

V. Slyusar

Central Research Institute of Armaments and Military Equipment of Armed Forces of Ukraine
 Povytrophlotsky Ave, 28B, Kyiv, 03049
<https://orcid.org/0000-0002-2912-3149>

THE TEXT SEGMENTATION BY NEURAL NETWORKS OF IMAGE SEGMENTATION

Abstract. The article highlights the importance of text segmentation in the field of natural language processing (NLP), especially in light of the development of large language models such as GPT-4. It discusses the use of specialized segmentation neural networks for various tasks, such as processing passport data and other documents, and points out the possibility of integrating these technologies into mobile applications. The use of neural network architectures, geared towards image processing, for text segmentation is considered. The study describes the application of networks such as PSPNet, U-Net, and U-Net++ for processing textual data, with an emphasis on adapting these networks to text tasks and evaluating their effectiveness. The potential of the multimodal capabilities of modern neural networks and the need for further research in this field are emphasized.

Keywords: neural network, dataset, ImageNet, text segmentation, Word2vec, embedding.

Input

As is known, text segmentation is one of the basic technologies of natural language processing (NLP). In addition to text classification, it is very common in various areas of use and consists, in particular, in

dividing the text into semantically combined fragments (phrases, sentences, paragraphs, or thematic sections), which are marked with some color or underlined with different lines, etc. An example of performing the text segmentation procedure is shown in Fig. 1.

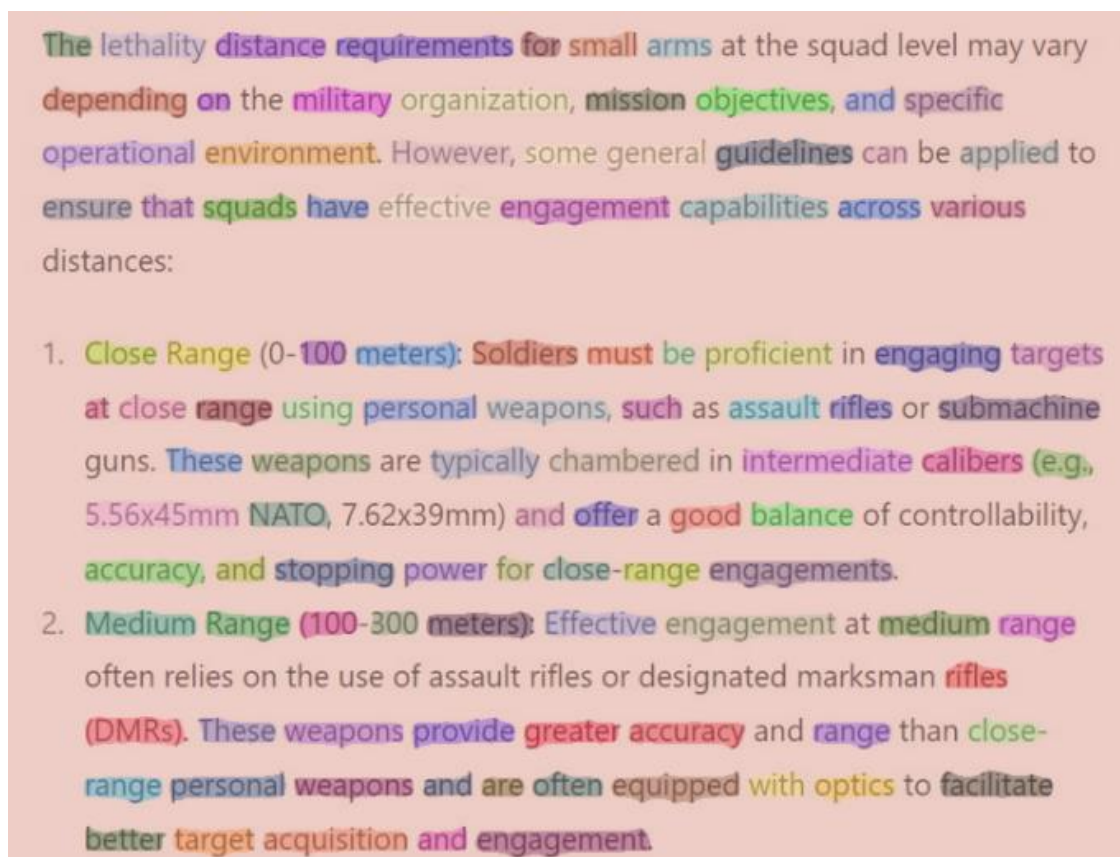


Fig. 1. An example of visualizing the results of text segmentation

The emergence of large language models (LLM) such as GPT-4 [1] and others has allowed achieving a deep understanding of texts and a completeness of context analysis, which are important elements of NLP. At the same time, there are certain niches where the use of overly complex LLM architectures would be excessive. In particular, we are talking about tasks of processing passport data, store receipts, cargo transport bills of lading, etc., where text segmentation is necessary before text recognition according to certain features. In this context, it is significant that a specialized segmentation neural network, thanks to its locality and modest hardware resource requirements, can be integrated into mobile applications that, instead of Internet access, are connected to official data transmission networks. Therefore, this direction should also be developed, regardless of the successes of LLM neurotransformer technologies, which makes relevant efforts in this direction.

Analysis of the latest research and publications

One of the important trends observed in the development of LLM is the improvement of the multimodal properties of the respective neural networks when the same neural network complex or hyperneural network performs various functions without changing its structure. In this regard, it would be interesting to investigate whether it is possible to use the same architecture of traditional neural networks for image processing, in particular their segmentation, for texts. Such an idea was expressed by the author in [2] based on the results of applying for text classification pretrained neural networks on the ImageNet classification datasets [3 - 6]. This approach showed very good results, in particular, it was possible to achieve text classification accuracy of 95% and above depending on the dataset. This encourages scaling of this concept to the field of segmentation of textual data arrays as well.

