SberQuAD – Russian Reading Comprehension Dataset: Description and Analysis

Pavel Efimov[1⋆], Andrey Chertok[2], Leonid Boytsov[3], and Pavel Braslavski[4,5][0000−0002−6964−458X]

[1] St. Petersburg State University, St. Petersburg, Russia pavel.vl.efimov@gmail.com
[2] Sberbank, Moscow, Russia achertok@sberbank.ru
[3] leo@boytsov.info
[4] Ural Federal University, Yekaterinburg, Russia pbras@yandex.ru
[5] JetBrains Research, St. Petersburg, Russia

Abstract. The paper presents SberQuAD – a large Russian reading comprehension (RC) dataset created similarly to English SQuAD. SberQuAD contains about 50K question-paragraph-answer triples and is seven times larger compared to the next competitor. We provide its description, thorough analysis, and baseline experimental results. We scrutinized various aspects of the dataset that can have impact on the task performance: question/paragraph similarity, misspellings in questions, answer structure, and question types. We applied five popular RC models to SberQuAD and analyzed their performance. We believe our work makes an important contribution to research in multilingual question answering.

Keywords: reading comprehension · evaluation · Russian language resources · multilingual question answering

1 Introduction

Automatic Question Answering (QA) is a long-standing important problem, which can be broadly described as building a system that can answer questions in a natural language. The modern history of QA starts from TREC challenges organized by NIST in 2000s [7] and extended by CLEF to a multilingual setting [10]. Reading comprehension (RC) is a subtask of QA, where the system needs to answer questions for a given document. This task has recently become quite popular with the introduction of an English large-scale Stanford Question Answering Dataset (SQuAD) [17].

In this paper, we present a large Russian RC dataset, which was created for a data science competition organized by Sberbank (hence SberQuAD) and is freely available for public.[6] The paper focuses on a post hoc analysis of the dataset properties and reports several baselines results. Given the importance of the RC task and scarcity of non-English resources, we believe it is an important contribution to research and evaluation in multilingual QA.

⋆ Work done as an intern at JetBrains Research.
[6] https://github.com/sberbank-ai/data-science-journey-2017

Table 1. Aggregate statistics of SQuAD and existing Russian RC datasets. LCMS stands for the longest contiguous matching subsequence.

	SberQuAD	*SQuAD 1.1* *train/dev*	*XQuAD (ru)*	*TyDi QA (ru)* *train/dev*
# questions	50,364	87,599 / 10,570	1,190	6,490 / 812
# unique paragraphs	9,080	18,896 / 2,067	240	6,490 / 812
Number of tokens				
avg. paragraph length	101.7	116.6 / 122.8	112.9	79.5 / 73.1
avg. question length	8.7	10.1 / 10.2	8.6	6.4 / 6.5
avg. answer length	3.7	3.16 / 2.9	2.9	3.9 / 3.9
avg. answer position	40.5	50.9 / 52.9	48.4	25.9 / 25.6
Number of characters				
avg. paragraph length	753.9	735.8 / 774.3	850.3	585.4 / 539.3
avg. question length	64.4	59.6 / 60.0	64.9	44.8 / 47.1
avg. answer length	25.9	20.2 / 18.7	21.4	25.7 / 26.5
avg. answer position	305.2	319.9 / 330.5	364.5	190.7 / 188.9
question-paragraph LCMS	32.7	19.5 / 19.8	20.1	12.4 / 14.9

2 Related Work

SQuAD [17] contains more than 100K questions posed to paragraphs from popular Wikipedia articles. Questions were generated by crowd workers. An answer to each question should be a valid and relevant paragraph span. Wide adoption of SQuAD led to emergence of many RC datasets. TriviaQA [12] consists of 96K trivia game questions and answers found online accompanied by answer-bearing documents. Natural Questions dataset [14] is approximately three times larger than SQuAD. In that, unlike SQuAD, questions are sampled from Google search log rather than generated by crowd workers. MS MARCO [2] contains 1M questions from a Bing search log along with free-form answers. For both MS MARCO and Natural Questions answers are produced by in-house annotators. QuAC [4] and CoQA [18] contain questions and answers in information-seeking dialogues. For a more detailed discussion we address the reader to a recent survey [23].

