Towards a Polish Question Answering Dataset (PoQuAD)

Ryszard Tuora(✉), Natalia Zawadzka-Paluektau, Cezary Klamra, Aleksandra Zwierzchowska, and Łukasz Kobyliński

Institute of Computer Science, Polish Academy of Sciences, Jana Kazimierza 5, 01–248 Warszawa, Poland
{r.tuora,natalia.zawadzka-paluektau,c.klamra,a.zwierzchowska, lkobylinski}@ipipan.waw.pl

Abstract. This paper presents the efforts towards creating **PoQuAD**, a dataset for training automatic question answering models in Polish. It justifies why having native data is vital for training accurate Question Answering systems. PoQuAD broadly follows the methodology of SQuAD 2.0 (including impossible questions), but detracts from it in a few aspects. The first of these concerns reducing annotation density in order to broaden the range of topics included. The second is the inclusion of a generative answer layer to better suit the needs of a morphologically rich language. PoQuAD is a work in progress and so far consists of over 29000 question-answer pairs with contexts extracted from Polish Wikipedia. The planned size of the dataset is over 50 thousand such entries. The paper describes the annotation process and the guidelines which were given to annotators in order to ensure quality of the data. The collected data is subjected to analysis in order to shed some light on its linguistic properties and on the difficulty of the task.

Keywords: Natural language processing · Question answering · Machine reading comprehension

1 Introduction

Automatic question answering (**QA**) is a burgeoning field within natural language processing (**NLP**). A robust QA system can be used to gather information from a digital library in a much more natural way than standard information-retrieval (**IR**) methods, i.e. by asking questions, as opposed to forming search queries. As in other domains of NLP, the best contemporary QA methods rely on utilizing general-purpose language models, which are then fine-tuned on question answering datasets. The models are usually embedded in a retrieve-and-read pipeline in which a document is first recovered using conventional IR techniques, and then the fine-tuned reading-comprehension model extracts an answer from the document. The latter task is data-heavy, and for this reason, high-quality data is crucial in achieving good performance of the entire system.

Y.-H. Tseng et al. (Eds.): ICADL 2022, LNCS 13636, pp. 194–203, 2022.
https://doi.org/10.1007/978-3-031-21756-2_16

This paper presents the ongoing efforts in creating **PoQuAD** — Polish Question Answering Dataset, a resource designed for training machine-learning QA models in Polish. First, the related datasets and experiments are discussed, then the annotation procedure is explained in detail, and lastly some analysis of the collected data is presented.

2 Previous Work

The paradigm dataset for the retrieve-and-read approach is SQuAD [18], which contains over 100 thousand question-answer pairs annotated by crowdsources to be answered based on articles collected from the English Wikipedia. This dataset has subsequently been extended to form SQuAD 2.0 [17] by adding so-called *impossible questions*, i.e. questions which are relevant to the text, but nevertheless cannot be answered based on the information within it.

When working with languages other than English, a range of approaches can be proposed. We distinguish:

1. Zero-shot transfer based on multilingual models
2. Training monolingual models on translated datasets
3. Training monolingual models on native datasets

It is generally the case that 3 is superior to 1 and 2. For French the best **F1** results for each paradigm are 86.1%, 87.5%, and 91.8%, respectively [8]. In the case of German, when models of similar size are compared, the **F1** scores for each category are 68.6%, 78.8%, and 88.1%, respectively [14]. These results suggest that providing native data is important in achieving high accuracy even when the native datasets are substantially smaller, as is the case for both of these studies (60k for French, and 14k for German). The general SQuAD formula was therefore used for preparing native datasets for other languages, such as Russian [9], Korean [10], Persian [1], Vietnamese [15], and Chinese [5].

