The Advantages of Human Evaluation of Sociomedical Question Answering Systems

Victoria Firsanova

***Abstract*—The paper presents a study on question answering systems evaluation. The purpose of the study is to determine if human evaluation is indeed necessary to qualitatively measure the performance of a sociomedical dialogue system. The study is based on the data from several natural language processing experiments conducted with a question answering dataset for inclusion of people with autism spectrum disorder and state-of-the-art models with the Transformer architecture. The study describes model-centric experiments on generative and extractive question answering and data-centric experiments on dataset tuning. The purpose of both model- and data-centric approaches is to reach the highest F1-Score. Although F1-Score and Exact Match are well-known automated evaluation metrics for question answering, their reliability in measuring the performance of sociomedical systems, in which outputs should be not only consistent but also psychologically safe, is questionable. Considering this idea, the author of the paper experimented with human evaluation of a dialogue system for inclusion developed in the previous phase of the work. The result of the study is the analysis of the advantages and disadvantages of automated and human approaches to evaluate conversational artificial intelligence systems, in which the psychological safety of a user is essential.**

***Keywords*—Autism Spectrum Disorder, Dialogue Systems, Natural Language Processing, Question Answering.**

I. Introduction

Question Answering (QA) systems deal with tasks from different research areas, such as Natural Language Understanding (NLU), Conversational AI (ConvAI) and Information Retrieval (IR) [1]. The QA task is to output a consistent answer to a user query in a form of a question. The answer of a QA system can be based on information from a knowledge base, such as a knowledge graph or a collection of texts [2]. Such strategies as Knowledge Base Question Answering (KBQA), Machine Reading Comprehension (MRC or Extractive QA) and Generative QA have proved to be effective on task-oriented, closed- and open-domain QA [2, 3]. Nevertheless, QA is a challenging research area.

There are some challenges related to user behaviour. The user behaviour is a priori uncontrollable; it is impossible to predict all the scenarios, which causes multiple problems. For example, any QA system might come across a so-called lexical gap when a user question contains some vocabulary that is not presented in a model database [4]. Similarly, it is difficult to foresee all the question types that a user would like to engage. The practice shows that it is easier to deal with factoid questions, like "*What is the capital of Russia?*". Many QA studies are focused on this question type. However, the practice shows that users also tend to ask other question types, like definitional ("*What is ASD?*") or list ones ("*List the earliest symptoms of flu.*") [1].

The development of a specific QA type brings more challenges. For example, Closed-Domain QA (CDQA) systems that deal with data on a particular topic [1], like COVID-19 or koalas, might have obstacles while executing their program if the knowledge base lacks some information from a user question. Another problem with CDQA is that some available training datasets can be low-resourced due to the domain specifics. As a result, the model development involves issues of low-resourced Natural Language Processing (NLP). A lexical gap problem might become acute due to the knowledge base volume and possible rare domain-specific words that could be ignored during the model training.

This paper focuses on the challenges of sociomedical QA on the example of a QA system development process. The study describes experiments on a dialogue system for the inclusion of people with Autism Spectrum Disorder (ASD). The basic idea of the study is that such a QA system should be psychologically safe by not providing misleading answers that could frighten or disturb a user. The study shows that the main challenge of such a system is its controllability. According to [5], a sociomedical system should properly perform its program while managing domain requirements. It can be assumed that combining different frameworks might be applied to reach this goal.

In this paper, the evaluation approaches are in the spotlight. Considering the idea that human evaluation (evaluation with possible users of a new QA model) might shed light on problems of a new sociomedical dialogue system, the author aims to find out if this approach is indeed necessary to measure the model quality. The results of the author's approach to human evaluation are compared with the results of the automated evaluation with F1-Score metrics. The data for the experiments is based on an MRC dataset about ASD [6] described in Section III. The experiments conducted during the study are based on state-of-the-art Transformer-based NLP models listed and described in Section VI. The process of human evaluation is given in Section VIII. The research results include tables comparing the performance of different models and approaches and the analysis of the advantages and disadvantages of automated and human evaluation methods.

Manuscript received October 14, 2021.

V. Firsanova is with the Computer and Applied Linguistics Department, Saint Petersburg State University, Saint Petersburg, Russia, ORCID 0000-0002-8474-0262 (e-mail: vifirsanova@gmail.com).

