Effective extraction of textual data from document images using transformer architecture of deep neural networks

V.A. Vykhodtseva [1], G.V. Popova [2], Y.A. Vais [2]
[1] Kazakh-American Free University, 070000, Kazakhstan, Ust-Kamenogorsk, 76 M. Gorky Street;
[2] D. Serikbaev East Kazakhstan State Technical University, 070004, Kazakhstan, Ust-Kamenogorsk, 19 Serikbayev Street

Abstract

In the context of modern digital document management, the automation of document pro-cessing, particularly in accounting, is a crucial factor in enhancing the efficiency of business pro-cesses. However, automated document processing encounters a range of specific challenges, both linguistic and structural characteristics of the data. Traditional text processing methods that rely on classical optical character recognition (OCR) algorithms do not provide sufficient accuracy in extracting data from document images, which limits their use in automated accounting systems. These challenges are particularly evident when processing documents with complex structures, specific element placement, and text content. This paper proposes a solution to this problem by applying a model based on a transformer neural network architecture, specifically adapted for working with document images. Within the scope of this study, the transformer model is trained on a dataset of accounting document images with varying element placements and text with Cy-rillic characters. The focus on Cyrillic text is particularly relevant, as research in this area has pre-dominantly concentrated on documents in English or other Latin-based scripts. This article in-cludes the results of training evaluated through specialized performance metrics. As a result of the experiment, at the final stage of training the model, the confidence loss was 0.156, which indicates that the model effectively minimizes the prediction error. The obtained accuracy of 0.868 showed a relatively high accuracy of forecasts. The Recall value of 0.905 indicates that the model effectively identifies most of the positive examples. The indicator F1=0.886 reflects a good balance between accuracy and memorability. The accuracy of 0.96798 indicates that the model's predictions are highly accurate. The use of the transformer model significantly improves the accuracy of extracting key in-formation, such as date, number, and organization name, from accounting documents containing Cyrillic text. The findings of this study affirm the potential of this method for implementation in automated accounting systems, contributing to enhanced efficiency and precision in processing accounting documents.

Keywords: attention mechanism, deep learning, document intelligence, neural network, optical character recognition, transformer.

Citation: Vykhodtseva VA, Popova GV, Vais YA. Effective extraction of textual data from document images using transformer architecture of deep neural networks. Computer Optics 2026; 50(2): 1744. DOI: 10.18287/COJ1744.

Introduction

The increase in the volume and diversity of data across various domains, coupled with the concurrent advancement of computational capabilities, has catalyzed the rapid progress of the expansive field of artificial intelligence. Initially, the prevailing paradigm was symbolic artificial intelligence, which involved the processing of data defined as input parameters according to predefined rules. This paradigm was particularly widespread from the 1950s to the 1980s, a period during which the resolution of many tasks increasingly shifted from manual execution to software-based solutions [1]. However, over time, significant challenges in executing specific operations, such as image classification and speech recognition, became increasingly evident. As a result, the limitations of symbolic artificial intelligence prompted the creation and advancement of a new domain within artificial intelligence known as machine learning. This domain is characterized by the automatic generation of solution rules for specific tasks based on data and examples of expected outcomes.

Deep learning, a subset of machine learning, provides an advanced yet more intricate approach to addressing these challenges. This methodology is fundamentally supported by neural networks, which enable multilayered model training, thereby enhancing the accuracy of results when the model is applied to new data. The capabilities afforded by deep learning facilitate the automation of processing large volumes of data, thus enabling effective application across a variety of tasks, including image and speech recognition. Numerous researchers investigating the historical development and trends within artificial intelligence emphasize the growing complexity of machine learning methods and neural network architectures [2], [3].

Subsequently, these advancements have contributed to the expansion of the range of tasks in which these technologies, along with the results of numerous studies, can be applied. Specifically, Cui et al. [4] explore the concept of Document Intelligence, or Document AI, which encompasses technologies for document layout processing, data extraction, document classification based on images, and visual question answering. In their work, the authors explore the role of these technologies, the challenges associated with implementing universal tools, and available neural network architectures and

models that enable the automatic processing of various types of documents based on images. It is important to note that this field emerged as a result of extensive research in both Natural Language Processing (NLP) [5] and Computer Vision (CV) [6], [7], which has led to the integration of certain concepts from both domains within Document Intelligence. Furthermore, the research of Kastanas et al. [8] provides a comparative analysis of graph-based models and neural network architectures, such as Convolutional Neural Networks (CNNs) and Transformers, in the context of document layout analysis. This work also addresses the challenges of processing documents in multiple languages.

