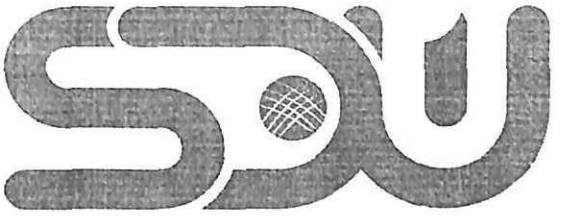
Ministry of Education and Science of the Republic of Kazakhstan
Suleyman Demirel University

UDC 50.10



Aimoldir Aldabergen

Image scene understanding by using Recurrent Neural Networks

THESIS

Presented in Partial Fulfillment for the
Degree of Master of Science in Computing Systems and Software
(degree code: 6M070400)
Department of Computer Sciences
Faculty of Engineering and Natural Sciences

Supervisor: **Abay Nussipbekov**

Kaskelen, 2020

Abstract

This work proposes an implementation of Recurrent Neural Networks (RNN) for image scene understanding. Task is clear: given an image and the system should provide an accurate description for the given image. The novelty of the work is that this system is realized on Telegram Bot. Fine tuned model learns where to look, its focus is shifted across the image by the help of attention mechanism. Thus the model was able to find the most relevant parts of the image and find out most relevant words that describe the scene. It has an encoder-decoder architecture. As own contribution, transfer learning is implemented on pre-trained model. The significance of the work is that this kind of system can be easily implemented in bunch of areas of our life rather than other capturing applications.

Аңдатпа

Бұл жұмыс бейненің көрінісін түсіну үшін қайталанатын нейрондық желілерді (RNN) енгізуді ұсынады. Тапсырма айқын: берілген сурет пен жүйе берілген суреттің нақты сипаттамасын беруі керек. Жұмыстың жаңалығы - бұл жүйе Telegram Bot-та іске асады. Жақсы баптalған модель қайда қарауға болатындығын біледі, назар фокустау тетігінің көмегімен кескін бойына ауысады. Осылайша модель кескіннің ең маңызды бөліктерін тауып, көріністі сипаттайтын ең қажетті сөздерді таба алды. Онда декодер-архитектурасы бар. Өз салымы ретінде трансферттік оқыту алдын-ала дайындалған үлгі бойынша жүзеге асырылады. Жұмыстың маңыздылығы мынада, мұндай жүйені біздің өміріміздің көптеген салаларында оңай қолдануға болады.

Аннотация

В этой работе предлагается реализация Recurrent Neural Networks (RNN) для понимания сцены изображения. Задача ясна: дано изображение, и система должна предоставить точное описание для данного изображения. Новизна работы заключается в том, что эта система реализована на Telegram Bot. Тонко настроенная модель узнает, где искать, ее фокус перемещается по изображению с помощью механизма внимания. Таким образом, модель смогла найти наиболее важные части изображения и найти наиболее подходящие слова, которые описывают сцену. Он имеет архитектуру кодер-декодер. В качестве собственного вклада, трансферное обучение осуществляется по предварительно обученной модели. Важность этой работы заключается в том, что такого рода системы могут быть легко реализованы во многих областях нашей жизни, а не в других приложениях захвата.

Acknowledgements

I want to thank my thesis supervisor, Nussipbekov Abay, for directing me and constant support, useful discussions, that helped to significantly improve the current work. Thanks to my university that provided me with all these opportunities. Also, thanks a lot to my spouse for all kinds of support.

Contents

Introduction **7**
1.1 Background 7
1.2 Significance of Research 12
1.3 Motivation 14
1.4 Aims and Objectives 14
1.5 Thesis Outline 15

Literature review **17**
2.1 Existing Knowledge 17
2.2 Significant Prior Research 22

Materials and Methods **34**
3.1 Detailed model description 34
3.2 Attention mechanisms 37
3.3 Dataset 40

Results and Discussion **46**
4.1 Encoder 46
4.2 Decoder 47
4.3 Attention mechanism 48
4.4 Overall model workflow 49

Further research and future work **51**
5.1 Methods 52
5.2 Results and discussion 54
5.3 Conclusion 56

1. Introduction

1.1 Background

In the past several decades, deep learning, which can be described as a class of machine learning algorithms, has shown noticeably good results across a variety of problems. Nowadays, computer vision systems are tested by their accuracy in detecting and localizing instances of objects. It is one of the popular topics in the field of deep learning. Specifically, this work's resulting product is a system that is used for image scene understanding, by generating description based on the image given. Image understanding will be realized by using LSTM. Image is a depiction of visual perception or object in a binary form that carries a very large amount of information. However, for computers, to parse all that information is not as easy as for humans, but possible. The concept is very simple: formatting input into a data structure that concisely summarizes the visual concept in the given image by the help of attention mechanism and beams search. In one word it is all about accuracy. Nowadays, recognition accuracy of deep learning have reached the highest plane than ever before. For instance, on a daily life DL can help electronics satisfy user demands in much more better level, it is used in automation of processes in almost every field of life. In one of the recent articles about DL it is stated that latest approaches in DL has reached the point where it outperforms humans, in some tasks like object classification [1]. Deep learning has recently arose as a highly successful paradigm for big data. By the means of technological advances deep learning has given noticeable contributions in several basic artificial intelligence tasks in the fields of image processing and computer vision. Image is a depiction of visual perception or object in a binary form that carries very large amount of information. However, for computers, to parse all that information is not as easy as for humans, but possible.

Image captioning or image scene understanding is one of the most important components of automatic systems, artificial intelligence, computer vision, supervision and scouting or intelligence service. Particularly this kind of applications and systems can enormously profit because of the opportunity that they are able to classify content of an image or any scene automatically.

For instance, an Uninhibated Airy Vehicle (UAV) in an Intelligence Supervision Reconnaissance would afterward be able to process the video-streams automatically and dispatch information only about frame description to a demander, which is much more faster and efficient that sending whole video through downlink.

Image scenes contained of different objects and environment (background) that surround these objects, where both of them are important, informative and useful for tasks of computer vision systems that are aimed to distinguish the content of the image scene.

For this kind of tasks most succeed and efficient methods are based on Recurrent Neural Networks, namely Convolutional Neural Networks, they show state of the art results on image scene clustering by the help of all the features that can be extracted from image features on enormous datasets of images, however they do not precisely make use of information about objects while they actually can be made use of.

Objects in the image are able to give a lot of information about the general scene of the image. For example if there is a car, it means that there is a road on the image scene. In addition to that these objects that occur on the image are able to help to classify and recognize image scenes which can seem to be similar, while looking through feature level of the whole image. For instance, the images where in first case an airplane is on a runway and in the second case an airplane is on the road. Unfortunately, in such cases only detection of an object can be not so helpful and not enough to classify the whole image scene. Also, second problem is when objects are not required to be detected, like in images of big desert or ocean, In this cases RNN based models can help to solve them and show state of the art results compared to other models.

There are a number of large data-sets of images that are used for deep learning tasks like classification of images, recognition of objects. As a result, according to the accuracy on performing such kinds of tasks computers can perform better

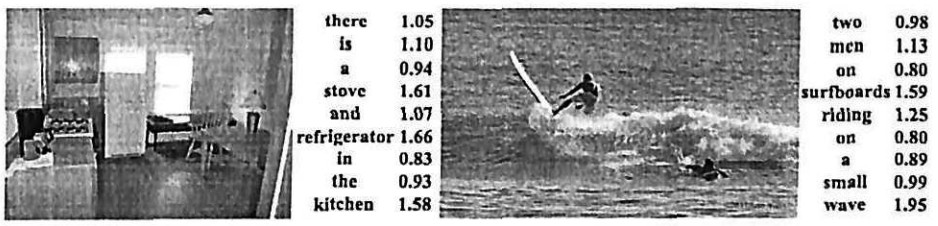
than humans. In these latter days, generating the description of an image automatically is one of the most attractive tasks of artificial intelligence (AI). This task, alias image scene captioning, takes a significant role in AI, for example, empowering technology (computers) describe and understand image scenes can be implemented in wide range of applications, like tracking video streams, retrieval cross-view, analysis of sentiment, education and descriptive decline exoneration. However, for computers, to parse all that information is not as easy as for humans, but possible, as it covers two very complex fields of AI: natural language processing and computer vision technologies. Besides the task of object recognition in an image, the system must also analyze the states of these identified objects, identify the relationship and connection between them, and eventually provide with description about all of them in natural human language.

There are number of methods and techniques that are used for compilation of this task. For instance, previous attempts on the task of image scene understanding mostly assume the methods based on templates, where recognition of objects and their attributes, relationships are required on the first phase. Afterward all identified elements are required to be organized into sentences based on either pre-defined language models [2] or templates [3]. The next method is a true to type which can be referred to transfer-based ones, where nearest neighbor algorithm is implemented to extract a scene specification from the input image [4]. However this method cannot generate any novel sentence, it suggests that NN can indeed provide valuable information.

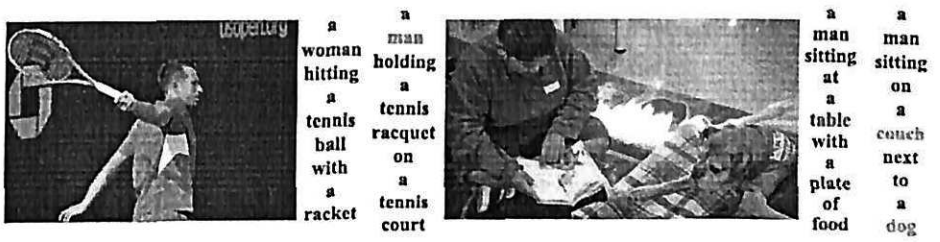
Thanks to the last achievements in CV and Machine Translation [5], methods that are based on neural networks, were widely implemented for image captioning and image scene understanding tasks [6] and shown state of the art results. Most of them are constructed with two steps, in forms of encoder and decoder. On the first step features of the image are retrieved by the help of convolutional neural networks. Secondly, recurrent neural network (RNN), namely LSTM [7] is used for decoding the output of the encoder, which constructs a natural language description according to extracted image features. This kind of methods are flexible because of combination of 2 steps described above. In addition to that, nowadays attention mechanism also helped to achieve better results. Number of different attention mechanisms were implemented for image scene understanding, like visual, semantic, spatial, region-based, local, global, channel-wise attention

mechanisms [8].

According to the researchers result, even if there are a number of outstanding achievements, currently constructed methods, that include CNN and RNN have several drawbacks, illustrated in Fig. 1:



(a) Information inadequateness problem



(b) Misrecognition problem

Figure 1.1: Drawbacks of existing methods

- Inadequacy of the retrieved information: current methods see the words of an image description equally, even if they are different and thus make differentiation of main image description parts hard.
- Problem of misrecognition: important scenes and subjects can be misrecognized by implementing methods that are traditional.

Actually, words are not important while creating a caption for given image. Even if the subject or scene is not recognized correctly, the errors that are generated accumulate and cannot simply be corrected. The next word in the title can be interrupted by these irrelevant contexts. Therefore, system can be suggested, that uses visual features and exercise picture inscriptions to solve the above problems. References are included in the training and creation phase of the LSTM framework and form the new R-LSTM model. While training part, words with different weights are listed according to the target image, part of the language and the meaning of this synonym. Words with a high relevance rating are important for the image description, so more weight is given for loss computation. Such

way, the model can know more about inscriptions, defining important objects, important properties and their relationship to each other.