II. Related Work

The history of automated question answering starts in the '60s and '70s with dialogue systems like BASEBALL [7] and LUNAR [8]. Both systems represented natural language interfaces for closed-domain knowledge bases developed to answer user questions by reading a user question from punched cards, processing the input with dictionaries and parsers and printing output. The BASEBALL was based on the concept of specification list (a list representing the information in the form of attribute-value pairs, for example, "*Team = Red Sox, Month = May*"). The system answered questions about dates, places, teams and scores of baseball games. The LUNAR consisted of a transition network parser, semantic interpretation and retrieval components; the system gave information about lunar geological samples.

Since then, the understanding of question answering has changed. In 2019, Gao et al. in [2] listed Knowledge Base Question Answering (KBQA) and neural text-QA agents among modern question answering applications. KBQA systems are based on structured databases like DBPedia [9] that are often called knowledge graphs. According to [2], the core of neural text-QA agents is Machine Reading Comprehension (MRC) task. The task is to answer questions posed on text passages.

Recent papers on question answering focus on data- and model-centric challenges such as human-augmented Reading Comprehension (RC) dataset annotation methodologies [10], the development of computationally cheaper state-of-the-art QA models [11], RC models for questions with several non-contiguous answers in a reading passage [12], and others. Attention mechanisms allowing quantifying the interdependence between the elements of input and output (if it is General Attention) or within the input only (if it is Self-Attention) became an efficient tool for boosting the model performance in both KBQA and text-QA [2]. For example, different models based on the Transformer architecture [13] achieve state-of-the-art performance in MRC. That makes them more and more common in dialogue systems.

The Transformer architecture is based on a Feed-forward Neural Network (FFN) consisting of two linear transformations and an activation function as in

$$FFN(x) = max(0, xW_1 + b_1)W_2 + b_2, \quad (1)$$

and scaled dot-product attention units consisting of queries (Q), keys (K), and values (V) that is calculated as

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

Multi-head attention used in the Transformer allows applying attention function in parallel by projecting and concatenating queries, keys and values as

$$MultiHeadAttention(Q, K, V) = Concat(head_1, \dots, head_h)W^o,$$
$$\text{where } head_i = Attention(QW_i^Q, KW_i^K, VW_i^V). \quad (3)$$

The success of the development of a QA model depends on the properties of training data. The structure and design of QA datasets are differentiated according to the type of QA task for which they will be used. For example, Figure 1 illustrates types of question answering according to the domain coverage. Open-Domain Question Answering (ODQA) aims to answer natural language questions using retrieval algorithms to extract information from large-scale databases [14]. The purpose of Closed-Domain Question Answering (CDQA) is to give answers to natural language questions under a specific domain, for example, by extracting information from a domain-specific knowledge base.

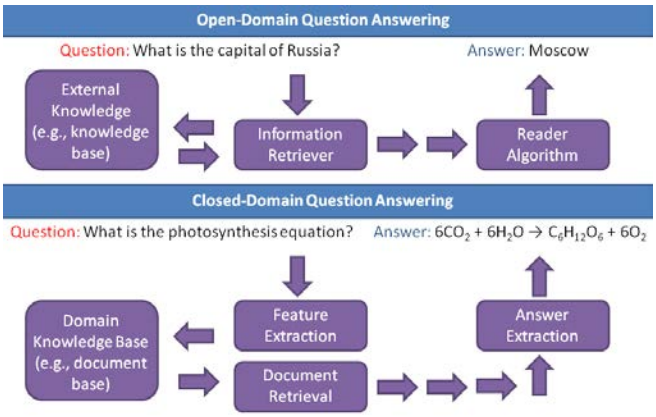

Figure 1. Types of question answering according to the domain coverage

Machine Reading Comprehension (MRC) task can be applied to both closed- and open-domain question answering. Datasets play a crucial role in the success of such systems. For example, Stanford Question Answering Dataset (SQuAD) [15, 16] is a large-scale MRC dataset consisting of more than 100,000 questions posed by crowdworkers on texts from around 500 Wikipedia articles. The structure of this MRC dataset is that the answer to every question is a piece of text from the corresponding paragraph. This one of the most representative datasets is now being widely used as a reference for building other MRC datasets. Those datasets can be open-domain MRC datasets in different languages, like SberQuAD [17], a Russian MRC dataset based on SQuAD. Those can also be closed-domain MRC datasets. Moreover, those can be spoken datasets, like Spoken SQuAD, a listening comprehension dataset [18].

