УДК 004.8
https://doi.org/10.47533/2020.1606-146X.150

Y. AMIRGALIYEV*, MATEUS MENDES, K. MUKHTAR, R. JANTAYEV, CH. KENCHIMOV

Suleyman Demirel University, Kaskelen, Kazakhstan
Professor Politechnic Institute of Coimbra(ISES), Portugal

RESNET50+TRANSFORMER: KAZAKH OFFLINE HANDWRITTEN TEXT RECOGNITION

Nowadays, due to the transition to digital data storage, there is a need to implement handwritten text recognition (HTR), which is an automatic translation of handwritten characters into a machine format. Handwriting recognition is complicated by the fact that there are many languages and it is possible to write the same character in different ways. In this regard, we conducted a study of a machine learning model for recognizing handwritten characters using databases of the Kazakh language. We trained the ResNet50 + Transformer deep learning model using two published databases of the Kazakh language: KOHTD and HKR. In the course of the study, these databases were studied on the component and qualitative sides with a comparison of the results of validation of the trained model. As a result, the KOHTD database showed results in the form of CER-9.46% and WER-20.18%, while the HKR database showed results in the form of CER-6.08% and WER-15.51%.

Keywords*: ResNet50, Transformer, HTR, KOHTD, HKR, CNN, Kazakh HTR.*

INTRODUCTION. Handwritten text recognition (HTR), in view of its growing importance and the enthusiasm of many scholars, is gaining traction in academic study. In today's digital age, HTR is in high demand in the industry and is used to convert paper data to digital media in either an online or offline format [1]. The primary source for offline text translation into digital format is scanned or picture documents in image format [2]. Bank checks, medical documents, and postal documents are all examples of these documents. All this creates a great need to create curpnomastable HTR systems with the ability to function with a large number of documents in different languages. However, there are challenges linked with the fact that handwritten words vary in characteristics according on the author and linguistic traits such as slanted and rounded characters, diacritical dots, transverse stripes, and curved letters, as in any industry. The accuracy with which characters of varied

* E-mail корреспондирующего автора: 201107062@stu.sdu.edu.kz

complexity are identified and, as a result, the level of discovering the most appropriate words, determines the effectiveness of the handwriting recognition system.

In the past years, the field of deep learning has achieved very good results in the field of optical character recognition, and many methods have proved to be very effective for many tasks, such as image classification, object detection and pattern recognition [3]. The field of research is handwriting recognition, which has not been left aside having demonstrated significant progress. Thanks to the use of Convolutional Recurrent Neural Networks (CRNN) [4]-[6], more and more effective recognition models have been identified as optical models. These networks use convolutional layers that are responsible for extracting objects from text images. Then the extraction result is fed to repeating layers that propagate and decode objects using Connectionist Time Classification (CTC), which leads to the final result. Inside the CNN there is a Long-term Short-Term Memory (STM) which is often used as a sequence decoder. Many more approaches, such as Multidimensional LSTM (MDLSTM), have been proposed in the future to improve the accuracy of this decoder [7], extending the capabilities of Recurrent Neural Network (RNN) architectures for multidimensional data. However, while MDLSTM [8], [9] is ineffective due to its computational cost and complexity, new studies have emerged in which simpler optical models are presented [12]. A Bidirectional Long-Term Short-Term Memory (BLSTM) [10] is one of such models. This model comes close to MDLSTM in terms of results. Although the results using such architectures seem promising, optical recognition models have some degrees of difficulty remembering long contexts. This is due to problems with vanishing gradients. In addition, to get better results, these models employ millions of trainable parameters. As a result, many real-world applications find it challenging to apply them [11].

In this article, we propose a study of the classification problems in handwritten input images of the Kazakh language based on a comparison of two databases of Cyrillic characters using the residual neural network (ResNet) image classification architecture. The Kazakh language differs from English in that it has a large number of letters and, accordingly, a large number of words, which in turn complicates recognition. And also, this complexity is manifested in the collection of databases, in which it is difficult to balance the qualitative side, which is directly proportional to the number and variety of handwritten words. The first KOHTD database has a very large database of handwriting scans in the form of individual words of the Kazakh language. It differs in that more than 95% consists exclusively of words of the Kazakh language. The second database differs in that it has a smaller number of scans but it has better quality. On the other hand, in contrast to the first base, preference is given more towards the Russian language, which can be seen in the ratio of words: 95% Russian words, 5% Kazakh words. This is explained by the fact that the Kazakh language in its arsenal of Cyrillic symbols differs from the Russian language in only 9 letters. Further in the study, there is a preprocessing of scans of these databases and subsequent training of the recognition model. The ResNet architecture was chosen as the model, which has become popular in the image classification environment and has many varieties. In our case, ResNet50 is a variety that forms 50 layers of a neural network and is well suited for large databases.