Similar task for image scene captioning is visual question answering, which is one of the most popular tasks of Computer Vision nowadays, as it includes image captioning in itself. The second reason why most of the previous works that are done on this task were not very accurate is that they used to deal with exactly defined syntax (sentences) and hard-coded features [2]. The reason for that is the model which was built on exact descriptions for exact images. In order to overcome such a problem it is necessary to make the model independent of such hard-coded descriptions. Main purpose of this work is to develop a model that can perform this task as accurate as possible. If normally humans use their languages for image captioning process, machines are supposed to use it also. Image is a visualization of perception or object in a binary form that carries very large amount of information. However, for computers, to parse all that information is not as easy as for humans, but possible. There are huge project works that are done on this topic. Actually this kind of systems may differ according to the type of the questions and images. For instance, the work "Visual Question Answering" done by Aishwarya Agrawal, Jiasen Lu et al. proposes a system that can answer natural language questions about the content of given image based on its features. Their proposed system can be given one image and may be asked about every concept of that image including background (Fig. 1) and such kind of questions are called open ended free-form questions [9]. The next kind of system

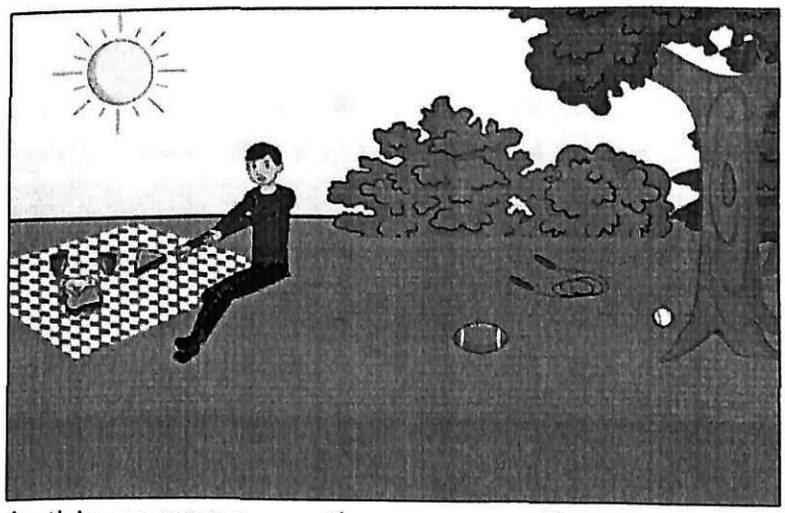


Figure 1.2: Open ended free-form questions sample

proposed by Yash Goyal, Tejas Khot et al. is about asking one question and giving similar but counter images, in order to get different answers (Fig. 2) [10]. Also

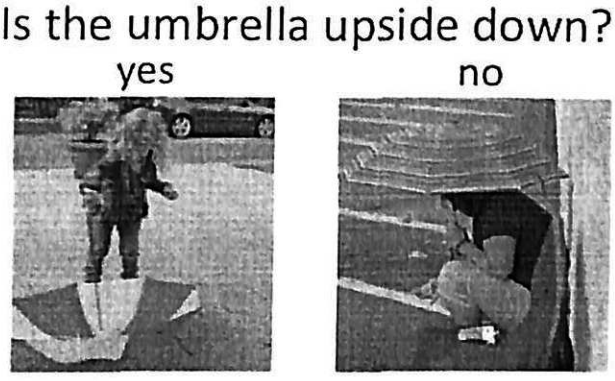


Figure 1.3: Counter images and questions sample

the third way is asking "yes" or "no" questions (Fig. 3) which is the result of the work called "Yin and Yang: Balancing and Answering Binary Visual Questions". The concept is very simple: it can be realized by formatting the question into a data structure (eg. tuple) that concisely summarizes the visual concept to be detected in the given image. If the visual concept and the converted question words intersect with each other (if the concept can be found in the image), the answer will be positive, otherwise will be negative [11]. There are also other plenty

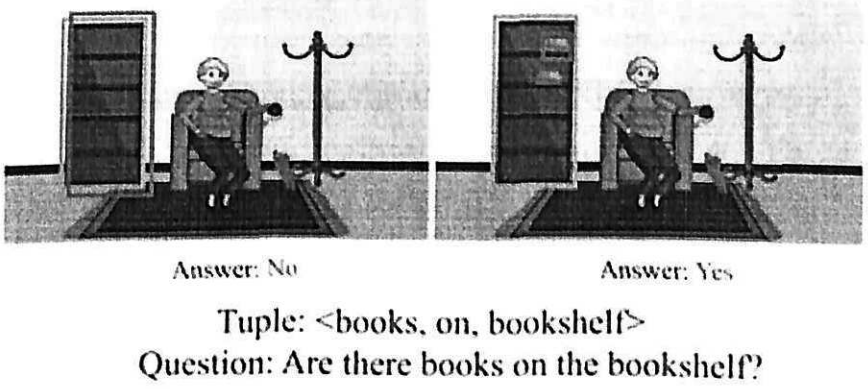


Figure 1.4: Binary Visual Questions sample

of works that are done on this field of research.

1.2 Significance of Research

This work proposes an implementation of Recurrent Neural Networks (RNN) for image scene understanding. Task is clear: given an image and the system should

provide an accurate description for the given image. The system is realized on Telegram Bot. Fine tuned model learns where to look, its focus is shifted across the image by the help of attention mechanism. Thus the model was able to find the most relevant parts of the image and find out most relevant words that describe the scene. It has an encoder-decoder architecture and transfer learning is implemented on pre-trained encoder.

In order to overcome problem that is mentioned above, about hard-coded descriptions, it is necessary to make the model independent of such hard coded descriptions. Therefore question organizing will be realized by the help of LSTM layers and image operations are handled by using CNN model where also attention mechanism will be applied. The key point is top-down attention mechanism, that makes it available to caption the given image and compare given question and image features (detected objects). This model attention mechanism consists of two layers: language LSTM and attention LSTM: it is very simple and effective which will definitely increase the accuracy and decrease time consuming processes like preparing hard coded descriptions.

Image scene understanding can be classified in different ways according to its purpose. It is known that image scene understanding is about the objects inside the image as well as their positions and relations to each other, about background of an image. Proposed long short term memory model is evaluated on two datasets: COCO and Flicker dataset with 30 000 images. The results of the comparison show the remarkably better result of LSTM. The results of the MS COCO Image descriptioning method is also reported. If to compare to other current methods, this model shows indicators that are comparable. This proposed model has three main aspects:

- Reference is image that is used for training and new model is developed for generating image scene descriptions
- Words and their weights in the image description are determined based on events. Such type of training helps to understand the information about the description in more details.
- The new target function is defined by estimates with the help of visual references. Neighbors target images and traditional logarithmic probability. Making connections can resolve misunderstandings and make the caption

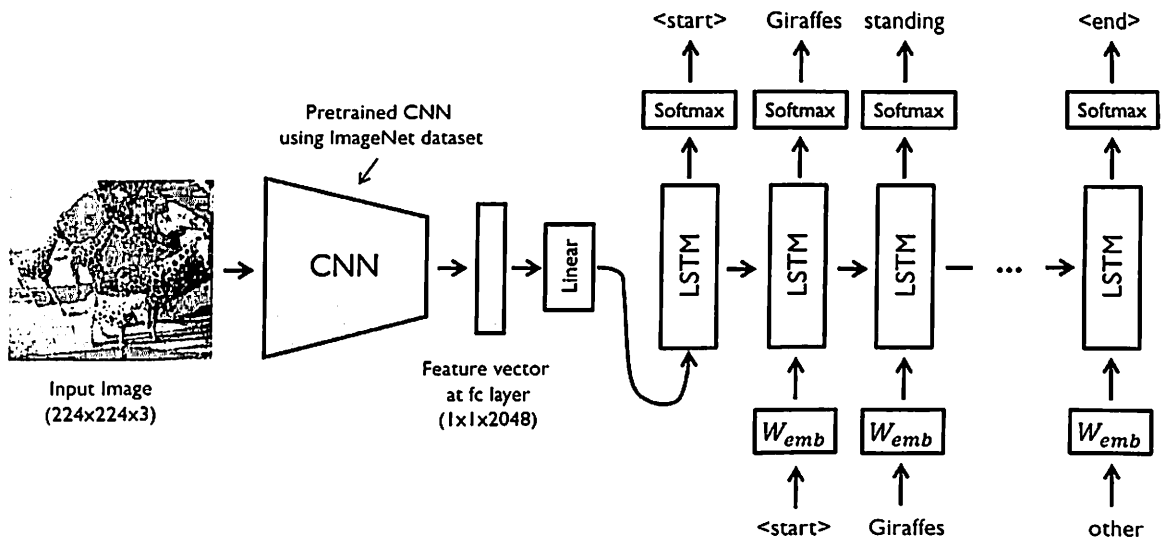


Figure 1.5: LSTM model

much more realistic, like real human language.

1.3 Motivation

Image scene understanding is one of the very important components of computer vision, automatic systems, artificial intelligence, supervision and scouting or intelligence service. Particularly this kind of applications and systems can enormously profit because of the opportunity that they are able to classify content of an image or any scene automatically.

My own contribution is a system itself. First of all, there will be another approach for solving this trending problem and it can lead others to other research works, they can see new ideas from my work. The result of the work cancan enormously profit because of the opportunity that they are able to classify content of an image or any scene automatically.

1.4 Aims and Objectives

Aim of the work is to develop a system that will create an accurate description for a given image and to construct a new model based on the old one through implementing transfer learning. In order to reach the aim the objectives are:

- Search for redoable works on image scene understanding and research

- Find suitable dataset (balanced, not biased)
- Define the methods and materials
- Construct the model. implement the methods, algorithms
- Test and evaluate
- Optimize
- Implement attention mechanism, fine-tuning
- Test final model
- Optimize
- Define further works and derive conclusions

Potential outcome of this dissertation wok is a system that provides a description for different kind of images. So, in this work, implementation of deep learning for image scene understanding with the help of Recurrent Neural Networks is developed.

1.5 Thesis Outline

The current chapter is Introduction chapter. It contains insight into the general work done. It includes the background information about the problem, what is the significance of the research work, motivation for doing this work (based on real world facts and researches of other scientists) and aims, objectives of doing this research work. the next chapter is "Literature Review", where some of the related works on image scene understanding and image captioning were briefly reviewed and it also includes significant prior research . Chapter 3 called "Methods and Materials" gives general overview of constructed model and explanation of the structure of the model and used materials: dataset. Also detailed algorithms are described in this section and possible solution to the problem is described. This chapter is important as by the help of detailed information here the current work can be redone by other interested researchers. All results and analysis of the research work are presented in Chapter 4: :Results and Discussion which is

followed by the last and concluding chapter, which includes summary of the work and possible estimations for the further works.

2. Literature review

2.1 Existing Knowledge

As computers do not have intelligent reasoning, detecting objects on the image and determining their relationship by constructing the context of the image is a challenging task for them. Indeed there are a lot of works done on this topic as an intersection of vision and language is now one of the significant questions of modern research.

There are two most common ways of doing image captioning: generative and retrieval based methods. As an example of the retrieval-based method we can take Im2Txt model [12]. The general system is contained of two parts: matching the image and generation of the caption. When an image is given to the system it will be matched with the image and its caption which is in the database. After the images are matched, received objects of high level will be compared with original input data. One of the disadvantages of this model is being overfitted, it can construct captions only from already prepared ones in the dataset. However, this problem is solved in generative models. Because generative models let to construct new sentences (descriptions) for the model.

The model proposed in the work of Vinyals et al. is one of the examples of the generative type of models. This model implements neural machine translation and newest image recognition techniques, by using LSTM and model of inception [13].

If to look at researches on the task of image scene understanding and visual question answering (where the main task is image captioning) they range from constrained settings to freeform natural language questions and answers. For example, the authors of research work "A visual Turing test for computer vision systems" propose a system to generate binary questions from templates using

a fixed vocabulary of objects, attributes, and relationships between objects [14]. They constructed a "visual Turing test": an operator-assisted device that produces a stochastic sequence of binary questions from a given test image. The engine will construct some question, afterward question will be checked by the operator and either the answer will be given or will state the question was ambiguous.

Indeed there are a lot of works done on this topic as intersection of vision and language are now one of the significant questions of modern research. If to look at researches on the task of visual question answering they range from constrained settings to free form natural language questions and answers. For example, the authors of research work "A visual Turing test for computer vision systems" propose a system to generate binary questions from templates using a fixed vocabulary of objects, attributes, and relationships between objects [14]. They constructed a "visual Turing test": an operator-assisted device that produces a stochastic sequence of binary questions from a given test image. The query engine proposes a question; the operator gives the correct answer otherwise rejects the question as not clear, ambiguous; and the system will request the next question.

Another research work called "Video And Text Parsing for Understanding Events and Answering Queries" presents a framework that is does multimedia analysis like processing video streams and text together in order to understand what is happening on frame, what is the event [15]. The system constructs a graph that shows the composition structures of scenes and objects, events and actions, causalities between events in text and video.

Generally recent papers are mostly about neural network models for Visual Question Answering systems that also implement Long Short Term Memory and neural networks. For instance, research group lead by Nobuyuki Shimizu, from Yahoo Japan Corporation, have created a dataset for Visual Question Answering system by the help of descriptions with images from VG dataset. As another contribution, they present a method that works multi-lingually in order to take an advantage of English language models in order to improve Japanese systems [16].

Indeed there are a lot of works done on this topic as intersection of vision and language are now one of the significant questions of modern research. As computers do not have intelligent reasoning, detecting objects on the image and determining their relationship by constructing the context of the image is chal-

lenging task for them.

There are two most common ways of doing image captioning: generative and retrieval based methods. As an example of retrieval based method we can take Im2Txt model [17]. This model was constructed by Vicente Ordonez, Tamara L Berg and Girish Kulkarni. General system is contained of two parts: matching the image and generation of the caption. When an image is given to the system it will be matched with the image and its caption which is in the database. After the images are matched, received objects of high level will be compared with original input data. One of the disadvantages of this model is being overfitted, it can construct captions only from already prepared ones in the dataset. However, this problem is solved in generative models. Because, generative models let to construct new sentences (descriptions) for the model. Model proposed in the work of Vinyals et al. called ""Show and tell: Lessons learned from the 2015 mscoco image captioning challenge" is one of the examples of generative type of models. This model implements neural machine translation and newest image recognition techniques, by using LSTM and model of inception-v3 [18].

If to look at researches on the task of visual question answering (where the main task is image captioning) they range from constrained settings to free form natural language questions and answers. For example, the authors of research work "A visual Turing test for computer vision systems" propose a system to generate binary questions from templates using a fixed vocabulary of objects, attributes, and relationships between objects [15]. They constructed a "visual Turing test": an operator-assisted device that produces a stochastic sequence of binary questions from a given test image. Engine will construct some question, afterwards question will be checked by the operator and either the answer will be given or will state the question was ambiguous. Another research work called "Video And Text Parsing for Understanding Events and Answering Queries" propose a multimedia analysis framework to process video and text jointly for understanding events and answering user queries [19].