This paper presents a study based on the Autism Spectrum Disorder and Asperger Syndrome Question Answering Dataset 1.0 (ASD QA) [6] collected by the author. The structure of this closed-domain dataset is based on SQuAD but has some modifications. The description of the dataset structure is in Section III. The paper's contributions are following.

Firstly, the author tunes the dataset design besides traditional model training and hyperparameter optimization experiments to achieve higher metric scores. The author analyses the results of such a twofold model development. Secondly, the author proposes a human evaluation methodology complementary to the traditional automated evaluation techniques used in MRC. Finally, the author analyses the advantages and disadvantages of automated and human evaluation techniques in light of conversational AI issues that appear when building intelligent systems where

user psychological safety is essential.

III. DATASET

The Autism Spectrum Disorder and Asperger Syndrome Question Answering Dataset 1.0 (ASD QA) [6] structure is similar to the one in SQuAD v2.0 [16]. Figure 2 illustrates the ASD QA structure. The dataset consists of 1,134 sets of question-answer pairs and corresponding reading passages. The author collected the data from a Russian informational source about Autism Spectrum Disorder and Asperger Syndrome available online on https://aspergers.ru/. Articles from the website were extracted with BeautifulSoup HTML parser using Python 3.6.9. The author divided all the texts into reading passages up to 512 symbols.

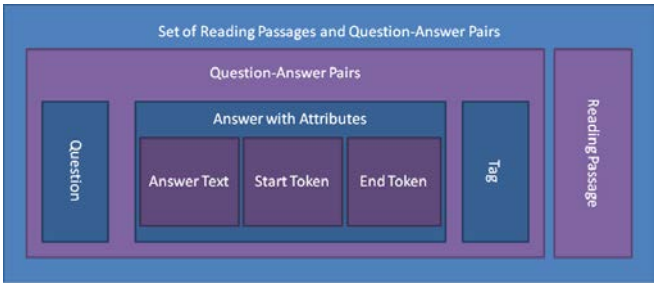

Figure 2. The ASD QA dataset structure

The author provided each reading passage with several sets of question-answer pairs. These sets contained from 5 to 15 pairs according to the volume and content of each reading passage. The author manually posed questions to the content of a reading passage. The answer to each question should have been a span from a passage paragraph. Apart from the answer text, the author provided the dataset with position numbers of the first and last span tokens.

The author provided each question-answer pair with a tag reflecting if a question is relevant to a reading passage. 5% of questions from the dataset are deliberately unanswerable and irrelevant to the topic of a reading question. For example, "*How neural networks work?*" is an irrelevant question in the dataset about autism.

The idea of this step is to train a dialogue system to ignore user questions provoking chitchat. A dialogue system about autism spectrum disorder should be educational and not entertaining. However, there is a probability that some users would like to ask such a system some questions to entertain themselves. The chitchat in the educational system aiming to inform people about autism spectrum disorder might become destructive. Provocative questions might cause uncontrollable text generation. That might end with creating false information and misconceptions about autism and building a negative attitude towards people with special needs.

The ASD QA dataset has four modifications developed for the data-centric experiments (see Section VII). All the modifications are available on FigShare [6]. Figure 3 illustrates their structure.

The Original, Half-Sized and Shortened versions of the dataset have the same structure; see the top right-hand corner in Figure 3. The version called Half-Sized includes 50% of the original shuffled data. The Shortened version is a copy of the original dataset with answers shortened where possible. For example, if a question from the dataset could have several answers of different lengths, the author appended only the shortest one to the Shortened dataset modification. For instance, of two answer variants, "*autism is a lifelong state of being*" and "*a lifelong state of being*", the author appended only the second shorter one.

The structure of the Multiple version of the dataset is on the left-hand side in Figure 3. Unlike in the other three versions, in this one, a question could have several (up to four in order not to overload the dataset) answers of different lengths and contents if possible. The structure of the version called No Impossible is on the lower left-hand corner in Figure 3. This version is a copy of the original one without irrelevant (or impossible) questions. The author filtered the dataset by the tag showing the irrelevance of a question to a reading passage. If the tag value was True (irrelevant), the author omitted the question-answer pair from the dataset.