II. RELATED WORKS

1. KOHTD. Kazakh Offline Handwritten Text Dataset (KOHTD) [13] is the first database of handwritten characters consisting only of the Kazakh language. In this paper, the authors proposed an extensive dataset that contains 3,000 handwritten exam papers and more than 140,335 segmented images, as well as approximately 769,000 characters. The database was compiled from the examination answers of students of Satpayev University and Al-Farabi University. It consists of scans divided into individual words and has an extensive collection of handwriting. The researchers also trained models such as Puigcerver [14], Bluche [15], Flor [16] and Abdallah [17] as a test of their own database. As a result, a good indicator of CER and WER in the Flor model was obtained.

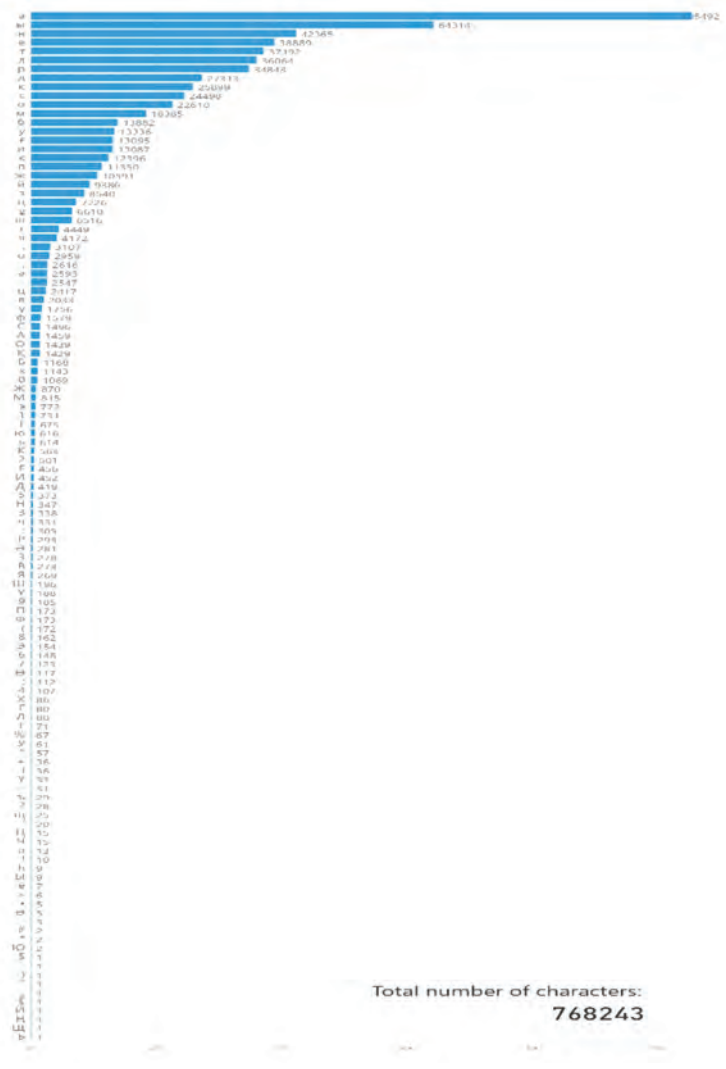

Figure 1 – The number of characters in the KOHTD database.

2. HKR. Russian and Kazakh databases for autonomous handwriting recognition are presented by the authors in the article "HKR for a handwritten database in Kazakh and Russian languages"[18]. The database contains about 95% of Russian and 5% of Kazakh words/sentences, respectively. And it consists of more than 1400 completed forms. There are about 63,000 sentences, more than 715,699 characters, created by about 200 different authors. The database mainly contains samples of data (Forms) of keywords in Kazakh and Russian languages (Districts, Cities, Villages, etc.), handwritten Kazakh and Russian Cyrillic alphabet, handwritten samples (Forms) of poems in Russian. The database is divided into two parts: the first contains scans of completed forms, the second contains prepared cuts of these forms. In this study we will use the second part. But unlike the previous database, scans are collected in the form of 1-3 words in one scan with dozens of repetitions from different writers. This, in turn, gives the database a certain quality and is very useful when teaching the recognition model. But at the same time, there are very few symbols of the Kazakh language in this database, or even some do not exist at all. In the picture below, we can notice this and also see the total number of characters.