There are several monolingual non-English RC datasets, e.g. for Chinese [11] and French [9]. Recently, Artetxe et al. experimented with cross-language transfer learning and prepared XQuAD dataset containing 240 paragraphs and 1,190 Q&A pairs from SQuAD v1.1 translated into 10 languages, including Russian [1]. MLQA [15] covers seven languages with over 12K English Q&A instances and 5K in each other languages. Yet, the Russian data is missing. TyDi QA [6] covers 11 typologically diverse languages with over 200K Q&A instances. However, there are only about 7K Russian items. Two latter papers [15,6] provide a good overview of non-English RC resources. Statistics of Russian RC datasets are summarized in Table 1.

P6418 The term “computer science” appears in a 1959 article in Communications of the ACM, in which Louis Fein argues for the creation of a Graduate School in Computer Science . . . Louis Fein’s efforts, and those of others such as numerical analyst George Forsythe, were rewarded: universities went on to create such departments, starting with Purdue in 1962.
Q11870 When did the term “computer science” appear?
Q28900 Who was the first to use this term?
Q30330 Starting with wich university were computer science programs created?

Fig. 1. A translated sample SberQuAD entry: answers are underlined and colored. The word **which** in *Q30330* is misspelled on purpose to reflect the fact that the original has a misspelling.

3 Dataset

SberQuAD contains 50,364 paragraph–question–answer triples and was created in a similar fashion to SQuAD. First, Wikipedia pages were selected, split into paragraphs, and paragraphs presented to crowd workers. For each paragraph, a Russian native speaking crowd worker had to come up with questions that can be answered using solely the content of the paragraph. In that, an answer must have been a paragraph span, i.e., a contiguous sequence of paragraph words. The tasks were posted on Toloka crowdsourcing platform.[7] SberQuAD has always only one correct answer span, whereas SQuAD can have multiple answer variants (1.7 *different* answers for each question on the development set).

Examples and basic statistics. Figure 1 shows a translated sample SberQuAD paragraph with three questions: Gold-truth answers are underlined in text. Generally, the format of the question and the answers mimics that of SQuAD. Note, however, the following peculiarities: Question *Q30330* contains a spelling error; Question *Q28900* references prior question *Q11870* and cannot, thus, be answered on its own (likely both questions were created by the same crowd worker).

Basic dataset statistics is summarized in Table 1: SberQuAD has about twice as fewer questions compared to SQuAD. However, the number of Russian questions in SberQuAD is substantially higher compared to XQuAD and TyDi QA. The average lengths of paragraphs, questions, and answers are similar across three datasets – SberQuAD, SQuAD, and XQuAD. TyDi QA stands out due to a different approach to data collection: Annotators generated questions in response to a non-restrictive prompt, then a top-ranked Wikipedia article for each question is retrieved. Finally, annotators were presented with articles split into paragraphs and had to choose a relevant paragraph and an answer within. This annotation scheme led to shorter questions and paragraphs, and more importantly – to a lower question/paragraph overlap. In SberQuAD, there are 275 questions (0.55%) having at least 200 characters and 374 answers (0.74%) that are longer than 100 characters. Anecdotally, very long answers and very short questions are

[7] `https://toloka.yandex.com`

frequently errors. For example, for question *Q61603* the answer field contains a copy of the whole paragraph, while question *Q76754* consists of a single word 'thermodynamics'.