All the features of transformer architecture are detailed in the work titled "Attention Is All You Need" by Vaswani et al. [9], which serves as the original presentation of the transformer's revolutionary capabilities and its advantages over other existing neural network architectures. Subsequently, an increasing number of studies have been dedicated to exploring the application of transformers in natural language processing tasks, with particular emphasis on the attention mechanism. These studies highlight the high efficiency and accuracy of these models [10], [11]. The transformer architecture, as previously mentioned, has introduced a more refined approach to data processing compared to its predecessors, particularly recurrent neural networks (RNNs). While RNNs were predominantly used in the past, transformers have demonstrated significantly superior accuracy in many applications [12]. In the work by Islam et al. [13], in addition to tasks related to natural language processing, other categories of tasks are explored where transformer-based models are applied, such as audio and speech recognition. The study also offers a comprehensive overview of the models currently available, along with a comparative analysis of the approaches proposed by other researchers in the field.

An essential aspect of the functioning of transformers is tokenization and embedding generation, which are areas that have also been the subject of numerous studies and scientific experiments [14], [15]. A substantial study has also been conducted by Sajun et al. [16], in which the development of transformer models is analyzed in a chronological sequence.

Over time, alongside natural language processing tasks, the potential of transformer architecture and its components in the field of computer vision has been actively explored [17 – 21]. The results have shown that transformers are also well-suited for this category of tasks, enabling image classification and the extraction of relevant information. The study by Gheini et al. [22] focuses on machine translation, implemented through the training of a transformer model that employs the concept of cross-attention.

Despite the relatively recent introduction of transformer architecture, the high performance of these models has led many researchers to highlight their fundamental significance in deep learning [23], [24]. One of the transformer-based models used in the field of Document Intelligence for document classification based on images and relevant data extraction is Language-Independent Layout Transformer (LiLT), as presented in the work by Wang et al. [25]. This model eliminates language dependence, enabling the correct recognition of identical documents in different languages.

An important component in the training process of transformer models is coordinate normalization, a concept that has also been addressed in recent studies [26], [27].

In recent years, the automation of document processing has emerged as a critical task across various domains, particularly in accounting. With the increasing volume of digital data and the high demand for optimized processing of document content for information exchange, retrieval, and record-keeping, the effective extraction of information from documents has become both an important and challenging problem. This issue is exacerbated by the complex structure and arrangement of elements, such as dates, numbers, organization names, and linguistic peculiarities. Solutions designed for documents using Latin characters may not be applicable for Cyrillic characters. Consequently, this creates a need to develop an alternative approach that accommodates the format specific to a given domain and takes into account linguistic features.

The purpose of this paper is to provide a comprehensive review of transformers, with a particular emphasis on its applications in accounting document processing. This includes exploration of fundamental principles of transformer models and the key architectural components that have established them as a powerful tool for solving different tasks. Furthermore, this research is dedicated to a detailed examination of the LiLT model, focusing on its approach to processing accounting document images in order to prevent manual data extraction and ensure the minimization of mistakes during the subsequent analysis of the obtained data. The novelty of this research is the implementation of the LiLT model for processing images of accounting documents containing Cyrillic text, a domain that has not been extensively explored within the context of deep learning. While much of the existing studies have focused on Latin-script documents, the application of advanced deep learning techniques to Cyrillic text remains insufficiently explored. This study makes a significant contribution to the field of automated document processing by introducing a novel approach designed to enhance the accuracy and efficiency of information extraction from complex accounting documents. The study also presents the results of training the LiLT model on accounting document images, demonstrating its potential and effectiveness in tasks related to document understanding and applicability for integration into automated accounting systems.

1. Architecture of the transformer model

The transformer architecture comprises the processes of sequential encoding and decoding [9]. The encoder, as a component of this architecture, converts the input sequence into a vector of higher dimensionality. This resulting vector is then passed to the decoder to generate the output sequence. Unlike recurrent neural networks, the transformer preserves information about long-term dependencies through an attention mechanism, which is computed using dot product operations.

Earlier architectures do not allow for the retention of contextual information regarding sequences. Specifically, recurrent neural networks store sequence information in a hidden state, which is updated at each time step. As a result, information from

previous steps is lost because retaining it would require an excessively large hidden state [10]. In transformers, on the other hand, the solution to this problem is achieved through the attention mechanism, which is one of the key features of the transformer model, as evidenced by its high efficiency in processing and analyzing input sequences [11]. Furthermore, model training based on recurrent neural networks is not adapted for parallel computation. This implies that to compute the state of a layer, for instance, at step i+1, it is necessary to compute the state for step i, resulting in interdependent operations that are executed sequentially and do not permit parallelization, thereby impacting optimization during training [12].

The transformer model receives input in the form of tokens that represent the original sequence. The procedure by which textual data is divided into discrete components is referred to as tokenization [13]. At the subsequent stage, these tokens are converted into embeddings, which are vectors that provide a compressed representation of the original data, facilitating more effective interaction with machine learning algorithms. Initially, these embeddings consist of a random sequence of numbers, which evolves into a meaningful representation during the model training process [14].