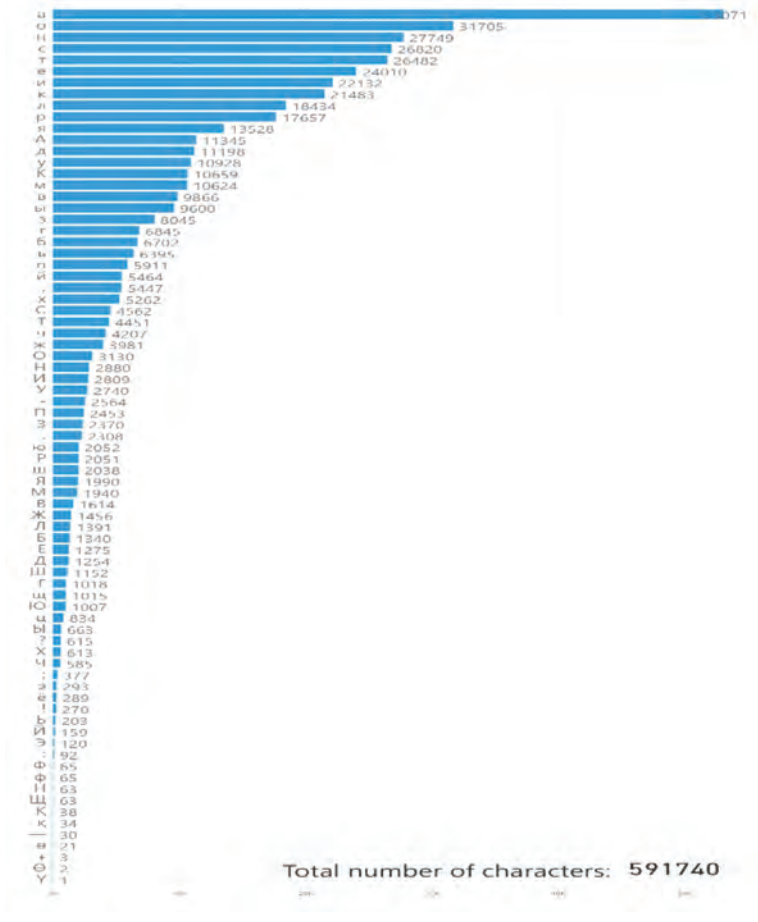

Figure 2 – The number of characters in the HKR database.

III. METHODS

3.1 Model. When studying the literature, several approaches were found to implement the above-mentioned system using various CNN and RNN structures. But more attention was attracted by the ResNet50 architecture in conjunction with the Transformer model [19]. At the moment, the architecture, which is no less popular for its effectiveness from other architectures, is used in research both in the field of classification of objects with images and in character recognition. This architecture also differs in that it does not have RNN layers, which makes it possible to train the model in an unconventionally parallel method, as opposed to sequential in RNN layers.

The main idea of ResNet is to overcome the notorious vanishing gradient problem by introducing a «fast connection to the identifier", which is carried out by skipping one or more layers. The notorious vanishing gradient problem occurs during model training when increasing CNN layers, which seems logical when increasing the efficiency of the model. Due to the fact that the gradient spreads back to earlier layers, this leads to the fact that the model is limited to a certain number of CNN layers, an increase in which leads to an increase in training losses. As a consequence of this factor, well-known models such as VGG network [22] have only 19 convolutional layers, the AlexNet [20] and Google Net [21] have only 5 and 22 layers respectively. The method of solving this problem in the ResNet architecture is illustrated in the figure 3.

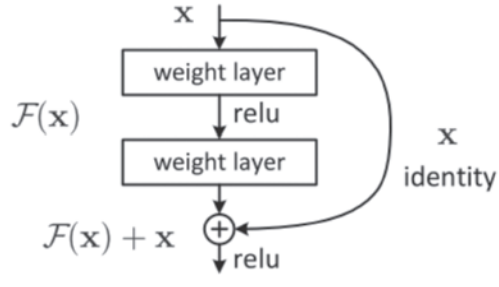

Figure 3 – A residual block

Stacking layers, according to the authors [23] should not impair network performance since we could just superimpose identifier mappings on the present network and the new design would perform similarly. This means that a more complex model should not result in a larger learning error. In the mathematical equation of identity mapping with the residual network F(x,{Wi}) denotes the residual mapping to be learned across the stacked-layer, and x denotes a shortcut connection to be inserted with residue, and both dimensions must be the same.

$$y = F(x,\{Wi\}) + x \quad (1)$$

In our case, we use a model with 50 layers. In Figure 4, we can see the architecture of the model in which the embedded identification blocks are shown with many layers of

convolutional blocks, the structure of which is shown in the right part of the image. “ReLU” is used as the activation layer.