It should be noted that a search for publications in this direction did not reveal previous developments that would indicate the presence of work on a similar topic. This is evidence of the novelty of the approach

proposed in [2] and further developed below for solving the task of text segmentation.

The aim of the article is to investigate the effectiveness of using neural networks for text segmentation, which, in their architecture, coincide with neural networks for image segmentation.

Presentation of the main material

Research on text segmentation began with the use of the PSPNet neural network as the simplest in its architecture [7]. In this case, a collection of texts in the form of contracts was chosen as the dataset for its training, which were presented in different variants: as a bag of words (Bag of Words) [2] or a vector (Word2vector), and also using the Embedding operation [2].

Fig. 2 shows a simple modification of the PSPNet neural network structure, which allows moving from processing a one-dimensional text array to a two-dimensional (2D) architecture using layers typically used for image processing: Conv2D, Average2D, etc.

A feature of this architecture that allowed adapting the 2D-oriented PSPNet for processing 1D arrays is the use of the Reshape layer, which converts an input vector of 80 elements into a tensor with dimensions $80 \times 1 \times 1$.

A similar Reshape layer is also included in the output segment of the neural network for reverse transition from the image format to one-dimensional segmented text, in this case segmented into 6 categories. In this case, the data format at the output of Reshape is 80×6 , and the output Conv2D layer has 6 filters (for the number of text categories).

Fig. 3 presents a more complex architecture, which, in addition to the Reshape layer ($80 \times 16 \times 1$), includes the Embedding operation in the format of 5000×16 before it, which allows increasing the data volume by scaling the input vector. In the subsequent segment of the neural network, the same architecture is used that is characteristic of image processing. Another difference in this version of the neural network construction is the additional inclusion of the Dropout operation in the output segment with the aim of

discarding excess connections between neurons.

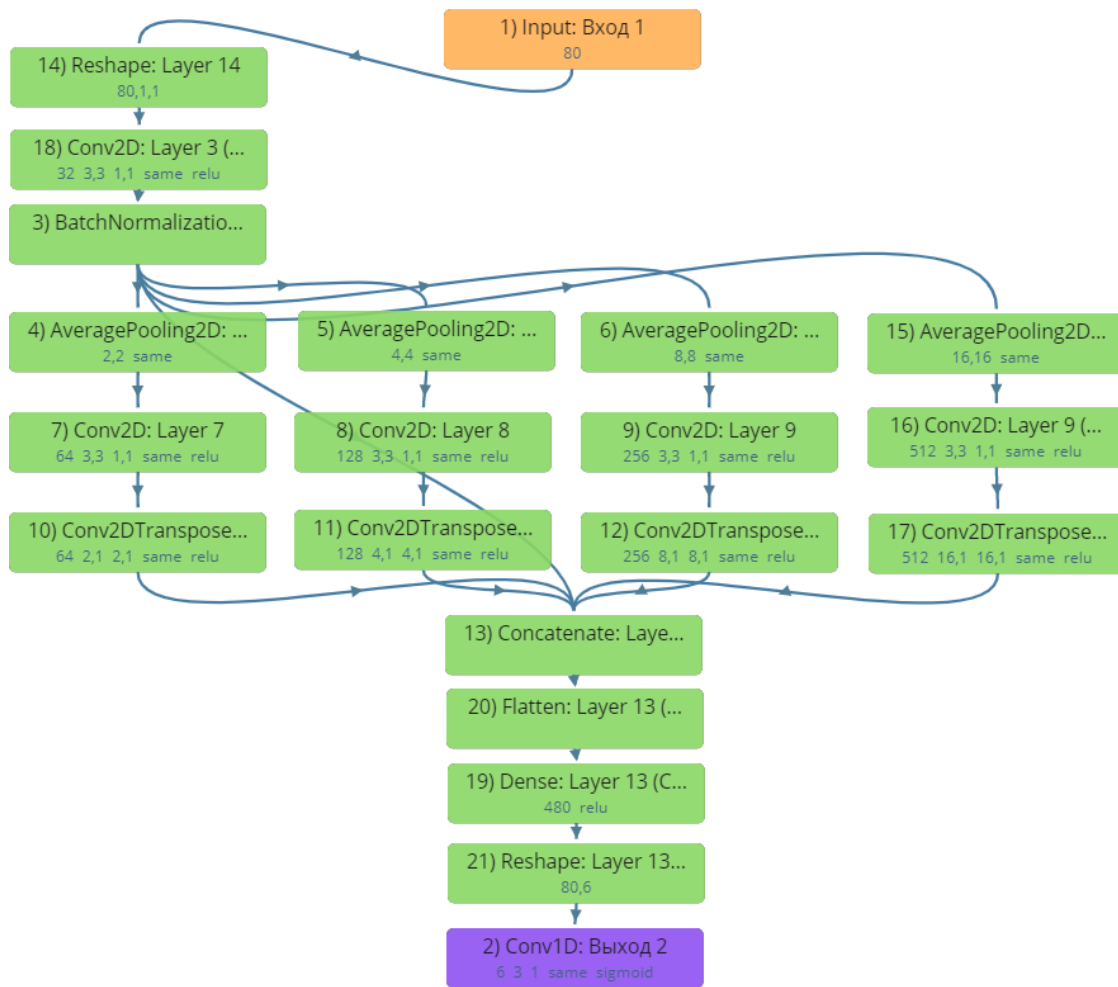


Fig. 2. Variant of modifying the 2D architecture of PSPNet for processing one-dimensional text arrays

To evaluate the effectiveness of the proposed PSPNet architecture, the balanced Dice coefficient (BDC) was used, which is used in text segmentation to evaluate text splitting methods. The BDC is an adaptation of the traditional Dice coefficient [8], which was proposed as a measure of statistical similarity between two data sets and is often used in information retrieval systems to compare queries and documents or to analyze interdependencies between them.

