Developing and Optimizing a Model for Creating a Learning Chatbot Using the Example of Technical Support

A A Tishchenko¹, P A Tishchenko¹, T A Shestakov¹

¹Bryansk State Technical University, 241035, Central Federal district, Bryansk region, Bryansk city, Boulevard 50 let Oktyabrya, house 7

E-mail: karkuc@yandex.ru

Abstract. A model for building a chatbot to automate technical support work was developed, the model was optimized and the developed chatbot was tested. This development will help to reduce the load on the existing technical support system and to solve users’ routine tasks without human intervention.

1. Introduction

Chatbots are a relevant tool for communicating with users; this is due to the fact that chatbots are widely applied in many subject areas to solve everyday questions and users’ tasks. The popularity peak of chatbots can be considered the time when they began to be used in social networks and instant messengers, such as Facebook, WhatsApp, Vkontakte and others. The importance of this theme is due to the high popularity of instant messengers and automation tools such as chatbots among Internet users. Chatbots make it easier to perform daily trivial tasks such as getting information about the weather, traffic jams, the latest news and others. And also the advantage of developing chatbots against classic applications is the ability to provide services on a single messenger platform or social network.

2. Theoretical part

A chatbot is a special program that simulates human speech and behaviour and acts as a virtual interlocutor in a chat with one or more users.

A chatbot for technical support is an important task that allows you to provide technical support around the clock and reduce the load on call centres [1]. There are several commercial solutions for this task, which were considered namely VirtualSpirits Chatbot which is a chatbot platform for technical support and IT support, 1C Connect which is a chatbot platform for technical support and IT support [2].

The analysis of the considered solutions has shown the presence of neural networks in each of them, in one form or another, which suggests that this tool allows you to efficiently process natural language texts and, therefore, this mechanism should be applied in this study as well.

In this work, a learning chatbot was chosen because it is necessary to answer users’ questions that describe problems in a natural language and script bots are not suitable in this case, since they do not allow understanding the meaning of what the user has written, in contrast to what students have written. In this case, the tasks of automating technical support activities will be solved, and the results can also be used as a system to simplify the work of technical support operators when answering
users’ typical questions.

To train the developed chatbot, it is necessary to define the data corpus. This study used Twitter customer support dataset, which is a large, modern set of tweets and responses to help to fulfill innovations in a natural language understanding and in conversational patterns, as well as to explore customers’ modern support practices and their impact [3].

The dataset contains responses from many companies, such as: Apple, Amazon, Uber, PlayStation and others. The dataset is a file of CSV format, where each line represents a tweet. The original dataset is 500Mb.

Before training the model, you need to build and clear the dataset. Not all fields in the current set are useful for creating a chatbot and therefore they need to be processed or deleted. To train the chatbot, it was decided to consider users’ questions and answers from technical support from only one company, therefore, automating the company’s technical support was chosen for training and testing Apple, since this company has the most tweets.

Next, the dataset was converted to Question-Answer format, that is, the answer was selected for each tweet, if any. After selecting messages from only one company, the fields such as tweet_id, author_id, inbound, created_at, response_tweet_id and in_response_to_tweet_id were deleted since they are no longer useful.

The next step was to delete links from the text. In most cases, links in this set are outdated and there is no sense in sending them to the user, but if the company has a list of relevant pages, this stage can be modified and leave links from the white list that lead to the materials which are useful for solving users’ problems.

Further, all punctuation was deleted from the text, since the model has no idea about English grammar rules and can place punctuation marks only based on the statistics of their usage with words, which can negatively influence the work result [4].

Next, it was necessary to process users’ questions. To do this, it was significant to choose between word processing with the help of stemming or lemmatization. Having studied the approaches and features of these ways of processing words, it was decided to use stemming, and more specifically, Porter’s stemmer [5]. The next step was to delete the most rarely used words, tags were added to each tweet <start> and <end>, which will be used by the model to understand where the question starts and ends. In addition, all words have been digitized using a tokenizer.