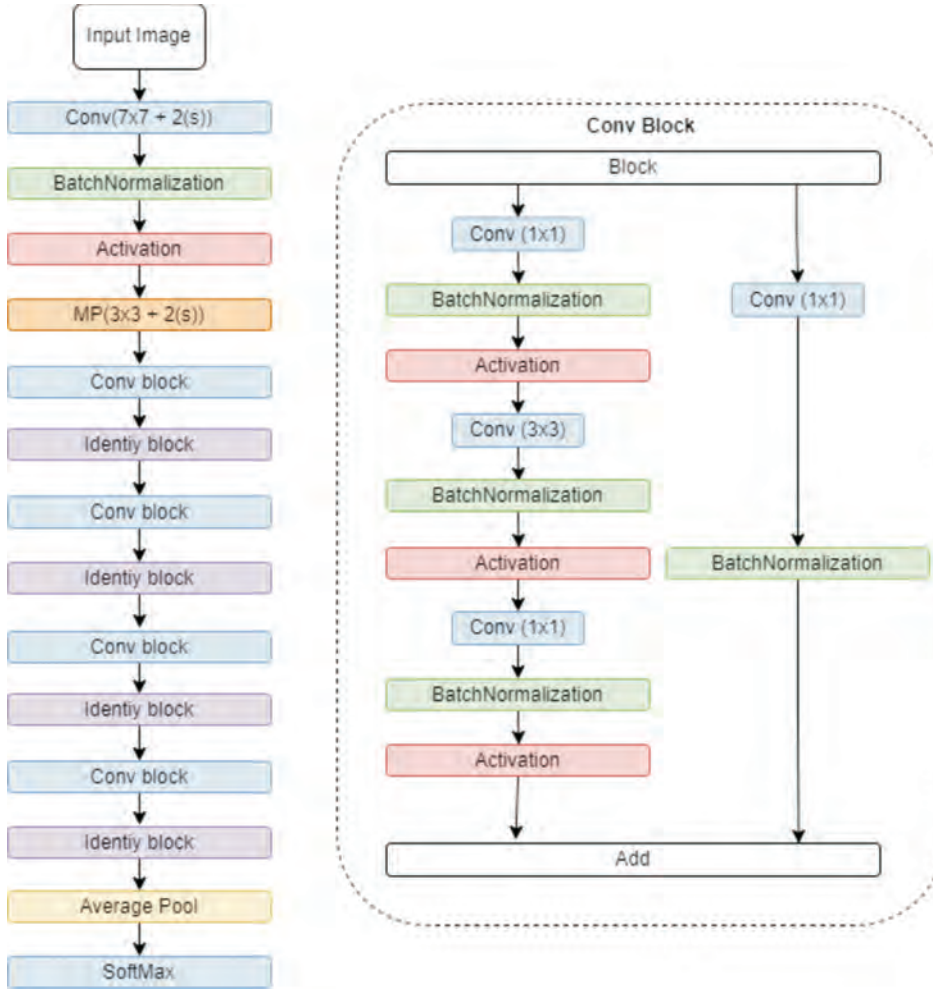

Figure 4 – ResNet50 model

Next, we need to consider the Transformer model. Before the evolution of the Transformer, sequential patterns were learnt using a notion known as the RNN network. RNN, on the other hand, struggled to remember prior information from long word sequences and so failed to predict the following words in the series. Long Short-Term Memory (LSTM), which features an inbuilt forget gate and addition gate, was created to solve this problem. With the addition of the attention notion, this idea can handle the long-term relationships between the phrases. It was still unable to capture the dependencies of long phrases, such as those of 1000 words. Furthermore, we know that sentence length changes from one phrase to the next, therefore training time varies as well. Because we have to unroll the LSTM network for each input sentence and compute the gradient at each time step while

backpropagating gradients, it takes a long time to train. To overcome all of these issues, researchers developed "The Transformer," a robust yet simple network design that is built on attention and has the same property as the recurrence model. When performing tasks like as machine translation, question and answer modeling, and so on, the attention mechanism works well. It's a straightforward recurrent attention system with an end-to-end memory network. It achieves superior outcomes without the use of sequence-aligned RNNs or convolution networks.

