PolEval 2021 Task 4: Question Answering Challenge

Maciej Ogrodniczuk, Piotr Przybyła
(Institute of Computer Science, Polish Academy of Sciences)

Abstract

We introduce the Question Answering Challenge – a shared task organised at the PolEval 2021. The task involves answering open-domain free-form questions in Polish through an automatic system, without human intervention or accessing external services. We describe the motivation behind the problem, explore various question types and formulations, and lay out the rules of the competition. The solutions submitted by 7 teams are discussed in terms of their evaluation results and the techniques applied.

Keywords

question answering, PolEval, shared task

1. Introduction

The idea of an open domain question answering (QA) system has always seemed more attractive than keyword-based search since it promises a compelling perspective of a seemingly intelligent agent answering users' questions instantaneously rather than requiring them to browse through documents to find the answer to their question. But the QA setup is also natural for knowledge exchange between humans: for years it was used to test general knowledge in quiz shows such as *Fifteen to One* (PL: *Jeden z dziesięciu*) or *Jeopardy!* (PL: *Va banque*). In 2011, this dream came closer to realisation when IBM Watson (Ferrucci et al. 2010) successfully competed with human contestants in a series of *Jeopardy!* games.

The purpose of the work described here is to encourage a similar development for Polish. The Question Answering Challenge (QAC) involves developing a solution capable of providing answers to general-knowledge questions, typical for popular TV quiz shows. With numerous new language models and deep neural network-powered improvements in almost every natural language processing task, the QAC results can show where the state-of-the-art stands and further motivate development and testing of new solutions for languages other than English.

2. Related work

A task, in which a computer system is expected to formulate an answer in a natural language (e.g. English) that addresses a question in a natural language (typically the same one) is described as *Question Answering* (QA). Given the flexibility of natural languages, a great variety of problems might be formulated as QA tasks, but what they all have in common is the need to apply NLP methods to understand the question and generate an answer.

2.1. QA systems

BASEBALL (Green et al. 1961), the first QA system developed, was only capable of answering questions regarding a strictly defined domain, namely results of baseball matches in a US league during one season. A set of rules was used to convert a question to a query for a structured database, from which an answer was retrieved. Similar approaches were popular in the following years (Simmons 1970), but the manual effort necessary to prepare rules was limiting their abilities. The availability of vast open-domain knowledge on the Internet triggered the second wave of QA systems; e.g. *START* (Katz 1997) converted web text to a set of triples, in which the one matching a query was sought.

The standard architecture for modern open-domain QA includes three stages: question analysis, document retrieval and answer extraction. As in other areas of NLP, current QA approaches employ neural networks trained on large datasets for each of these steps (Zhu et al. 2021). More recently, advances in natural language generation based on the Transformer architecture (Vaswani et al. 2017) have allowed to encode significant amount of knowledge in a pre-trained language model. As a result, solutions like GPT-3 (Brown et al. 2020) can achieve competitive QA performance given a prompt with just a few examples.

2.2. QA data

Given the importance of ML approaches in the QA solutions, it is no surprise that availability of training data has played a critical role in the research so far. The first datasets were made openly available through the QA shared tasks organised within the Text Retrieval Conference (TREC) from 1999 (Voorhees 1999) to 2007 (Dang et al. 2007).

Since then, many more collections, built from vastly different perspectives, have been published and used for evaluation efforts – see review by Zhu et al. (2021). Some aspects in which these datasets differ are (1) whether the knowledge base to extract answers from is provided; (2) whether questions belong to a particular topic or are open-domain and (3) whether an expected answer is a span from the knowledge base, a selected option (true/false or multiple choice), or a free-form text. Datasets that, like in our task, contain free-form answers from any domain, include *MARCO* (Nguyen et al. 2016), *Quasar-T* (Dhingra et al. 2017) and *DuReader* (He et al. 2018).

In case of many datasets, including the three mentioned above, each question is also accompanied by a few text passages (usually results from a search engine) that should contain the answer. The task of answering a question based on such text is known as Machine Reading Comprehension (MRC). Many QA formulations differ from ours in that they rely on the provided snippets even further, requiring an answer to be a span within one of these passages, rather than a free-form string. A well-known example for this is Stanford Question Answering Dataset (SQuAD), which includes 100,000 questions prepared by crowdworkers based on articles from Wikipedia (Rajpurkar et al. 2016).

2.3. QA for Polish

Similarly to other NLP tasks, the resources and solutions for English are by far the most numerous, but not the only ones available. The first QA system for Polish was developed by Vetulani (1988) as a natural-language interface to *ORBIS* database, implemented in *PROLOG*. As with English, all initial attempts were limited to a fixed domain, such as business information (Duclaye et al. 2002) or public safety (Vetulani et al. 2010). To answer open-domain questions, some systems relied on the structure of Wikipedia; for example RAFAEL (Przybyła 2016) uses it both as a knowledge base and to build a resource for precise entity recognition (Przybyła 2015). Other approaches used a corpus of web pages gathered based on the questions in the evaluation set (Walas and Jassem 2010, Marcińczuk et al. 2013).

Regarding datasets, two of the approaches mentioned above were accompanied by substantial question collections in Polish. The first system used 4721 questions gathered from the *Did you know?* panel on Wikipedia page (Marcińczuk et al. 2013). An important limitation of the collection is the lack of explicit answers – instead, each question is associated with a Wikipedia page with a selection of relevant fragments. The second system, RAFAEL (Przybyła 2016), used a collection of 1130 questions from a Polish quiz TV show *Jeden z dziesięciu* (Karzewski 1997). These questions were included with explicit answers and could thus be used for the present work (see Section 4).