The Dice coefficient formula can be represented as [8]:

$$D(A, B) = 2 |A \cap B| / (|A| + |B|),$$

where A and B are the compared sets, $|A \cap B|$ is the size of their joint part, and $|A|$ and $|B|$ are the volumes of these sets.

The balanced variant of the Dice coefficient introduces corrections to this formula, taking into account the peculiarities

of text segmentation, where the size of segments can vary significantly. It does this by introducing a corrective coefficient that gives more weight to smaller segments, thus providing a more accurate assessment for text segmentation tasks, where the main goal is to identify small but important segments within a larger text.

In the case of using a Dropout coefficient of 0.1, a batch of 32, and a training step of the neural network of 0.001 with such an architecture, a text segmentation accuracy of 67.2% was obtained on the 108th epoch according to the BDC indicator. This example shows that adding an Embedding layer at the input significantly improves the efficiency of the neural network, thanks to a larger data array at the input of the pyramidal segment.

Thus, the main thing in solving the task of adapting 2D neural network architectures is

the transformation of text into such an array of numbers that would be appropriate for image processing and coincide with them in format. By the way, the Embedding operation in LLM transformers is performed in a similar way. Their use was once characteristic only of the text processing field, but now they have spread

to the processing of images and videos. In this case, several different data arrays are combined together, and as a result of their further joint processing, for example, images or videos are generated from texts and so on.

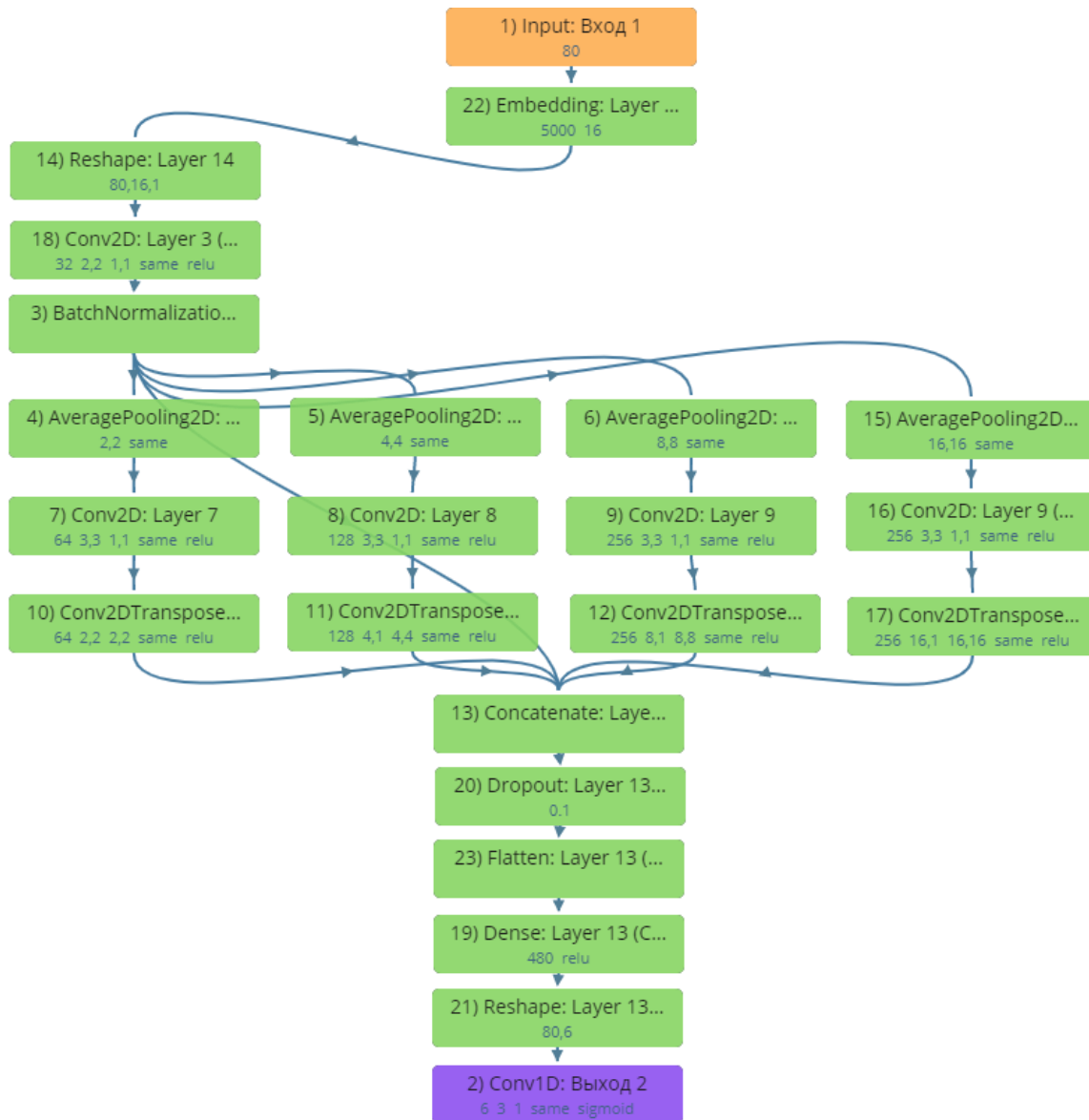


Fig. 3. 2D architecture of PSPNet using the Embedding layer

A more detailed study of the neural network with the architecture shown in Fig. 3 revealed that its text segmentation accuracy can be increased in case of expanding the vocabulary in Embedding. For example, by increasing the number of input tokens in this layer to 10,000, reducing the number of filters in the Conv2D and Conv2DTranspose layers in the rightmost branch of the pyramidal segment from 512 to 256 (Fig. 4), and

increasing the sizes of the kernels in the two central Conv2DTranspose layers (Fig. 4), a segmentation accuracy of 71.1% was achieved on the 185th epoch. It should be noted that the considered neural networks are trained quite a long time, so it is necessary to have at least a hundred epochs to achieve the indicated results.