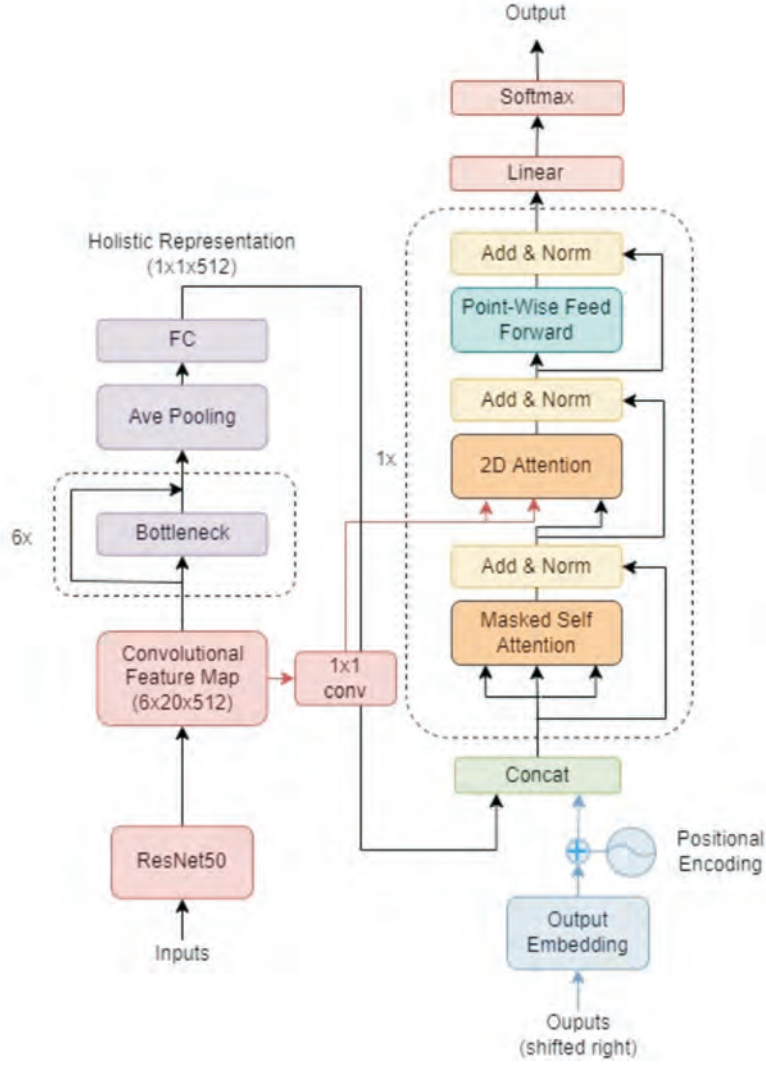

Figure 5 – The Transformer Architecture.

In this study, we used an architecture that is a combination of ResNet50 as an encoder and Transformer as a decoder. In the figure 6 we can see that there are two sections of the architecture. The encoder is on the left, while the decoder is on the right.

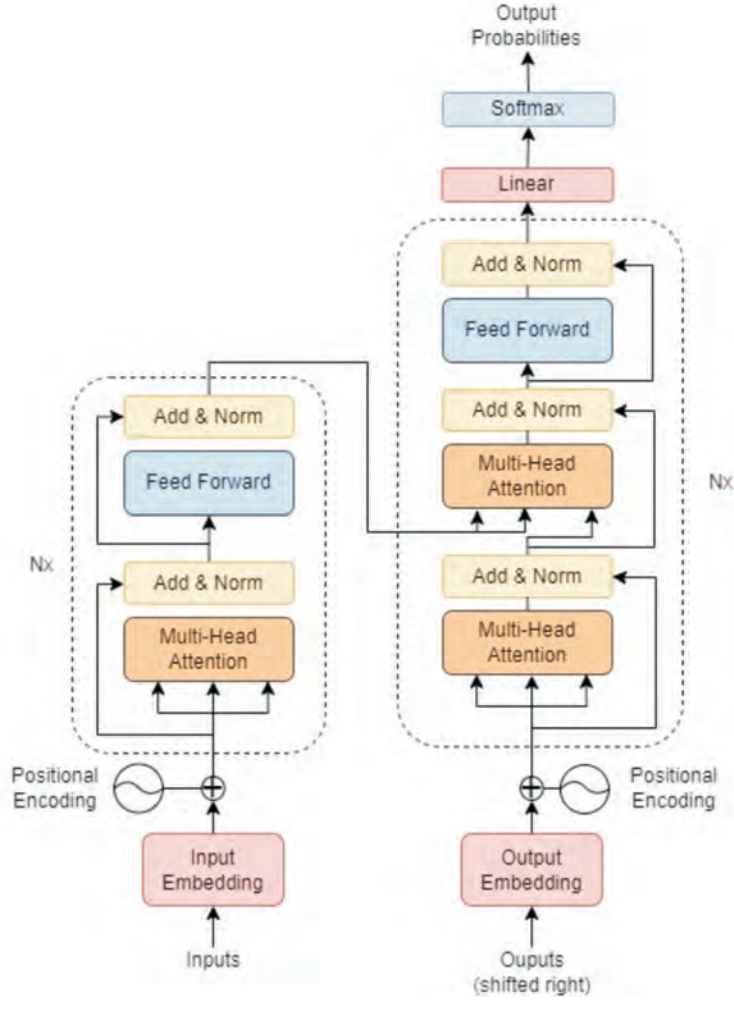

Figure 6 – Model architecture

In our experiment, in the encoder part, we used a ResNet50 for the deeper network. Feature mapping is then sent through two networks at the same time: a convolution layer (1x1) and a bottleneck. The outputs of the (1x1) convolution layer are sent into the decoder sublayer, which is the second multi attention mechanism, and are regarded as a query and key vector.