To train a chatbot, it is necessary to correctly divide the sample, select the model architecture and criteria for evaluating the results. The BLUE metric was used to evaluate the results. The result of the BLUE metric is always a number between 0 and 1. This value indicates how similar the candidate’s text is to help texts, with the values closer to 1 representing more similar texts. Few human translations will receive a score of 1, as this would mean that the candidate is identical to one of the control translations. For this reason, there is no need to score 1 point. The advantage of the BLUE algorithm is that it correlates well with users’ ratings. It averages the number of errors in individual sentences for the entire test suite, instead of trying to develop a user’s accurate score for each sentence.

\[
p_n = \frac{m_n}{l_n} \quad (1)
\]

\[
BP = \min(1.0, \exp(1 - \frac{\text{len}(R)}{\text{len}(T)})) \quad (2)
\]

\[
\text{BLEU} = BP \cdot \exp\left(\sum_{n=1}^{N} W_n \log p_n\right) \quad (3)
\]

where \(m_n\) is the number of n-grams that have matched between translation \(T\) and the correct result \(R\), \(l_n\) is the total amount of n-gram, \(W_n\) is match weight of n-gram (the sum should be equal to one) [2].
3. Practical significance and the results of implementation
To solve the problem, a model was chosen that is most frequently used for machine translation, but can also be used for question-answer systems. This model is called seq2seq. Seq2seq models use two recurrent neural networks (RNN).

An interpreted programming language Python was used to process the data, as well as to build a chatbot model. This language is chosen because many libraries have been written for it that allow you to quickly and efficiently work with data and use popular algorithms and models of computational linguistics and neural networks.

In the course of this research, the following Python libraries were used: NumPy, Pandas, NLtk, Langdetect, Pickle, Matplotlib and TensorFlow [7,8,9].

The software system includes four scripts, each of which performs its own functions. Each task is moved to separate scripts instead of a sequential program, because this allows you to develop the program in stages and, if changes are necessary not to re-execute all the scripts but to carry out only the scripts following the current one.

1. The script “Prepare_data” is designed to prepare data for training. This script performs the following actions: it loads the original data in the CSV format, deletes the data from missing values, filters messages only from the necessary authors, performs text processing and tokenizes words. After executing the script, we get two files with the processed text in question-answer format, where each word in the sentence is digitized, which can already be used for training. The first file contains the training dataset and the second one contains the test dataset.

2. Script “Neural_net” is auxiliary for the script “Train”. It contains classes that describe the architecture of the neural network.

3. Script “Train” is designed to train a neural network. In this script, the training and test data obtained from the first script are loaded. Next, the parameters for training the neural network are set, such as the number of training epochs, the neural network optimizer, the path to save the key training points for subsequent network recovery to continue training or using the network. After executing the script, we have files with the weights of the neural network, which can later be restored and applied for the chatbot.

4. Script “Chatbot” is designed to predict answers to users’ questions. And also it is possible to build attention maps to better understand the model results.

To determine the optimal parameters of the developed model, it was tested with various values of such parameters as: the size of the vector representation of the word, the size of the attention layer, the size of the GRU output layer.

First, consider the effect of the size of the word vector representation on the value of the loss function (Loss) and the value of the BLUE metric. The model is tested for the following values of this parameter: 128, 256, 512, which are the sizes of vector representation, see Fig. 1 and Fig. 2.
As it can be seen from Fig. 1, increasing the dimension of the output vector does not give an increase in the learning rate, therefore, value 128 is a sufficient size of the word representation.

Looking at Fig. 2, it becomes clear that value 128 is sufficient for the model and it shows better results compared to the values of higher dimensions. This is most likely due to the fact that large models take longer to be trained. Therefore, in the final model, the size of the vector representation of the word will be 128.

Next, the effect of the attention layer size on the model results was examined. The model was
tested for the following values of this parameter: 3, 7, 12 neurons in the attention layer, see Fig. 3 and Fig. 4.

As it can be seen from Fig. 3, the optimal value for the model is 7 neurons in the attention layer.

Looking at Fig. 4, it becomes clear that the value of 7 neurons allows the model to be in a state of balance and to obtain better accuracy values than the models having more and fewer neurons.

Next, influencing the size of the GRU output layer on the model results was considered. The model is tested for the following values of this parameter: 256, 512, 1024, 1536 neurons, see Fig. 5 and Fig. 6.
Figure 5. Comparative graph of the loss function when training the model.