For experiments described in this paper, we used the SberQuAD split into a training and testing sets (45,328 and 5,036 items, respectively) made by `DeepPavlov` team.[8]

Analysis of questions. Most questions in the dataset start with either a question word or preposition: ten most common starting words are *что* (*what*), *в* (*in*), *как* (*how*), *кто* (*who*), *какие* ($what_{\text{adj}}$), *когда* (*when*), *какой* ($what_{\text{adj}}$), *где* (*where*), *сколько* (*how many*), *на* (*on*). These starting words correspond to 62.4% of all questions. In about 4% of the cases, an interrogative word is not among the first three words of the question, though. Manual inspection showed that in most cases these entries are declarative statements, sometimes followed by a question mark, e.g. *Q15968 'famous Belgian poets?'*, or ungrammatical questions.

While manually examining the dataset, we encountered quite a few misspelled questions. To estimate the proportion of questions with misspellings, we verified all questions using Yandex spellchecking API.[9] The automatic speller identified 2,646 and 287 misspelled questions in training and testing sets, respectively. We also found 385 and 51 questions in training and testing sets, respectively, containing Russian interrogative particle *ли* (*whether/if*). This form implies a yes/no question, which is generally not possible to answer in the RC setting by selecting a valid and relevant paragraph phrase. For this reason, most answers for these yes/no questions are fragments supporting or refuting the question statement. In addition, we found 15 answers in the training set, where the correct answer 'yes' (Russian *да*) can be found as a paragraph word substring, but not as a valid/relevant phrase. Thus, we estimate that in the testing set, 5.7% of the questions have misspellings and 1% of questions cannot be answered using a paragraph.

Analysis of answers. Following [17], we analyzed answers presented in the dataset by their type. To this end, we employed a NER tool from `DeepPavlov` library.[10] In our analysis, we focus on the following NEs: DATE, NUMBER, PERSON, LOCATION, and ORGANIZATION. In total, almost 43% of answers in testing set contain NEs, while about 14% are exact NEs. Obtained information is used to evaluate models' performance on different answer types (see Tables 3 and 4). We complemented our analysis of answers with syntactic parsing. To this end we applied the rule-based constituency parser AOT[11] to answers without detected NEs. AOT parser supports a long list of phrase types (57 in total), we

[8] `http://docs.deeppavlov.ai/en/master/features/models/squad.html`

[9] `https://yandex.ru/dev/speller/` (in Russian)

[10] The multilingual BERT model is trained on English OntoNotes corpus and transferred to Russian, see `http://docs.deeppavlov.ai/en/master/features/models/ner.html`

[11] `http://aot.ru`

grouped them into conventional high-level types, which are shown in Table 5.[12] Not surprisingly, noun phrases are most frequent answer types (24%), followed by prepositional phrases (10.5%). Verb phrases represent a non-negligible share of answers (7.1%), which is quite different from a traditional QA setting where answers are predominantly noun phrases [16].

Question/paragraph similarity. We further estimate similarity between questions and paragraph sentences containing the answer: The more similar is the question to its answer's context, the simpler is the task of locating the answer. In contrast to SQuAD analysis [17] we refrain from syntactic parsing and rely on simpler approaches. First, we compared questions with complete paragraphs. To this end, we calculated the length of the longest contiguous matching subsequence (LCMS) between a question and a paragraph using the `difflib` library.[13] The last row in Table 1 shows that despite similar paragraph and question lengths in both SQuAD and SberQuAD, the SberQuAD questions are more similar to the paragraph text. Second, we estimated similarity between a question and the sentence containing the answer. First, we applied `DeepPavlov` tokenizer[14] to split the dataset into sentences. Subsequently, we lemmatized the data using `mystem`[15] and calculated the Jaccard coefficient between a question and the sentence containing the answer. The mean value of the Jaccard coefficient is 0.28 (median is 0.23). Our analysis shows that there is a substantial lexical overlap between questions and paragraph sentences containing the answer, which may indicate a heavier use of the copy-and-paste approach by crowd workers recruited for SberQuAD creation.[16]

4 Employed Models

We applied the following models to SberQuAD: 1) two baselines provided by the competition organizers; 2) four pre-BERT models that showed good performance on SQuAD and were used in a study similar to ours [21] – BiDAF, DocQA, DrQA, and R-Net; and 3) BERT model provided by the `DeepPavlov` library.

