 6. The pretrained models are initialized from Bert-base-uncased except for specified positional encoding weights.

4.2 Experimental Settings

Then we fine-tune and evaluate the performance of our pretrained models on three datasets: FUNSD (Guillaume Jaume, 2019), SROIE (Huang et al., 2019), and an In-house Invoice Dataset, which are benchmark datasets for entity extraction in form, receipt, and invoice domains.

FUNSD [3] consists of noisy scanned documents. There are 149 scanned forms for training and 50 scanned forms for testing with more than 31K words, 9.7K entities, and 5.3K relations in combination. For more fair comparison, we refer the evaluation results from LayoutLM, DocFormer, and BROS with the same text and spatial features as input and similar model size architecture. The evaluation result of LayoutLMv2 is conducted by the same settings of our methods but without visual feature inputs.

SROIE [4] attracts a lot of attention from both research and industry community as an open-source OCR and information extraction benchmark for receipt understanding. The dataset consists of 626 receipt images for training and 347 receipt images for testing with four predefined entities which are $company$, $date$, $address$, and $total$. There is no post-processing strategy before evaluation as we tend to compare the performance gap caused by different positional encodings only. We also experiment with official pretrained LayoutLM and LayoutLMv2[5] on the same fine-tuning hyper-parameters but without visual feature inputs for a fair comparison.

In-house Invoice Dataset To further evaluate the effectiveness of our positional encoding method on large scale document understanding tasks, we collect a large English inovice dataset with 24175 training and 643 testing invoices and 14 annotated fields. We test our approach on this in-house invoice dataset. (More detailed information of dataset and evaluation results are listed in Appendix A).

We use entity recognition evaluation metrics including entity-level precision, recall, and F1-score for each experiment with the default settings of seqeval package (Nakayama, 2018).

4.3 Experimental Results

As shown in Table 1, on FUNSD dataset, our $LSPE$ model achieves 82.04 F1-score and outperforms other baseline methods. The $Sine$ model achieves low performance and $LSPE_{SC}$ is worse than $LSPE$ which indicates the sinusoidal function cannot represent layout positional information with skip connection structure. The small performance gap between our $LearnVec$ and official LayoutLM model with shared model structure might be from different pretraining data and settings since our pretraining experiments run on a 1M subset training data and fewer pretraining steps.

We observe similar trend on SROIE as shown in Table 2. $LSPE$ model achieves F1 score of 93.87 with text and spatial features. With larger scale of training size on SROIE, the performance gap is narrowed down between $LearnVec$ and $LSPE$ in the testing dataset.

These results illustrate the effectiveness of our $LSPE$ on document understanding tasks with dif-

[3] https://guillaumejaume.github.io/FUNSD

[4] https://github.com/zzzDavid/ICDAR-2019-SROIE

[5] https://github.com/microsoft/unilm/tree/master

ferent data scale. And the ability of positional representation affects the final performance significantly on document understanding models.

Method	P(%)	R(%)	F1(%)
$LayoutLM$(2020b)	75.97	81.55	78.66
$DocFormer$(2021)	77.63	83.69	80.54
$BROS$(2021)	80.56	81.88	81.21
$LayoutLMv2_{base}$(2020a)	80.26	83.26	81.73
$LearnVec$	75.97	80.04	77.95
$Sine$	72.8	77.24	74.95
$LSPE_{SC}$	78.25	82.79	80.46
$LSPE$	80.4	83.74	**82.04**

Table 1: Entity level evaluation results on FUNSD dataset. All models utilize input features of text and spatial information with "Base" model size architecture. The evaluation result of $LayoutLMv2$ is reproduced without visual inputs.