In the initial stage, the original sequence is processed by the encoding component, which consists of a variable number of encoders depending on the specific implementation. Each encoder is composed of two layers: a multi-head attention layer and a feed-forward neural network [15]. Each encoder outputs a sequence of the same length as the input sequence. After passing through each of the described layers, the output of the layer is augmented with the input information. This is managed by another component of the transformer known as skip connection. This mechanism helps mitigate the vanishing gradient problem and ensures the stability of the model's training process. Subsequently, the activations of the sequences pass through a normalization layer to regulate their scale, thereby maintaining training efficiency and preventing issues related to vanishing or exploding gradients. The decoding component, in turn, consists of several decoders, each incorporating a feed-forward neural network and two multi-head attention layers, one of which utilizes the output data obtained from the encoder [16]. The components of the transformer architecture are illustrated in Figure 1.

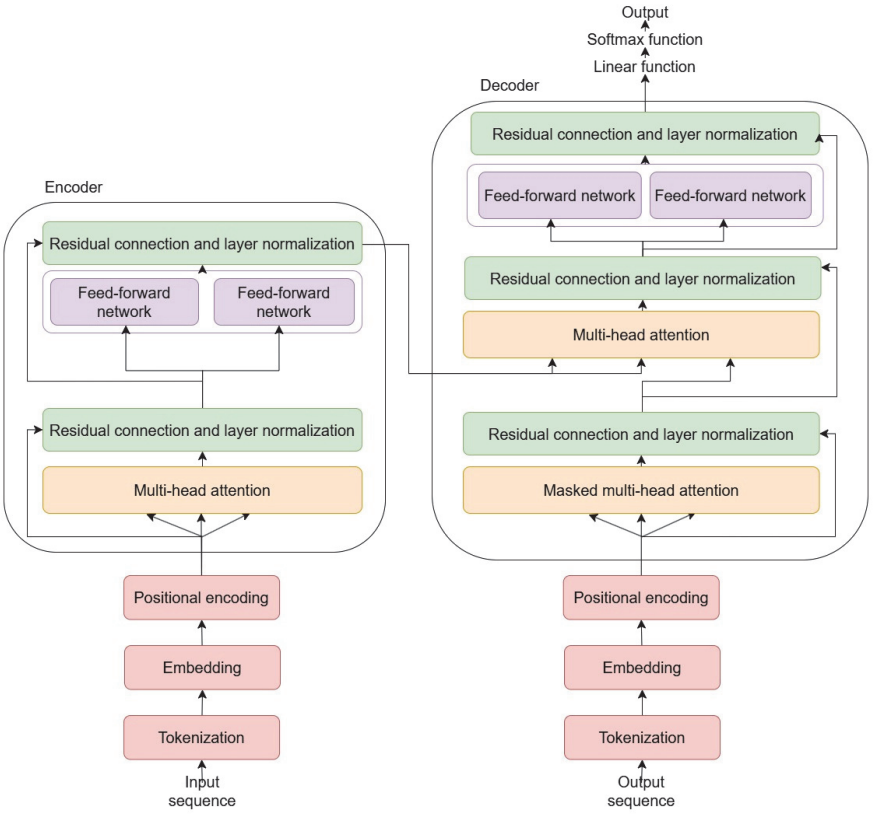

Fig. 1. Components of transformer architecture

It is important to note that embeddings have a limitation in that they do not account for the context in which the tokens were generated. This is due to the addition of positional embeddings to the base token embeddings during the positional encoding process, which aims to preserve the order of the tokens [17]. In addition, words may be represented by the same embedding if they consist of the same sequence of characters but have different meanings and are adjacent to different tokens, provided that the tokenization method does not account for this distinction. The attention mechanism incorporated in transformers, which is one of the key distinguishing features of this architecture, is designed to address this limitation. This allows the neural network to take into account the context relative to the current token in the input sequence [18]. In other words, the attention mechanism enables the identification of which tokens throughout the entire sequence may be relevant within the context of the current token. This element can be viewed as a method for transforming the standard

token embedding into an embedding that incorporates information about neighboring tokens, thus capturing its context. Consequently, this eliminates the duplication of embeddings that arises from the loss of context. As a result, it becomes feasible to compute this context either as a linear combination of embeddings or through a weighted average of the corresponding vectors [19]. Moreover, processing sequences through the attention mechanism prevents issues associated with hidden states that are updated at each step, as seen in recurrent neural networks. In this case, each token has direct access to any part of the sequence, since the attention layer is applied to the entire sequence simultaneously, which also facilitates the parallelization of the operations being performed.

When applying the attention mechanism, embeddings corresponding to the most relevant parts of the sequence for the given token are assigned larger weights. In this context, the weights represent an identity matrix. These values can be computed using various methods, but the most commonly used approach involves calculating weights based on the scalar product. Therefore, to obtain embeddings that contain information about their current context, it is essential to determine which tokens possess the highest relevance. In this case, the degree of similarity between two embeddings is determined by the scalar product of the considered embedding and another embedding from the input sequence. The greater the scalar product, the more contextually similar the embeddings are.