Preprocessing and training. We tokenized text using `spaCy`.[17] To initialize the embedding layer for BiDAF, DocQA, DrQA, and R-Net we use Russian case-sensitive `fastText` embeddings trained on Common Crawl and Wikipedia.[18]

[12] Table 5 provides data for the testing set, but the distribution for the training set is quite similar.

[13] `https://docs.python.org/3/library/difflib.html`

[14] `https://github.com/deepmipt/ru_sentence_tokenizer`

[15] `https://yandex.ru/dev/mystem/` (in Russian)

[16] Note that in the interface for crowdsourcing SQuAD questions, prompts at each screen reminded the workers to formulate questions in their own words; in addition, the copy-paste functionality for the paragraph was purposefully disabled.

[17] `https://github.com/buriy/spacy-ru`

[18] `https://fasttext.cc/docs/en/crawl-vectors.html`

This initialization is used for both questions and paragraphs. For BiDAF and DocQA about 10% of answer strings in both training and testing sets require a correction of positions, which can be nearly always achieved automatically by ignoring punctuation (12 answers required a manual intervention). Models were trained on GPU nVidia Tesla V100 16Gb with default implementation settings.

Baselines. As a part of the competition two baselines were made available.[19] *Simple baseline:* The model returns a sentence with the maximum word overlap with the question. *ML baseline* generates features for all word spans in the sentence returned by the simple baseline. The feature set includes TF-IDF scores, span length, distance to the beginning/end of the sentence, as well as POS tags. The model uses gradient boosting to predict F1 score. At the testing stage the model selects a candidate span with maximum predicted score.

Gated Self-Matching Networks (R-Net): This model, proposed by Wang et al. [22], is a multi-layer end-to-end neural network that uses a gated attention mechanism to give different levels of importance to different paragraph parts. It also uses self-matching attention for the context to aggregate evidence from the entire paragraph to refine the query-aware context representation. We use a model implementation by HKUST.[20] To increase efficiency, the implementation adopts scaled multiplicative attention instead of additive attention and uses variational dropout.

Bi-Directional Attention Flow (BiDAF): The model proposed by Seo et al. [20] takes inputs of different granularity (character, word and phrase) to obtain a query-aware context representation without previous summarization using memory-less context-to-query (C2Q) and query-to-context (Q2C) attention. We use original implementation by AI2.[21]

Multi-Paragraph Reading Comprehension (DocQA): This model, proposed by Clark and Gardner [5], aims to answer questions based on entire documents (multiple paragraphs). If considering the given paragraph as the document, it also shows good results on SQuAD. It uses the bi-directional attention mechanism from the BiDAF and a layer of residual self-attention. We also use original implementation by AI2.[22]

Document Reader (DrQA): This model proposed by Chen et al. [3] is part of the system for answering open-domain factoid questions using Wikipedia. The Document Reader component performs well on SQuAD (skipping the document retrieval stage). The model has paragraph and question encoding layers with RNNs and an output layer. The paragraph encoding passes as input to RNN a sequence of

[19] `https://github.com/sberbank-ai/data-science-journey-2017/tree/master/problem_B/`
[20] `https://github.com/HKUST-KnowComp/R-Net`
[21] `https://github.com/allenai/bi-att-flow`
[22] `https://github.com/allenai/document-qa`

feature vectors derived from tokens: word embedding, exact match with question word, POS/NER/TF and aligned question embedding. The implementation is developed by Facebook Research.[23]

Bidirectional Encoder Representations from Transformers (BERT): Pre-trained BERT models achieved superior performance is a variety of downstream NLP tasks, including RC [8]. The Russian QA model is obtained by a transfer from the multilingual BERT (mBERT) with subsequent fine-tuning on the Russian Wikipedia and SberQuAD [13].[24]

Evaluation. Similar to SQuAD, SberQuAD evaluation employs two metrics to assess model performance – 1) the percentage of system's answers that exactly match (EM) any of the gold standard answers and 2) the maximum overlap between the system response and ground truth answer at the token level expressed via F1 (averaged over all questions). Both metrics ignore punctuation and capitalization.