Method	P(%)	R(%)	F1(%)
$LayoutLM_{base}$	91.4	94.24	92.8
$LayoutLMv2_{base}$	92.3	94.16	93.22
$LearnVec$	92.57	94.31	93.43
$Sine$	87.72	90.06	88.87
$LSPE_{SC}$	89.89	92.87	91.35
$LSPE$	92.94	94.81	**93.87**

Table 2: Results on SROIE datasets. All above experiments are fine-tuned with the same hyper-parameter setting and training environments. We evaluate the official $LayoutLM_{base}$ and $LayoutLMv2_{base}$ on the same settings without visual features.

4.4 Ablation Study

In real-world application, the reading order of text blocks is not always reliable and consistent. The incorrect reading order harms the performance of existing document language models and leads to embarrassing error of predictions in downstream tasks. We conduct three ablation experiments to simulate the impact of such error with the following augmentation methods.

Neighbor Block Swapping and Global Block Shuffling We apply these methods to training data only during fine-tuning which simulates impact of incorrect block order data. The testing set is kept as original which allows us to compare the performance with original reading order in Table 1. The σ of neighbor block swapping is set to 1 in all experiments. Note that the augmentation methods in this paper require block information of each token, and that might cause leaking of block boundary information during the model training indirectly. Besides of data impact, the model receives inconsistent reading order during training and it might benefit the evaluation performance by eliminating the over-fitting from 1D positional embedding, and tent to learn more information of relative token order inside block and 2D spatial information.

In Table 3, with synthetic noisy data generated by two augmentation methods, our $LSPE$ method shows better performance than existing discrete $LearnVec$ embedding and sinusoidal function $Sine$ consistently on FUNSD data. Similar observations can be found on the In-house Invoice dataset in Appendix A. The global block shuffling is harmful for all positional encoding methods while the performance impact of neighbor block swapping is marginally. The discrete positional encoding method shows more sensitive with significant performance drop by global block shuffling augmentation.

Removing 1D Position Input We throw the 1D positional input and only consider the 2D positional representation $\mathcal{R}^{2D}$ in eq. 4 in composed positional representation for both training and testing datasets. The model does not receive word order information on both text block and sub-token levels. We refer the performance result from BROS (Hong et al., 2021) with similar settings for comparison.[6]

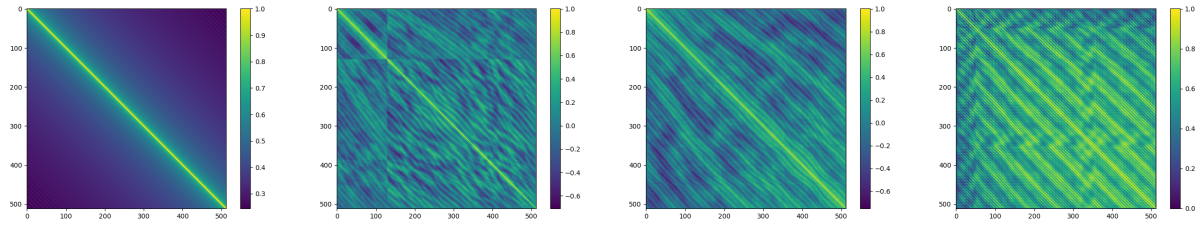
On FUNSD dataset, we observe a significant performance degradation across all positional methods in Table 4. The $LearnVec$ leads a huge drop from approximately 79% to 49% on F1 score which indicates the discrete 2D embedding is not well represented without optimal order information. The continuous 2D positional encoding methods perform better relatively. $LSPE_{SC}$ performs the best with only 2.67% F1 drop, and keeps a reasonable performance even with none order information.