The character string is the input to the decoder embedding layer. The input string is character tokenized using the characters '<PAD>' at the start and '<EOS>' at the end. The preceding layer's output is fed into a masked multi-layer attention model, which is then normalized by adding a residual network. The output is then input into two-dimensional attentional layers, together with the output from feature mapping, and the layer is normalized using a residual network. The output of layer normalization is fed into a position-wise feed-forward network, which is then followed by layer normalization using a residual network, and 'Softmax' activation using 2-dimensional dense layers.

IV. RESULTS. The experiment was conducted on a computer with an intel core i5-10500 processor with 16 GB of RAM. At the beginning of the experiment, both bases were divided into three parts: for training, for validation and for the test, respectively. The volume of parts was taken from the general database through random selection, depending on the name of the scans.

Table 1 – Division results

Dataset	Training	Validation	Test	Total
KOHTD	130 697	7778	1880	140 355
HKR	60 396	3571	976	64 943

First, training was conducted on the KOHTD database. Learning parameters learning rate is 0.0001, dropout is 0.1 and batch size is 100. In case of immutability in the value loss parameter for 3 epochs, training stops. As a result, the training lasted more than 12 hours and stopped at 18 epochs. During the training, after each epoch, validation was carried out using the part of the database that is allocated for validation. With this, the following parameters such as loss, validation loss and CE (Character Error Rate) were calculated. Changes in this parameter can be seen in Figure 7 and 8, respectively.

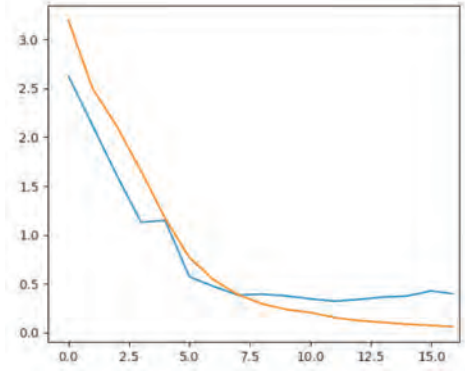

Figure 7 – Loss (orange) and validation loss (blue) values.

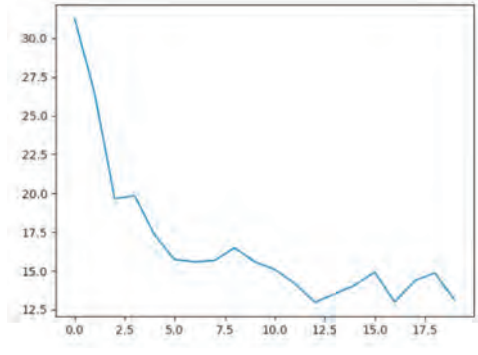

Figure 8 – CER loss value.

Secondly, the training was conducted using the HKR database. The parameters were identical, that is, the learning rate is 0.0001, the dropout is 0.1, only the packet size was reduced by 50 since the image size was larger. As a result, the training lasted more than 10 hours and stopped at 17 epochs. During training changes in the parameters of the loss, validation loss and CER can be seen in Fig. 9 and 10, respectively.

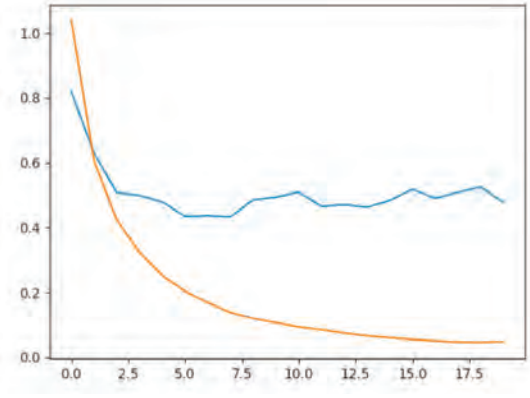

Figure 9 – Loss (orange) and validation loss (blue) values.

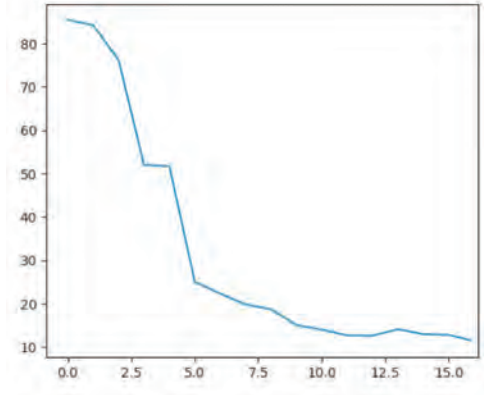

Figure 10 – CER loss value.

After the models were trained, a test was conducted using the part of the database that is intended for the test.

Figure 11 – Samples from prediction of KOHTD (1,2) and HKR (3,4) dataset.

For the sake of interest, the models were also tested on each other's test parts. The parameters CER and WER (word error rate) was chosen for validation. The results can be seen in Table 2.