A case which deserves a more detailed discussion is that of Czech because of its linguistic proximity to Polish. The first Czech dataset for QA, named SQAD [12], was created in 2014 (and therefore the naming similarity to SQuAD is purely coincidental). It has been iteratively enriched in [19,22], and it has a different approach to annotation, more suited to Slavic languages. Additionally, [11] represents the efforts to use machine-translated data for training QA models. In that study, the best model trained on translated data fares worse on Czech (79.2%) than the same architecture on the original data (86.2%), which suggests that translation does lead to a substantial data degradation. [11] also raises an important point, namely the fact that SQuAD's overrepresentation of named entities may artificially inflate the results, as these are usually represented more uniformly across different languages. For a more general task, these cross-lingual strategies might fare even worse than ones which use native data.

With respect to Polish data, [16] is the sole native resource, but the questions are not paired with the relevant paragraphs, and therefore it cannot be used for extractive QA as is. [2] offers a machine-translation of the SQuAD 2.0 dataset, but the reported top score of 61.9%, when evaluated on the translated dev-set, is not satisfactory. The translated data is of lower quality because of 1. translation errors, and 2. additional difficulties in aligning answer spans between the translated documents. Additionally, the data in SQuAD are biased towards the anglophone culture, with questions about American pop stars, cities in the US, or intricacies of the political systems of the anglosphere being much more frequent than what would be of interest to the average Polish reader. The previous work is therefore insufficient with respect to providing satisfactory question answering capabilities for Polish, and a resource filling that gap would be an important addition to Polish NLP. **PoQuAD** is planned as a means of bridging this gap.

3 Data Gathering

3.1 Textual Data

Textual data was obtained from Polish Wikipedia by scraping articles falling into one of three categories which are recognized by the Wikipedia community: 1. Featured Articles, 2. Good Articles, or 3. Most popular articles.

These criteria were imposed in order to ensure the quality of textual data and also relevance to the interests of an average Polish reader. Articles were then divided into a summary (usually everything before the first header) and the rest of the article. The remaining paragraphs were narrowed down by imposing the criterion of length (over 500 characters). Subsequently, textRank algorithm was used to rank the centrality of these paragraphs, and only the top scoring paragraph of each article was selected for annotation.

This is a substantial difference from the original SQuAD approach, where entire articles were annotated. The original method is perhaps more cost efficient as it does not require annotators first familiarizing themselves with the article, and usually makes it impossible for a paragraph to be incomprehensible for the annotator. On the other hand, this method focuses on the more interesting paragraphs (as per textRank), and covers a broader range of topics. However, because even with the summary available, a paragraph can be incomprehensible without the fuller context, we allow annotators to entirely skip paragraphs if they are unable to ask questions about them. Including such a possibility can also nudge them against forcing trivial questions.

3.2 Annotation Process

Four in-house annotators were given instructions about the desiderata for the data, which were mostly similar to those from SQuAD, i.e. emphasized lexical differences between the question and context, and encouraged asking interesting, hard questions. The proportion between possible and impossible questions was to be kept roughly around 4:1. All the annotations were done *via* LabelStud.io [20],

with a custom interface built for the task. After collection, the annotations were validated by both automatic methods and manual supervision by a linguist. The automatic validation relied on using a custom Polish spaCy model[1], and it aimed to identify: 1. technical errors, e.g. missing labels or questions, 2. misspellings in question or generative answer text, 3. questions with high lexical similarity to the corresponding text fragment, and 4. overrepresentation of a particular type of questions or answers, e.g. yes-no questions or dates. Validation results served to identify systematic problems, before proceeding onto manual curation, which included marking incorrect annotations for a random sample of each tranche (200 paragraphs) as falling into one of error types, e.g. WRONG EXTRACTIVE ANSWER SPAN, or MULTIPLE PLAUSIBLE ANSWERS. The annotators were then asked to correct their annotation based on the results of both phases of validation, and only the tranches which passed both phases were admitted into the dataset.