From Table 5, we observe our $LSPE$ model achieves 89.98 F1 score with 3.89% absolute drop (4.14% relatively) from Table 2. The performance of $LSPE_{SC}$ drops 3.2% relatively which shows better robustness on such extreme condition. There is significant performance regression with discrete $LearnVec$ method on this receipt understanding data set. The $LSPE_{SC}$ performs better with global block shuffling method on the FUNSD dataset which might be beneficial from regularization ad-

[6] Result from text line in their ablation study paragraph

Method	Neighbor Block Swapping			Global Block Shuffling		
	P(%)	**R**(%)	**F1**(%)	**P** (%)	**R**(%)	**F1** (%)
$LearnVec$	76.43	79.49	77.93	72.32	69.78	71.03
$Sine$	73.77	78.24	75.94	74.1	74.99	74.54
$LSPE_{SC}$	78.72	81.79	80.23	77.09	80.14	**78.59**
$LSPE$	79.9	82.14	**81.01**	78.03	78.34	78.18

Table 3: Comparison on **FUNSD** dataset for four positional encoding methods by applying **Neighbor Block Swapping** and **Global Block Shuffling** on training dataset. Evaluation results clearly demonstrate our methods show stable and robustness under unreliable order information.



(a) $Sine$ (b) official $BERT$ (c) $LearnVec$ (d) $LSPE$

Figure 4: Similarity of 1D position embedding from pretrained $Sine$, official BERT, $LearnVec$, $LSPE$ models.

vantage of skip connection structure. Similar observations can be found on the In-house Invoice dataset in Appendix A.

Ablation study results further prove that better learnability and spatial correlation of positional representation are essential factors of existing document Transformer model. By comparing with other positional encoding methods and other recent pretrained Transformer based solutions, our methods demonstrate optimal performance and robustness on noisy data with unreliable order information.

Method	**P**(%)	**R**(%)	**F1**(%)
$BROS$(2021)	–	–	70.07
$LearnVec$	44.66	54.63	49.14
$Sine$	69.4	73.74	71.5
$LSPE_{SC}$	75.71	79.99	**77.79**
$LSPE$	72.2	77.19	74.61

Table 4: Experimental results by removing 1D position inputs on training and testing sets of FUNSD. The BROS performance is referenced from their ablation study with similar experimental setting.

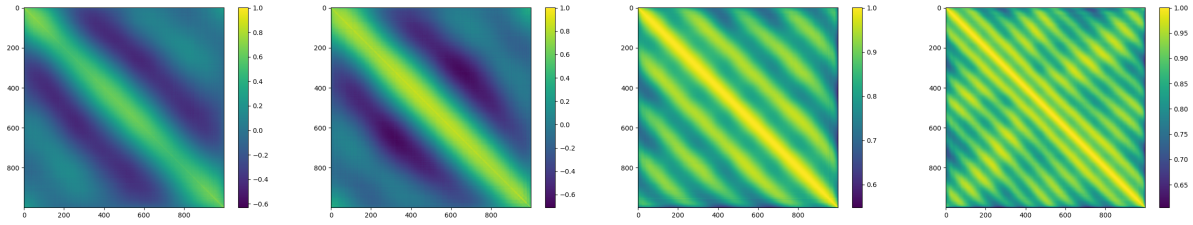
5 Position Embedding Similarity Analysis

To further investigate what Transformer encoders capture about positions after pretraining, we visualize the position-wise cosine similarity of each position embedding (Wang and Chen, 2020) in the pretrained models. Figure 4 shows the position-wise cosine similarity of 1D position embedding in our pretrained models with $Sine$, $LearnVec$, $LSPE$ and in the official BERT model. The point at (i, j) indicates the similarity between the i-th position and the j-th position. (i and j are from 0 to 512). First, with regard to $Sine$, we can only observe that embedding vectors are similar to the positions nearby. Both Bert and $LearnVec$ can observe similar embedding vectors nearby, but have no or very limited explainable patterns in long-term relations. Our $LSPE$ shows obvious periodic patterns along with position orders, which displays its embedding can actually capture the meanings of positions in the long-term relations.