In the computation of attention for a sequence, three trainable matrices are required: W_q W_k W_v [20]. These matrices are initialized according to the selected approach and are subsequently updated during the model training process. The representation x_i of each element in the sequence is multiplied by each of the specified matrices. As a result of this operation, the following vector rows are obtained: q_i is the query to the database; k_i is the keys of the values stored in the database, which are used for searching; v_i is the values. The attention weight or compatibility coefficient between the query and the key is computed using the scalar product as shown in (1):

$$Attention\ weights_i = softmax(\frac{q_i k_1^T}{\sqrt{d_k}}, \frac{q_i k_2^T}{\sqrt{d_k}}, \dots) , \quad (1)$$

where d_k represents the dimension of the keys and values.

After that, the values v_i are weighted by the obtained coefficients as shown in (2):

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V , \quad (2)$$

where Q, K, and V are the matrices of queries, keys, and values, with each row containing q_i, k_i, and v_i, respectively [21].

In the decoder, one of the attention layers is a cross-attention layer. The query comes from the output sequence, while the keys and values come from the encoder as illustrated in Figure 2 [22].

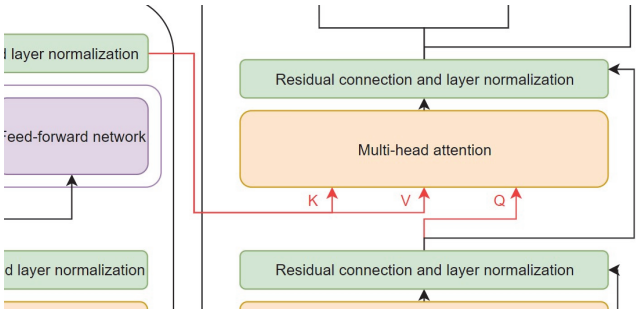

Fig. 2. Cross-attention layer

Multi-head attention addresses the issue of capturing multiple dependencies between tokens, which arises from the limited information contained in the set of matrices. Instead of a single attention layer, multiple parallel layers with different weights are used, and the resulting outputs are then aggregated as schematically shown in the Figure 3 [23].

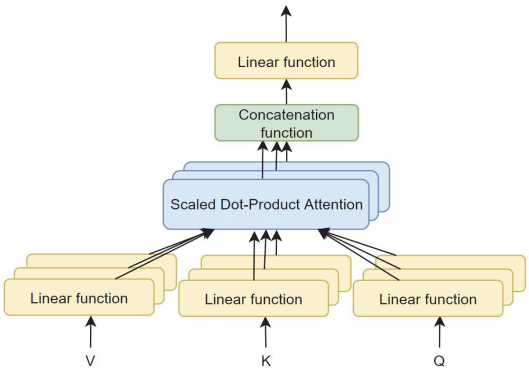

Fig. 3. Multi-head attention mechanism

A feed-forward neural network consists of two fully connected layers, which are applied independently to each element of the input sequence [24].

2. Language-independent layout transformer model recognition algorithm

One of the transformer-based models, Language-independent Layout Transformer (LiLT) employs a parallel two-stream transformer architecture. Initially, the input document image is processed by an optical character recognition (OCR) mechanism to extract the bounding boxes of the recognized text and their corresponding content. Subsequently, the extracted text and positional information are processed independently and converted into embeddings. Each embedding is processed by a specialized transformer-based architecture to generate meaningful representations. To facilitate effective interaction between the textual representations and the document layout features, a bi-directional attention complementation mechanism (BiACM) is used. In the final stage, the encoded textual representations and layout features are fused to produce a unified representation for subsequent processing [25]. The overall architecture of the model is illustrated in Figure 4.

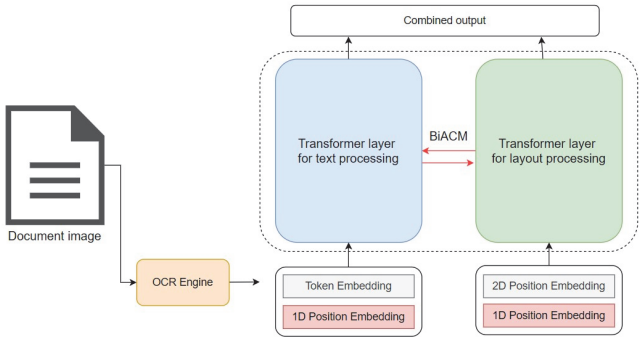

Fig. 4. LiLT transformer-based model architecture

The language model is used to generate a textual embedding, which is combined with a one-dimensional positional embedding. The result is then normalized using the LayerNorm function [26]. The creation of the embedding is shown in (3):

$$E_T = LN(E_{token} + P_{1D}) , \quad (3)$$

where E_{token} is the token-based embedding, P_{1D} is the one-dimensional positional embedding, and LN is the LayerNorm function.

