Large Sequence Representation Learning via Multi-Stage Latent Transformers

Ionut-Catalin Sandu  Daniel Voinea  Alin-Ionut Popa
Amazon Inc.
{sannion, dvoinea, popaaln}@amazon.com

Abstract
We present LANTERN, a multi-stage transformer architecture for named-entity recognition (NER) designed to operate on indefinitely large text sequences (i.e. \( \gg 512 \) elements). For a given image of a form with structured text, our method uses language and spatial features to predict the entity tags of each text element. It breaks the quadratic computational constraints of the attention mechanism by operating over a learned latent space representation which encodes the input sequence via the cross-attention mechanism while having the multi-stage encoding component as a refinement over the NER predictions. As a proxy task, we propose RADAR, an LSTM classifier operating at character level, which predicts the relevance of a word with respect to the entity-recognition task. Additionally, we formulate a challenging novel NER use case, nutritional information extraction from food product labels. We created a dataset with 11,926 images depicting food product labels entitled TREAT dataset, with fully detailed annotations. Our method achieves superior performance against two competitive models designed for long sequences on the proposed TREAT dataset.

I. Introduction. Information extraction from images and unstructured text plays a key role in natural language understanding (NLU) with direct impact in domains such as news content synthesis (Ozsoy et al., 2011; Moratanch and Chitrakala, 2017; Foong et al., 2015), query search (Choi et al., 2018) or knowledge-base systems (Agirre et al., 2018). The transformer architecture (Vaswani et al., 2017) played a pivotal role in advancing the state-of-the-art due to its robustness with respect to tasks related to sequential data manipulation and understanding. Its success relies mostly in the attention mechanism, considered as a key component for its versatility and its ability to self-specialize and filter the input information flow.

At the same time, the attention is a curse for transformers, as it scales quadratically with respect to the size of the input sequence. To overcome this limitation, we propose a sequence parsing framework for textual information understanding allowing for indefinitely large input sequences, with focus on the NER task. Our method is entitled LANTERN (i.e. LAarge sequeNce TransformER for NER). This is achieved via a multi-stage transformer approach over a latent space representation of the input sequence inspired from (Jaegle et al., 2021a) and adapted to NER. The key innovation we propose is a framework which analyses the entire sequence via a latent space attention-based modelling and leverages a multi-stage prediction setup.

II. Related Work. The transformer based architecture led to significant advances across major areas of NLU such as text classification (Lin et al., 2021), question answering (Choi et al., 2018), sequence to sequence translation (Rae et al., 2019; Lewis et al., 2019) and sentiment analysis (Zhang et al., 2018).

Pretraining (Yang et al., 2019; Devlin et al., 2018; Brown et al., 2020) proved extremely useful for generic language understanding tasks. These are unsupervised learning models which produce a vocabulary representation of the unlabelled training corpus via a masked language modelling task. One such model is Bidirectional Encoder Representation from Transformers (BERT) (Devlin et al., 2018). Given the pretrained weights of the model obtained from a large unlabelled data corpus, it has proven quite impactful in terms of peak performance and versatility for different downstream linguistic tasks, such as question answering (Garg et al., 2020; Laskar et al., 2020), NER (Xu et al., 2020b,a; Li et al., 2021; Zhang et al., 2020; Hwang et al., 2020; Lee et al., 2022; Huang et al., 2022; Sandu et al., 2022) or sentence / document classification (Lin et al., 2021).

Most of these approaches are constrained by
the quadratic computational limitation of the attention mechanism (sequences of up to 512 elements), thus bounded to subcontext. Several approaches (Dai et al., 2019; Goyal et al., 2021; Beltagy et al., 2020; Zaheer et al., 2020; Ainslie et al., 2020) considered modelling the sequence as a whole from the prism of the attention mechanism, even for sequence lengths $\gg 512$.

In (Beltagy et al., 2020), the authors propose a sparse attention mechanism to analyse larger sequences of up to 4096 elements, while keeping the computational time of the attention matrix linear with respect to the input size. A similar setup is also proposed in (Zaheer et al., 2020) using an additional random iterative attention mechanism to parse the entire graph of the sequence. Different from these approaches, our proposed pipeline can process indefinitely large sequences and has a stage-wise refinement mechanism of the NER predictions.