5 Analysis of Model Performance

Main experimental results are shown in Table 2. It can be seen that all the models perform worse on the Russian dataset than on SQuAD. In that, there is a bigger difference in exact matching scores compared to F1. For example, for BERT the F1 score drops from 91.8 to 84.8 whereas the exact match score drops from 85.1 to 66.6. The relative performance of the models is consistent for both datasets, although there is a greater variability among four neural "pre-BERT" models. One explanation for lower scores is that SberQuAD has always only one correct answer. Furthermore, SberQuAD contains many fewer answers that are named entities than SQuAD (13.8% vs. 52.4%), which—as we discuss below—maybe another reason for lower scores. Another plausible reason is a poorer quality of annotations: We have found a number of deficiencies including but not limited to misspellings in questions and answers.

Table 2. Model performance on SQuAD and SberQuAD; SQuAD part shows single-model scores on test set taken from respective papers.

Model	**SberQuAD**		**SQuAD**	
	EM	F1	EM	F1
simple baseline	0.3	25.0	–	–
ML baseline	3.7	31.5	–	–
BiDAF [20]	51.7	72.2	68.0	77.3
DrQA [3]	54.9	75.0	70.0	79.0
R-Net [22]	58.6	77.8	71.3	79.7
DocQA [5]	59.6	79.5	72.1	81.1
BERT [8]	66.6	84.8	85.1	91.8

Figure 2 shows the relationship between the F1 score and the question-answer similarity expressed as the Jaccard coefficient. Note that 64% of question–sentence

[23] `https://github.com/facebookresearch/DrQA`

[24] `http://docs.deeppavlov.ai/en/master/features/models/squad.html`

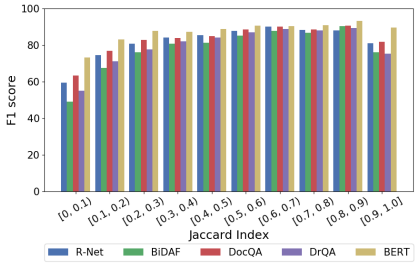

Fig. 2. Model performance depending on Jaccard similarity between a question and the sentence containing an answer.

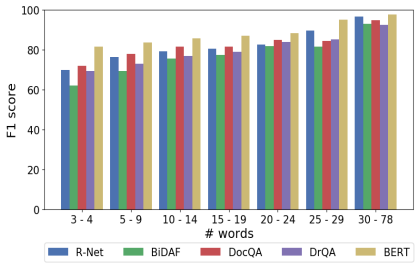

Fig. 3. Model performance depending on question length (# of words).

pairs fall into first three bins. As expected, a higher value of the Jaccard coefficient corresponds to higher F1 scores (with the exception of 14 questions where Jaccard is above 0.9).[25] Furthermore, in the case of the high similarity there is only a small difference among model performance. These observations support the hypothesis that it is easier to answer questions when there is a substantial lexical overlap between a question and a paragraph sentence containing the answer.

Longer questions are easier to answer too: the F1 score increases nearly monotonically with the question length, see Figure 3. Presumably, longer questions provide more context for identifying correct answers. In contrast, dependency on the answer length is not monotonic: the F1 score first increases and achieves the maximum for 2-4 words. A one-word ground truth constitutes a harder task: missing a single correct word results in a null F1 score, whereas returning a two-word answer containing the single correct word results in only $F1 = 0.67$. F1 score also decreases substantially for answers above average length. It can be explained by the fact that models are trained on the dataset where shorter answers prevail, see Table 1. Models' average-length answers get low scores in case of longer ground truth. For example, a 4-word answer fully overlapping with a 8-word ground truth answer gets again only $F1 = 0.67$.