Before generating the textual embedding, the resulting lines obtained from the optical character recognition mechanism are sorted based on the coordinates of the bounding boxes.

The layout embedding is a two-dimensional embedding formed using the bounding boxes of the recognized lines. In other words, the positional embedding is generated through the four coordinates of the bounding box along with its width and height. These data are concatenated and added to the positional embedding to obtain the resulting layout embedding. The bounding box coordinates are normalized and discretized, meaning they are transformed into integer values within the range of 0 to 1000 [27].

Embeddings for height and width are generated as shown in (4) and (5):

$$h\_position\_embeddings = self.h\_position\_embeddings(y_{max} - y_{min}) , \quad (4)$$

$$w\_position\_embeddings = self.w\_position\_embeddings(x_{max} - x_{min}) . \quad (5)$$

The final formula for generating the positional embedding is shown in (6):

$$P_{2D} = Linear(CAT(E_{xmin}, E_{xmax}, E_{ymin}, E_{ymax}, E_{width}, E_{height}) , \quad (6)$$

where CAT denotes the concatenation operation along the channels.

The created embeddings are concatenated to form the final two-dimensional embedding. The two-dimensional positional embedding is then concatenated again with the one-dimensional embedding and normalized using the LayerNorm function as shown in (7).

$$E_L = LN(P_{2D} + P_{1D}) . \quad (7)$$

As mentioned earlier, the architecture incorporates a bi-directional attention complementation mechanism. Attention complementation means that the attention layers in the text and layout streams also work in parallel and, at the same time, are applied to each other's streams. This enables the exchange of attention values obtained in the same part of the same layer within both the text and layout networks.

The values of effective attention are computed using the corresponding attention values from both streams. During pretraining, the attention values are decoupled using the DETACH function to prevent them from influencing each other's

gradients and to facilitate the use of other language models. During fine-tuning, these attention values are merged, allowing the network to learn the relationships between linguistic and visual features as demonstrated in (8) [25].

$$a_{ij}^{\tilde{L}}\begin{cases}a_{ij}^{L}+DETACH(a_{ij}^{T});\ if\ task\ ==\ pretrain\\ a_{ij}^{L}+a_{ij}^{T};\ if\ task\ ==\ finetune\end{cases}. \quad (8)$$

The core idea of the architecture of this model is to separate the neural networks responsible for processing the layout information of document elements from the linguistic neural networks, while ensuring the model can be quickly adapted to other documents with similar layouts and in different languages. In other words, the model allows for the use of a layout processing neural network in conjunction with various language models [25].

Thus, LiLT emerges as an innovative tool in the field of document image processing technologies, where the combination of architectural components ensures adaptability, performance, and language independence of the document features.

3. Results and discussion

In order to train the Language-independent Layout Transformer model on documents specific to a particular domain, such as accounting, several key steps must be undertaken. First, it is essential to perform image annotation, followed by the configuration of the model's training parameters. Finally, the model's performance should be evaluated through inference on a test dataset.

For the training data, images of document samples in Russian were selected, representing various operations in accounting. The document samples are filled with random data, but in the format required for these types of documents. Additionally, the images were pre-annotated using a specialized tool called «Label Studio». The annotations included key fields such as: organization, identification number, document number, document date, title, and additional information. Upon completion of the annotation process, the data was exported in JSON format, which contained information about the properties of the original image, as well as the coordinates of the bounding boxes and the associated tags. In the subsequent stage, the content of the JSON file was processed, and textual data was extracted using the Tesseract OCR engine. The prepared data was split into training and test sets with a 70 – 30 ratio.

The model fine-tuning involves a combined model that includes LiLT and the XLM-RoBERTa model, a multilingual version of the RoBERTa transformer that supports multiple languages, including Russian. As for the model training parameters, the number of epochs is set to 20, and the learning rate is set to 5e – 5, which is the recommended value for training transformer models. Model evaluation is performed every 10 steps. Table 1 presents the training results of the LiLT transformer model.

Tab. 1. Training Results of the LiLT+RoBERTa Model (XML)

Step	Validation Loss	Precision	Recall	F1	Accuracy
10	1.750508	0.100752	0.141350	0.117647	0.568488
20	1.048503	0.230047	0.310127	0.264151	0.721474
30	0.674874	0.379048	0.419831	0.398398	0.828463
40	0.463021	0.542443	0.552743	0.547544	0.879034
50	0.356668	0.602687	0.662447	0.631156	0.910292
60	0.295223	0.752083	0.761603	0.756813	0.929352
70	0.223563	0.812757	0.833333	0.822917	0.956036
80	0.221583	0.832998	0.873418	0.852729	0.959848
90	0.179204	0.858012	0.892405	0.874871	0.965947
100	0.164847	0.871690	0.902954	0.887047	0.967217
110	0.169306	0.870637	0.894515	0.882414	0.966963
120	0.154504	0.877551	0.907173	0.892116	0.967726
130	0.157613	0.874494	0.911392	0.892562	0.968488
140	0.156380	0.868421	0.905063	0.886364	0.967980

The Validation Loss of 0.156, achieved at the final step of model training, indicates that the model has effectively minimized prediction error. This value suggests good generalization to unseen data. The Precision of 0.868 demonstrates a relatively high accuracy in predictions. The Recall of 0.905 shows that the model effectively identifies most positive examples, minimizing the number of missed cases. The F1-Score of 0.886 reflects a good balance between precision and recall, indicating that the model provides a satisfactory level of both accuracy and completeness in its predictions. Finally, the Accuracy of 0.96798 confirms the overall high correctness of the model's predictions.