3.3 Differences with Respect to SQuAD

As stipulated, there were some deviations from the original SQuAD formula of the task. The most important ones are as follows:

Ambiguous Questions. Because a satisfactory question answering system should be able to answer ambiguous questions in context (e.g. in a series of questions about the same topic, or based off the metadata about the user, e.g. which page they are currently on), some degree of ambiguity in the data would be essential for training. For this reason, annotators are not discouraged to ask such ambiguous questions as long as it is clear, for an average reader, how the ambiguities should be resolved based on the paragraph. For example:

Czy Jerzy Płażewski wydał negatywną opinię o filmie Wajdy?
[Did Jerzy Płażewski review Wajda's film negatively?]

It is only in the context of the paragraph, which is wholly devoted to the film "Popiół i Diament", that it becomes clear what film is the subject of the question. This is an acceptable level of ambiguity. On the other hand:

Jak on ocenił to dzieło?
[How did he rate this piece?]

is too ambiguous and therefore would not be accepted into the dataset, as it would introduce noise into the training process.

Generative Question Answering. In English QA, a fragment extracted from text can usually be used as an answer without any alterations. This does not apply to the morphologically rich Polish. A word or an entire phrase can appear in the text in an inflected form. Returning it as is can be ungrammatical and confusing, as shown in Fig. 1. In such cases a generative method is needed.

[1] https://github.com/ipipan/spacy-pl-trf.

Context: [...] *Ministerstwo Skarbu Pastwa wystpio do Centralnego Biura Antykorupcyjnego z prob o podjcie dziaa sprawdzajcych proces pozyskiwania rodków przez Stpnia na potrzeby finansowania komercyjnej produkcji filmowej z udziaem Anny Szarek, czyli yciowej partnerki Prezesa GPW Luawika Sobolewskiego.* [...]
Question: *Z kim w zwizku bya Anna Szarek?*
[With whom was Anna Szarek in a relationship?]
Extractive Answer: *Ludwika Sobolewskiego* [GEN case]
Generative Answer: *z Ludwikiem Sobolewskim* [INSTR with a preposition added]

Fig. 1. Differences between both annotation layers

For this reason, similarly to [19], we add a second layer of annotation, which is done by hand, and includes answers in a "normalized" form. The cue for the annotator is to convert the extracted fragment into a form which would be most natural and grammatical to use while answering the question during, for example, a conversation. This operation usually involves making necessary inflections, but can also require adding words (e.g. prepositions), subtracting words (e.g. interjections), or expanding abbreviations. Additionally this layer can be used to store answers to yes-no questions; in this case, the extracted answer is usually a sentence which clearly supports "yes" or "no" (which rarely occur explicitly in the contexts) as a generative answer. In these more nuanced cases, the skills demanded by the generative task are not limited to purely linguistic matters, but also to being able to determine which elements are superfluous, what might an abbreviation corefer with, and whether a given fragment supports or contradicts a supposition.

4 Data Analysis

A random sample of 100 question and answer pairs has been analyzed manually (largely following the methodology of [18]). The answers to each question have been grouped into the following categories: common noun phrase, person, other proper nouns, adjective phrase, verb phrase, date, and other numeric answers, as well as yes/no for polar questions. As can be observed in Table 1, noun phrases account for more than half of all the answers in the sample (similar results have been reported by [8,18]). Among them, proper nouns not referring to people prevail, followed by common noun phrases, and references to people. Numerical answers are three times less frequent than noun phrases. Among them, other numbers are slightly more common than dates. The least frequently selected answers are those forming adjectival and verb phrases. Finally, with respect to the polar questions, it can be observed that the annotators had a preference for questions that could be answered affirmatively.