Following our analysis of the dataset, we break down model scores by the answer types. Tables 3 and 4 summarize performance of the models depending on the answers containing named entities of different types. Table 3 represents answers that contain at least one NE, but which are not necessarily NEs themselves (42.7% in the test set). Table 4 represents answers that are NEs (13.8% in test). A common trend for all models is that F1 scores for answers mentioning dates, persons, locations, and organizations are higher than average. NUMBER is an exception in this regard, probably due to a high variability of contexts might

[25] Among these 14 questions the majority are long sentences from the paragraph with a single word (answer) substituted by a question word; there is an exact copy with just a question mark at the end; one question has the answer erroneously attached after the very question.

Table 3. Model performance (F1) on answers containing named entities.

NE	% test	R-Net	BiDAF	DocQA	DrQA	BERT
Date	12.2%	88.0	86.6	90.0	88.9	91.3
Number	9.6%	73.1	69.1	75.5	72.5	80.4
Person	8.8%	78.3	73.1	81.0	77.7	86.6
Location	7.6%	79.8	75.7	81.1	77.8	85.8
Organization	4.1%	79.0	77.3	82.3	78.3	88.2
Other NE	2.1%	72.7	59.4	73.6	64.7	80.9
Any NE	42.7%	80.3	76.4	82.6	79.7	87.0
Test set		77.8	72.2	79.5	75.0	84.8

Table 4. Model performance (F1) on answers matching NER tags.

NE	% test	R-Net	BiDAF	DocQA	DrQA	BERT
Date	2.2%	87.1	87.3	90.8	87.5	95.0
Number	3.3%	78.2	72.4	80.1	77.7	90.2
Person	4.2%	83.2	74.0	85.1	82.9	91.4
Location	1.7%	78.3	72.8	82.1	77.9	88.6
Organization	1.5%	80.7	76.5	81.6	79.2	91.8
Other NE	0.9%	80.9	54.9	78.1	66.4	88.9
Any NE	13.8%	81.6	74.5	83.6	80.2	91.2
Test set		77.8	72.2	79.5	75.0	84.8

contain numerals both as digits and words. Answers containing *other NEs* also show degraded performance – probably, again due to their higher diversity and lower counts. The scores are significantly higher when an answer is exactly a NE. This is in line with previous studies that showed that answers containing NEs are easier to answer, see for example [19].

For about 48% of the answers in the testing set that do not contain NEs we were able to derive their syntactic phrase type, see Table 5. Among them, non-factoid verb phrases stand out as most difficult ones— all models perform worse on such questions.[26] In contrast, answers expressed as prepositional phrases are easier to answer compared to both noun and verb phrases. Noun phrases—most common syntactic units among answers—are second-easiest structure among others to answer. However, F1 scores for noun phrases are lower than average.

The models behave remarkably differently on questions with and without detected misspellings, see Table 6. DrQA seems to be most sensible to misspellings: The difference in F1 is almost 8% (scores are lower for misspelled questions). DocQA has most stable behavior: The difference in F1 scores is about 2%.

Questions with interrogative *ли*-particle represent around 1% in the whole dataset. Although score averages for such small sets are not very reliable, the decrease in performance on these questions is quite sharp and consistent for all models: It ranges from 8.5% in F1 points for DocQA to 18.7% for BiDAF, see

[26] Adverbial phrases appears to be even harder, but they are too few to make reliable conclusions.

Table 5. Model performance (F1) on answers not containing NEs by constituent type (NP – noun phrase, PP – prepositional phrase, VP – verb phrase, ADJP – adjective phrase, ADVP – adverb phrase, non-R – words in non-Russian characters; None – not recognized).

Type	% test	R-Net	BiDAF	DocQA	DrQA	BERT
NP	24.0	77.5	70.3	78.2	73.5	84.5
PP	10.5	83.1	78.6	84.9	81.4	89.1
VP	7.1	61.9	54.0	62.7	55.5	71.6
ADJP	5.9	73.0	65.3	75.5	67.2	80.5
ADVP	0.3	67.9	45.3	70.7	51.2	76.6
non-R	0.3	91.7	88.2	98.2	92.9	95.1
None	9.1	75.7	69.0	77.1	70.1	83.0
Test set		77.8	72.2	79.5	75.0	84.8

Table 6. Model performance (F1) on misspelled (upper part) and yes/no (lower part) questions.