Upon completion of the model training, the process of applying it to make predictions on new, unseen data is carried out. In this case, the model is evaluated on a pre-prepared test set of document images, which are filled with random data for experimental purposes. For a clearer representation of the results, bounding boxes with predicted labels are displayed on the document images.

Figure 5 demonstrates the results of entity recognition on a document image confirming the transfer of goods or materials to counterparties.

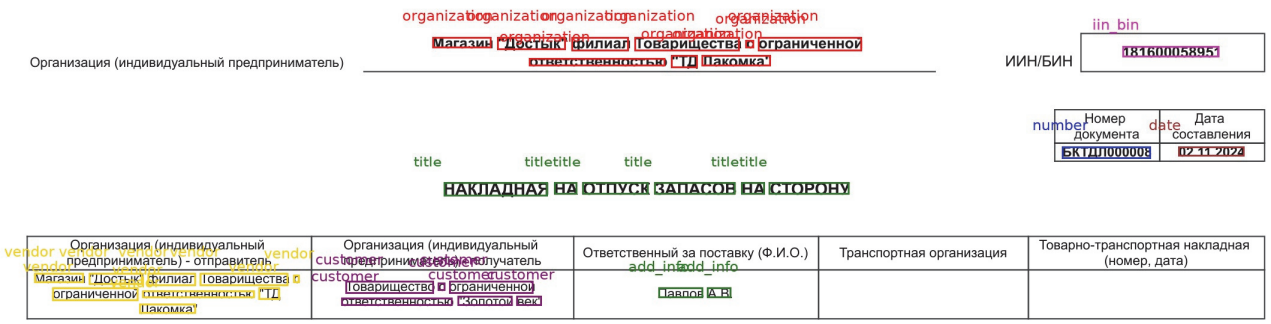

Fig. 5. First type of documents from the test set

Figure 6 shows the results of recognizing relevant data on a document image used to confirm the receipt of goods in warehouse inventory management.

Fig. 6. Second type of documents from the test set

As shown in the presented images, the trained model correctly recognized the data in the test set.

Thus, one of the main advantages of the LiLT transformer model is flexibility since it demonstrates its seamless integration with any pre-trained text encoder, such as RoBERTa. In other words, this model is able to correctly process documents with the same structure, but in different languages, ensuring that the layout information remains unchanged when the language is switched. In addition, it should be noted that LiLT demonstrates strong performance, sometimes even outperforming its predecessors, on widely used machine learning benchmark datasets across various languages.

The presented results extend existing research in the field of recognition and information extraction from document images, as they explore the potential for adapting a transformer model to work with accounting documents containing Cyrillic characters. Specifically, the application of the pre-trained LiLT model in the context of processing documents with these characteristics is examined. Previous studies have generally focused on working with documents containing Latin characters within standard and publicly available datasets and have not addressed custom samples of accounting documents. The results obtained from the trained LiLT model are structured in a way suitable for integration into automated accounting systems.

Considering that these accounting automation programs are widely utilized in countries where documents are written in Cyrillic, particularly in Russian, this functionality would provide substantial benefits, facilitating the automation of accounting documents processing in these regions. The transformer model's output consists of a collection of predicted tags corresponding to the text fragments provided as input, along with their coordinates. These tags can be matched to the corresponding text fragments and formatted into a JSON or XML structure, which serve as a universal data exchange format. This structure can subsequently be transmitted to accounting software, where, within the application, the text can be mapped to specific database fields based on the assigned tags, thereby automating the process of creating or updating records within the system. Thus, the information extracted using the transformer model can reduce the number of errors during data entry into the system when processing large volumes of paper documents. However, it should be noted that the model's performance and results largely depend on how correctly the data for model training is prepared. Furthermore, the training process itself is also computationally demanding, requiring substantial processing power and memory. Therefore, for these purposes, it is recommended to use pre-trained models, utilize tools for automating data annotation, and leverage cloud computing platforms.