Method	**P**(%)	**R**(%)	**F1**(%)
$LearnVec$	75.12	79.18	77.1
$Sine$	83.71	87.03	85.34
$LSPE_{SC}$	87.46	89.41	88.42
$LSPE$	87.9	92.15	**89.98**

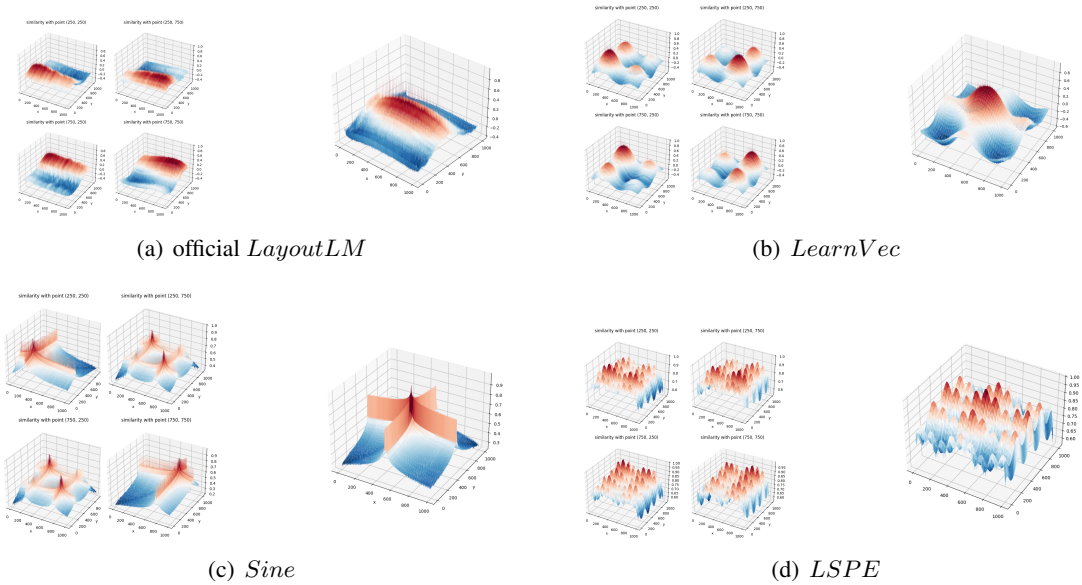
Table 5: Experimental results by removing 1D position inputs on training and testing sets of SROIE. The $LSPE$ achieves best performance and $LSPE_{SC}$ keeps lowest relative performance drop with this extra settings.

The 2D positional representation plays an essential role in document Transformer models with spatial information. Figure 5 shows position-wise cosine similarity of each position embedding of



(a) PE_x of $LearnVec$ (b) PE_y of $LearnVec$ (c) PE_x of $LSPE$ (d) PE_y of $LSPE$

Figure 5: Similarity of x and y axes in 2D positional embedding from our pretrained $LearnVec$ and $LSPE$ models.



(a) official $LayoutLM$ (b) $LearnVec$

(c) $Sine$ (d) $LSPE$

Figure 6: Similarity of 2D positional representation on 5 fixed points ((250, 250), (250, 750), (750, 250), (750, 750), (500, 500)) to the rest positions from official LayoutLM and $LearnVec$, $Sine$, $LSPE$ positional encoding methods.

x- and y- axes in 2D positional embedding in our petrained $LearnVec$ and $LSPE$ models. We can find our $LSPE$ has obvious periodic patterns along with both x- and y- position orders in the long-term relations than the $LearnVec$, which can mostly capture similar embeddings nearby.

Figure 6 demonstrates the position-wise cosine similarity of $\mathcal{R}^{2D}$ representation of five specific points to the rest positions in our pretrained models and in the official LayoutLM. $Sine$ captures close similar embeddings only, where its 2D similary map decays rapidly from central point and shows sharp edge on the border. The official LayoutLM model shows boarder vision horizontally with proper spatial correlation, but still fail to capture long-term relations. Our $LSPE$ shows higher wave frequency on both x- and y- axes which tend to capture the long distance signals with obvious periodic pattern.