The indicated accuracy levels were actually the maximum for the PSPNet

architecture, although they can probably be improved by further optimization of the settings of the sizes of the convolutional layer kernels. At the same time, a significant

drawback of the considered PSPNet structures is the noticeable stratification of the accuracy curves into two clusters.

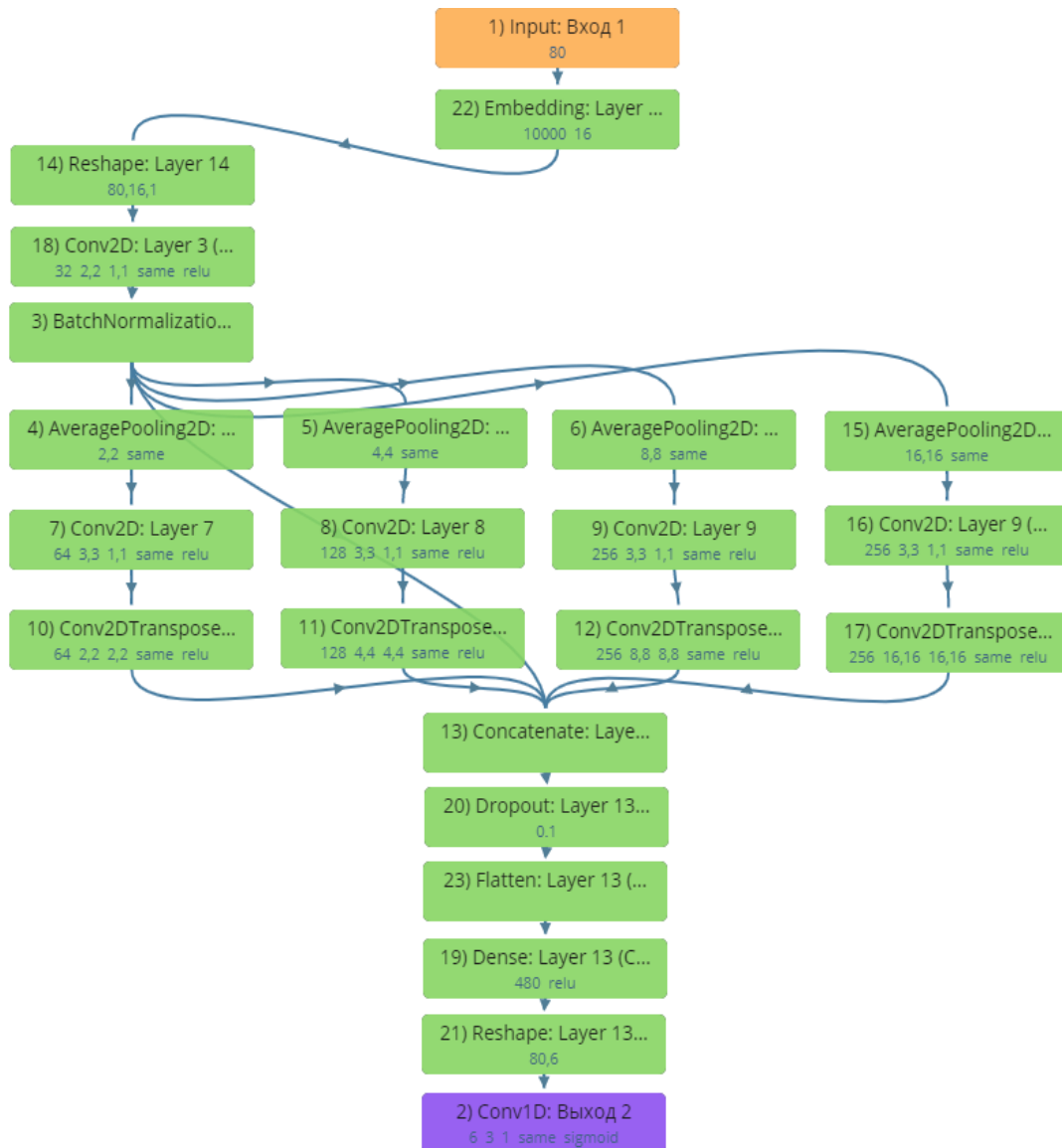


Fig. 4. Improved PSPNet with an expanded Embedding vocabulary

More promising hopes were placed on the results to be obtained using the U-Net neural network [8]. Fig. 5 shows its architecture in the form of an autoencoder, when the U-Net removed the connections for flipping images from the input to the output. As a result, the typical U-Net neural network structure has transformed into a combination of encoder and decoder.

Figs. 6 and 7 show a close-up of the input and output segments of this structure, respectively. Here, the input text array is combined in the form of 80 word vectors, then

their dictionary of 20,000 tokens for embedding is formed, and the Reshape operation is performed to obtain an image format of 80×80 pixels. The result of the training was a text segmentation accuracy of 82.7% at 60 epochs with a batch of 32 and a training step of 0.001.