Conclusion

This paper presents an optimized approach for the effective extraction of data from images of accounting documents with varying structures and content, using a deep learning model based on transformer architecture. This method ensures high accuracy due to an advanced attention mechanism and the model's ability to analyze not only the text but also its placement within the image. Moreover, the core model architecture can be combined with a separate tokenizer, enabling the processing of documents in multiple languages. The results of the study expand upon previous research in the

application of transformer models, which primarily focused on document images in Latin script, showing that transformer models can also be successfully applied to texts using other scripts, particularly Cyrillic. The contribution of this study is in broadening the scope of document processing and introducing a novel approach designed to address the needs of the field of accounting. It should be mentioned that the use of transformer architecture can significantly improve the accuracy and speed of data processing not only in accounting documents but also in other fields such as legal contracts, financial reports, medical certificates, and other types of documents. These findings are of considerable importance for the automation and optimization of business processes in areas where electronic document management is required and complex documents must be processed quickly and accurately. Although the preparation of training data and the training process itself are relatively complex and resource-demanding, the results of the trained model compensate for the challenges, demonstrating high efficiency and adaptability to documents with diverse content. It provides competitive advantages across various industrial sectors. Future research in this area is aimed at training the transformer model on more diverse document datasets, as well as addressing document classification tasks. Another important direction is the integration of the model's functionality with third-party systems for accounting automation.

References

[1] Yu L, Zhao X, Huang J, Hu H, Liu B. Research on machine learning with algorithms and development. J Theory Pract Eng Sci (JTPES) 2023; 3(12): 7-14. DOI: 10.53469/jtpes.2023.03(12).02.

[2] Xu Y, Zhou Y, Sekula P, Ding L. Machine learning in construction: From shallow to deep learning. Dev Built Environ 2021; 6: 100045. DOI: 10.1016/j.dibe.2021.100045.

[3] Kühl N, Goutier M, Hirt R, Satzger G. Machine learning in artificial intelligence: Towards a common understanding. Proc 52nd Hawaii Int Conf on System Sciences 2019: 5236-5245.

[4] Cui L, Xu Y, Lv T, Wei F. Document AI: Benchmarks, models and applications. arXiv Preprint. 2021. Source: <https://arxiv.org/abs/2111.08609>. DOI: 10.48550/arXiv.2111.08609.

[5] Chakkarwar V, Tamane S, Thombre A. A review on BERT and its implementation in various NLP tasks. In Book: Tamane S, Ghosh S, Deshmukh S, eds. Proceedings of the international conference on applications of machine intelligence and data analytics (ICAMIDA 2022). Atlantis Press; 2022: 112-121. DOI: 10.2991/978-94-6463-136-4_12.

[6] Kameswari ChS, et al. An overview of vision transformers for image processing: A survey. Int J Adv Comput Sci Appl 2023; 14(8): 273-289. DOI: 10.14569/IJACSA.2023.0140830.

[7] Pereira GA, Hussain M. A review of transformer-based models for computer vision tasks: Capturing global context and spatial relationships. arXiv Preprint. 2024. Source: <https://arxiv.org/abs/2408.15178>. DOI: 10.48550/arXiv.2408.15178.

[8] Kastanas S, Tan S, He Y. Document AI: A comparative study of transformer-based, graph-based models, and convolutional neural networks for document layout analysis. arXiv Preprint. 2023. Source: <https://arxiv.org/abs/2308.15517>. DOI: 10.48550/arXiv.2308.15517.

[9] Vaswani A, et al. Attention is all you need. In Book: von Luxburg U, Guyon I, Bengio S, Wallach H, Fergus R, eds. NIPS'17: Proceedings of the 31st international conference on neural information processing systems. Red Hook, NY: Curran Associates Inc; 2017: 6000-6010.

[10] Gillioz A, Casas J, Mugellini E, Khaled OA. Overview of the transformer-based models for NLP tasks. In Book: Ganzha M, Maciaszek L, Paprzycki M, eds. Proceedings of the 2020 Federated Conference on Computer Science and Information Systems, September 6–9, 2020. Sofia, Bulgaria. New York City: Institute of Electrical and Electronics Engineers; 2020: 179-183. DOI: 10.15439/2020F20.

[11] Shirahatti A, Rajpurohit V, Sannakki S. Transformer-based multi-head attention network for aspect-based sentiment classification. Indones J Electr Eng Comput Sci 2022; 26(1): 472-481. DOI: 10.11591/ijeecs.v26.i1.pp472-481.

[12] Lakew SM, Cettolo M, Federico M. A comparison of transformer and recurrent neural networks on multilingual neural machine translation. In Book: Bender EM, Derczynski L, Isabelle P, eds. Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018. Association for Computational Linguistics; 2018: 641-652.

[13] Islam S, et al. A comprehensive survey on applications of transformers for deep learning tasks. Expert Syst Appl 2024; 241: 122666. DOI: 10.1016/j.eswa.2023.122666.

[14] Erkan A, Gungor T. Analysis of deep learning model combinations and tokenization approaches in sentiment classification. IEEE Access 2023; 11: 134951-134968. DOI: 10.1109/ACCESS.2023.3337354.