### III. Dataset.

Additionally, we introduce a new language modelling problem setup from the umbrella of NER tasks, namely nutritional information extraction from images of food products. This was achieved by collecting a dataset of food product images, entitled TREAT (i.e. naTRiEnt fAcTs) with 11,926 images depicting food products with fully extracted text information, bounding box image localization and complete NER class label annotations with sequences of up to $\approx 2,000$ words ($\approx 5000$ tokens) to be parsed. Labelling was performed using a web interface highlighting words. Annotators had to individually assign classes to words of interest. The proposed NER use-case is formulated on a linguistically diverse dataset in terms of semantics and specialisation. The collected data covers multiple European languages (EN, FR, DE, ES, IT, etc.) containing the nutritional information expressed with different languages within the same input text sequence. We further explore the specialisation in the compliance domain, learning to reason within the vocabulary of text content from food product labels. They must contain information related, but not limited, to ingredients, storage instructions, manufacturer addresses, nutritional information, allergen statements, advertising and production process details. Every piece of information can be found in standalone paragraphs, lists or table structures. The general layout is rich in both structure and visual cues. In terms of textual information, the collected data has both semantic text with full paragraphs or text structures (e.g. instructions, tables, lists or advertisement sections) and independent sentences.
TREAT Test Set - Word Level Results

Let \( I \) denote an image containing a list \( X = (x_1 \ldots x_{M_1}) \) of \( M_1 \) words, with \( x_i = (w_i, c_i) \) where \( w_i \) is the OCR (AWS-Texttract, 2019) extracted word and \( c_i \in [0, 1]^4 \) represents the bounding box coordinates relative to dimensions of \( I \). Additionally, let \( Y = (y_1 \ldots y_{M_1}) \) denote the class labels for each word \( w_i \) in \( X \).

Given the high inflation of \([\text{OTHER}]\) inside TREAT (see table 1) we constructed RADAR (RelevAnt wordD clAssifieR) using a bidirectional LSTM model (Hochreiter and Schmidhuber, 1997) at character level to predict if the word is relevant or irrelevant \(([\text{OTHER}])\). Thus, we have \( X_{\text{filtered}} = \{x_i \mid \text{RADAR}(x_i) > \alpha, i = 1 \ldots M_1\} \) where \( \alpha \) is validated to obtain a recall of 1 for relevant words. To simplify notation, we identify \( X_{\text{filtered}} \) with \( X \).

The language based embeddings for \( w_i \) are supplemented with positional encodings (Vaswani et al., 2017). Thus, we obtain the embedding representation \( E = (e_1 \ldots e_{M_1}) \) of \( X \) where \( e_i = (e^{w}_{i1} \ldots e^{w}_{i4}, e^{x}_{i1} \ldots e^{x}_{i4}, e^{y}_{i1} \ldots e^{y}_{i4}) \).
This leads to an input sequence \( \mathbf{E} \in \mathbb{R}^{M \times d} \), where \( d = d_w + d_x + d_y \). The dimensions \( d_w, d_x \) and \( d_y \) denote language embedding sizes while \( x \) and \( y \) denote positional variables with respect to image width and height, respectively. Additionally, a latent block representation denoted with \( \mathbf{L}^{\text{INIT}} \in \mathbb{R}^{N \times d} \) is learned. The intuition behind the latent block is to learn a projection of the most relevant information with respect to the NER task.

The \textsc{LANTERN} module is applied in a stage-wise manner. Each stage receives as input the embedded input sequence, \( \mathbf{E} \), and a latent block, \( \mathbf{L}' \in \mathbb{R}^{N \times d} \), and it predicts an array, \( \mathbf{Y}' \). In the following, we will describe the computational stages \( t \in \{0 \ldots T\} \). For \( t = 0 \), we have \( \mathbf{L}^{\text{INIT}} \in \mathbb{R}^{N \times d} \).

**STAGE \( t = 0 \):** we start by applying a cross-attention mechanism \( \Theta : \mathbb{R}^{M \times d} \rightarrow \mathbb{R}^{N \times d} \), over \( \mathbf{E} \) and \( \mathbf{L}^{\text{INIT}} \). Functions \( k(\cdot), q(\cdot) \) and \( v(\cdot) \) represent the keys, queries and values, respectively.

\[
\Theta(\mathbf{E}, \mathbf{L}) = \text{softmax}(q(\mathbf{L})k(\mathbf{E})^\top \sqrt{d})v(\mathbf{E}) \quad (1)
\]

This will basically result in a projection of the input sequence \( \mathbf{E} \) to the latent space \( \mathbb{R}^{N \times d} \) through \( \mathbf{L}^{\text{INIT}} \). We will denote the resulted projected latent block with \( \mathbf{L}^{\Theta} = \Theta(\mathbf{E}, \mathbf{L}^{\text{INIT}}) \).