Table 2 – Models validation results.

Model	Train Data	Test Data	CER	WER
ResNet50+Transformer	KOHTD	KOHTD	9.465%(±0.5%)	20.187%(±0.3%)
ResNet50+Transformer	HKR	HKR	6.088%(±0.3%)	15.513%(±0.2%)
ResNet50+Transformer	KOHTD	HKR	14.44%(±1.5%)	30.235%(±2.1%)
ResNet50+Transformer	HKR	KOHTD	87.19%(±3%)	97.929%(±2%)

V. DISCUSSION. Checking the trained model base by means of another was not originally part of the research plans. But they were carried out for more specific validation. As a result, at the first check of the KOHTD model based on HKR, the results were equal to CER-44.4% and WER-65%. This was due to the fact that the words in the KOHTD database are divided into separate scans and do not have a space character, and therefore this led to a recognition error. To improve the result, the words in the HKR test database were divided into separate scans. After that, we received the described results. As for the results of checking the KOHTD-based HKR model, we can say that they were expected, since basically the KOHTD database consists of Kazakh words and symbols. What can not be said about the HKR database, as a result of which the model was not able to recognize many Kazakh characters. But the test results on the base itself are very good. And also the KOHTD databases can be compared with the results in the article itself, which were equal to FLOR CER-6.52%, WER-24.52% [13]. In our case, CER turned out to be worse, but instead it turned out to improve the WER indicator.

And there is also a problem in the similarity of Cyrillic characters, such as ‘м’ and ‘ш’, ‘л’ and ‘е’, ‘қ’ and ‘к’, etc. This, in turn, makes character recognition easier not only for the machine but also for the person himself. This problem can be considered one of the most important problems of recognizing handwritten characters, if possible, eliminating which can achieve very good recognition results.

('[ERR:1]', '"жал"', '->', '"жап"')

('[ERR:1]', '"беруден"', '->', '"берудеп"')

Figure 12 – Recognition errors.

VI. CONCLUSION AND FUTURE WORK. As a conclusion, we can say that the initial goals were fully achieved and the model showed good results in validating both databases. We can notice that both the model and the quality of the collected database affect the result. Therefore, in the future it is possible to conduct experiments with newer models with an improved base. And also, as an improvement in the work, there is possibility to

add a linear separation algorithm for the full implementation of the model as recognition of entire pages of handwriting. There is also an idea about the possibility of improving the model by adding a search from a ready-made dictionary of words to improve and avoid errors of similar characters. This would make it possible to increase the recognition quality of many models by several times.

This researh was supported by the Science Committee of the Ministry of Education and Science of the Republic of Kazakhstan, project GF AR 08053034.

REFERENCES

1 R. Reeve Ingle. et. al., A Scalable Handwritten Text Recognition System, IEEE Trans. Pattern Anal. Mountain View, CA 94043, USA, 2019.

2 Tsochatzidis L, Symeonidis S, Papazoglou A, Pratikakis I. HTR for Greek Historical Handwritten Documents. J Imaging. 2021.

3 Phillip Benjamin Ströbel. et. al., Fink G.A., Evaluation of HTR models without Ground Truth Material, University of Zurich, University of Bern), 2022.

4 Chieh-Chi Kao. et. al., R-CRNN: Region-based Convolutional Recurrent Neural Network for Audio Event Detection, Amazon Alexa, 2019.

5 Xinyu Fu et. al., CRNN: A Joint Neural Network for Redundancy Detection, Solution Architect and Engineering Asia Pacfic and Japan, Nvidia, 2017.

6 Baoguang Shi et al, An End-to-End Trainable Neural Network for Image-based Sequence Recognition and Its Huazhong University of Science and Technology, 2015.

7 Alex Sherstinsky et al. Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network, Physica D: Nonlinear Phenomena, 2018.

8 Has¸im Sak, et. al. Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling, Google, USA, 2018.

9 Haowei Jiang et. al., Recurrent Neural Network from Adder's Perspective: Carry-lookahead RNN. Whiting School of Engineering, 2021.

10 Ralf C. Staudemeyer et. al., Understanding LSTM – a tutorial into Long Short-Term Memory Recurrent Neural Networks Last accessed November, Schmalkalden University of Applied Sciences, 2019.

11 Tomohiro Tanaka et. al., Evolution Strategy Based Neural Network Optimization and LSTM Language Model for Robust Speech Recognition, Tokyo Institute of Technology, Japan, 2018.

12 Daniel Hopp., Performance of LSTM Neural Networks in Nowcasting during the COVID-19 Crisis, UNCTAD Research Paper, 2021.

13 Nazgul Toiganbayeva, et al.KOHTD: Kazakh Offline Handwritten Text Dataset, Cornell University, 21(2021).

14 J. Puigcerver. et. al., Are Multidimensional Recurrent Layers Really Necessary for Handwritten Text Recognition, ICDAR, 2017.