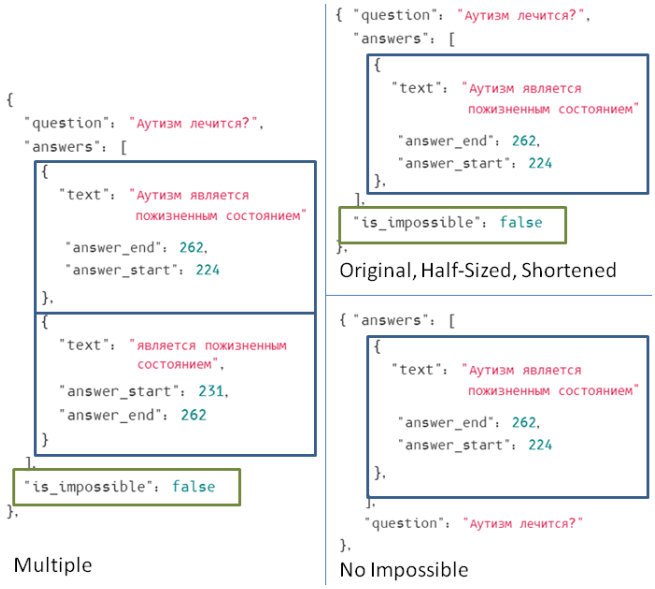

Figure 3. The structure of the ASD QA dataset modifications

The dataset covers three topics. The topics are the following: (1) "The general information about autism spectrum disorder and Asperger syndrome"; (2) "Communication between neurotypical and autistic people"; (3) "Guidelines for parents of autistic children: Sport and autism spectrum disorder". The topics are based on the information from https://asperger.ru/.

The topic coverage is random and incomplete because the ASD QA is a work in progress, and to date, the author compiles the dataset manually alone. The experiments described in Sections VI and VII are conducted before the dataset is complete deliberately. The author believes that studies on conversational AI at the early stages of dataset development might shed light on problems that might arise later during the production. Experiments on a small, low-resourced dataset might allow solving some data-centric issues that would be much harder to solve working later on larger datasets.

The author used the original dataset and its modifications for the experiments described in Sections VI and VII. The statistics of the dataset are following. The dataset contains 96 reading passages. The overall length of the reading

passages is 45,400 symbols, 6,578 words. The maximum volume of a reading passage is 512 symbols. The dataset includes 1,134 question-answer pairs. The length of sequences of questions and answers is 179,174 symbols, 26,269 words.

IV. Technical Setup

The author fine-tuned all the Transformer-based models (see Subsection SOTA-Models in Section VI) with NVIDIA Tesla K80 Graphics Processing Unit (GPU) provided by Google Colab. The code for data pre-processing, model fine-tuning and evaluation, and output extraction was created in Python 3.6.9. The code is available in the study repository on https://github.com/vifirsanova/ASD-QA. The whole program was implemented on Google Colab environment.

The dataset (see Section III) is a JavaScript Object Notation (JSON) object, and it was processed with the eponymous Python library. The data was split into three sets with the Scikit-learn train-test-split method. 70% of the dataset was used for model training; 15% was used for model evaluation; 15% was used for testing.

The model training was performed with the PyTorch open-source machine learning library. The author used the HuggingFace Transformers package to fine-tune chosen Transformer-based models (see Subsection SOTA-Models in Section VI).

V. Automated Evaluation

The author has chosen Precision, Recall and F1-Score for the automated evaluation of the systems. According to [15, 16], developers usually evaluate machine reading comprehension systems using F1-Score (a harmonic mean of the Precision and Recall) and Exact Match (EM) metrics.

A system gets 1 EM point for each answer that exactly matches a corresponding sample from the evaluation dataset. Otherwise, it gets 0 points. The author did not use EM metrics in this study. Unlike SQuAD or other MRC datasets, the ASD QA dataset has longer answers (for example, one- or two-sentence long), which do not necessarily require an exact match because they can be losslessly truncated.