[15] Kanjirangat V, Mellace S, Antonucci A. Temporal embeddings and transformer models for narrative text understanding. arXiv Preprint. 2020. Source: <https://arxiv.org/abs/2003.08811>. DOI: 10.48550/arXiv.2003.08811.

[16] Sajun AR, Zualkernan I, Sankalpa D. A historical survey of advances in transformer architectures. Appl Sci 2024; 14(10): 4316. DOI: 10.3390/app14104316.

[17] Xu P, Zhu X, Clifton DA. Multimodal learning with transformers: A survey. IEEE Trans Pattern Anal Mach Intell 2023; 45(10): 12113-12132. DOI: 10.1109/TPAMI.2023.3275156.

[18] Hafiz AM, Parah SA, Bhat RUA. Attention mechanisms and deep learning for machine vision: A survey of the state of the art. arXiv Preprint. 2021. Source: <https://arxiv.org/abs/2106.07550>. DOI: 10.48550/arXiv.2106.07550.

[19] Lin T, Wang Y, Liu X, Qiu X. A survey of transformers. AI Open 2022; 3: 111-132. DOI: 10.1016/j.aiopen.2022.10.001.

[20] Bai G, Guo H, Xiao C. Research on the application of transformer in computer vision. J Phys Conf Ser 2023. DOI: 10.1088/1742-6596/2649/1/012033.

[21] Papa L, Russo P, Amerini I, Zhou L. A survey on efficient vision transformers: algorithms, techniques, and performance benchmarking. IEEE Trans Pattern Anal Mach Intell 2024; 46(12): 7682-7700. DOI: 10.1109/TPAMI.2024.3392941.

[22] Gheini M, Ren X, May J. Cross-attention is all you need: Adapting pretrained transformers for machine translation. Proc 2021 Conf on Empirical Methods in Natural Language Processing 2021: 1754-1765. DOI: 10.18653/v1/2021.emnlp-main.132.
[23] Chitty-Venkata KT, Emani M, Vishwanath V, Somani AK. Neural architecture search for transformers: A survey. IEEE Access 2022; 10: 108374-108412. DOI: 10.1109/ACCESS.2022.3212767.
[24] Sonkar S, Baraniuk RG. Investigating the role of feed-forward networks in transformers using parallel attention and feed-forward net design. arXiv Preprint. 2023. Source: <https://arxiv.org/abs/2305.13297>. DOI: 10.48550/arXiv.2305.13297.
[25] Wang J, Jin L, Ding K. LiLT: A simple yet effective language-independent layout transformer for structured document understanding. 60th Annual Meeting of the Association for Computational Linguistics (ACL 2022) 2022: 7747-7757. DOI: 10.18653/v1/2022.acl-long.534.
[26] Menary S, Kaski S, Freitas A. Transformer normalisation layers and the independence of semantic subspaces. arXiv Preview. 2024. Source: <https://arxiv.org/abs/2406.17837>. DOI: 10.48550/arXiv.2406.17837.
Nguyen TQ, Salazar J. Transformers without tears: Improving the normalization of self-attention. 2019 16th International Workshop on Spoken Language Translation 2019: 1-9. DOI: 10.5281/zenodo.3525484.

Authors' information

Victoria Aleksandrovna Vykhodtseva (b. 2001) graduated from Kazakh-American Free University with a Bachelor's degree in Information Systems in 2023 under the American program offered by the university. She is currently pursuing a Master's degree at the same university in the same field. Research interests: artificial intelligence, computer vision, automation of accounting operations, information and analytical systems. E-mail: *vykhodtseva.va@gmail.com*

Galina Vladimirovna Popova (b. 1964) graduated from Tomsk Polytechnic University, Faculty of Automation and Computer Engineering in 1986. In 2006, she defended her PhD thesis at Altai State Technical University in the field of Computer Modeling of Physical Processes. She is the author of over 40 scientific publications, has published 2 monographs and 4 textbooks. She supervises Master's and Doctoral students. She has published 4 articles in the Scopus database. Research interests: mathematical modeling, knowledge bases, distributed systems, information and analytical systems, artificial intelligence. E-mail: *gal.tomsk64@gmail.com*

Yuriy Andreevich Vais (b. 1973) graduated from S.Amanzholov East Kazakhstan State University in 2001 with a Master degree in Matematika. In 2010 he received the degree of candidate of technical sciences in the dissertation council at the D. Serikbaev East Kazakhstan State Technical University, specialty 05.25.05 - "Information systems and processes, legal aspects of informatics". He is the author of more than 50 scientific publications. He has published 3 monographs, 2 educational aids. He has published 3 articles in the Scopus database. Research interests: mathematical modeling, knowledge bases, distributed systems, information and analytical systems, artificial intelligence. E-mail: *vais.jura.73@gmail.com*

Code of State Categories Scientific and Technical Information (in Russian – GRNTI)): 28.23.37
Received June 03, 2025. The final version – September 06, 2025.