As it can be seen from Fig. 5, 512 neurons in the GRU output layer are sufficient for the model. The value of 256 neurons is not sufficient since the value of the loss function after 25 iterations does not decrease. And the values of 1024 and 1536 neurons in this case are redundant so they do not allow achieving better results, but only significantly increasing the training time and the final model weight.

Figure 6. Comparative graph of the BLUE metric when training the model.

Looking at Fig. 6, it becomes clear that the value of 512 neurons allows the model to obtain better metric accuracy than the model having more or fewer neurons.

As a result of studying the model optimal parameters, the following values were chosen: 128 neurons is the size of the vector representation of the word, 7 neurons is the size of the attention layer, 512 neurons is the size of the GRU output layer.

To start working with the chatbot, you must first train the model using the script “Train” and then run the script to work directly with “Chatbot” itself.

After a short boot, the console will display the message “Ask your questions:” after which you must enter your question and press the Enter key. Next, the bot will generate its answer and print it onto the console. An example of generated answers to questions to the Twitter account of AskPlayStation is presented in Table 1.
Table 1

| Question                                                                 | Generated answer                                                                 |
|--------------------------------------------------------------------------|----------------------------------------------------------------------------------|
| My account is telling me to reset my password but i dont know my date of birth but i know my password for the account | no worries we have sent you a direct message via twitter with further instructions |
| what do i do, when my playstation is connected to the internet, but won't log in to psn? | lets check it out what console are you using what error message or error code do you see any kind of error message or error code |
| I bought a year worth of playstation plus and the code isn't working?    | hi there please let us know the exact error code or message that you receive when you try to redeem the code |
| hi, my playstation 4 freezes everytime i play a game. i've tried everything, i don't know how to fix it, what do i do? | sorry to know that please power cycle your network devices and try again |

As it can be seen from Table 1, in most cases, technical support tries to translate the dialogue into private messages and not to make the solution to the problem public. Or the second most likely outcome of the events is that technical support is trying to find out the error code to provide additional information.

Next, answers were generated for some questions from the test dataset (see Table 2).

Table 2

| Question                                                                 | Real answer                                                                 | Generated answer                                                                 | Is the answer correct? |
|--------------------------------------------------------------------------|------------------------------------------------------------------------------|----------------------------------------------------------------------------------|------------------------|
| plea help get account back                                               | happy to help check your messages                                            | hi there lets look into that please check your dms for further instructions      | yes                    |
| old playstat account cant reset pw say wrong dob dont help               | totally understand please follow us for further assistance                   | sorry to know that please follow us via twitter and let us know so we can send you a dm with further instructions thanks | yes                    |
| hi help problem ps turn ps power button doesnt work control isnt link usb port cant read control adaptor reason help somehow | thats not good please let us know what is the color of the power light you see on the console | hello check out the next article to troubleshoot the ps                                      | no                     |
| guy psn work problem server                                              | sure we can help please try the following steps related to the error code    | here to help psn is up and running do you receive a look at the exact error code   | yes                    |
| bought minecraft ps want let play surviv onlin friend get refund         | hi nick for refund info please check the following link                       | hello for refund info please check the following link                            | yes                    |

From Table 2, you can see that the model roughly understands the meaning of what is written and the generated responses are close to the real answers of technical support. In addition, this research has
attached and reviewed questions and answers for another company namely AskSpectrum.

Unfortunately, there are no publicly available data on technical support dialogues with customers in private messages. These data would help to better solve users’ questions, and not to answer with sufficiently general phrases.

The model in its current state can be used to automate users’ initial requests, to reduce the receipt of primary information, or to serve as a support system for technical support operators, which will simplify routine questions and actions in their work. In addition, when creating a similar corpus of data from Russian technical support companies, this model can also be applied to build a chatbot using the Russian language. This research is interesting in terms of working out the approach and developing a methodology for creating a chatbot for your company, as well as for teaching students when they build their own learning chatbots.