Moreover, usually, machine reading comprehension studies do not take into consideration Precision and Recall scores. In this study, the author considered these two metrics to analyse how accurate and complete can the system perform. Precision (P) showing a proportion of correct positive outputs is the fraction of true positive model answers among all the retrieved positives. Recall (R) showing a proportion of positives identified correctly is the fraction of true positive model answers among true positives and false negatives. P and R are usually calculates as follows:

$$P = \frac{true\ positives}{true\ positive + false\ positives}, \quad (4)$$

$$R = \frac{true\ positive}{true\ positives + false\ negatives}. \quad (5)$$

In machine reading comprehension, the answer to which output or separate token is positive or negative is unobvious. According to the SQuAD [15, 16] evaluation script, the automated system evaluation should be done on a token level. For example, true positives are tokens shared between a correct (gold) answer from the evaluation dataset and an output. Then false positives are output tokens absent in the gold answers, and false negatives are gold tokens absent in the shared token set. Modified in such a way P and R are calculated as follows:

$$PModified = \frac{shared}{shared + (predicted - shared)}, \quad (6)$$

$$RModified = \frac{shared}{shared + (gold - shared)}. \quad (7)$$

The F1-Score was not modified for the study. The author used a harmonic mean of modified Precision and Recall that is calculated as follows:

$$F = \frac{2PModifiedRModified}{PModified + RModified}. \quad (8)$$

The author also used the human evaluation technique described in Section VIII (after the experiments on model fine-tuning).

VI. Model-Centric Approach

The author sequentially applied different approaches to build and evaluate question answering systems. Sections VI and VII describe how the author has fine-tuned and optimized state-of-the-art pre-trained neural models. Section VI describes a model-centric approach, and Section VII describes a data-centric one. During the model-centric approach implementation, the author fine-tuned several state-of-the-art Transformer-based models and optimized their hyperparameters to achieve the best results. During the data-centric approach implementation, the optimized models were fine-tuned with different modifications of the training dataset (see Section III). Then the most efficient models were chosen to apply a human evaluation technique (see Section VIII).

A. Models

The author chose four base models for the model-centric experiments. All the models are based on the Transformer architecture [13]. All of them are pre-trained models that can be fine-tuned on a custom dataset using transfer learning capabilities. Transfer learning allows one to train a model on some task or language to transfer knowledge gained during the model training into another task or language in the fine-tuning process [19].

The Bidirectional Encoder Representations from Transformers (BERT) [20] is the first language model of the chosen ones. This model can be used for solving machine reading comprehension or extractive question answering tasks. The BERT base model consists of 12 Transformer encoders with 12 bidirectional self-attention heads. The model was pre-trained on the BooksCorpus and English Wikipedia.

BERT was pre-trained on two base tasks. The first one was masked language modelling or fill-in-the-gap task to

predict masked tokens by their surroundings. The second task, next sentence prediction, was to predict if one sentence is the next to another in some context. The knowledge of BERT is contextual word embeddings.

The distilled version of BERT (DistilBERT) [21] was the second model of the chosen ones. This model was obtained from BERT knowledge distillation. The size of the original BERT model was reduced by 40%, which make DistilBERT computationally cheaper and faster to fine-tune with around the same performance efficiency.

The third model is XLM-RoBERTa [22] based on masked language modelling. XLM-RoBERTa is a model for one hundred languages trained on CommonCrawl data. According to the developers, this model improved the performance on low-resource languages of other cross- and multilingual models developed earlier. This model is considered to be competitive with strong monolingual models.

The last model was Generative Pre-Trained Transformer 2 (GPT-2) [23]. GPT-2 is a traditional language model trained to predict the next token in a given sequence. Zero-shot learning capabilities, meaning that the model can solve some tasks without explicit training on them, allow GPT-2 to implement generative question answering. When being fine-tuned on a dataset that consists of question-answer pair sequences, the model can memorize context and answer questions just by addressing its memory. However, due to its generative properties, the outputs of this model might be repetitive or implausible.

Table 1 shows the configurations of each model used in the experiments. As the table shows, GPT-2 is significantly larger than BERT-based models due to the volume of its vocabulary, the number of heads and parameters. Nevertheless, GPT-2 is the only model from this selection that uses the generative approach, which is considered to be less reliable than extractive or retrieval ones due to its repetitiveness and uncontrollability.