Currently existing systems that are aimed to generate a description for image scenes can be divided into three types according to their way of working (how do they generate the descriptions) and structures:

- methods that are based on transformation
- methods that are based on templates

- methods that are based on neural networks

The methods that are based on transformation use generally approaches that are retrieval, where descriptions are extracted directly from the given input image. Some of the accesses take the given input scene like a query and select some description in space embedding of joined image and sentence. Actually this kind of images and their descriptions have one characteristic that extracted images are the very similar with given input images and retrieved segments/phrases from their captions. Eventually these segments are collected into one description that is called an image caption. Descriptions that are generated with the help of methods that are based on transformation often have right grammar. However, this kind of methods can have errors while looking at content of an image visually. Also one of the limitations of these methods are that their models are unable to construct new phrases and sentences. However, despite this fact, researchers state that there is an advantage of having similar images to the inputs, because they are used for implementation for performing different tasks like evaluating the descriptions constructed by other models.

The methods that are based on templates need either a designed model for language or some kind of template. This kind of model works in a following way: template slots are filled based on relations of points, their co-occurrence, or random fields that are conditional or multi dimensional data that are scales of web. There are some other models that are complicated and which is able to construct agile sentences.

Mitchell et al. propose trees that are syntactic that construct model that is driven by data. For representation of the relationships between objects is used dependency that is visual [20]. This kind of methods that are based on templates are very user friendly and not complex. However, there is a disadvantage, because they are hard-coded, hand-designed and not expressive. Therefore they are not flexible, which would help to construct sentences with meaning. Methods that are constructed on the basis of NNs are showing state of the art results because of a big development in RNN based machine translation.

There is an another work that proposes a model with layers with multiple modes, which connect CNN and RNN and lets to construct a description word by word given the image and generated previous word. In machine learning there is a model that has encoder-decoder architecture. Group of researchers lead by

Vinyals constructed a model that uses CNN in order to encode the input image. Also the model includes long short term memories layer [21]. As a decoder which transforms vector to the description of an image RNN is used.

Actually there are other works that implement this idea of application of attention mechanism, as generating a description by extracting it from layers that are convolutional is more effective than layers that are connected fully. Model is able to focus on different locations at an image or on a different objects.

Attention mechanism can be driven also by combining features of the whole image and description words extracted from that image by using detectors of an attribute. Attention mechanism can be different, researcher Li combined two types which are global and local, thanks to it the model looks at image on level of an object and also globally.

There is an another type of learning, known as reinforcement learning and it can be implemented for encoder-decoder architectures models. Usage of this kind of architecture will reduce the loss value and the bias.

Similarly, in this research work the model that is developed is also constructed by using encoders and decoders. Short description of the model is as following:

- Encoder-Decoder architecture. It encodes given RGB colored image into smaller images with "learned" channels. The resulted image contains the main summary information from the original one. There is a fine-tuning implemented to improve its performance, as the encoder is already CNNs trained. The used model is 101 layered Residual Network trained on the ImageNet classification task, already available in PyTorch.

- Attention. One of the most widespread techniques in deep learning is the Attention Mechanism. It allows a model to focus on and choose only those parts of the encoding that it thinks are relevant to the task at hand. In image captioning some pixels should be considered more important than others. It is composed of only linear layers and a couple of activations.

- Transfer Learning. This is when a new model is constructed based on the old one. It is better than training a new model from scratch, which can be efficient by means of time and memory.

- Beam Search. This lets the Decoder find the most efficient sequence of words, not only based on the words' score at each decode-step.

- Telegram Bot: Bots are run inside Telegram and can be defined as third-party applications, with the help of which the users are able to use and work with bots by sending some kinds of messages, inline requests and commands. It is controlled using HTTPS requests to bot API.

2.2 Significant Prior Research

Indeed there are a lot of works done on this topic as intersection of vision and language are now one of the significant questions of modern research. As computers do not have intelligent reasoning, detecting objects on the image and determining their relationship by constructing the context of the image is challenging task for them. There are two most common ways of doing image captioning: generative and retrieval based methods.

As an example of retrieval based method we can take Im2Txt model [16]. This model was constructed by Vicente Ordonez, Tamara L Berg and Girish Kulkarni. General system is contained of two parts: matching the image and generation of the caption. When an image is given to the system it will be matched with the image and its caption which is in the database. After the images are matched, received objects of high level will be compared with original input data. One of the disadvantages of this model is being overfitted, it can construct captions only from already prepared ones in the dataset. However, this problem is solved in generative models. Because, generative models let to construct new sentences (descriptions) for the model. Model proposed in the work of Vinyals et al. called ""Show and tell: Lessons learned from the 2015 mscoco image captioning challenge" is one of the examples of generative type of models. This model implements neural machine translation and newest image recognition techniques, by using LSTM and model of inception-v3 [16].

If to look at researches on the task of visual question answering (where the main task is image captioning) they range from constrained settings to free form natural language questions and answers. For example, the authors of research work "A visual Turing test for computer vision systems" propose a system to generate binary questions from templates using a fixed vocabulary of objects, attributes, and relationships between objects [8]. They constructed a "visual Turing test": an operator-assisted device that produces a stochastic sequence of binary

questions from a given test image. Engine will construct some question, afterwards question will be checked by the operator and either the answer will be given or will state the question was ambiguous. Another research work called "Video And Text Parsing for Understanding Events and Answering Queries" propose a multimedia analysis framework to process video and text jointly for understanding events and answering user queries [15].

However, for my idea I have taken as an example the work cited before, about "Binary Visual Questioning" [22] system and or attention mechanism implementation the work of research group lead by Peter Andreson called "Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering" [10] will be used. The first work has the gap that it uses pre-defined hard coded features, however he model and the dataset are very good. so I will use this work for my structure of the model and their data for training my constructed model. In order to realize the first part, pre-trained VGG16 model with under 3 000 000 trainable parameters will be used for image captioning process. Manually constructed model will have the following structure and summary (Fig. 2.1 and Fig. 2.2):

```
!pip install torchsummary
from torchsummary import summary
model = customCNN()
summary(model,(3,224,224))
```

```
Requirement already satisfied: torchsummary in /root/anaconda3/lib/
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1         [-1, 10, 220, 220]             760
            Conv2d-2         [-1, 20, 108, 108]           1,820
            Conv2d-3           [-1, 30, 52, 52]           5,430
            Conv2d-4           [-1, 40, 24, 24]          10,840
            Linear-5                 [-1, 500]       2,880,500
            Linear-6                 [-1, 100]          50,100
            Linear-7                  [-1, 20]           2,020
            Linear-8                   [-1, 7]             147
================================================================
Total params: 2,951,617
Trainable params: 2,951,617
Non-trainable params: 0
```

Figure 2.1: CNN model summary

The data used have over 8000 instances with their 5 different descriptions. Second part of the work is focused on implementing it in another dataset and trying to construct own CNN model which recognizes an object and captions it

questions from a given test image. Engine will construct some question, afterwards question will be checked by the operator and either the answer will be given or will state the question was ambiguous. Another research work called "Video And Text Parsing for Understanding Events and Answering Queries" propose a multimedia analysis framework to process video and text jointly for understanding events and answering user queries [15].

However, for my idea I have taken as an example the work cited before, about "Binary Visual Questioning" [22] system and or attention mechanism implementation the work of research group lead by Peter Andreson called "Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering" [10] will be used. The first work has the gap that it uses pre-defined hard coded features, however he model and the dataset are very good. so I will use this work for my structure of the model and their data for training my constructed model. In order to realize the first part, pre-trained VGG16 model with under 3 000 000 trainable parameters will be used for image captioning process. Manually constructed model will have the following structure and summary (Fig. 2.1 and Fig. 2.2):

```
!pip install torchsummary
from torchsummary import summary
model = customCNN()
summary(model,(3,224,224))
```

```
Requirement already satisfied: torchsummary in /root/anaconda3/lib/
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1         [-1, 10, 220, 220]             760
            Conv2d-2         [-1, 20, 108, 108]           1,820
            Conv2d-3          [-1, 30, 52, 52]            5,430
            Conv2d-4          [-1, 40, 24, 24]           10,840
            Linear-5                [-1, 500]        2,880,500
            Linear-6                [-1, 100]           50,100
            Linear-7                 [-1, 20]            2,020
            Linear-8                  [-1, 7]              147
================================================================
Total params: 2,951,617
Trainable params: 2,951,617
Non-trainable params: 0
```

Figure 2.1: CNN model summary

The data used have over 8000 instances with their 5 different descriptions. Second part of the work is focused on implementing it in another dataset and trying to construct own CNN model which recognizes an object and captions it

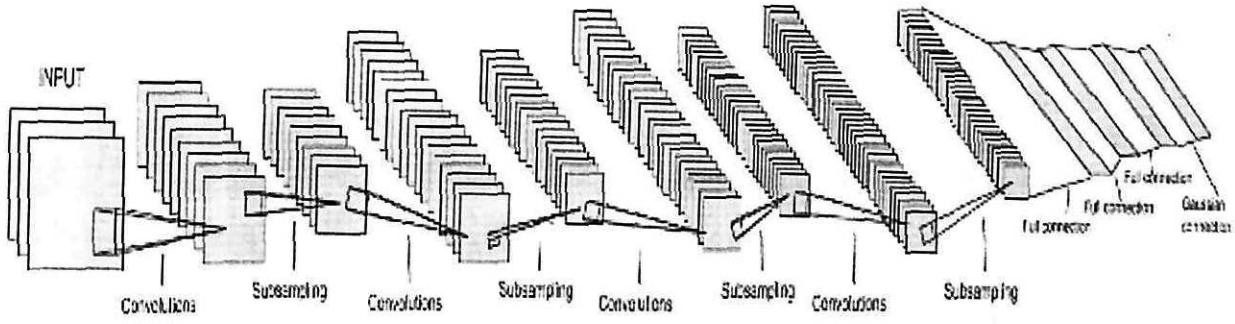
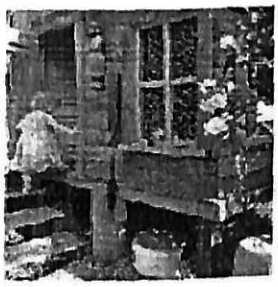


Figure 2.2: CNN model structure

(Fig. 2.3).



a little girl in a pink dress going into a wooden cabin .
a little girl climbing the stairs to her playhouse .
a little girl climbing into a wooden playhouse .
a girl going into a wooden building .
a child in a pink dress is climbing up a set of stairs in an entry way .

Figure 2.3: Sample from the dataset

Neural network architectural image captioning model is implemented in keras framework. Keras is one of the simplest, fastest and mostly used human oriented APIs. The VGG16 model imported from keras has 16 layers and over 138 000 000 parameters. First of all distribution of the words in the text was visualized in order to prepare more accurate and informative text for captioning. Afterwards, all the images were linked with their descriptions data were divided into training and testing sets.

Pre-trained with weights model was trained with images and their descriptions. The next step was about testing the constructed ready model. Randomly chosen images are given as an input one by one and the model returns one description for each. Some captions were not right and some were right. Eventually the model was evaluated by calculating the loss value on each epoch, from 5.4 till 3.3. There is a metrics called Bilingual Evaluation Understudy (BLEU) that is used for measuring the likeliness of one sentence with other given sentences. Its value varies between 0 and 1. BLEU is very well-known and easily calculated accuracy metric. While evaluating the constructed model BLEU values were good enough, it has shown result over 0.75, which means a good captioning (Refer to Fig. 2.4).

true: man and baby are in yellow kayak on water

pred: man in blue shirt is sitting in the water

BLEU: 0.759835685652

Figure 2.4: BLEU accuracy scores

It was not so accurate in describing the image with complex background, but identifies objects and describes separately well. Model has shown an accuracy about 0.75. However, in some cases model have shown very irrelevant descriptions also, it can be caused by the size of the data given for training the model, because 7000 is not a big amount for model training in images classification problems. This limitation was caused by the lack of RAM memory on the laptop, only about 7000 images could be processed and sent for training the model. In further works, it will be more appropriate to connect additional GPU, or to buy a server in some large servers like Amazon or to process the images in a PC with larger amount of memory.

There is an another work that proposes an automated fusion system that performs image scene understanding developed by researchers of Yang Chen Information and System Sciences Library. Their system works differently than previously developed ones. Their architecture combines data from 2 different and parallel processing ways, where the first one works with image itself and the second one pays attention to separate objects. It also can work in 4 different ways: classification based on image, classification based on objects, feature based fusion and class based fusion (Refer to Fig. 2.5, 2.6, 2.7).