Subsequently, the architecture was further complicated, in particular, the UNet++ neural network [9] (Fig. 8) was used, which is one of the most complex and had previously been used for image processing.

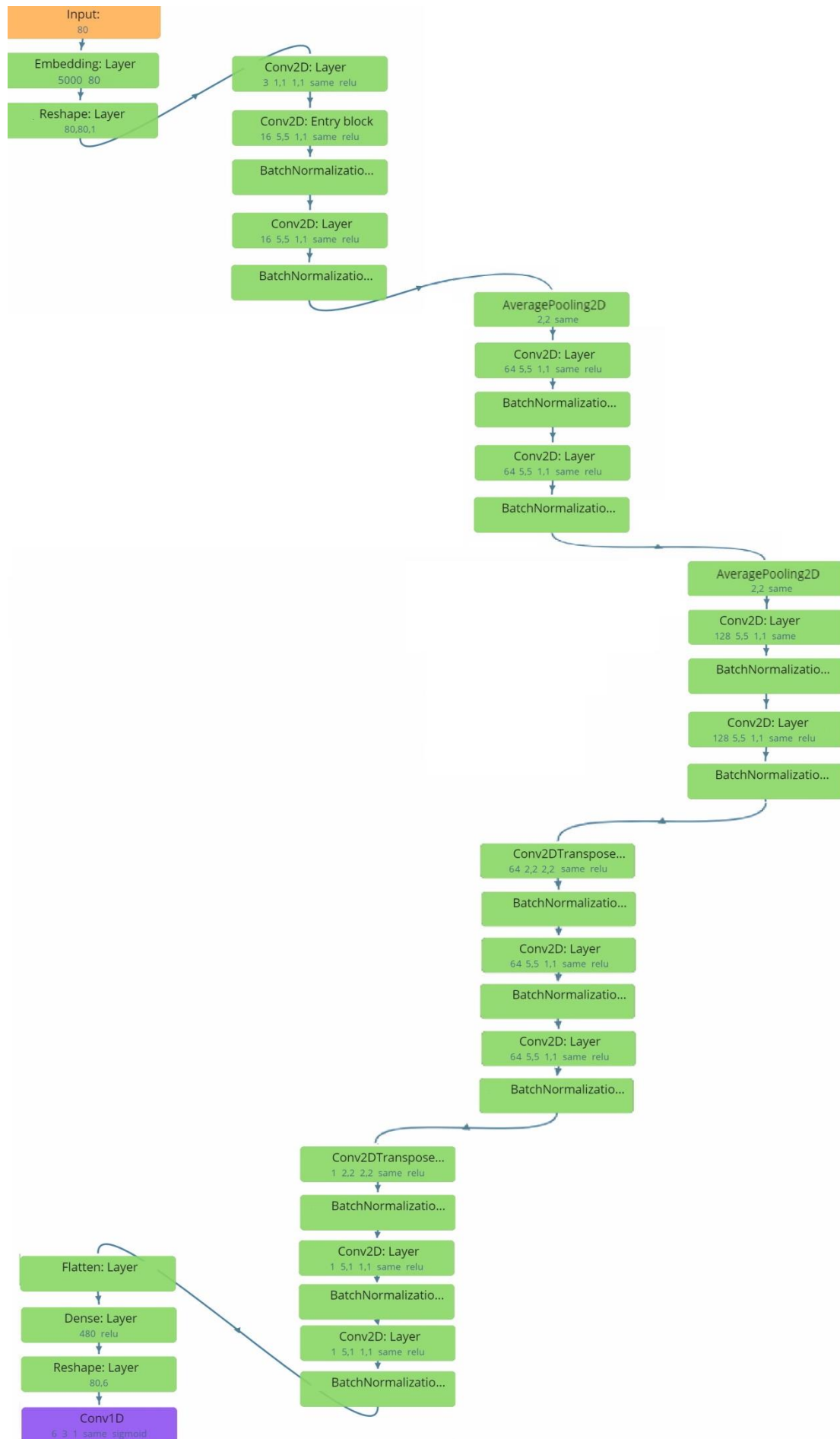


Fig. 5. Autoencoder variant of U-Net

In this case, a similar structure of input and output segments was integrated to move

from the text array, make embedding, and then manipulate quasi-images (Figs. 9, 10).