Table 1. Model configuration

Parameter	BERT	DistilBERT	XLM-RoBERTa	GPT-2
Dropout ratio	0.1			
Activation function	GELU			
Vocabulary size	30522			50257
Heads	12			16
Layers	12	6	12	24
Embeddings	512			1024
Parameters	110M	66M	125M	355M

B. Fine-Tuning

Table 2 shows the parameters that were optimized during the model fine-tuning. For example, the batch size is the number of samples in a model training epoch, and the learning rate is the iteration step size.

The generated sequence length was set only for the generative model output. Because BERT-base models could only extract spans from reading passages, there was no need in setting the sequence length. An extractive model could output the whole reading passage (which maximum length was 512 symbols) as an answer, but the probability of such a result was low.

The temperature and top k are the GPT-2 parameters. The temperature controls the output randomness. The lower it is, the less random model completions are. A temperature value close to zero might lead to repetitive model outputs. The top k controls the output diversity. The value of this parameter reflects the number of words considered for each step.

The input type is attributable to the question answering mode implemented by a model. Because BERT-based models are robust at solving machine reading comprehension, and this task was a base one for the extractive question answering in this study, the input for these models required a question and a reading passage. However, because GPT-2 is a generative model, as input, it required a prefix with a question only.

Table 2. Training and generation procedures configuration

Parameter	BERT	DistilBERT	XLM-RoBERTa	GPT-2
Batch size	1			
Generated sequence length	None (512)			100
Learning rate	3e-5	1e-5	3e-5	1e-4
Epochs	10	20	10	30
Save checkpoint steps	12	6	12	500
Temperature	None			0.7
Top K	None			40
Number of generated samples	1			
Input type	A question and a reading passage from the dataset			A prefix with a question from the dataset

C. Results

Table 3 presents Precision, Recall and F1-Score points achieved by each model fine-tuned with the parameters from Tables 1 and 2. The author chose for the experiments two BERT and one DilstilBERT modifications found in the HuggingFace repository. Those modifications were already fine-tuned on multilingual (mBERT and mDistilBERT) and Russian (ruBERT) data.

GPT-2 became the most efficient model according to the automated evaluation results due to the high Precision score. XLM-RoBERTa became the most efficient among the BERT-based models. Its Precision and Recall do not differ as much as in other models.

Table 3. Results of the model-centric approach experiments

Base Model	Precision	Recall	F1-Score
mBERT	0.42	0.25	0.31
ruBERT	0.45	0.28	0.35
mDistilBERT	0.51	0.24	0.33
XLM-RoBERTa	0.39	0.36	**0.37**
GPT-2	0.78	0.41	**0.54**

VII. DATA-CENTRIC APPROACH

The author has decided to apply a data-centric approach to improve the performance of the extractive model. XLM-RoBERTa was chosen as a base model for the experiments

according to its Precision, Recall and F1-Score (see Table 3). The author used the ASD QA dataset modifications described in Section III to fine-tune the optimised XLM-RoBERTa model by changing the training data structure and design.

A. Results

Table 4 shows the results of the data-centric fine-tuning. The dataset modification without impossible (irrelevant) question-answer pairs allowed the author to get the highest metric scores on the extractive approach. The dataset version that contained only 50% of the training data led to a high Precision score and extremely low Recall. This result shows the influence of the dataset volume on model performance.

Table 4. Results of the data-centric approach experiments based on XLM-RoBERTa

Dataset Version	Precision	Recall	F1-Score
Short	0.37	0.29	0.33
Multiple	0.39	0.36	0.38
No Impossible	0.44	0.40	**0.42**
Half-Sized	0.72	0.04	0.07

VIII. Human Evaluation

The author used the human evaluation stage to check whether the generative approach is truly more robust than the extractive one. The author has shuffled the GPT-2 and XLM-RoBERTa outputs manually, chosen ten good-looking samples from each set of outputs, and proposed 122 informants to evaluate answers to question and explain their choice. The respondents were not informed about the purpose of the procedure and the fact that given question-answer pairs are the inputs and outputs of two automated systems.

Figure 4 shows the examples of outputs generated by GPT-2 and XLM-RoBERTa in Russian. The upper part of Figure 4 shows the output of the extractive model. It is informative but overdriven by its volume and formal style. The translation is the following: "*Question: What coach should teach an autistic child how to swim? Answer: Apart from providing the safety of an autistic child, swimming might become for him or her a pleasant and even therapeutic experience. If you are the parent of an autistic child but not a professional swimmer or lifeguard, please do not try to teach your child swimming on your own.*"

The lower part of Figure 4 shows the output of the generative model. It is short and easy to understand, although it does not answer the question, it is generic and even contains inappropriate punctuation. The translation is the following: "*Question: Why is bowling the right sport for people with ASD [Autism Spectrum Disorder]? Answer: You and your child should decide whether this type of participation is a plus or minus;*".