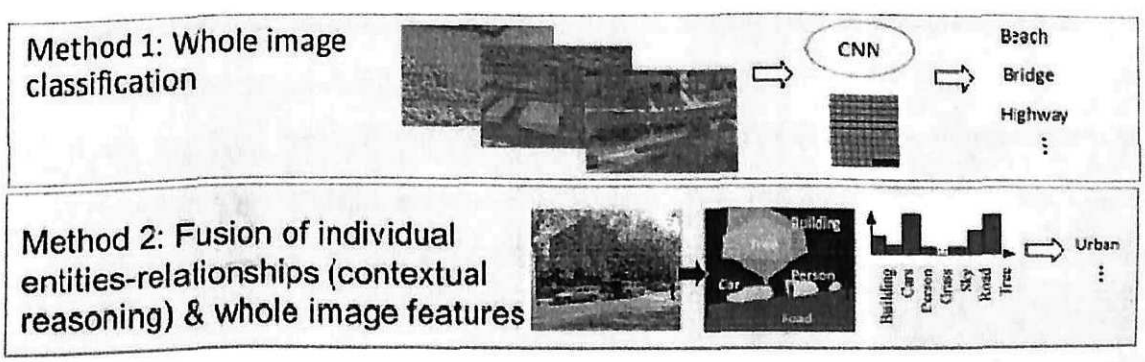


Figure 2.5: Image and object based scene captioning

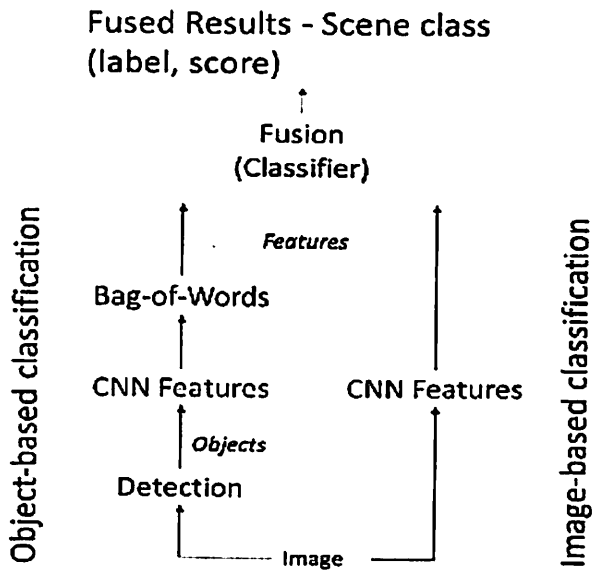


Figure 2.6: Feature based fusion

- Method 1 Image-based Classification (Bottom-up) Whole image features are extracted with a deep convolutional neural network, which consists of several layers performing filter convolution, spatial pooling, and nonlinear rectification [23]. The network has been trained by supervised learning on a large image dataset with class labels. The class labels used in training the network do not necessarily have to correspond to the classes of scenes to be classified by the whole architecture, as it is expected that most parts of the earlier layers of the network are learning generalizable, low-level visual features. The final layer of the network that assigns a class label to the final feature space can be replaced with another classifier that is later trained on the scene classes. This last layer is replaced with a linear support vector machine [24].

- Method 2 Object-based classification (Top-down) The entity processing pipeline scans the image, identifies and segments potential object locations, and assigns an entity class label to each potential object. The entity classifier can also be a convolutional neural network whose last layer has been trained on the entity types of interest. We have previously used and developed a pipeline for this in the DARPA Neovision2 project [1,2]. Thus for each image, the entity processing pipeline produces a list of all entities in the image and their types. In the current embodiment, this is encoded into a bag-of-words histogram feature. This feature has a number of dimensions

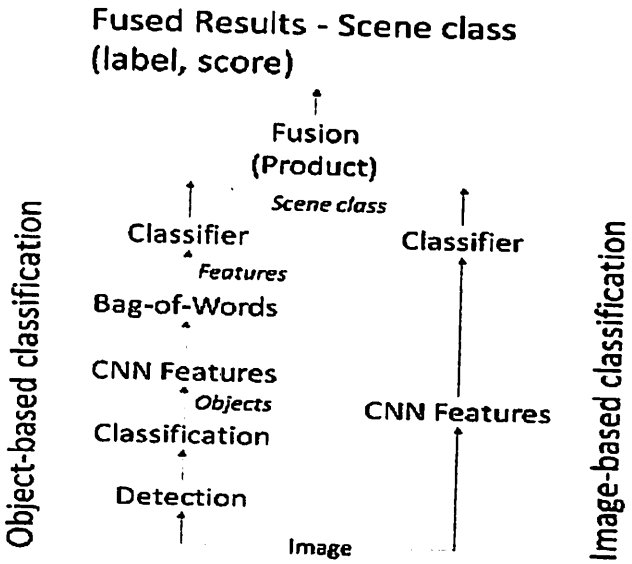


Figure 2.7: Class based fusion

equal to the number of entity classes, and the value at each dimension is the frequency (number) of such entities detected. Other implementations that encode both the spatial and frequency information are just as possible. The image can be divided into N grids and the bag-of-words histogram can be generated for each grid. The information from all grids is then combined by concatenating each grid bag-of-words into a single bag-of-word histogram. This preserves spatial information that can be helpful in scene classification. For example, with a fixed camera orientation such as in autonomous driving, roads are at the bottom of the scene, vehicles/pedestrians/street signs and trees are above that and finally the sky is at the top. This spatial structure is encoded in this implementation and can be used in scene classification [24].

- Method 3 In this approach, the two feature spaces are simply concatenated. For example, a 4096-dimensional image feature vector and a 10-dimensional entity bag-of-words feature are combined into one 4106-dimensional feature. Then a classifier is trained on the combined feature space. In other embodiments, weighted or biased fusion of these features can be done [24].

- Method 4 In this approach, information is combined at the class probability level. Two classifiers are trained separately for the entities and whole image features. Each classifier produces a class probability distribution over the

scene types. These distributions are combined, e.g., multiplied and renormalized to produce the final classifier result [24].

Most of the works that are done on the problem of image scene understanding or recognition of image scene are constructed around this problem: possibility of defining the image scene, giving the description of an image without paying attention first to objects. There are several reasons for not doing recognition of separate objects. It is known that there are very large number of sensible object recognition algorithms and models. Using whole image scene recognition with object recognition can be complex and problematic as object recognition algorithms error can influence image scene recognition. Also in object recognition there can be a lightening problem. In order to solve such kinds of problems the model must be architectured in a such way that there are less number of feature levels. The main problem is to define that features which are not necessary and only can harm the overall performance of the algorithm.

Indeed there are number of different kinds of image features that can be extracted from them, however some of them are classified as most meaningful and useful ones according to the experiments and researches of many scientists. For instance features like orientation, color or texture are being used widely in the problems of image scene understanding.

The model proposed by Malik and Reninger implements a method and an algorithms that mainly focuses on texture of the input image [10]. It is very simple but also had shown a good performance. The idea of deriving textns from the textures are taken from another related work, where textns show and determine to what extent the texture is strong. Textns are such particles of an image which controls texture perception. Their values are computed by implementing some filters to a given image - by doing some image preprocessing. Models that are based on textns focuse on local ones, to textures that show and describe different kinds of categories of an image scene.

For storing the feature frequency simple histogram is used for each and every image separately. In order to identify the most relevant and high level features a machine learning algorithm called K-means clusering. In process of identification of a new feature all the sorted ones are compared with its histogram. As a result of this research it can be stated that earl scene recognition can be done by using texture features. This kind of algorithms and models work as human.

Another paper proposes orientation of features which are multi-scale and they were compared to human studies [11]. The researchers say that one of the most informative and important features of image is orientation. The model is trained in a such way that its algorithm can define orientation locally and find weights of each and every pixel of an image. results of this work show that performance of orientation focused algorithm is very high, it has shown over 90%.

The work done by Oliva et al.proposes similar model but their model implement extended verion of orientation-focus: selecting most efficient scale for classification task [25]. Their feature is called as local dominant orientation (LDO) which are the features for clustering photos from the real-world into 4 different classes (indoor, outdoor, closed and open places). They propose an optimization of categorization by using Fourier series and thus they do not directly use local dominant orientation features. Strength of an orientation is computed for each and every scale and location. Among all scales and ratios the most efficient ones were with the value of spatial median scale, which also means the combination of several scales.

Research paper written by Csurka et al., propose number of different key approaches for categorization visually [25]. However it requires description and recognition of image patches. After creating the patches, image captions vocabulary is constructed by implementing an algorithm of vector quantifiying. SWIFT decriptions were implemented for this model that is followed by generating other keypoints which is responsible for the counting the patches for each and every cluster. Eventually, Support Vector Machines algorithm is implemented as an algorithms that is able to classify into multiple classes. Group of points were used as an input for the SVM algorithm as a vector.

The use of computer vision is very wide and includes such areas as agriculture, augmented reality, autonomous vehicles, biometrics, character recognition, forensic medical examination, industrial quality inspection, face recognition, gesture analysis, image recovery, medical image analysis, robotics and much more. Given this, it is not surprising that, being a relatively young science – the roots go back to the end of the 50s of the last century – great attention is paid to it in the academic environment. At the beginning of its development, computer vision tried to solve simple problems in controlled conditions, like understanding the shape of the figure, finding angles and edges objects, recognition of simple

printed characters, etc.

At the end of the 90s of the last century, the theoretical base already made it possible to solve the problems of finding and recognizing more complex objects, such as a person's face, a human model, a car, a car number, etc. However, it should be noted that the methods and algorithms for recognizing each object could radically differ, in the best case, when the general approach coincided, it was necessary to carefully work out the details and adjust the parameters of the algorithms for each recognizable object. The tasks of understanding the image at a high level are somehow related to classification and by categorizing the entire image, its parts, or each pixel. Therefore, in the further development of computer vision, machine learning and its universal classification, clustering, and artificial neural network algorithms played a huge role.To get good results on the accuracy of object recognition, machine learning algorithms require a large amount of labeled data for training.

With the advent of a huge amount of data on the Internet, this has become possible. The first initiative, the purpose of which was to collect and mark up a huge number of images for the further development of computer vision algorithms to human capabilities, i.e. capable of recognizing many thousands of objects, ImageNet was. Every year, competitions of methods and algorithms are held for the best accuracy indicators on this set of images. ImageNet contains over fourteen million images, divided into over twenty thousand objects and visual concepts. The presence of such a quantity of annotated data played a crucial role in the development of not only computer vision, but also the entire field of artificial intelligence.

The first algorithm, which showed a fairly good result by image classification in ImageNet, was the Bagof Visual Words algorithm. This algorithm is based on the "bag of words" method, which is widely used in textual semantic analysis and textual search. Its essence is to build a dictionary of visual words, and describe any image with a histogram of occurrences of words from a dictionary in a given image. Due to the tolerance of this algorithm to changes in angle and illumination, it was widely used in various tasks of computer vision. A breakthrough in the accuracy of ImageNet was in 2012, when Alex Krizhevsky was able to build a deep convolutional neural network, which was more than doubled results of previous years. Since then, classification algorithms based on deep neural networks have

been steadily improving performance on the ImageNet suite every year.

In general, neural networks with many hidden layers improved the results in many related areas of artificial intelligence, such as speech recognition, speech synthesis, natural language processing, game theory, etc., debunking the belief prevalent in the early 2000s that that neural networks are not able to solve the complex tasks of the real world. This belief was justified, and consisted in the fact that, so far, the 4 following aspects of training a neural network were performed incorrectly: the network was trained on the amount of data per thousand fewer times than required; insufficient computing resources were used to complete the training at any acceptable time; the strategy for choosing the initial value of the weights was chosen incorrectly; the activation function of neurons was not chosen correctly. To date, all methods and algorithms that allow you to train neural networks with many hidden layers are combined under the same term "deep learning". The significant progress we are gaining today through deep learning is that it allows computational models consisting of several processing layers to study data representations with several levels of abstraction. As we noted earlier, conventional machine learning methods were limited in their ability to process raw data in their original form.

For decades, building an image recognition system or machine learning has required careful development and considerable experience in the field of knowledge for which a feature extractor was developed that converted raw data (image pixel values) into a suitable internal representation or feature vector from which the training subsystem is often a classifier , could detect or classify patterns at the input. Learning a view consists of a set of met rows which allow the machine to load the raw data and automatically find representations necessary for the detection or classification. Deep learning methods are methods of teaching presentation with several presentation levels, obtained by compiling simple but non-linear modules, each of which transforms the presentation at one level (starting from the initial input) into a presentation at a higher, slightly more abstract level. By combining enough of these transformations, very complex functions can be trained. For classification tasks, higher levels of presentation reinforce input characteristics that are important for discrimination and suppress irrelevant change and noise. The image, for example, is represented as an array of pixel values, and the images obtained with the first layer of the presentation usually describe the presence or

absence of edges in certain orientations and locations in the image. The second layer, as a rule, detects groups and the structure of interconnected edges, regardless of small changes in their position in the image. The third layer can collect groups on the previous layers larger combinations that correspond to parts of familiar objects, and subsequent layers will detect objects as combinations of these parts. A key aspect of deep learning is that the features in these layers have not been developed by engineering scientists: they learn from the provided data using a common well-known learning algorithm. Deep learning, to date, has achieved great success in solving problems that for many years could not find their optimal solution with the best minds of the artificial community intelligence.

It has proven to be a very convenient tool for detecting complex structures in multidimensional data and therefore is applicable to many fields of science, business and economics. In addition to achieving better results in pattern recognition and speech recognition, it also outperformed other machine learning methods in predicting the activity of potential molecule interactions, analyzing data in a particle accelerator, reconstructing brain circuits, and predicting the effects of mutations in non-encoded DNA during gene and disease expression. Deep learning has yielded extremely promising results for various tasks in understanding natural language, in particular for classifying topics, analyzing feelings, answering questions, and translating into a language. In addition, in the interdisciplinary areas where computer vision is widely used, significant advances are being made in solving , problems that were previously considered difficult to solve.