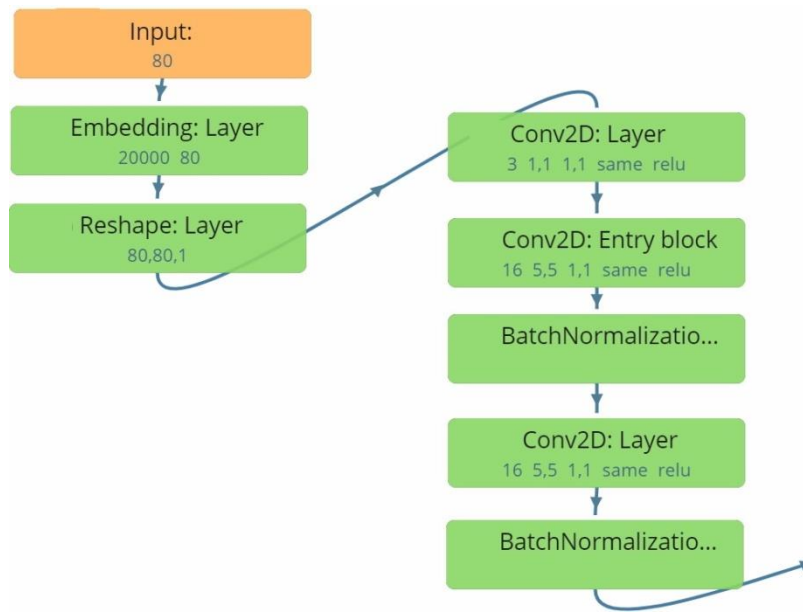


Fig. 6. Input cascades of the autoencoder variant of U-Net

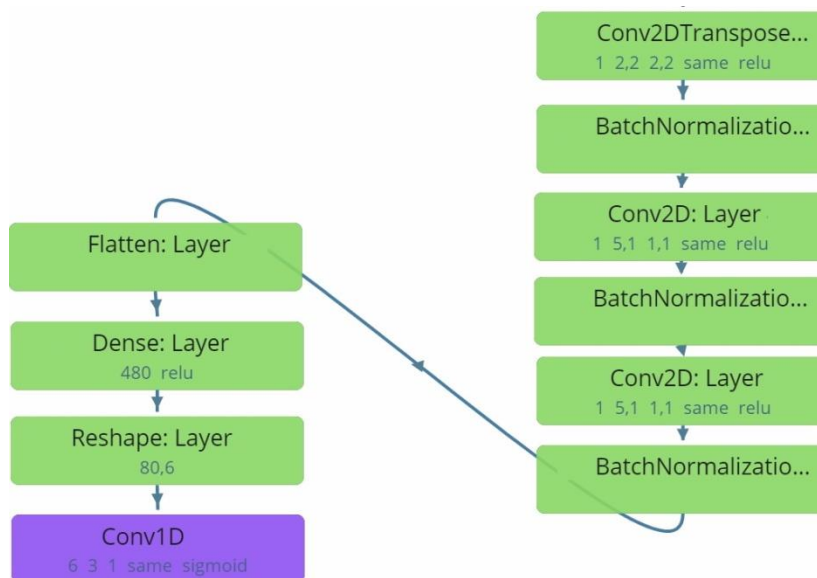


Fig. 7. Output cascades of the autoencoder variant of U-Net

The Embedding layer provided the formation of a data array in the format of 5000×80. The return to a smaller vocabulary size in embedding was due to the increased computational complexity of the overall neural network architecture, which exhausted the available resources of the Google Colab environment where calculations were conducted.

Meanwhile, the inclusion of a Conv2DTranspose layer with 3 filters and a

kernel size of 2x2 and the same stride right after the Reshape operation (Fig. 9) allowed for processing quasi-images of 160x160x3 pixels. It should be noted that for the neural network, the nature of the data array does not matter. It does not orient itself in this at all, the main thing is that the user can then correctly identify the results. In other words, the neural network does not care what is at the input: text or image, it will work the same according to its purpose.

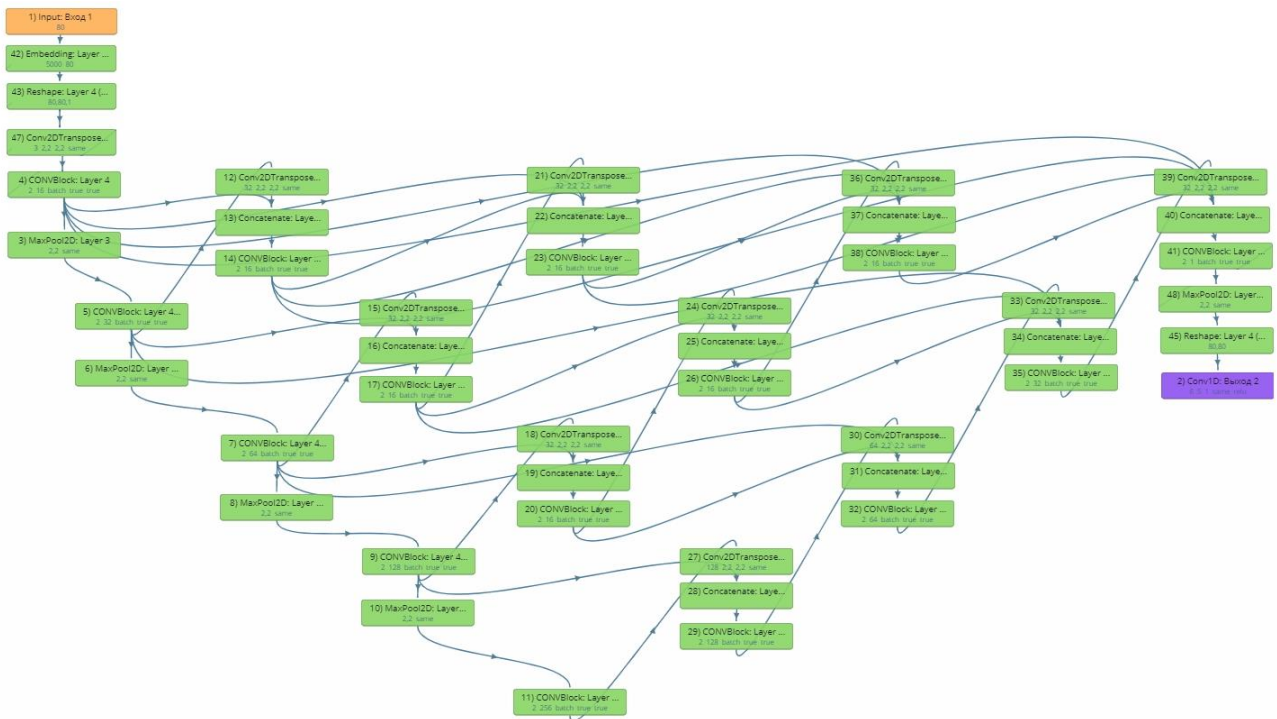


Fig. 8. Structure of U-Net++

The transition to the UNet++ architecture in this case allowed achieving a record accuracy of 87.3% for the segmentation of the test sample of the specified dataset. This was achieved on the 671st epoch of training, the chosen strategy of which anticipated an initial step of changing the weight coefficients of 0.001, moving to a training step of 0.0001 on the 220th epoch. Significantly, the training dataset provided an accuracy of 93.1% for segmentation. Such a lag behind 100% indicates the potential for further increasing accuracy on the test sample, until the accuracy on the training array reaches 100%. The recorded difference of 5.8% in accuracies on the test and training samples allowed