	% test	R-Net	BiDAF	DocQA	DrQA	BERT
w/ typos	5.7	74.1	66.7	77.5	67.5	81.1
correct	94.3	77.1	72.5	79.6	75.4	85.0
Test set		77.8	72.2	79.5	75.0	84.8
w/ *ли*	1.0	66.6	53.7	71.0	57.5	73.3
other	99.0	77.9	72.4	79.6	75.2	84.9
Test set		77.8	72.2	79.5	75.0	84.8

Table 6. We hypothesize that these questions are substantially different from other questions and are poorly represented in the training set.

Finally, we sampled 100 questions where all models achieved zero F1 score (i.e., they returned a span with no overlap with a ground truth answer). We manually grouped the sampled questions into the following categories (number of questions in each category in parentheses; questions can be assigned to more than one category):

- An entire paragraph or its significant part can be seen as an answer to a *broad/general question* (12).
- An answer is *incomplete* (29), because it contains only a part of an acceptable longer answer. For example for *Q31929 'Who did notice an enemy airplane?'* only the word *pilots* is marked as ground truth in the context: *On July 15, during a reconnaissance east to Zolotaya Lipa, pilots of the 2nd Siberian Corps Air Squadron Lieutenant Pokrovsky and Cornet Plonsky noticed an enemy airplane.*
- *Vague questions* (19) are related to the corresponding paragraph but seem to be a result of a misinterpretation of the context by a crowd worker. For example, in *Q70465 'What are the disadvantages of TNT comparing to dynamite and other explosives?'* the ground truth answer *'a detonator needs to be used'* is not mentioned as a disadvantage in the paragraph. A couple of

these questions use paronyms of concepts mentioned in the paragraph. For example, *Q46229* asks about *'discrete policy'*, while the paragraph mentions *'discretionary policy'*.
- *No answer in the paragraph* (3) and *incorrect answer* (14) constitute more straightforward error cases.
- Some questions require *reasoning* (10) and *co-reference resolution* (12).
- A small fraction of questions uses *synonyms and paraphrases* (3) that are not directly borrowed from the paragraph.
- A relatively large fraction of 'difficult' questions contains *misspellings* (6) and imply *yes/no* (3) answers.

One can see from the list that most potential causes of degraded performance can be attributed to poor data quality: Only 25% of cases can be explained by a need to deal with linguistic phenomena such as co-reference resolution, reasoning, and paraphrase detection.

6 Conclusions

We presented a large Russian reading comprehension dataset SberQuAD, which is nearly seven times larger compared to the next competitor. The SberQuAD was created similarly to SQuAD, but as our analysis shows, SberQuAD has a higher lexical overlap between questions and sentences with answers; not all questions are well-formed. At the same time, SberQuAD has a lower proportion of named entities as answers and a non-negligible share of answers that are verb phrases.

We applied five RC models to SberQuAD. Expectantly, a BERT-based model outperforms its predecessors. All models perform better on questions with higher overlap with paragraph text, on longer questions, on average-length answers, as well as when an answer contains a named entity. Despite the similarities between SQuAD and SberQuAD, all the models perform worse on Russian dataset than on its English counterpart, which can be attributed to smaller training set, having only a single answer variant in SberQuAD (as opposed to SQuAD, which has at least two variants) and fewer answers that are named entities. Furthermore, SberQuAD annotations might have been of poorer quality, but it is hard to quantify. These observations can be used to guide a creation of more difficult RC data sets. We believe that our work constitutes an important contribution to research in multilingual QA and will lead to a wider adoption of SberQuAD by the community.

Acknowledgments. We thank Peter Romov, Vladimir Suvorov, and Ekaterina Artemova (Chernyak) for providing us with details about SberQuAD preparation. We also thank Natasha Murashkina for initial data processing. PB acknowledges support by Ural Mathematical Center under agreement No. 075-02-2020-1537/1 with the Ministry of Science and Higher Education of the Russian Federation.