Autonomous systems operating in complex, unstructured and dynamic environments have high requirements for localization capabilities, to be able to show resistance to accumulated odometer errors. A general way to increase such reliability is the terrain recognition mechanism, that is, a system that uses the detection mechanism of previously viewed scenes to correct the current alleged odometry or restore the position of the robot in localization failure scenarios. The terrain recognition mechanism is based on visual perception algorithms, usually classified as visual terrain recognition (VRM). In order to provide increased reliability, BPM methods should demonstrate immutability with respect to lighting, viewing angle and environmental differences. In particular, the difficulties arising in scenarios with long trajectories, which are subject to extreme changes in the appearance of the scene due to a change in the time of day (day and night

conditions) or a change in seasons (summer and winter), have made BPM one of the most difficult tasks in computer vision. In view of the foregoing, research in this dissertation related to the development of new methods and algorithms for the classification and categorization of visual data obtained in uncontrolled, naturally s environmental conditions, it is relevant.

3. Materials and Methods

3.1 Detailed model description

Once the Encoder generated the encoded image, it is transformed to construct first hidden stat for the Long Short Term Memory Decoder. At every decoding stage antecedent hidden layer and image that is encoded are used to construct wights for each pixel in network mechanism of attention. The previously outcomed word and the weightd average of the encoding are fed to the Long Short Term Memory Decoder to give the next word. Afterward a linear layer to transform the decoder's output into a score for each word is used. The greedy option would be to choose the word with the highest score and use it to predict the next word. However it can be not optimal, as the rest of the sequence hinges on that first word chosen. Each word in the sequence has consequences for the ones that succeed.A while later a straight layer to change the decoder's yield into a word score. The ravenous choice is pick the word with the most elevated score and use it to anticipate the following word. Anyway it very well may be not ideal, as the remainder of the succession relies on that first word picked. It would be ideal if the last form would not be chosen until translating is done and pick the succession that has the most noteworthy generally speaking score from a container of competitor groupings.

In this section, 2 different versions of our model is described with attention mechanism. Difference between these two versions is their function f. In the figure below (Refer to Fig. 3.1), long Short Term Memory cell is depicted. Generated weights, that are vectors are depicted in lines with squares inside. Each cell is responsible for computing the weight for its input and modulation of the memory for it. In addition to this these cells are responsible for deleting from the memory and output gate, where memory emition is controlled. The main components of the model are encoder-decoder, attention mechanism,

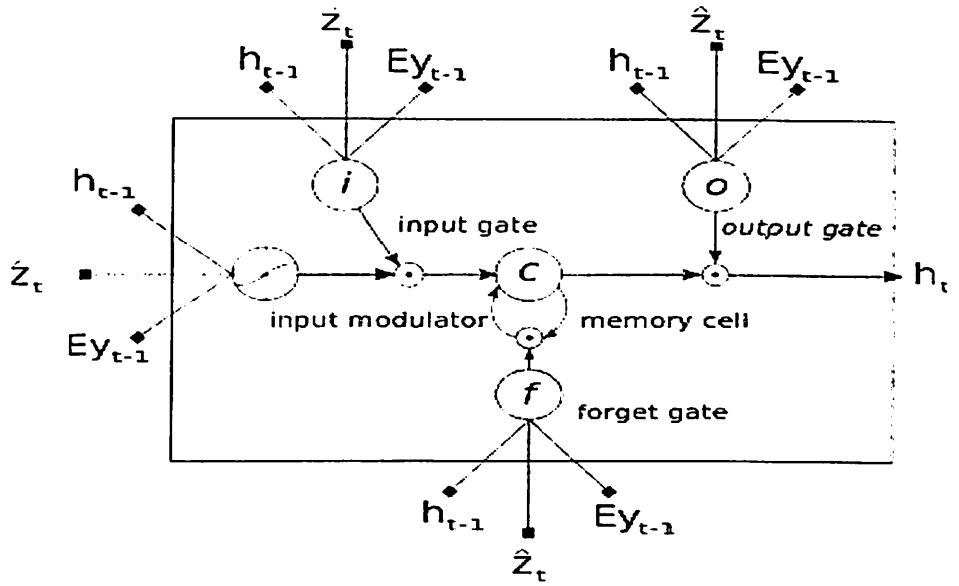


Figure 3.1: LSTM cell

- Encoder: It encodes given RGB colored image into smaller images with "learned" channels. The resulted image contains the main summary information from the original one. There is a fine-tuning implemented to improve its performance, as the encoder is already CNNs trained. The used model is 101 layered Residual Network trained on the ImageNet classification task, already available in PyTorch.

  Model is given one image as an input and the description of that image is constructed as a set of K words that are encoded.

$$y = y^1, ..., y^C, y^i \epsilon R^K$$

  Therefore, the size for generated vocabulary - K and the longitude of produced caption - C. In order to reveal a subsequence of feature vectors CNN is used. Revealer generates L number of vectors and each and every of them is dimensionality of D, which refers to some specific place on a given input image.

$$a = a^1, ..., a^L, a^i \epsilon R^D$$

.

Eventually, features in the form of vectors and parts of the image are combined by revealing some other features from small convolutional layers.

- Decoder: Long short-term memory constructing a caption by producing words one by one on vector is used in decoder. It learns from the previous constructed context and states. Similar kind of implementation is proposed in the research work of Zaremba, however this model is fine tuned [21]. Here,i^t, h^t, o^t, f^t stand for each states of the LSTM layer:

$$\begin{pmatrix} \mathbf{i}_t \\ \mathbf{f}_t \\ \mathbf{o}_t \\ \mathbf{g}_t \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} T_{D+m+n,n} \begin{pmatrix} \mathbf{E}\mathbf{y}_{t-1} \\ \mathbf{h}_{t-1} \\ \hat{\mathbf{z}}_t \end{pmatrix}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t).$$

  - i - input
  - h - hidden
  - f - forget
  - o - output

There is a vector z which is responsible for catching data visually and $z \epsilon R^D$. There is also a matrix E - embedding which belongs to $R^m * K$. If to say it in more simple way, all the three equation given above are active looking of image partitions that are most relevant. There is also a mechanism that calculates a vector z by the help of other vectors listed above (a). Some positively valued weight it computed for each and every location in an image. This value can also be taken as a probability of being an important place to be focused on or not. It also can be used as simple importance status of the location. These weights are computed by the help of attention mechanism. The model has multilayer. Also the focus of the model for each location and the object inside an image is computed based on set of previously generated words. After computing of the news are done, main vector for the whole

$$e_{ti} = f_{\text{att}}(\mathbf{a}_i, \mathbf{h}_{t-1})$$

$$\alpha_{ti} = \frac{\exp(e_{ti})}{\sum_{k=1}^{L} \exp(e_{tk})}.$$

description is created in a following way:

$$z_i = f(a^i, \alpha^i)$$

. Where f is a vector constructed from annotation vectors. Memory for hidden state can be predicted with the help of small vectors.They are generated in 2 separate perceptrons:

$$\mathbf{c}_0 = f_{\text{init.c}}(\frac{1}{L}\sum_i^L \mathbf{a}_i)$$

$$\mathbf{h}_0 = f_{\text{init.h}}(\frac{1}{L}\sum_i^L \mathbf{a}_i)$$

- Attention. One of the most widespread techniques in deep learning is the Attention Mechanism. It allows a model to focus on and choose only those parts of the encoding that it thinks are relevant to the task at hand. In image captioning some pixels should be considered more important than others. It is composed of only linear layers and a couple of activations.
- Transfer Learning. This is when a new model is constructed based on the old one. It is better than training a new model from scratch, which can be efficient by means of time and memory.
- Beam Search. This lets the Decoder find the most efficient sequence of words, not only based on the words' score at each decode-step.
- Telegram Bot: Bots are run inside Telegram and can be defined as third-party applications, with the help of which the users are able to use and work with bots by sending some kinds of messages, inline requests and commands. It is controlled using HTTPS requests to bot API.

Structure of the model is depicted in Figure 3.2 below:

3.2 Attention mechanisms

One of the most widespread techniques in deep learning is the Attention Mechanism. It allows a model to focus on and choose only those parts of the encoding

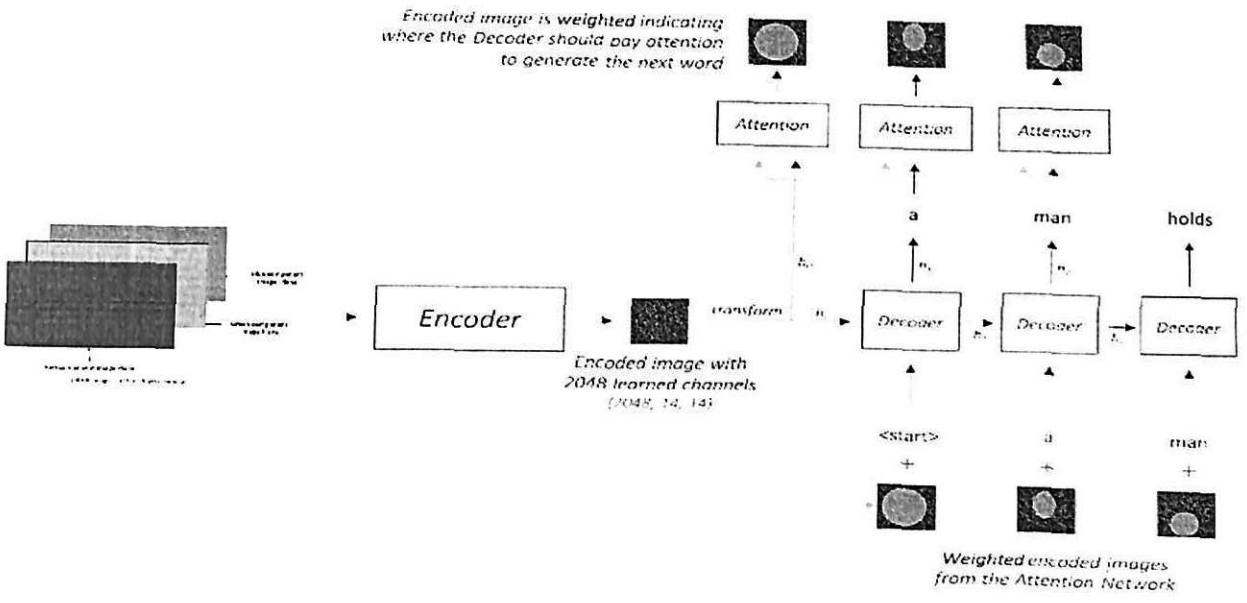


Figure 3.2: LSTM cell

that it thinks are relevant to the task at hand. In image captioning some pixels should be considered more important than others. It is composed of only linear layers and a couple of activations. There can be a mistake when simple model without attention mechanism is used for image scene understanding. For example, here on the Figure 3.3 it can be seen that it need the attention in it.

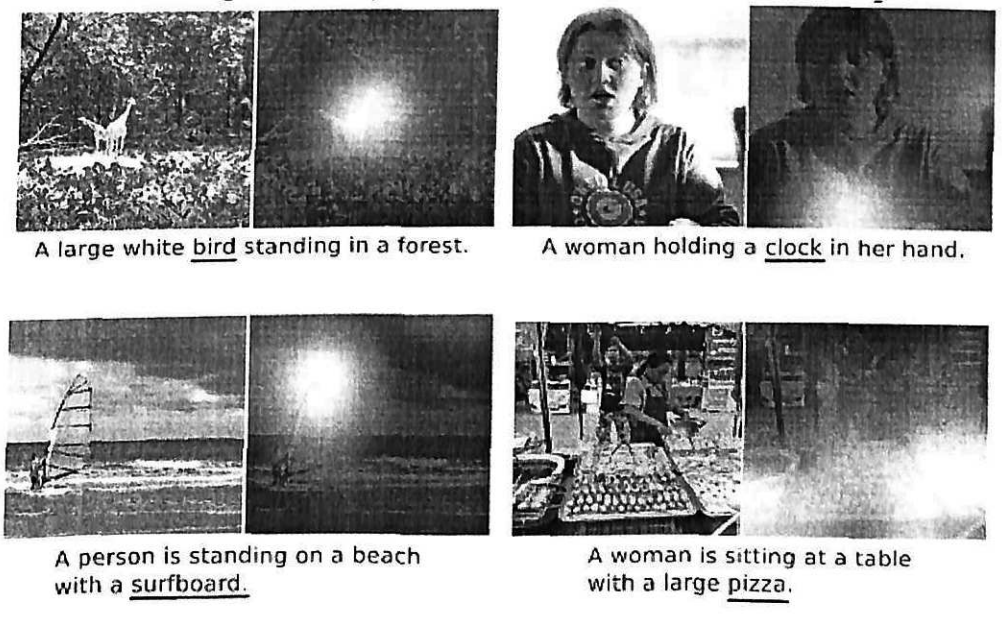


Figure 3.3: Example of the model prediction without attention mechanism

This subsection is about different kinds of attention mechanisms, namely they are: hard stochastic and soft determenistic types of this mechanism.