6 Conclusions

In this paper, we propose a simple but effective learnable positional encoding method $LSPE$ to improve the positional representation in Transformer based models. By building a sinusoidal position feed-forward network, our method has better learnability and extrapolability in position representation. Experimental results on FUNSD, SROIE and an in-house Invoice datasets clearly show the effectiveness of our method on document understanding tasks. By leveraging global and local shuffling augmentation methods and removing order information from inputs, we demonstrate our method substantially outperforms other positional encoding methods on noisy data with unreliable reading order.

For future research, we will employ and evaluate our method on other tasks or modalities such as Vision Transformer (Dosovitskiy et al., 2020).

References

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

Zewen Chi, Shaohan Huang, Li Dong, Shuming Ma, Saksham Singhal, Payal Bajaj, Xia Song, and Furu Wei. 2021. Xlm-e: Cross-lingual language model pre-training via electra.

Christian Clausner, Stefan Pletschacher, and Apostolos Antonacopoulos. 2013. The significance of reading order in document recognition and its evaluation. In *2013 12th International Conference on Document Analysis and Recognition*, pages 688–692. IEEE.

Lei Cui, Yiheng Xu, Tengchao Lv, and Furu Wei. 2021. Document ai: Benchmarks, models and applications. *arXiv preprint arXiv:2111.08609*.

Xiang Dai and Heike Adel. 2020. An analysis of simple data augmentation for named entity recognition.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Philipp Dufter, Martin Schmitt, and Hinrich Schütze. 2021. Position information in transformers: An overview. *Computational Linguistics*, pages 1–31.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. In *International Conference on Machine Learning*, pages 1243–1252. PMLR.

Jean-Philippe Thiran Guillaume Jaume, Hazim Kemal Ekenel. 2019. Funsd: A dataset for form understanding in noisy scanned documents. In *Accepted to ICDAR-OST*.

Adam W Harley, Alex Ufkes, and Konstantinos G Derpanis. 2015. Evaluation of deep convolutional nets for document image classification and retrieval. In *International Conference on Document Analysis and Recognition (ICDAR)*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention.

Ruining He, Anirudh Ravula, Bhargav Kanagal, and Joshua Ainslie. 2020. Realformer: Transformer likes residual attention. *arXiv preprint arXiv:2012.11747*.

Dan Hendrycks and Kevin Gimpel. 2020. Gaussian error linear units (gelus).

Teakgyu Hong, DongHyun Kim, Mingi Ji, Wonseok Hwang, Daehyun Nam, and Sungrae Park. 2021. {BROS}: A pre-trained language model for understanding texts in document.

Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and CV Jawahar. 2019. Icdar2019 competition on scanned receipt ocr and information extraction. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1516–1520. IEEE.

Zhiheng Huang, Davis Liang, Peng Xu, and Bing Xiang. 2020. Improve transformer models with better relative position embeddings.

David Lewis, Gady Agam, Shlomo Argamon, Ophir Frieder, David Grossman, and Jefferson Heard. 2006. Building a test collection for complex document information processing. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 665–666.

Chenliang Li, Bin Bi, Ming Yan, Wei Wang, Songfang Huang, Fei Huang, and Luo Si. 2021a. Structurallm: Structural pre-training for form understanding. *arXiv preprint arXiv:2105.11210*.

Yang Li, Si Si, Gang Li, Cho-Jui Hsieh, and Samy Bengio. 2021b. Learnable fourier features for multi-dimensional spatial positional encoding.

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 arXiv:1907.11692*.

Hiroki Nakayama. 2018. seqeval: A python framework for sequence labeling evaluation. Software available from https://github.com/chakki-works/seqeval.

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. In *Workshop on Document Intelligence at NeurIPS 2019*.

Thang M. Pham, Trung Bui, Long Mai, and Anh Nguyen. 2021. Out of order: How important is the sequential order of words in a sentence in natural language understanding tasks?

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Wei Wang, Bin Bi, Ming Yan, Chen Wu, Zuyi Bao, Jiangnan Xia, Liwei Peng, and Luo Si. 2019. Structbert: Incorporating language structures into pre-training for deep language understanding.