Table 1. Answer type by frequency (in a sample of 100)

Answer type	Freq.	Example
Other proper nouns	28	**Q:** *Jakiego zespołu album jest uważany za najważniejszy w historii MTV Unplugged* [Which band's album is thought to be the most important album in the history of MTV Unplugged]?, **A:** *Alice in Chains*
Common noun phrase	19	**Q:** *Jakie są cechy charakterystyczne klimatu oceanicznego* [What are the characteristics of the oceanic climate]?, **A:** *Wysokie opady* [High precipitation]
Person	10	**Q:** *Jaki naukowiec, między innymi, prowadził badania nad RNA* [Which scientist, among others, did studies on RNA]?, **A:** *Kreiter*
Other numeric	10	**Q:** *Jak wysoko zbudowano miasto Pompeje* [How high was Pompeii located]?, **A:** *40 m n.p.m.* [40 m above sea level]
Date	8	**Q:** *Kiedy trwał konflikt zbrojny pomiędzy Rosją a Japonią* [When were Russia and Japan in conflict]?, **A:** *W 1905 r.* [In 1905]
Adjective phrase	7	**Q:** *Jakie zdolności posiadał przyszły mąż Krystyny z dynastii Wazów* [What talents did the future husband of Christina of the House of Vasa possess]?, **A:** *Wojskowe* [Military]
Verb phrase	7	**Q:** *Jakie są cele "lustitii"* [What are Iustitia's goals]?, **A:** *Umacnianie niezależności sądów i niezawisłości sędziów* [Strengthening the autonomy of courts and the independence of judges]
Yes/No	8/3	**Q:** *Czy chciano wybudować port lotniczy* [Did they want to build an airport]? **A:** *Tak* [Yes]

Additionally, the relationship between the question and the answer for each of the pairs from the same sample was analyzed (see Table 2) in order to shed light on the type of reasoning required to arrive from one to the other. This has shown that lexical and, to a slightly lesser extent, syntactic variation, were the two most frequently adopted procedures in the formation of questions. The following is an example of both lexical and syntactic variation (it also illustrates the fact that some question and answer pairs fall into more than one category):

W jakim miejscu Dee Dee miała ***spotkać*** *chłopaka swojej córki?*
[Where was Dee Dee supposed to **meet** her daughter's boyfriend?]

Context:

Wedle jej planu miał ***wpaść na*** *nią, gdy ona z Dee Dee były w kinie w kostiumach.*
[According to her plan, he was supposed to **bump into** her when she and Dee Dee were at the cinema in costumes.]

With respect to lexical choices, the original *wpaść na* [bump into] is replaced in the question by the more neutral *spotkać* [meet]. As regards the syntactic variation, the question swaps the original text's subject and object and requires the original complex sentence to be restructured into a simple one:

(He was supposed to bump into her when she and Dee Dee were at the cinema in costumes → He was supposed to bump into her at the cinema)

The question and answer pair analysed above also provides an example of another type of reasoning – multiple sentence reasoning as knowledge that *her* refers to Dee Dee and that *her daughter's boyfriend* is the elided subject of the original sentence needs to be accessed from the preceding sentence:

> *Rok później Gypsy zaaranżowała spotkanie matki z Godejohnem oraz zapłaciła mu, gdy ten przybył do Springfield.*
> [A year later, Gypsy arranged a meeting between her mother and Godejohn and paid him when he arrived in Springfield.]

Table 2. Reasoning required to answer questions

Type of reasoning	Frequency
Lexical variation	47
Syntactic variation	43
World knowledge	11
No reasoning	11
Multiple sentence reasoning	8
Ambiguous	7

Other, less populous categories include World knowledge where the lexical gap to be bridged is less about linguistic knowledge, and more based in knowledge about the world, No reasoning where the answer is explicitly stipulated as such in the text, and Ambiguous which includes questions where it is not entirely clear whether the annotated answer is the correct one.

4.1 Evaluation

The 29k collected questions were divided into train, dev and test sets in a 8:1:1 proportion. 3 paradigms of training are considered: training an extractive model on PoQuAD, training an extractive model on translated SQuAD-PL [2], and training a generative model on PoQuAD. In all paradigms, the test set of PoQuAD was used for evaluation. Two metrics are employed: **EM** which requires gold and system answers to be identical, and a macro average of token-wise **F1** coverage between these. Results on answerable and impossible questions are also considered separately, as **HasAns** and **NoAns** respectively.