References

1. Artetxe, M., Ruder, S., Yogatama, D.: On the cross-lingual transferability of monolingual representations. arXiv preprint arXiv:1910.11856 (2019)
2. Bajaj, P., et al.: MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. arXiv preprint arXiv:1611.09268 (2016)
3. Chen, D., Fisch, A., Weston, J., Bordes, A.: Reading wikipedia to answer open-domain questions. arXiv preprint arXiv:1704.00051 (2017)
4. Choi, E., He, H., Iyyer, M., Yatskar, M., Yih, W.t., Choi, Y., Liang, P., Zettlemoyer, L.: QuAC: Question Answering in Context. In: EMNLP. pp. 2174–2184 (2018)
5. Clark, C., Gardner, M.: Simple and effective multi-paragraph reading comprehension. arXiv preprint arXiv:1710.10723 (2017)
6. Clark, J.H., Choi, E., Collins, M., Garrette, D., Kwiatkowski, T., Nikolaev, V., Palomaki, J.: TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. arXiv preprint arXiv:2003.05002 (2020)
7. Dang, H.T., Kelly, D., Lin, J.J.: Overview of the TREC 2007 Question Answering Track. In: Proceedings of the 16th TREC (2007)
8. Devlin, J., et al.: BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
9. d'Hoffschmidt, M., Vidal, M., Belblidia, W., Brendlé, T.: FQuAD: French question answering dataset. arXiv preprint arXiv:2002.06071 (2020)
10. Giampiccolo, D., et al.: Overview of the CLEF 2007 Multilingual Question Answering Track. In: Advances in Multilingual and Multimodal Information Retrieval. pp. 200–236 (2008)
11. He, W., et al.: DuReader: a Chinese machine reading comprehension dataset from real-world applications. arXiv preprint arXiv:1711.05073 (2017)
12. Joshi, M., et al.: TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In: ACL. pp. 1601–1611 (2017)
13. Kuratov, Y., Arkhipov, M.: Adaptation of deep bidirectional multilingual transformers for Russian language. arXiv preprint arXiv:1905.07213 (2019)
14. Kwiatkowski, T., et al.: Natural Questions: a benchmark for question answering research. TACL **7**, 453–466 (2019)
15. Lewis, P., Oğuz, B., Rinott, R., Riedel, S., Schwenk, H.: MLQA: Evaluating cross-lingual extractive question answering. arXiv preprint arXiv:1910.07475 (2019)
16. Prager, J.M.: Open-domain question-answering. Foundations and Trends in Information Retrieval **1**(2), 91–231 (2006)
17. Rajpurkar, P., Zhang, J., Lopyrev, K., Liang, P.: SQuAD: 100,000+ questions for machine comprehension of text. arXiv preprint arXiv:1606.05250 (2016)
18. Reddy, S., Chen, D., Manning, C.D.: CoQA: A Conversational Question Answering Challenge. TACL **7**, 249–266 (2019)
19. Rondeau, M.A., Hazen, T.J.: Systematic Error Analysis of the Stanford Question Answering Dataset. In: MRQA Workshop (2018)
20. Seo, M., Kembhavi, A., Farhadi, A., Hajishirzi, H.: Bidirectional attention flow for machine comprehension. arXiv preprint arXiv:1611.01603 (2016)
21. Wadhwa, S., Chandu, K.R., Nyberg, E.: Comparative analysis of neural QA models on SQuAD. arXiv preprint arXiv:1806.06972 (2018)
22. Wang, W., Yang, N., Wei, F., Chang, B., Zhou, M.: Gated self-matching networks for reading comprehension and question answering. In: ACL. pp. 189–198 (2017)
23. Zhang, X., Yang, A., Li, S., Wang, Y.: Machine reading comprehension: a literature review. arXiv preprint arXiv:1907.01686 (2019)