Table 5. Results of the human evaluation experiment

Base Model	Score
GPT-2 (generative approach)	46.00
XLM-RoBERTa + No Impossible ASD QA version (extractive approach)	**71.65**

Table 5 shows the results of the human evaluation procedure. The mean of all the scores given on a five-point scale to answers was converted into a percentage. The human evaluation results show that from the point of view of the end-user, the extractive approach deals with the question answering better.

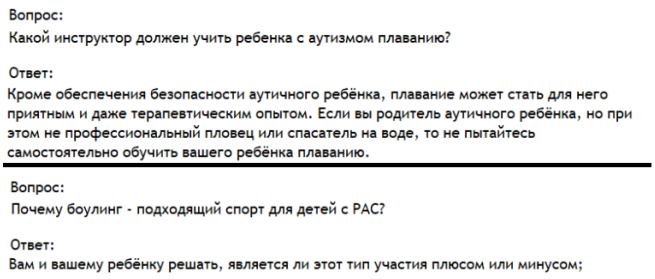

Figure 4. The model output samples

Table 6 contains the summary of the commentaries made by informants on extractive and generative model answers. The summary shows that for the end-user the relevance of the topic is one of the most important criteria. The consistency and grammatical correctness are also valuable. Table 6 shows the advantages and disadvantages of both systems, although the generative approach, according to the commentaries, is not only less robust but might even be dangerous as a part of the informational system about people with special needs.

Table 6. Commentaries Summary

Approach	Commentaries Summary
Extractive	Most of the answers reflect the question topic, are grammatically correct.
	Some answers are too long, hard to read, their style is too official.
	The answer is logical but I am not sure if this information could be applied to real life.
	The answer gives too common information.
	The answer text is inconsistent; it requires rephrasing or sentence order changing.
	The answer is like from another context.
Generative	Many answers cover absolutely another topic than those covered in questions.
	Vague, too common answers.
	Misleading answers.
	Incomplete answers.
	Some answers make no sense.

IX. Conclusion

The study describes several approaches to building and optimizing question answering systems for the sociomedical domain. The author conducted a series of model- and data-centric experiments to choose two efficient models representing two popular approaches to building question answering models. The extractive approach is similar to the machine reading comprehension task, which is to extract a span from a reading passage that would be the answer to a user question. The generative approach allows generating the answer based on the knowledge gained by a model during the training, for example, by zero-shot learning

The best model for the extractive approach implementation became XLM-RoBERTa fine-tuned on a modification of the author's custom dataset about autism

spectrum disorder built for this study. XLM-RoBERTa showed the highest F1-Score among the other extractive BERT-based models with similar Precision and Recall scores. This model pre-trained for one hundred languages showed the best performance for the Russian language dataset. XLM-RoBERTa and robust GPT-2 were chosen for the human evaluation of the extractive and generative approaches respectively.

The purpose of the human evaluation was to find out, which criteria are the most important to building sociomedical dialogue systems. The human evaluation aimed to find out if the generative approach is truly more efficient than the extractive one, because according to the automated evaluation, GPT-2 was more efficient, which contradicts a widespread opinion that extractive models are more reliable in question answering. After all, they do not use zero-shot learning but have special architecture.

122 informants were asked to evaluate the model answers on a five-point scale without being informed of the fact that the answers were generated or extracted by an automated system. Most of the informants rated the answers of the extractive system higher than the answers of the generative model.

According to the commentaries left by the informants, the answers of a sociomedical dialogue system should strictly reflect the question topic, be logical and grammatically correct. The extractive system deals well with these challenges. However, this system outputs might be too formal and hard to read. Although the answers of the generative system are easier for perception, the abilities of zero-shot learning are not enough to build a safe sociomedical system. A generic answer might cause negative emotions, create misconceptions and false information. This might be dangerous if a dialogue system would be later incorporated into the educational processes.