Next, \( \mathbf{L}^{\Theta} \) is passed through a transformer encoder, \( \Gamma : \mathbb{R}^{N \times d} \rightarrow \mathbb{R}^{N \times d} \), to learn an implicit statistic between the elements of the latent block elements. The initial latent block information is added as a residual information on the resulting encoded information from \( \Gamma \), thus having

\[
\mathbf{L}^{\text{INIT}}_t = \Gamma(\mathbf{L}^{\Theta}^{\text{INIT}}) + \mathbf{L}^{\text{INIT}} \quad (2)
\]

Lastly, we apply a reversed cross-attention mechanism, \( \Theta_{\text{rev}} : \mathbb{R}^{N \times d} \rightarrow \mathbb{R}^{M \times d} \) over the input sequence \( \mathbf{E} \) and the latent block \( \mathbf{L}^{\text{INIT}}_t \) processed through transformer \( \Gamma \).

\[
\Theta_{\text{rev}}(\mathbf{E}, \mathbf{L}) = \text{softmax}(q(\mathbf{L})k(\mathbf{E})^\top \sqrt{d})v(\mathbf{L}) \quad (3)
\]

The reversed cross-attention operation provides a reprojection of the transformed latent block information \( \mathbf{L}^{\text{INIT}}_t \) to the input space, \( \mathbb{R}^{M \times d} \). Thus, the resulted information from the reversed cross-attention will be, \( \mathbf{L}^{\Theta}_{\text{rev}} = \Theta_{\text{rev}}(\mathbf{E}, \mathbf{L}^{\text{INIT}}_t) \).

\( \mathbf{L}^{\Theta}_{\text{rev}} \) is next passed through a feed-forward network, \( \Psi(\cdot) \), which provides the class predictions for the sequence’s entities, \( \hat{\mathbf{Y}}^{t=0} \) (i.e. \( t = 0 \) denotes the first stage), and it is being optimized using a cross-entropy loss function.

**STAGE \( t > 0 \):** This process is repeated for several iterations, in a multi-stage setup. For the next stages (i.e. stage \( 0 < t \leq T \)), the latent sequence \( \mathbf{L}' \) is initialized with the latent block information obtained from equation 2 of stage \( t - 1 \).

\[
\mathbf{L}' = \begin{cases} \mathbf{L}^{\text{INIT}}, & \text{if } t = 1 \\ \mathbf{L}^{t-1}, & \text{otherwise} \end{cases}
\]

The prediction of stage \( t - 1 \) of the NER task, \( \mathbf{Y}^{t-1} \), is fed to the feed-forward module of the current stage, \( t \), as a weight factor. Also, \( \mathbf{Y}^{t-1} \) is used as a weight to the attention-matrix computed at the current stage, \( t \), obtained via a projection function \( \Phi : \mathbb{R}^{M \times d_{\text{target}}} \rightarrow \mathbb{R}^{M \times N} \) where \( d_{\text{target}} \) is the total number of entities to be predicted

\[
\Theta_{\text{rev}}(\mathbf{E}, \mathbf{L}, \mathbf{A}) = \text{softmax}(q(\mathbf{E})k(\mathbf{L})^\top \sqrt{d})v(\mathbf{L}) \quad (4)
\]

where \( \odot(\cdot) : \mathbb{R}^{M \times N} \rightarrow \mathbb{R}^{M \times N} \) represent the Hadamard product and \( \mathbf{A} \in \mathbb{R}^{M \times N} \) is a weight matrix. Thus, the reprojected latent block information from stage \( t \) becomes, \( \mathbf{L}^{\Theta}_{t_{\text{rev}}} = \Theta_{\text{rev}}(\mathbf{E}, \mathbf{L}'^{t-1}, \Phi(\mathbf{Y}^{t-1})) \).