Generally, native models are significantly superior to multilingual models, and models trained on native data outperform ones trained on the translated dataset. It may be argued that the latter fact stems from the detours from the original SQuAD formula, nevertheless the translated dataset is much larger than PoQuAD, which should at least partialy counteract this factor. All this amounts to a strong argument in favour of working with native data. For extractive QA, the best performer is **HerBERT-large**, with 76.36% **Total F1**, whereas in the generative paradigm, **plT5 large** scores the highest. The extractive results are around 12 p.p. lower than those reported for datasets of similar size (e.g. **F1** of 88.1% in [14], or 87.02% in [15]). The likely cause of this is that these datasets

do not include impossible questions, which, as [17] shows, substantially raise the difficulty level of the task. Although not directly comparable, the generative paradigm, as expected, leads to lower results. What is surprising is the particularly weak performance on impossible questions, which might be due to the inherent bias for producing text, as opposed to returning empty sequences. A full evaluation of both of these hypotheses would require obtaining human performance metrics.

Table 3. Evaluation results on PoQuAD

QA paradigm	Train set	Model	HasAns		NoAns	Total	
			EM	F1	EM	EM	F1
Extractive	PoQuAD	mBERT [7]	52.14	67.26	53.74	52.42	64.90
		XLM-R base [4]	55.93	70.77	48.03	54.55	66.80
		XLM-R large [4]	59.76	75.38	56.89	59.26	72.15
		HerBERT base [13]	59.26	73.64	56.50	58.78	70.65
		HerBERT large [13]	**63.59**	**78.90**	**64.37**	**63.72**	**76.36**
	SQuAD-PL	XLM-R base	36.45	54.25	47.44	38.37	53.06
		HerBERT base	42.99	63.80	42.13	42.84	60.01
		HerBERT large	48.27	70.85	41.34	47.06	65.70
Generative	PoQuAD	mT5 base [21]	51.85	66.34	14.96	45.41	57.37
		BART [6]	48.77	64.32	28.74	45.28	58.11
		PlT5 base [3]	55.89	69.71	17.32	49.16	60.57
		PlT5 large [3]	**67.08**	**80.27**	**36.22**	**61.70**	**72.58**

5 Conclusions

This paper presents a work in progress concerning the creation of a native resource for QA in Polish — PoQuAD. The motivation for the project, annotation methods, aims, and preliminary results were discussed. It is proposed and argued that this work will be an important step in enriching Polish QA and NLP in general. The dataset is available[2] in the SQuAD JSON format, with some additional keys storing the extra annotation layers. As of September 2022, PoQuAD consists of 29k questions, but in the following months, the threshold of 50k is to be reached. The final dataset, besides increased number of examples, would benefit from additional annotation for estimating human performance and robust error analysis, with respect to question types and their quantitative properties.

[2] The repository at https://github.com/ipipan/poquad will be continually updated with new data. It is licensed on **GNU GPL 3.0** license.

Acknowledgements. This work was supported by the European Regional Development Fund as a part of the 2014–2020 Smart Growth Operational Programme: (1) Intelligent travel search system based on natural language understanding algorithms, project no. POIR.01.01.01–00-0798/19; (2) CLARIN — Common Language Resources and Technology Infrastructure, project no. POIR.04.02.00-00C002/19.