Hard stochastic

We represent the location Variable that is location s_t shows the point where the model should pay more attention and focus while constructing the image description. s_t stands for the location and it is an indicator of generated features in the future. By regarding the consideration areas as middle of the road dormant factors, multi conveyance can be allocated as α_i, and z_t can be viewed like an irregular variable:

$$p(s_{t,i} = 1|s_{j<t}, a) = \alpha_{t,i}$$

$$z_t = \sum s_{t,i} a_i$$

Another target work L_s is characterized that is a lower part on the minor log-probability $\log p(y|a)$ of watching the words succession y of input picture a . The calculation of an algorithm for W parameters of constructed architecture is inferred by legitimately enhancing L_s:

$$L_s = \sum p(s|a) \log p(y|s, a)$$

$$\leq \log \sum p(s|a) p(y|s, a)$$

$$= \log p(y|a)$$

$$\frac{\partial L_s}{\partial W} = \sum p(s|a)[\frac{\partial \log(p(y|s, a))}{\partial W} + \log p(y|s, a) \frac{\partial \log(p(s|a))}{\partial W}]$$

This equation is an approximation by sampling based on Monte Carlo: gradients are approximated based on parameters of the model. It is realized by location sampling from the value computed by equation

$$p(s_{t,i} = 1|s_{j<t}, a) = \alpha_{t,i}$$

A dynamic normal benchmark is utilized to decrease the fluctuation in an estimator of Monte Carlo of the slope. Comparable, however progressively confused change decrease procedures have recently been utilized. After observing the batch (eg. k-th) than normal cluster, the moving normal benchmark is assessed as an

aggregated entirety of the past log probabilities with exponential rot:

$$b_k = 0.9 * b_{k-1} + 0.1 * \log p(y|s_k, a)$$

To additionally diminish the estimator fluctuation, multi conveyance $H[s]$ is included with term entropy. Likewise, with likelihood 0.5 for an input picture, the inspected consideration area s to its normal value α is set. The two methods improve the vigor of the consideration learning calculation. The last learning requirement for the model is then the accompanying:

$$\frac{\partial L_s}{\partial W} = \frac{1}{N}\sum[\frac{\partial \log p(y|s,a)}{\partial W} + (\lambda_r(\log p(y|s,a) - b))\frac{\partial \log(p(s|a))}{\partial W} + \lambda_e\frac{\partial H[s]}{\partial W}]$$

where λs are hyper parameters of the model which are generated while doing crosvaldation.

Soft determenistic

Stochastic consideration learning needs inspecting the consideration areas s_t, rather it can be taken the desire for the setting vector z_t legitimately,

$$\mathbb{E}_{p(s_t|a)}[z_t] = \sum \alpha_{t,i}a_i$$

and construct attention mechanism architecture by calculating a vector special for attention weight $\phi(a_i, \alpha_i) = \sum \alpha_{t,i}a_i$.

Learning by using usual back propagation is obvious for this kind of smooth models with deterministic mechanism of attention. It can also be explained like optimization of marginal probability under the place of deterministic attention. It can be also implemented doubly, as using it like this will promote the model to focus at each and every location of an image given. It obviously will increase BLEU score and thus the overall performance.

3.3 Dataset

The dataset that has been used for training of CNN model is MSCOCO (Microsoft Common Object in Context) '14 dataset: photographs of complex everyday scenes containing objects in their natural surroundings. 328,000 images (over 2.5 million

occurrences of objects), 91 classes [5]. It is a large-scale dataset for detecting and segmenting objects. The COCO Dataset consists of two parts: Images and Annotations

- Images: "Task + version" with the folder name (for example: train2014), which is the xxx.jpg image files;
- Annotations: folder containing the text format xxxx.json file (for example: intrain2014.json);

COCO has five types of annotations for different tasks:

- The task of finding objects in the image
- Key point detection. Detection of objects and localization of their key points.
- Segmentation of an environment (English Stuff Segmentation). Unlike the task of detecting objects (man, cat, car), here the focus is on what surrounds it (grass, wall, sky). Class labels are organized in a hierarchical order (e.g. stuff outdoor-stuff -> sky -> clouds). To achieve compatibility with the object discovery task, the following category identifiers are used:
  - 1-91 categories of objects (not used in environment segmentation)
  - 92-182 environment categories
  - 183 category "other" (selected for "objects")
- Panoptic Segmentation - the unification of the tasks of semantic segmentation (Image Segmentation) and object detection. The task is to classify all the pixels of the image as belonging to a certain class, and also to determine which of the instances of this class they belong to.
- Annotation of the image (English Caption Evaluation). Generation of an accompanying caption to the image.

There are five types of annotations COCO 01:

- object detection
- physical separation

- key detection
- panoramic image segmentation
- image description

Object detection: The mission to detect, for example, for each image, including at least one object, COCO data sets for each object are described, not the picture. Each object contains a set of fields, including object classes and mask identification code, code masks of the format with code division of channels depends on the number of objects in the image when the image of the object lane is moving (iscrowd = 0), the mask code using RLE format, when more than a single object (iscrowd = 1) using the polyhon format.

Key detection: As in the discovery task, a one-time image of a number of objects, an object corresponding to Keypoint annotation, all data containing annotations of the characteristic point of annotation of objects (including identifier, BBOX, etc.) and two additional fields. First, as a "key" of key points, the value is an array of length 3k, where k is the total number of critical points of certain categories (for example, the position of the body is a critical point of 17). Each key has a location index and $0, x, y$ and the flag v is visible ($v = 0$ to represent unlabeled, then $x = y = 0; v = 1$ is a label, but not visible, the reason is that the invisible is obscured, and the sign indicating when $v = 2$ is visible), if the key moment is within the target segment, it is considered visible.

Physical separation: To pan the segmentation task, each annotation structure is a comment for each image, and not the annotations of each object, the above three are different. Please note that each image has two parts: 1 PNG), stored regardless of the type of image segmentation; 2) saves semantic information about each JSON structure of the image segment.

1. To match the annotation of the image, using the imageid field (i.e. annotation.imageid == image.id);
2. For each annotation, the identifier for each pixel segment is stored as a separate PNG, PNG file located in the same name as the JSON folder. Each unit has a unique identifier; unlabeled pixels are 0;
3. For each annotation, each semantic information is stored in annotation.segmentsi Segmentinfo.id, storing a section stores a unique identifier, and is used to get

the corresponding mask (identifiers == segmentinfo.id) from PNG. iscrowd represents a segment containing a collection of objects. the field indicates the BBox area and additional information.

Image description: Note job for storing subtitle images of the video title, each title is described in the specified images, each of which is at least five titles.

All instances of MSCOCO dataset are resized to 256x256 for uniformity. Each instance of the dataset have 5 or more descriptions.In all testing cases vocabulary of length 10000 for words was used. There are some samples from the dataset with implemented operations listed above:

```
In [6]: # load and display image
        # I = io.imread('%s/images/%s/%s'%(dataDir,dataType,img['file_name']))
        # use url to load image
        I = io.imread(img['coco_url'])
        plt.axis('off')
        plt.imshow(I)
        plt.show()
```

Figure 3.4: Sample image from dataset

```
In [7]: # load and display instance annotations
        plt.imshow(I); plt.axis('off')
        annIds = coco.getAnnIds(imgIds=img['id'], catIds=catIds, iscrowd=None)
        anns = coco.loadAnns(annIds)
        coco.showAnns(anns)
```

Figure 3.5: Instance annotations.

```
In [8]: # initialize COCO api for person keypoints annotations
        annFile = '{}/annotations/person_keypoints_{}.json'.format(dataDir,dataType)
        coco_kps=COCO(annFile)

        loading annotations into memory...
        Done (t=0.58s)
        creating index...
        index created!

In [9]: # load and display keypoints annotations
        plt.imshow(I); plt.axis('off')
        ax = plt.gca()
        annIds = coco_kps.getAnnIds(imgIds=img['id'], catIds=catIds, iscrowd=None)
        anns = coco_kps.loadAnns(annIds)
        coco_kps.showAnns(anns)
```

Figure 3.6: COCO api for person keypoints annotations

```
In [11]: # load and display caption annotations
         annIds = coco_caps.getAnnIds(imgIds=img['id']);
         anns = coco_caps.loadAnns(annIds)
         coco_caps.showAnns(anns)
         plt.imshow(I); plt.axis('off'); plt.show()

         A man is skate boarding down a path and a dog is running by his side.
         A man on a skateboard with a dog outside.
         A person riding a skate board with a dog following beside.
         This man is riding a skateboard behind a dog.
         A man walking his dog on a quiet country road.
```

Figure 3.7: Keypoints annotations.

4. Results and Discussion

In this chapter result of the work is shown and some of the faced problems, possible errors and solutions are discussed. Problem statement and the task of the system is as following: given an image and the system should provide an accurate description for the given image. The system is realized on Telegram Bot. Fine tuned model learns where to look, its focus is shifted across the image by the help of attention mechanism. Thus the model was able to find the most relevant parts of the image and find out most relevant words that describe the scene. It has an encoder-decoder architecture and transfer learning is implemented on pre-trained encoder.

4.1 Encoder

It takes one RGB image as an input and divides it into small images where those three channels RGB are now become learned ones. Encoded small image can be considered as a representation of an original image in a summarized way. Convolutional Neural Networ is used in order to encode the image. The encoder is not trained from the start, as for this work already trained CNN models were used as encoders. Because, the task of image classification is popular and old enough and there are a lot of models that are constructed already. There are models that have shown state of art results and that is why it is not reasonable to build new one.

As it was stated before model which is trained on ImageNet dataset - 101 layered Residual Network was used, which can be downloaded from Pytorch. It was mentioned above, there we implement fine tuning - transfer learning. Here is the encoder (Refer to Fig. 4.1)

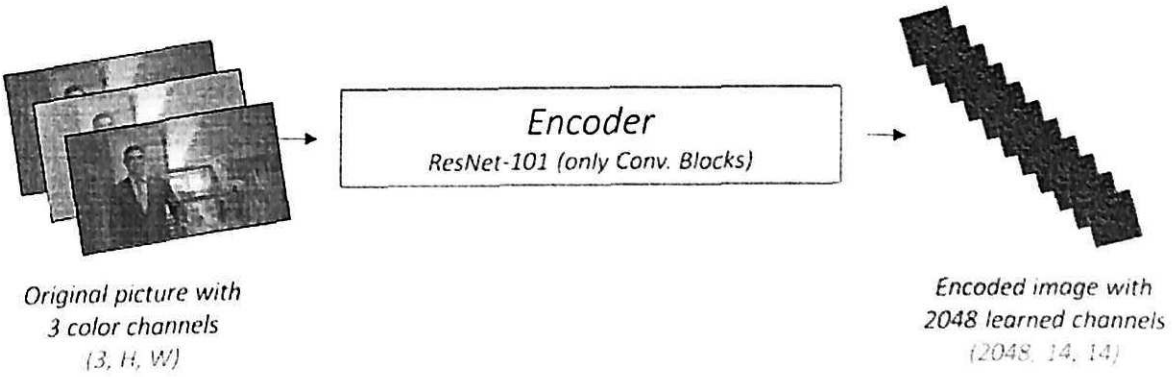


Figure 4.1: Encoder structure

As it can be seen the encoder create small types of the given image progressively. Where at each step the number of channels is getting bigger and more learned.

4.2 Decoder

Secondly the decoder takes that encoded image and constructs a caption by generating each word of it as a sequence. Therefore it has to be from the class of Recurrent Neural Networks. In this work Long Short Term memory is used.

If the attention mechanism is not used the decoder will find an average of all encoded images simply across each and every pixel of them. Afterwards, decoder is fitted with that and generate the description for an image word by word. However each word will be predicted based on the word before.

By implementing attention mechanism the decoder becomes able to focus on special parts of an image. For example, while generating the word suit in a man in a suit and tie standing in front of a store, the decoder knows that it should be focused on suit.

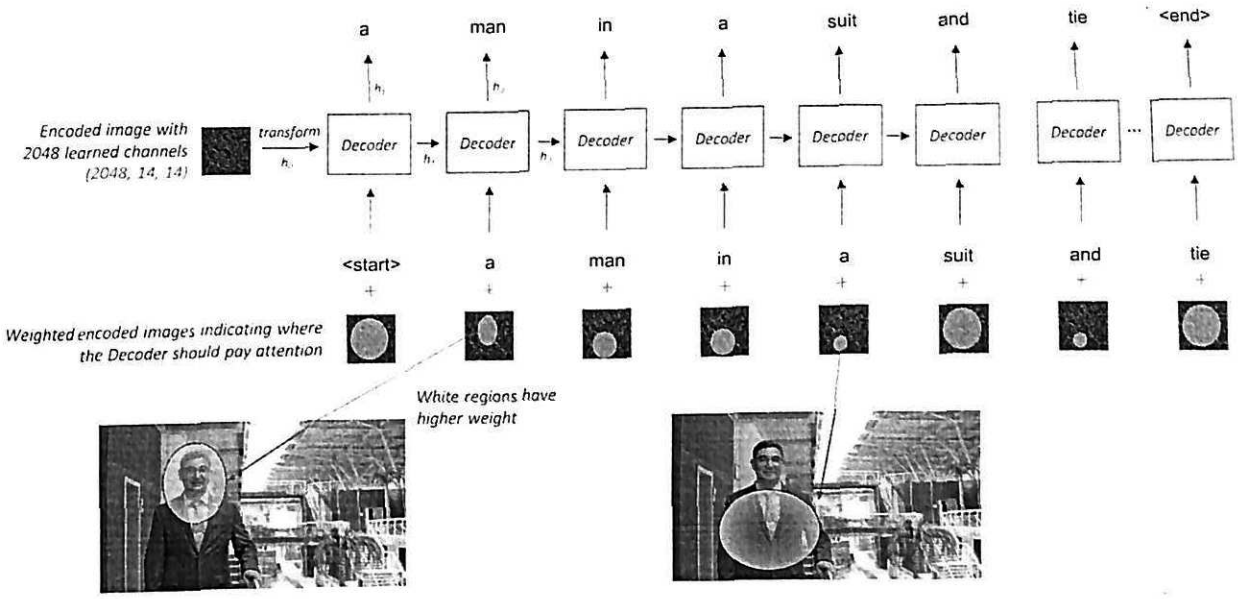


Figure 4.2: Decoder structure

4.3 Attention mechanism

Attention mechanism is needed for computing the weights of each word according to which the importance and relativity of the word is identified. Importance of some part of an image is estimated based on the generated sequence of words. It looks at an image and choose what should be described next. For instance, after it mentioned a man it will be logical to decide that he is in suit and tie. So that is what the attention mechanism is used for – mechanism considers the arrangement produced up to this point, and takes care of the piece of the picture that necessities portraying straightaway.