Yu-An Wang and Yun-Nung Chen. 2020. What do position embeddings learn? an empirical study of pre-trained language model positional encoding. *arXiv preprint arXiv:2010.04903*.

Zilong Wang, Yiheng Xu, Lei Cui, Jingbo Shang, and Furu Wei. 2021. Layoutreader: Pre-training of text and layout for reading order detection.

Jason Wei and Kai Zou. 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv preprint arXiv:1901.11196*.

Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, et al. 2020a. Layoutlmv2: Multi-modal pre-training for visually-rich document understanding. *arXiv preprint arXiv:2012.14740*.

Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2020b. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1192–1200.

Yiheng Xu, Tengchao Lv, Lei Cui, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, and Furu Wei. 2021. Layoutxlm: Multimodal pre-training for multilingual visually-rich document understanding.

A More Information on Training Hyperparameters

Pretraining We use PyTorch on Nvidia Tesla V100 GPU for all pretraining experiments. The training hyperparameters is listed in Table 6.

Finetuning For finetuning tasks, we use standard cross-entropy loss on the task-specific classification head layers over pretrained document transformer model outputs. To make fair comparisons on various positional encoding methods, we use same hyperparameters, same training data, and same running environment for each task. The learning rate is set to 3e-5 for FUNSD and 2e-5 for SROIE task with linear decay, and 10% of total steps are used for warm-up purpose. We use max_steps as $2k$ for FUNSD and $1.5k$ for SROIE task, and report the evaluation results on the finetuned models. We average evaluation results with different initial seeds to eliminate bias of shuffling augmentations.

Parameter Name	Value
max_steps	500K
per_device_train_batch_size	12
gradient_accumulation_steps	4
max_seq_length	512
max_2d_position_embeddings	1024
learning_rate	7e-5
warmup_ratio	0.1
fp16	true
fp16_backend	amp
fp16_opt_level	O1

Table 6: Pretraining hyperparameters for document Transformer model with our positional encoding methods.

Field Name	Training entity count	Testing entity count
BillingAddress	7515	198
CustomerAddress	19317	529
CustomerID	24927	643
DueDate	16319	701
InvoiceDate	26043	676
InvoiceNumber	21441	558
PONumber	2106	56
ShippingAddress	2486	74
Subtotal	6207	169
TotalInvoiceAmount	31075	853
TotalTax	11178	308
VendorAddress	29811	787
VendorName	45685	1208

Table 7: Per field statistics of Invoice dataset.

B Evaluation Result of In-house Inovice Dataset

To further analyze the effectiveness of various positional encoding methods on larger scale document understanding tasks, we collect a large English inovice dataset with 14 fields listed in Table 7. There are 24175 and 643 invoice documents in its training and testing sets.

We finetune the same pretrained document Transformer models from section 4.1 with $LearnVec$ and $LSPE$ positional encoding methods on this invoice dataset, and report their F1-Score in Table 8 with various 1D position inputs. We also apply global and neighbor shuffling augmentation methods on the training dataset from section 3.3. Then we evaluate the F1-Score performance on the testing dataset. $LSPE$ model shows consistent evaluation result and outperforms the baseline method on the original position inputs, no positional inputs, and various shuffling augmentation methods. The evaluation result clearly illustrates the effectiveness and robustness of $LSPE$ on handling unreliable reading order issues.

Model	**Original 1D Position**	**No 1D Position**	**Global Shuffling**	**Neighbor Swapping**
$LearnVec$	91.66	86.55	87.09	90.39
$LSPE$	92.17	92.27	92.16	91.71

Table 8: F1-Score comparison on the in-house **Invoice** testing dataset for two positional encoding methods, $LearnVec$ and $LSPE$, with **Original 1D Position**, **No 1D Position** inputs and applying **Neighbor Block Swapping** and **Global Block Shuffling** on the training data set.