References

1. Ayoubi, S., Davoodeh, M.Y.: PersianQA: a dataset for Persian question answering. https://github.com/SajjjadAyobi/PersianQA (2021)
2. Borzymowski, H.: Polish QA model (2020), model trained on HuggingFace. https://huggingface.co/henryk/bert-base-multilingual-cased-finetuned-polish-squad2
3. Chrabrowa, A., et al.: Evaluation of transfer learning for polish with a text-to-text model. arXiv preprint arXiv:2205.08808 (2022)
4. Conneau, A., et al.: Unsupervised cross-lingual representation learning at scale. CoRR (2019). https://arxiv.org/abs/1911.02116
5. Cui, Y., et al.: A span-extraction dataset for Chinese machine reading comprehension. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 5883–5889 Association for Computational Linguistics, Hong Kong, China (2019). https://doi.org/10.18653/v1/D19-1600
6. Dadas, S.: Polish BART. https://github.com/sdadas/polish-nlp-resources#bart
7. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. CoRR (2018). https://arxiv.org/abs/1810.04805
8. d'Hoffschmidt, M., Belblidia, W., Brendlé, T., Heinrich, Q., Vidal, M.: FQuAD: French question answering dataset (2020). https://arxiv.org/abs/2002.06071
9. Efimov, P., Chertok, A., Boytsov, L., Braslavski, P.: SberQuAD – Russian reading comprehension dataset: description and analysis. In: Arampatzis, A., et al. (eds.) CLEF 2020. LNCS, vol. 12260, pp. 3–15. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58219-7_1
10. Lim, S., Kim, M., Lee, J.: Korquad1.0: korean QA dataset for machine reading comprehension (2019). https://arxiv.org/abs/1909.07005
11. Macková, K., Straka, M.: Reading comprehension in Czech via machine translation and cross-lingual transfer (2020). https://arxiv.org/abs/2007.01667
12. Medved, M., Horak, A.: SQAD: Simple question answering database. In: RASLAN (2014)
13. Mroczkowski, R., Rybak, P., Wróblewska, A., Gawlik, I.: HerBERT: efficiently pretrained transformer-based language model for polish. In: Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing, pp. 1–10. Association for Computational Linguistics, Kiyv, Ukraine (2021). https://www.aclweb.org/anthology/2021.bsnlp-1.1
14. Möller, T., Risch, J., Pietsch, M.: GermanQuAD and GermanDPR: improving non-english question answering and passage retrieval (2021). https://arxiv.org/abs/2104.12741

15. Nguyen, K., Nguyen, V., Nguyen, A., Nguyen, N.: A Vietnamese dataset for evaluating machine reading comprehension. In: Proceedings of the 28th International Conference on Computational Linguistics. pp. 2595–2605. International Committee on Computational Linguistics, Barcelona, Spain (2020). https://doi.org/10.18653/v1/2020.coling-main.233
16. Ogrodniczuk, M., Przybyła, P.: PolEval 2021 task 4: question answering challenge (2021)
17. Rajpurkar, P., Jia, R., Liang, P.: Know what you don't know: Unanswerable questions for squad (2018). https://doi.org/10.48550/ARXIV.1806.03822
18. Rajpurkar, P., Zhang, J., Lopyrev, K., Liang, P.: Squad: 100,000+ questions for machine comprehension of text. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 2383–2392. Association for Computational Linguistics, Austin, Texas (2016). https://doi.org/10.18653/v1/D16-1264
19. Sabol, R., Medved' M., Horák, A.: Czech question answering with extended sqad v3.0 benchmark dataset. In: Horák, A., Rychlý, P., Rambousek, A. (eds.) Proceedings of the Thirteenth Workshop on Recent Advances in Slavonic Natural Languages Processing, RASLAN 2019, pp. 99–108. Tribun EU, Brno (2019)
20. Tkachenko, M., Malyuk, M., Holmanyuk, A., Liubimov, N.: Label studio: data labeling software (2020–2022). https://github.com/heartexlabs/label-studio
21. Xue, L., et al.: mT5: a massively multilingual pre-trained text-to-text transformer. CoRR (2020). https://arxiv.org/abs/2010.11934
22. Šulganová, T., Marek, M., Horák, A.: Enlargement of the Czech question-answering dataset to SQAD v2.0. In: Proceedings of the Eleventh Workshop on Recent Advances in Slavonic Natural Language Processing, RASLAN, pp. 79–84. Brno (2017)