At each stage, a cross-entropy loss is applied over the output of \( \Psi : \mathbb{R}^{M \times d} \rightarrow \mathbb{R}^{M \times d_{\text{target}}} \), which are cumulated until the final stage \( T \), \( \mathcal{L}_{\text{NER}} = \sum_{t=0}^{T} \mathcal{L}_{\text{NER}}(\mathbf{Y}^{t}, \mathbf{Y}) \).

Given an input image \( \mathbf{I} \) and a learned latent block \( \mathbf{L} \), the framework outputs an array \( \mathbf{Y} \) with class predictions for all the relevant words identified within image \( \mathbf{I} \). A step-by-step breakdown of our pipeline is illustrated in figure 2 and in algorithm 1.

V. Experiments. We conduct experiments on our proposed dataset \textsc{TREAT}. Prior to applying \textsc{LANTERN}, we filtered a part of the irrelevant word corpus (i.e. \{OTHER\}) using \textsc{RADAR}. The accuracy of \textsc{RADAR} is 85% and we filtered 48% of the irrelevant words. The reason behind filtering such a low quantity of \{OTHER\} with respect to the obtained accuracy is because the majority of the word corpus from \textsc{TREAT} represents numerical values (e.g. 100g, 30ml) which we marked with \{NUM\}. These words (\{NUM\}) cannot be filtered without leveraging their positional context as they can be linked with nutrient keys. Thus, we decided to introduce them as such inside \textsc{LANTERN} without
RADAR filtration. In table 2 we report the word-level precision, recall and F1-score. We use a total of 6 stages with a total embedding size of \(d = 64\), where \(d_w, d_e\) and \(d_y\) are set to the values of 32, 16 and 16, respectively. The dimension \(N\) of latent block \(L\) is 256. Our method is able to achieve superior results compared to similar NER frameworks (Xu et al., 2020b), (Beltagy et al., 2020) and (Zaheer et al., 2020), some of them being designed for long sequences. For a fair evaluation, we compare against other competitive NER baselines using text long sequences. For a fair evaluation, we compare against other competitive NER baselines using text against other competitive NER baselines against other competitive NER baselines against other competitive NER baselines against other competitive NER baselines.

<table>
<thead>
<tr>
<th>Method</th>
<th>Precision</th>
<th>Recall</th>
<th>F1-Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>LANTERN w/o RADAR</td>
<td>0.83</td>
<td>0.80</td>
<td>0.81</td>
</tr>
<tr>
<td>LANTERN w RADAR</td>
<td>0.84</td>
<td>0.87</td>
<td>0.86</td>
</tr>
<tr>
<td>LANTERN w/o weight sharing</td>
<td>0.83</td>
<td>0.80</td>
<td>0.81</td>
</tr>
<tr>
<td>LANTERN w weight sharing</td>
<td>0.82</td>
<td>0.82</td>
<td>0.82</td>
</tr>
<tr>
<td>LANTERN w Language</td>
<td>0.80</td>
<td>0.42</td>
<td>0.40</td>
</tr>
<tr>
<td>LANTERN w Language + Positional</td>
<td>0.83</td>
<td>0.80</td>
<td>0.81</td>
</tr>
</tbody>
</table>

Table 3: Ablation study on various components of LANTERN over TREAT dataset. The highest impact in terms of performance is obtained with the RADAR based filtration of \([\text{OTHER}]\). Parameter weight sharing across all the stages also impacts positively the performance and it helps the model to convergence faster.

In table 3 we analyse the impact of the various model components and embedding types used by our model, thus validating the architectural innovative aspects of our method. In figure 1 (c) we show the impact of the multi-stage component. The refinement aspect of the method is visible between consecutive stages leading to significant improvement between the first and last stages.

VI. Conclusions. We propose a novel method for NER, specifically designed for indefinitely large sequences. It leverages a multi-stage transformer-based pipeline which breaks the quadratic computational constraints of the attention mechanism by projecting the input sequence to a latent space representation and a stage-wise setup which acts as a context refinement of the entity predictions. The methodology was evaluated on a novel and difficult NER use case, nutritional information extraction, proving superior results over other strong baselines specifically designed for long sequence parsing.

Acknowledgements. We thank George Leotescu and Adrian-Mihai Alecu for helpful insights.

In figure 3 we showcase multi-stage predictions, thus highlighting the refinement aspect of our framework on a practical example.


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, page 1192–1200.