predicting that with an increase in the number of training epochs, the accuracy on the test sample might exceed 94%. To increase the likelihood of such an increase in accuracy, it would also be advisable to try to increase the size of the embedding dictionary to 20,000 tokens. Fulfilling these conditions confirmed the assumption made, but it required extending the training to 2000 epochs. As a result, with 20,000 tokens in the Embedding dictionary, an accuracy of 94.4% was achieved on the 1632nd epoch of training. This result almost equaled the maximum accuracy of 94.7%, which was provided by the application of the one-dimensional PSPNet architecture on the same dataset (Fig. 11).

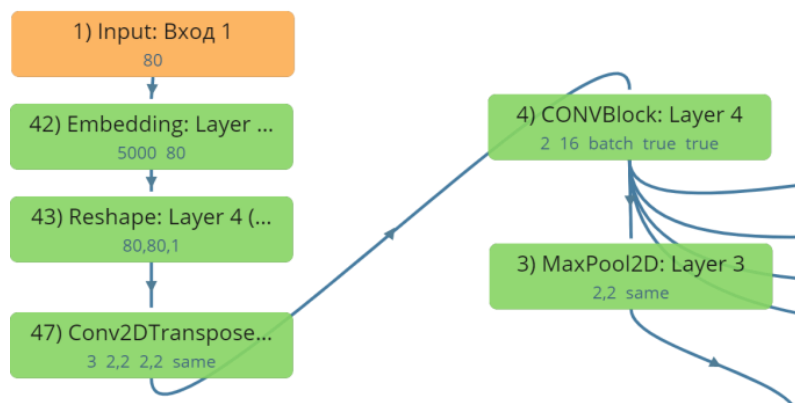


Fig. 9. Input cascades of the modified variant of U-Net++

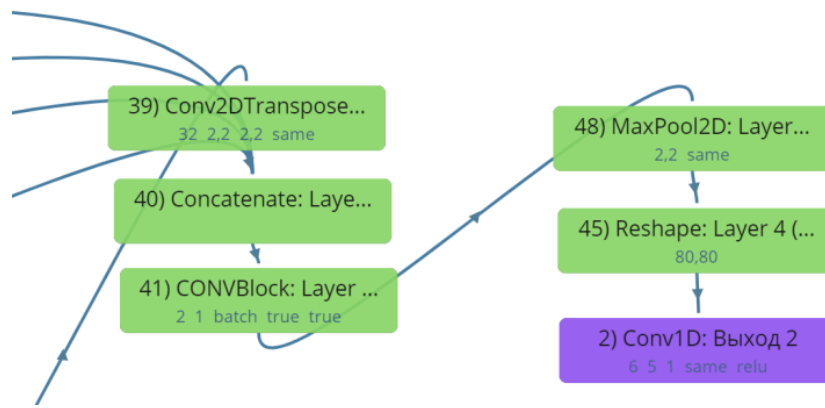


Fig. 10. Output cascades of the modified variant of U-Net++

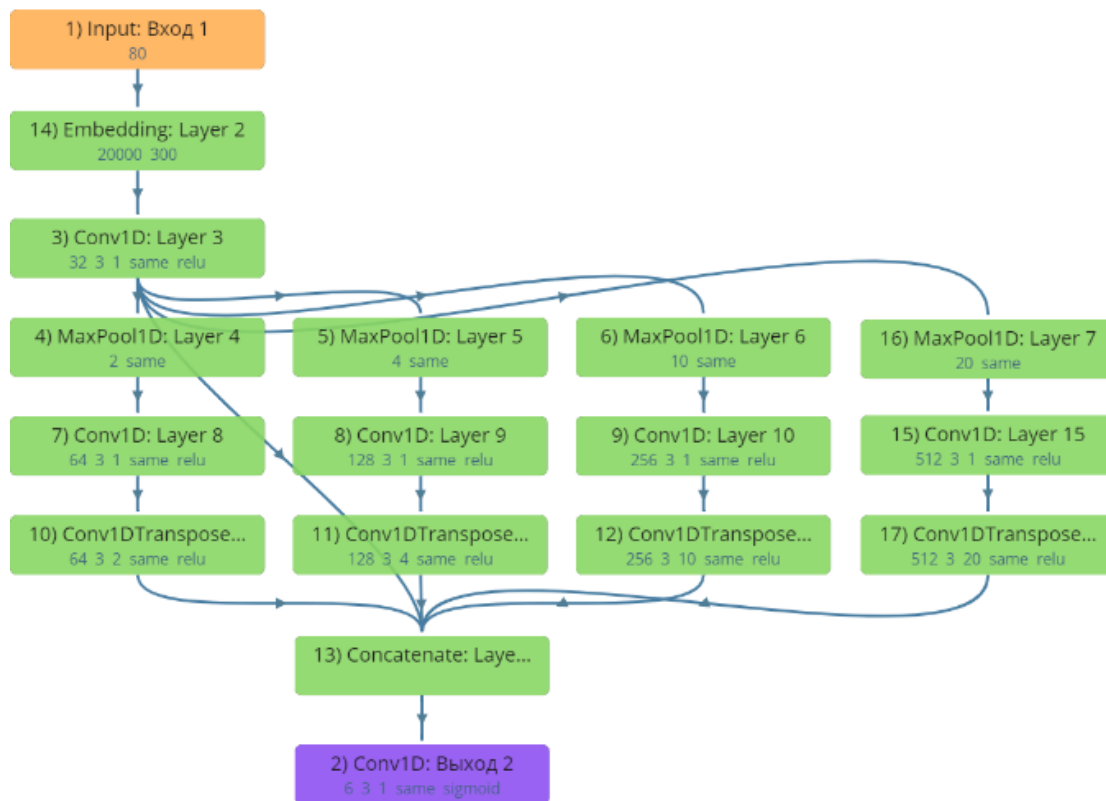


Fig. 11. Structure of the one-dimensional PSPNet specialized for text segmentation

Thus, the conducted studies have proven the possibility of effective segmentation of texts using neural network architectures adapted for image processing. In this case, as in [2], the transition from texts to images was made in a simpler way than was proposed in [10].

At the same time, the use of this approach did not allow exceeding the accuracy achieved on text segmentation neural networks based on one-dimensional Conv1D convolutions. Such networks, thanks to their simpler architecture, are subject to automatic hyperparameter optimization based on AutoML technologies [11], while it is quite

difficult to involve AutoML for optimizing the settings of structures like 2D UNet++.

Conclusions

The proposed approach of using image segmentation neural networks for text processing accelerates the speed of text data analysis by feeding the textual array to the neural network in a tensor-matrix form instead of a vector. In this way, segmentation of several different texts can also be carried out in parallel for further comparison of their segmentation results to determine similarity or to select a more relevant text for a given topic.

Moreover, the conducted research confirmed the hypothesis expressed in [2] that the same neural network can be effectively used both for text segmentation and for image processing. In this case, before changing the mode of use, a multimodal neural network should load the weight coefficients relevant to the specific task, with coordinated switching of the structure of the input segment of the neural network for feeding images, bypassing the Embedding and Reshape layers, which are only intended for text processing. A similar readjustment should also be provided in the output segment by removing from the data transfer chain the layers that convert 2D arrays into one-dimensional text. With sufficient computational resources, the Embedding operation at the input of the neural network can also be used for image processing, which will reduce the amount of intervention in the architecture when switching from text segmentation mode to similar image processing. This will additionally enable the segmentation of textual inscriptions on images, as well as carry out semantic or instance segmentation of images according to the given categories of text segmentation.