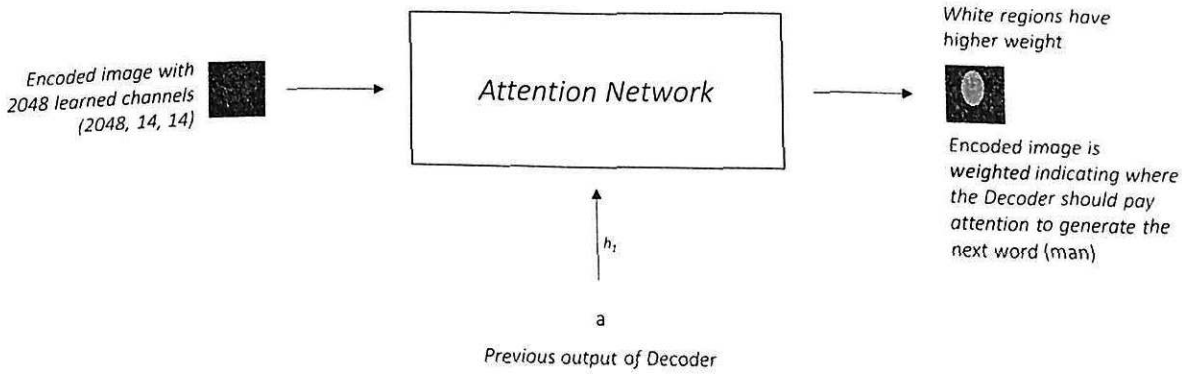


Figure 4.3: Attention mechanism structure

Here Soft attention is used: means that weights of each image point sums up to one:

$$\sum \alpha_{p,t} = 1$$

4.4 Overall model workflow

Overall model looks like as following:

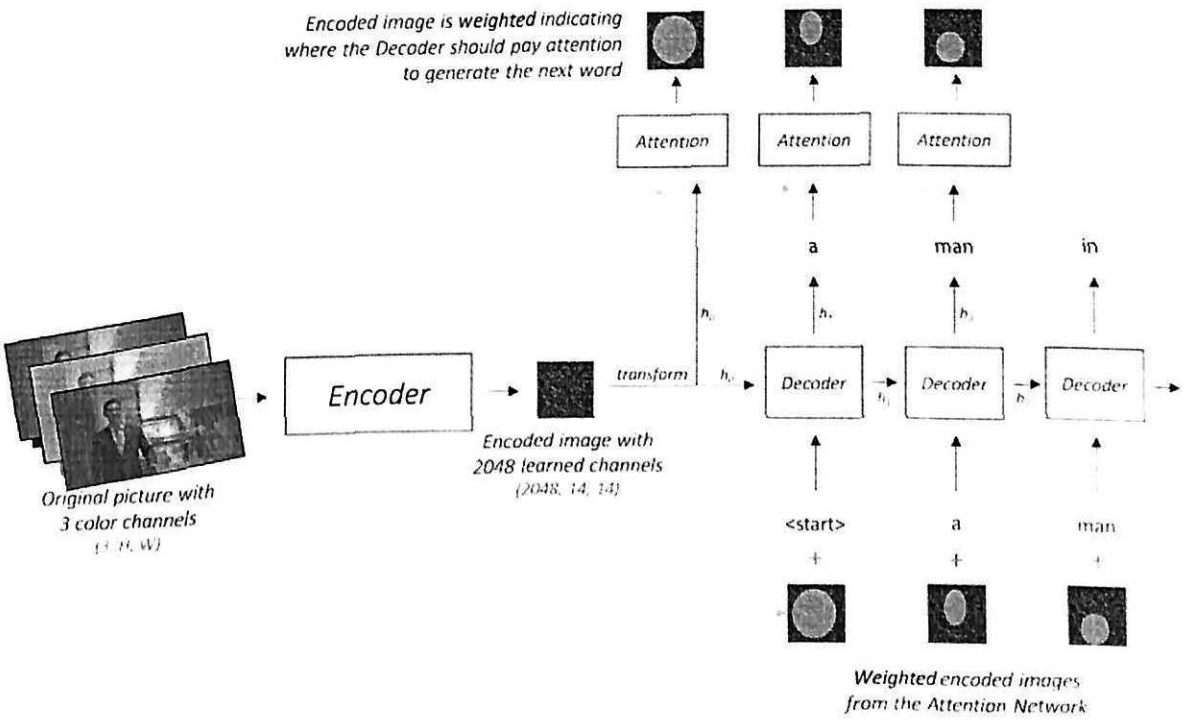


Figure 4.4: Full model structure

After encoded small images are constructed by the encoder, it is transformed for constructing the first hidden state of the decoder.

Every decoding stage includes:

- each pixel's weight is computed by using small images that are encoded and previous hidden state
- the next word is generated by sending the computed average of weight and the previous word to the Decoder to generate the next word

There are some more examples of the results of my model:

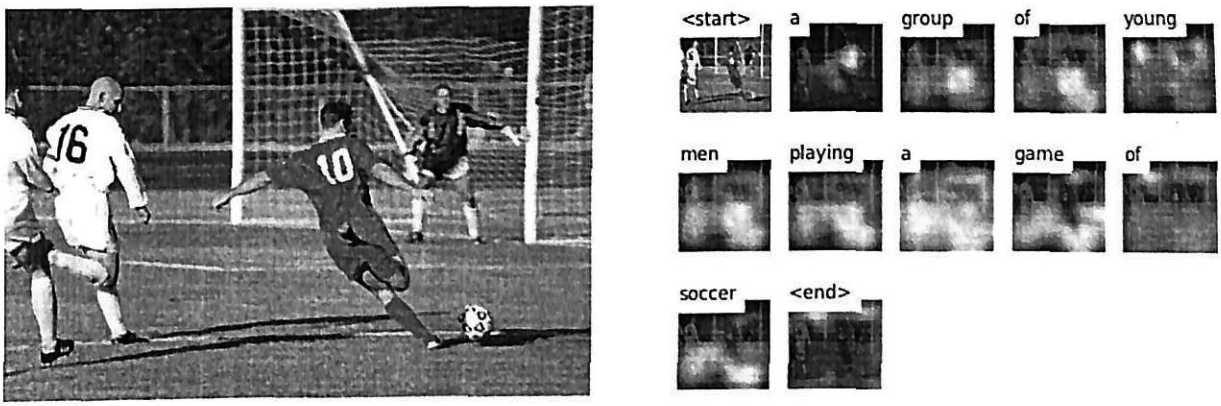


Figure 4.5: Testing results of the model

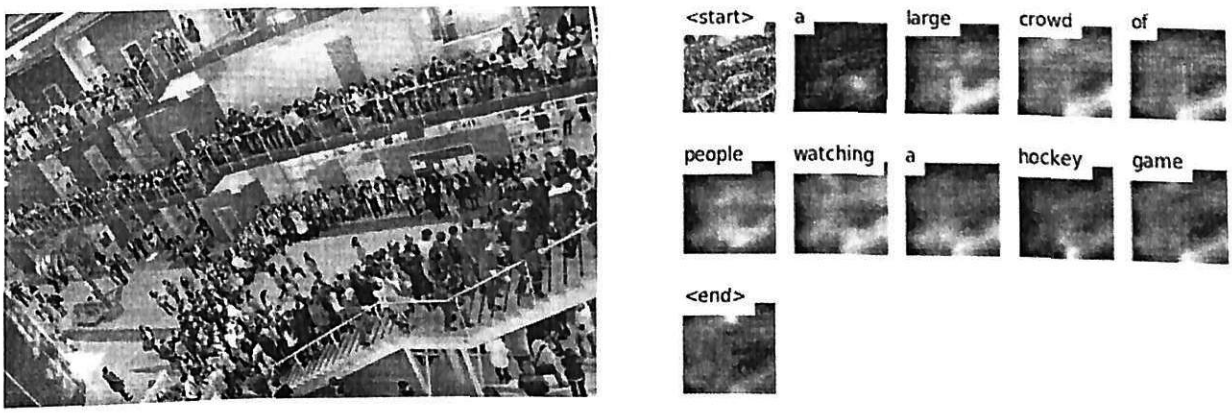


Figure 4.6: Testing results of the model

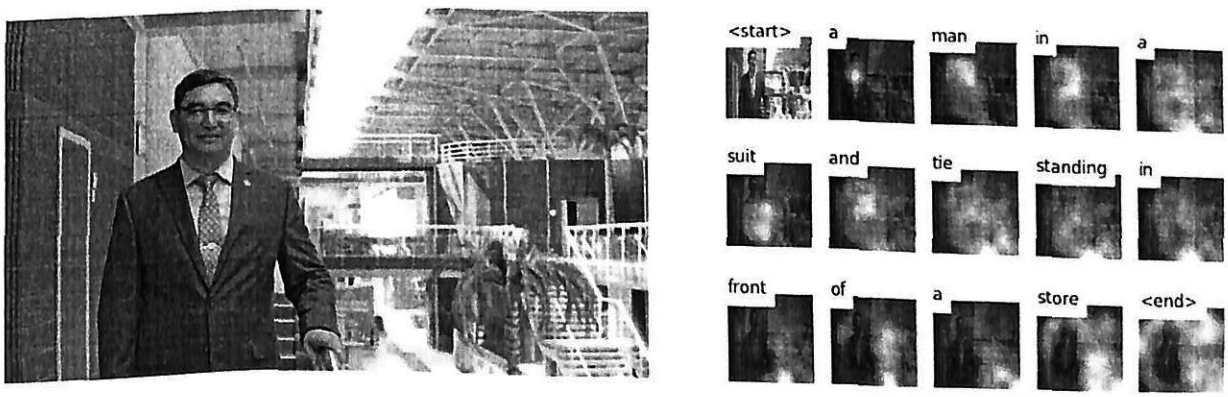


Figure 4.7: Testing results of the model

5. Further research and future work

The current research work can be expanded by using the system for Visual Question Answering tasks. It can be done by involving another big branch of Artificial Intelligence (AI) - Natural Language Processing (NLP). The description of an image can be used as a text for Question Answering system that will perform the second half of the problem: answering to questions. A fine tuned model that also contains attention mechanism is developed and explained in the current chapter. NLP problems are wider in comparison with other fields of AI, it ranges from studying meaning shades on natural communication levels, syntax, that includes morphology, semantic learning, that consists of words and phrases relation and etc. Current achievements and advancements in NLP have empowered noticeable progress in number of problems in the sphere of big data, information retrieval (IR) and AI in general [26]. In the past several decades, AI branches like NLP and Deep Learning (DL), which can be described as a class of machine learning algorithms, have showed noticeably good results across a variety of problems. In one word it is all about accuracy. Nowadays, accuracy of algorithms have reached the highest plane than ever before [26]. For instance, on a daily life it can help electronics satisfy user demands in much more better level, it is used in automation of processes in almost every field of life. In one of the recent articles about AI it is stated that latest approaches in AI has reached the point where it outperforms humans [26].

According to Alan F. Smeaton dovetailing of NLP and IR is the process that started nearly only few years ago [27]. Despite this fact, there are number of different NLP based approaches for modern IR tasks that are already implemented and have shown state-of-the-art results. For instance, Question Answering (QA)

can be stated as specific type of IR that also includes classification of texts, data extraction, summarization and semantic reasoning techniques. QAS is focused on finding the proper answer for natural language question of the user from the provided documents or source materials and this system is able to do that very well [28].

In this work pre-trained bidirectional encoder model was implemented for QA task by performing fine-tuning on it. For instance, in CV, fine-tuning or transfer learning is popular among researchers nowadays: like taking famous ImageNet model and fine-tuning it in order to create new model that is more accurate for specific task [29]. Transfer learning or fine-tuning refers to using pre-trained neural network model as a foundation for new one with concrete purpose by making some changes on its structure. During past several years, it was proven that same method can be used for number of NLP problems. The primary goal of this work is also to show that it is possible and how it is possible.