4. References
[1] Chatbot for technical support [Electronic resource] www.virtualspirits.com URL: https://www.virtualspirits.com/ru/chatbot-for-technical-support.aspx (access date: 01.12.2019)
[2] 1C Connect [Electronic resource] 1c-connect.com URL: https://1c-connect.com/ru/chatbot/ (access date: 01.12.2019)
[3] Customer support on Twitter [Electronic resource] www.kaggle.com URL: https://www.kaggle.com/thoughtvector/customer-support-on-twitter (access date: 01.12.2019)
[4] Boyarsky K K 2013 Introduction to computational linguistics [Electronic resource]: tutorial Electron. text data (St. Petersburg: ITMO University) 73 p Access mode: http://www.iprbookshop.ru/71485.html EBS “IPRbooks”
[5] Stemmer Porter [Electronic resource] wikiperdia.org URL: https://ru.wikipedia.org/wiki/Стеммер_Портера (access date: 05.12.2019)
[6] Yakhyaeva G E 2017 Fuzzy sets and neural networks [Electronic resource]: tutorial Electron. text data (Moscow) Saratov: Internet University of Information Technologies (INTUIT), University education 320 p Access mode: http://www.iprbookshop.ru/67390.html EBS “IPRbooks”
[7] NumPy [Electronic resource] wikipedia.org URL: https://ru.wikipedia.org/wiki/NumPy (access date: 10.05.2019)
[8] Nltk [Electronic resource] wikipedia.org URL: https://ru.wikipedia.org/wiki/Natural_Language_Toolkit (access date: 12.12.2019)
[9] TensorFlow [Electronic resource] wikipedia.org URL: https://ru.wikipedia.org/wiki/TensorFlow (access date: 17.05.2019)
[10] Saraev P V 2017 Machine learning methods [Electronic resource] Electron. text data (Lipetsk: Lipetsk State Technical University) ELS ABU 48 p Access mode: http://www.iprbookshop.ru/83183.html Electronic Library System “IPRbooks”
[11] Fedin F O 2012 Data analysis Part 1 Preparing data for the analysis [Electronic resource]: tutorial Electron. text data (Moscow: Moscow City Pedagogical University) 204 p Access mode: http://www.iprbookshop.ru/26444.html Electronic Library System “IPRbooks”
[12] Tsilikovsky I A 2010 Methods for analyzing knowledge and data [Electronic resource]: lecture notes Electron. text data (Novosibirsk: Novosibirsk State Technical University) 68 p Access mode: http://www.iprbookshop.ru/45385.html Electronic Library System “IPRbooks”
[13] A complete guide to developing chatbots [Electronic resource]: Electron. text data Access mode: https://ux.pub/polnoe-rukovodstvo-po-razrabotke-chatbotov/
[14] How to build a chatbot from scratch in Python: detailed instructions [Electronic resource]: Electron. text data Access mode: https://neurohive.io/ru/tutorial/kak-sozdat-chat-bota-s-nulja-na-python-instrukcija
[15] Leonov E A Intellectual subsystems for collecting information from the internet to create knowledge bases for self-learning systems
[16] Filippova In: Abraham A, Kovalev S, Tarassov V, Snasel V, Vasileva M, Sukhanov A (eds) Text : electronic Proceedings of the Second International Scientific Conference “Intelligent Information Technologies for Industry” (IITI’17) IITI 2017 Advances in Intelligent Systems and Computing vol 679 (Springer, Cham) pp 95-103 DOI:10.1007/978-3-319-68321-8_10

[17] Leonov, Y U 2019 Selection of rational schemes automation based on working synthesis instruments for technological processes (Yelm, WA, USA: Science Book Publishing House LLC) 192 p ISBN: 978-5-9765-4023-1 Text : direct

[18] Averchenkov A V 2018 Development of Mathematical Model of Information System for Inventory and Monitoring of Hardware and Software Based on Fuzzy Logic Text: direct Quality. Innovation. Education 7 (Moscow: European Centre for Quality) pp 105-112 ISSN: 1999-513X

[19] Kozhukhar V M 2010 Monitoring and Prediction of Regional Demand for Highest Scientific Degree Specialists: monograph Bryansk State Technical University (Bryansk)

[20] Kuzmenko A A 2018 Developing the structure of the WEB-resource based on the needs of the end user Text: direct In the collection: New information technologies in scientific research Materials of the XXIII All-Russian scientific and technical conference of students, young scientists and specialists Vol 2 (Ryazan: Ryazan State Radio Engineering University) pp 183-185 ISBN: 978-5-7722-0274-6