Further research should focus on using a wider range of text datasets with different sets of segmentation categories and considering transformer-informer architectures.

References

1. GPT-4. Technical Report by OpenAI, 27 March 2023. URL:<https://arxiv.org/pdf/2303.08774v3.pdf>.
2. Slyusar V. Classification of text as images using neural networks pre-trained on the ImageNet dataset. // *Artificial Intelligence*, 2023, №95(1). - Pp. 37- 47. - DOI: 10.15407/jai2023.01.037.
3. Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference

on computer vision and pattern recognition, pages 248 – 255. IEEE, 2009.

4. Krizhevsky, Alex; Sutskever, Ilya; Hinton, Geoffrey E. (2017). ImageNet classification with deep convolutional neural networks, *Communications of the ACM*. 60 (6): 84–90. doi:10.1145/3065386.

5. Slyusar V. Architectural and mathematical fundamentals of improvement neural networks for classification of images. // *Artificial intelligence*, 2022, №1.- Pp. 127 - 138. DOI: 10.15407/jai2022.01.127.

6. Slyusar V.I., Sliusar I.I. (2021) Lions of Neural Networks Zoo, Neyromerezhni tehnologiyi ta yih zastosuvannya NMTIZ-2021: zbirnik naukovykh prats XX Mizhnarodnoyi naukovoyi konferentsiyi «Neyromerezhny tehnologii ta yih zastosuvannya NMTIZ-2021», Kramatorsk: DDMA, 129 -133, DOI: 10.13140/RG.2.2.17187.58405.

7. Vadym Slyusar, Mykhailo Protsenko, Anton Chernukha, Vasyl Melkin, Olena Petrova, Mikhail Kravtsov, Svitlana Velma, Nataliia Kosenko, Olga Sydorenko, Maksym Sobol. Improving a neural network model for semantic segmentation of images of monitored objects in aerial photographs. // *Eastern-European Journal of Enterprise Technologies.- № 6/2 (114). – 2021. - Pp. 86 – 95*.DOI: 10.15587/1729-4061.2021.248390.

8. Dice, Lee R. “Measures of the Amount of Ecologic Association Between Species.” *Ecology*, vol. 26, no. 3, 1945, pp. 297–302. JSTOR, DOI: 10.2307/1932409.

9. Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang, “Unet++: A nested unet architecture for medical image segmentation,” in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pp. 3– 11. Springer, 2018.

10. Benarab Charaf Eddine. Classifying Textual Data with pretrained Vision Models through Transfer Learning and Data Transformations. // Feb. 7, 2022, 7 p. arXiv:2106.12479v4.

<https://arxiv.org/pdf/2106.12479.pdf>.

11. Vincent, A.M., Jidesh, P. An improved hyperparameter optimization framework for AutoML systems using evolutionary algorithms. *Sci Rep* 13, 4737 (2023). <https://doi.org/10.1038/s41598-023-32027-3>

The article has been sent to the editors 03.12.23.

After processing 25.12.23.

Submitted for printing 20.03.24.

Copyright under license CCBY-SA 4.0.