REFERENCES

[1] Allam A.M.N., Haggag M. H. The question answering systems: A survey // International Journal of Research and Reviews in Information Sciences (IJRRIS). Vol. 2 No. 3. 2012. P. 211–221.

[2] Gao J., Galley M., Li L., Neural approaches to conversational AI: Question answering, task-oriented dialogues and social chatbots. Now Foundations and Trends, 2019.

[3] Cheng H., Shen Y., Liu X., He P., Chen W., Gao J., UnitedQA: A hybrid approach for open domain question answering // Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online. 2021. P. 3080–3090.

[4] Alqifari R. Question answering systems approaches and challenges // Proceedings of the Student Research Workshop Associated with RANLP 2019, Varna, Bulgaria. 2019. P. 69–75.

[5] El-Hashem J. An ontology-driven sociomedical Web 3.0 framework. Ph.D. thesis, Concordia University. 2014.

[6] Firsanova V. Autism spectrum disorder and Asperger syndrome question answering dataset 1.0 // Figshare. 2021. DOI: 10.6084/m9.figshare.13295831.v10

[7] Green Jr. B.F., Wolf A.K., Chomsky C., Laughery K. Baseball: An automatic question-answerer // Western Joint IRE-AIEE-ACM Computer Conference. 1961. P. 219–224.

[8] Woods W.A., Kaplan R.B.F. Lunar rocks in natural English: Explorations in natural language question answering // Linguistic Structures Processing 5. 1977. P. 521–569.

[9] Auer S., Bizer C., Kobilarov G., Lehmann J., Cyganiak R., Ives Z. DBpedia: A nucleus for a web of open data // The semantic web. Springer. 2007. P. 722–735.

[10] Bartolo M., Roberts A., Welbl J., Riedel S., Stenetorp P. Beat the AI: Investigating adversarial human annotation for reading comprehension // Transactions of the Association for Computational Linguistics. Vol. 8. 2020. P. 662–678.

[11] Gao L., Dai Z., Callan J. Modularized Transfomer-based ranking framework // Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 2020. P. 4180–4190.

[12] Segal E., Efrat A., Shoham M., Globerson A., Berant J. A simple and effective model for answering multi-span questions // Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, 2020. P. 3074–3080.

[13] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser L., Polosukhin I. Attention is all you need // Advances in Neural Information Processing Systems. 2017. No 30. P. 5998–6008.

[14] Chen D., Fisch A., Weston J., Bordes A. Reading Wikipedia to answer open-domain questions // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Vancouver, Canada, 2017. P. 1870–1879.

[15] Rajpurkar P., Zhang J., Lopyrev K., Liang P. SQuAD: 100,000+ questions for machine comprehension of text // Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Austin, Texas, 2016. P. 2383–2392.

[16] Rajpurkar P., Jia R., Liang P. Know what you don't know: Unanswerable questions for SQuAD // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. Volume 2: Short Papers. Melbourne, Australia, 2018. P. 784–789.

[17] Efimov P., Chertok A., Boytsov L., Braslavski P. SberQuAD – russian reading comprehension dataset: Description and analysis // Experimental IR Meets Multilinguality, Multimodality, and Interaction. Springer, 2020, P. 3–15.

[18] Li C.-H., Wu S.-L., Liu C.-L., Lee H.-y. Spoken SQuAD: A study of mitigating the impact of speech recognition errors on listening comprehension // Interspeech, 2018, P. 3459–3463.

[19] Ruder S., M. Peters E., Swayamdipta S., Wolf T. Transfer Learning in Natural Language Processing // Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials, 2019.

[20] Devlin J., Chang M., Lee K., Toutanova K., Bert: Pre-training of deep bidirectional transformers for language understanding // Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Volume 1 (Long and Short Papers). 2019. P. 4171–4186.

[21] Sanh V., Debut L., Chaumond J., Wolf T. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter // 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing - NeurIPS, 2019.

[22] Conneau A., Khandelwal K., Goyal N., Chaudhary V., Wenzek G., Guzmán F., Grave E., Ott M., Zettlemoyer L., Stoyanov V. Unsupervised Cross-lingual Representation Learning at Scale // ACL. 2020.

[23] Radford A., Wu J., Child R., Luan D., Amodei D., Sutskever I. Language Models are Unsupervised Multitask Learners // OpenAI. 2019.