15 T. Bluche, R. Messina, Gated convolutional recurrent neural networks for multilingual handwriting recognition, 14th IAPR international conference on document analysis and recognition (ICDAR), 2017.

16 A. F. de Sousa Neto, B. L. D. Bezerra, A. H. Toselli, E. B. Lima, Htrflor: a deep learning system for offline handwritten text recognition, 33rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), 2020.

17 A. Abdallah, M. Hamada, D. Nurseitov, Attention-based fully gated cnn-bgru for russian handwritten text, Journal of Imaging 6 (12), 2020.

18 Daniyar Nurseitov, Kairat Bostanbekov, Daniyar Kurmankhojayev, Anel Alimova, Abdelrahman Abdallah, HKR For Handwritten Kazakh & Russian Database, Multimedia Tools and Applications, 2021.

19 Ashish Vaswani, et. al. Attention Is All You Need, Conference on Neural Information Processing Systems, 2017.

20 Alex Krizhevsky et al, ImageNet Classification with Deep Convolutional Neural Networks, 2012.

21 C Szegedy et al., Going Deeper with Convolutions, Google, USA, 2014.

22 Karen Simonyan et al, Very Deep Convolutional Networks for Large-Scale Image Recognition, ICLR, 2015

23 Kaiming He et al., Deep Residual Learning for Image Recognition, Microsoft Research, 2015.

Е. АМИРГАЛИЕВ, МАТЕУШ МЕНДЕС, К. МҰХТАР, Р. ДЖАНТАЕВ, Ч. КЕНШИМОВ

Сулейман Демирель Университеті, Қаскелең, Қазақстан
Коимбра политехникалық институтының профессоры (ISES), Португалия

RESNET50 + TRANSFORMER: ҚАЗАҚ ТІЛІНДЕГІ ҚОЛЖАЗБА МӘТІНДІ ОҚШАУ РЕЖИМДЕ ТАНУ

Қазіргі уақытта деректерді сандық сақтауға көшуге байланысты қолмен жазылған мәтінді тануды жүзеге асыру қажет, бұл қолмен жазылған таңбаларды машина форматына автоматты түрде аудару болып табылады. Қолжазбаны тану көптеген тілдердің болуымен және сол таңбаны әртүрлі жолмен жазуға болатындығымен қиындайды. Осыған байланысты біз қазақ тілінің дерекқорын пайдалана отырып, қолжазба таңбаларын тануға арналған машиналық оқыту моделіне зерттеу жүргіздік. Біз қазақ тілінің жарияланған екі: KOHTD және HKR дерекқорларын пайдалана отырып, resnet50+ Transformer тереңдетіп оқыту моделін оқыттық. Зерттеу барысында бұл мәліметтер базасы дайындалған модельдің валидация нәтижелерін салыстыра отырып, компоненттік және сапалық жағынан зерттелді. Нәтижесінде KOHTD дерекқоры CER-9,46% және WER-20,18% нәтижелерін көрсетті, ал HKR дерекқоры CER-6,08% және WER-15,51% нәтижелерін көрсетті.

Түйін сөздер*: ResNet50, Transformer, HTR, KOHTD, HKR, CNN, қазақша HTR.*

Е. АМИРГАЛИЕВ, МАТЕУШ МЕНДЕС, К. МУХТАР, Р. ДЖАНТАЕВ, Ч. КЕНШИМОВ

Университет Сулеймана Демиреля, Каскелен, Казахстан
Профессор Политехнического института Коимбры (ISES), Португалия

RESNET50+TRANSFORMER: РАСПОЗНАВАНИЕ РУКОПИСНОГО ТЕКСТА НА КАЗАХСКОМ ЯЗЫКЕ В АВТОНОМНОМ РЕЖИМЕ

В настоящее время, в связи с переходом на цифровое хранение данных, существует необходимость в реализации распознавания рукописного текста, который представляет собой авто-

матический перевод рукописных символов в машинный формат. Распознавание рукописного ввода осложняется тем фактом, что существует много языков, и один и тот же символ можно писать по-разному. В связи с этим мы провели исследование модели машинного обучения для распознавания рукописных символов с использованием баз данных казахского языка. Мы обучили модель глубокого обучения ResNet50+ Transformer, используя две опубликованные базы данных казахского языка: KOHTD и HKR. В ходе исследования эти базы данных были изучены с компонентной и качественной сторон со сравнением результатов валидации обученной модели. В результате база данных KOHTD показала результаты в виде CER-9,46% и WER-20,18%, в то время как база данных HKR показала результаты в виде CER-6,08% и WER-15,51%.

***Ключевые слова**: ResNet50, Transformer, HTR, KOHTD, HKR, CNN, казахский HTR.*