Generally QAS involves 3 main components that play significant role:

- Classification of the question
- IR
- Construction of the answer

Classification of the question is responsible for identifying the category of the question based on its entity type. While IR method is the way to obtain information in order to identify success by retrieving the appropriate response message using intelligent response systems. The last is answer construction, where ranking and affirming of an answer occurs. Literature review section has discussion of different works done on this task. Main workflow of the model for QAS is discussed in Methods section. Information about performance results of transfer learning applied model can be found in Results and Discussion section.

5.1 Methods

Basic model

Workflow and the structure of the basic and fine-tuned models are deeply discussed in this section. First of all, about bidirectional encoder model. This model involves

attention mechanism, that is responsible for tagging the most important parts of the sentence by identifying them through assigning relevant weights. In addition to that it uses transformer which consists of 2 parts: encoder and decoder. It is able to read the whole input sentence at once and make it possible for model to learn all the concept and be bidirectional. The structure of transform encoder is depicted below (Ref. to Fig. 5.1).

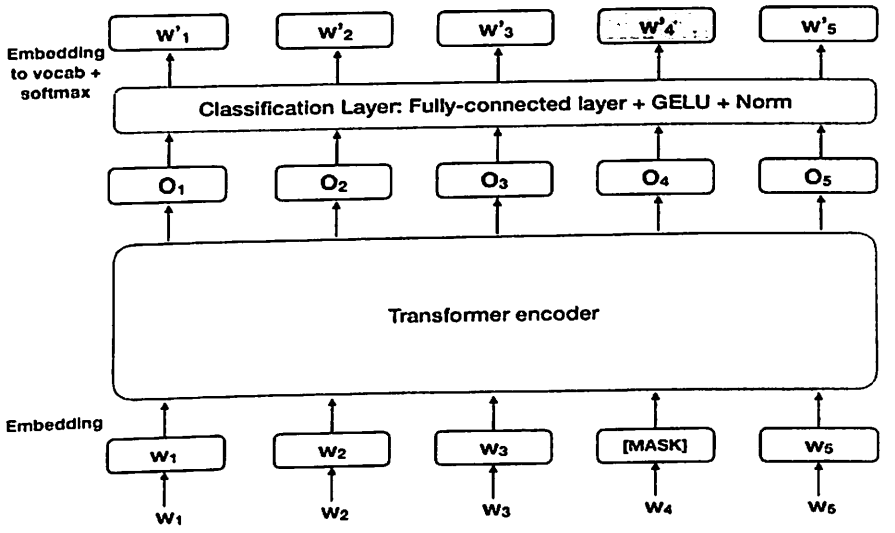


Figure 5.1: Transformer encoder

Fine-tuning the model

Transfer learning was applied to the model by using 2 unsupervised procedures:

- Masked Language Model

  It is performed in order to train bidirectional RNN model so it can be implemented as other language models, that can only be trained from right to left or vise versa. Input tokens were randomly hidden and then predicted. In general case it is efficient to hide about 15% of the input sentence and in order to save time not the whole sentence, but only hidden parts were predicted. Thus bidirectional pre-trained model is obtained.

- Next Sentence Prediction

  This task is involved in order to make it possible for model to understand two sentences' relationships from the given input.

In this work the source text and the question were given together but embedded differently. Only start and end vectors were provided while performing fine-tuning. The probability for the word that it can be inside the answer is calculated by dot product of word's tag with its cross entropy and start vector, followed by SoftMax loss for entire source text. Representation of an input for the model is depicted on the figure below (Ref. to Fig. 5.2)

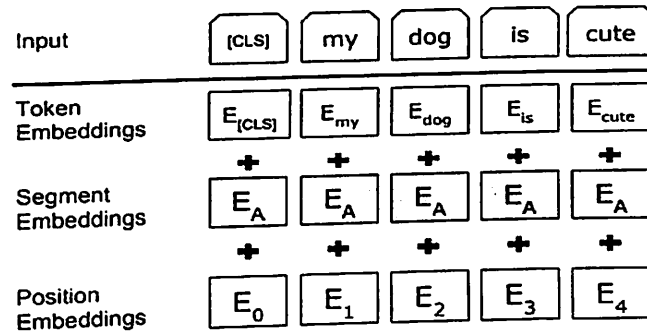


Figure 5.2: Model input sample

The model was trained by using Stanford Dataset for QA task that consists of 100 000 questions and answers. It is given a question and a text from where it should retrieve an answer.

5.2 Results and discussion

There are number of advantages were identified.

- After performing the fine-tuning the model became more simpler: number of parameters for initial model was 345 000 000 and after fine-tuning it became 110 000 000 only.
- After performing each tasks for transfer learning the performance of the model was only increasing.
- Using fine-tuned model is also efficient in terms of time, as initial model training requires several days, while training and importing the model after transfer learning can take several minutes.

Final results of the model are illustrated below (Ref. to Fig. 5.3, Fig. 5.4).

```
doc = "Victoria has a written constitution enacted in 1975, but based on \
    the 1855 colonial constitution, passed by the United Kingdom Parliament \
    as the Victoria Constitution Act 1855, which establishes the Parliament \
    as the state's law-making body for matters coming under state responsibility. \
    The Victorian Constitution can be amended by the Parliament of Victoria, \
    except for certain 'entrenched' provisions that require either an absolute \
    majority in both houses, a three-fifths majority in both houses, or the \
    approval of the Victorian people in a referendum, depending on the provision."

q = 'When did Victoria enact its constitution?'

answer = model.predict(doc,q)

print("Answer: " + answer['answer'])
print("Confidence: " + str(answer['confidence']))

Answer: 1975
Confidence: 0.9449190621565517
```

Figure 5.3: Result example

According to the results of the first testing sample, confidence level is high: nearly 0.95 and the answer is correct. However, one can deny that it is good only because of the fact they are standing together in one sentence.

However if to look at the second example (Ref. to Fig. 4), text is written for the peace of art "The path of Abai" of a famed Kazakh writer M. Auezov. The question is about the first love of Abai, however name of that person is not located in the same sentence with the word "love". Despite the fact that confidence level is low (around 0.4) the answer was constructed correctly by the model. It again shows the effectiveness of transfer learning and implementing Next Sentence Prediction task on the model.

```
doc = "Abai during his life used to speak on behalf of the people, although \
    his farther was very rich and powerful man. He had a very close friend \
    Yerbol who always helped him. Thanks to him, Abai had a chance to\
    meet with his first love, engaged at that time. Her name was Togzhan. \
    She taught him what is true love and \
    during all his life he kept the tender memories about her."

q = 'Who was Abai\'s first love?'

answer = model.predict(doc,q)

print("Answer: " + answer['answer'])
print("Confidence: " + str(answer['confidence']))

Answer: Togzhan.
Confidence: 0.39789274197056324
```

Figure 5.4: Result example 2

5.3 Conclusion

This work proposed an implementation of transfer learning on pre-trained bidirectional encoder representations from transformers model for question answering task, as it can be used in number of different language tasks like sentence classification, entity recognition and etc. Results have shown that this approach is certainly an achievement in the task of question answering that combines IR, NLP and DL. Most important factor is its approachable way of implementation. Having opportunity to perform transfer learning on this model is another advantage that makes it available for usage in other different tasks of NLP, which is not possible for all DL models. It can be stated as main contribution of this work. Further work can be about the involvement of Question Answering system to current model and probably implementation of other model structures and training the model for Kazakh language.

6. Conclusion

In this work, the implementation of Recurrent Neural Networks (RNN) for image scene captioning system is proposed. The system provides with a description for the given image. It is realized on Telegram Bot. The model was able to find the most relevant parts of the image and find out the most relevant words that describe the scene as it knows where to look, its focus is shifted across the image with the help of attention mechanism. It has an encoder-decoder architecture and transfer learning is implemented on the pre-trained encoder. Results have shown that it is possible to build an image captioning system with the help of DL tools and techniques. In some cases it was not so accurate in describing the image, but identifies objects and describes separately very well. Therefore the model might be fine-tuned again and optimized in the future. Further work may include the optimization of the model architecture and longer training.

References

[1] P. Shah, V. Bakrola, and S. Pati. (). Image captioning using deep neural architectures, [Online]. Available: https://arxiv.org/abs/1801.05568. (accessed: 08.10.2019).

[2] V. Ordonez, G. Kulkarni, and T. L. Berg. (2011). Im2text: Describing images using 1 million captioned photographs, [Online]. Available: https://papers.nips.cc/paper/4470-im2text-describing-images-using-1-million-captioned-photographs. (accessed: 10.10.2019).

[3] D. Geman, S. Geman, N. Hallonquist, and L. Younes. (2014). A visual turing test for computer vision systems, [Online]. Available: https://www.pnas.org/content/112/12/3618. (accessed: 14.10.2019).

[4] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. (2016). Show and tell: Lessons learned from the 2015 mscoco image captioning challenge, [Online]. Available: https://arxiv.org/abs/1609.06647. (accessed: 13.10.2019).

[5] G. Ding, M. Chen, S. Zhao, H. Chen, J. Han, and Q. Liu. (2019). Neural image caption generation with weighted training and reference, [Online]. Available: https://link.springer.com/article/10.1007/s12559-018-9581-x#Sec2. (accessed: 13.10.2019).

[6] ——, (2019). Neural image caption generation with weighted training and reference, [Online]. Available: https://link.springer.com/article/10.1007/s12559-018-9581-x#Sec2. (accessed: 13.10.2019).

[7] D. Khosla, R. Uhlenbrock, and Y. Chen. (2019). Automated scene understanding via fusion of image and object features, [Online]. Available: https://sci-hub.se/10.1109/THS.2017.7943474. (accessed: 10.02.2020).

[8] D. G. Vineeta Singh and A. Ralescu. (2017). Image understanding - a brief review of scene classification and recognition, [Online]. Available: http://ceur-ws.org/Vol-1964/ML2.pdf. (accessed: 10.02.2020).

[9] E. Frank, D. Frank, and K. Frank. (2013). Image description using vi-sual dependency representations.

[10] DFang, Hao, Gupta, Saurabh, Iandola, Forrest, Srivastava, Rupesh, Deng, Li, Doll, Piotr, Gao, Jianfeng, He, Xiaodong, Mitchell, Margaret, Platt, and John. (2014). From captions to visual conceptsand back, [Online]. Available: https://arxiv.org/abs/1411.4952. (accessed: 10.03.2020).

[11] S. Hochreiter and J. Schmidhuber. (1997). Long short-term memory. (accessed: 10.02.2020).

[12] Hodosh, Micah, Young, Peter, Hockenmaier, and Julia. (2013). Framingim-age description as a ranking task: Data, models and evalu-ation metrics. (accessed: 10.02.2020).

[13] Karpathy, Andrej, Li, and Fei-Fei. (2014). Deep visual-semantic align-ments for generating image descriptions, [Online]. Available: https://arxiv.org/abs/1412.2306. (accessed: 10.02.2020).

[14] Kingma, D. P., Ba, and Jimmy.Adam. (2019). A methodfor stochastic optimization, [Online]. Available: https://arxiv.org/abs/1412.6980. (accessed: 10.02.2020).

[15] Kiros, Ryan, Salahutdinov, Ruslan, Zemel, and Richard. (2014). Multimodal neural language model. (accessed: 10.02.2020).

[16] ——, (2014). Uni-fying visual-semantic embeddings with multimodal neural lan-guage models, [Online]. Available: https://arxiv.org/abs/1411.2539. (accessed: 10.02.2020).

[17] Kulkarni, Girish, Premraj, Visruth, Ordonez, Vicente, Dhar, Sagnik, Li, Siming, Choi, Yejin, Berg, A. C., Berg, and T. L. (2013). Babytalk: Understanding and generating simple image descriptions.

[18] Kuznetsova, Polina, Ordonez, Vicente, and Berg. (2012). Collective generation of naturalimage descriptions.

[19] Larochelle, Hugo, Hinton, and G. E. (2010). Learning to com-bine foveal glimpses with a third-order boltzmann machine.

[20] M. nad Junhua, Xu, Wei, Yang, Yi, Yuille, and Alan. (2014). Deep captioning with multimodal recurrent neural networks(m-rnn), [Online]. Available: https://arxiv.org/abs/1412.6632. (accessed: 10.02.2020).

[21] Mitchell, Margaret, Han, Xufeng, Dodge, Jesse, Mensch, and Alyssa. (2012). Generating image descriptions from computer vision detections.

[22] Mnih, Volodymyr, Hees, Nicolas, Graves, Alex, Kavukcuoglu, and Koray. (2014). Recurrent models of visual attention.

[23] Pascanu, Razvan, Gulcehr, Caglar, Cho, Kyunghyun, Bengio, and Yoshua. (2019). How to construct deep recurrent neural networks. (accessed: 10.02.2020).

[24] Rensink and R. A. (2000). He dynamic representation of scenes.

[25] S. K. and Z. A. (2014). Very deep convolu-tional networks for large-scale image recognition.

[26] S. Jasper, S. Kevin, Z. Richard, and A. Ryan. (2014). Input warping for bayesian optimization of non-stationary functions, [Online]. Available: https://arxiv.org/abs/1402.0929.

[27] S. A.F. (2019). Using nlp or nlp resources for information retrieval tasks.

[28] R. Horev. (2018). State of the art language model for nlp.

[29] A. Andrenucci and E. Sneiders. (). Automated question answering: Review of the main approaches.