3. QAC task

The problem in QAC is simple: given a text string, expressing an open-domain question in Polish, return a text string consisting a correct answer in Polish. According to the task's rules, the participants are free to choose any approach, as long as it answers the question automatically, without human assistance. The participants can build on top of existing resources, but they need to be downloaded and included locally, so the questions can be answered without access to the Internet. Thus, for example, pre-downloading Wikipedia and indexing it for search is acceptable (and applied by most participants), but querying Google online is not.

In order to facilitate the development of high-performing solutions by the participants, the following resources were made available:

- collections of questions with answers (Section 4), representing various types (Section 4.2) and split into development and test subsets (Section 4.4),
- an automatic evaluation method, allowing for some deviation between the user-supplied and gold-standard answers (Section 5),
- a simple baseline solution to compare against (Section 6).

4. Question collection

For the purposes of the task, a collection of 6000 questions and answers was created. Here we describe the data sources, various types of questions and answers observed and how the dataset was used in QAC.

4.1. Data sources

The data for the task was collected from various sources. The majority of question-answer pairs was collected from websites, where fans of the quiz shows collect or exchange questions[1], file sharing services[2], various online newspaper quizzes[3] and eventually from video-on-demand collections of actual quiz shows broadcast on the Polish television[4]. The 1130 questions gathered for the RAFAEL system (see Section 2.3) were also included.

The data was manually cleaned in numerous stages, including improving the phrasing, removing of duplicates, verification of answers in Wikipedia, adding variant answers or different formulations of the same answer (see Section 4.3).

The following types of questions were removed from the set:

- about issues dependent on time of asking the question or formulating the answer (e.g. the current president of Poland)
- seeking more than two entities in an answer (e.g. three out of five neighbouring countries)
- requesting to sort items (e.g. from the highest to lowest mountain)
- related to the spelling rules (e.g. *ch/h*, *rz/ż*, *ó/u* etc.)
- those requiring longer explanations (e.g. starting with *why*).

[1] Such as `http://zenzycia.blogspot.com/2018/08/pytania-z-teleturnieju-1-z-10-spis.html`, similar to `https://j-archive.com/` where fans of *The Jeopardy!* share questions and answers from episodes aired between 1983 and now.

[2] Such as `chomikuj.pl`.

[3] E.g. `onet.pl`, `kurierlubelski.pl`, `radiozet.pl`, `gazetawroclawska.pl`, `se.pl`, `polskatimes.pl`.

[4] E.g. *To był rok*, *Va Banque* i *Jeden z dziesięciu* na platformie `vod.tvp.pl`.

The dataset was then truncated to 6000 records and split into development and two test subsets.

4.2. Question types

Grouping questions into categories is helpful to understand the challenges posed by the task, but there are many possible ways to do this. Our categorisation is based on previous work for Polish (Przybyła 2016), but includes additional types we observed in our, significantly larger, dataset. Namely, we group question according to what information they seek:

- **Single entity**: a name of a specific entity:

  Q: *Jak nazywa się bohaterka gier komputerowych z serii Tomb Raider?* [Who is the hero in the Tomb Rider video game series?]

  A: *Lara Croft*

- **Multiple entities**: several entities, specified by names[5]:

  Q: *Które dwa morza łączy Kanał Koryncki?* [Which two seas are linked by the Corinth Canal?]

  A: *Egejskie i Jońskie* [Aegean and Ionian]

- **Entity choice**: one of the options, which are given in the question:

  Q: *Co zabiera Wenus więcej czasu: obieg dookoła Słońca czy obrót dookoła osi?* [What takes more time in case of Venus: revolving around the sun or rotating about its own axis?]

  A: *Obrót dookoła osi* [Rotating about its axis]

- **True/false**: veracity value:

  Q: *Czy w przypadku skrócenia kadencji Sejmu ulega skróceniu kadencja Senatu?* [When the term of office of the Polish Sejm is terminated, does it apply to the Senate as well?]

  A: *Tak* [Yes]

- **Other name**: alternative names for a given entity:

  Q: *Jaki przydomek nosił Ludwik I, król Franków i syn Karola Wielkiego?* [What was the nickname of Louis I, the King of the Franks and the son of Charlemagne?]

  A: *Pobożny* [The Pious]

- **Gap filling**: words that complete a given sentence, quote or expression:

  Q: *Proszę dokończyć powiedzenie: „piłka jest okrągła, a bramki są. . . ”* [Finish the adage: „the ball is round and the goals are. . . ”]

  A: *Dwie* [Two]

Moreover, the questions, where an answer involves one or more entities (Single entity, Multiple entities, Entity choice) can be also divided with respect to the type of the entities sought:

[5] In our dataset we only include questions that can be answered by providing two entities.

- **Named entities**: specific entities that are referred to through their names, e.g. people, countries, organisations. This also includes quantities, numbers and dates.
- **Unnamed entities**: more general categories of entities, such as objects, concepts or species:

  Q: *Paź królowej to gatunek których owadów?* [Old World swallowtail is a species of what type of insects?]

  A: *Motyli* [Butterflies]

Finally, the questions can be phrased through one of the following formulations:

- **plain question**,
- **command**: *Proszę rozwinąć skrót CIA.* [Expand the abbreviation 'CIA'.]
- **compound** consisting of more than one sentence: *Ten urodzony w XIX w. Nantes francuski pisarz uchodzi za prekursora literatury fantastycznonaukowej. O kogo chodzi?* [This French writer, born in the 19th century, is considered a pioneer of the sci-fi literature. Who is he?]

4.3. Answers

Answer strings are formulated in the same way, in which a Polish speaker would normally reply to a given query, e.g. when participating in a quiz. Namely, they contain just a few words that satisfy the question, for example a name of a person or place. This is different to many other QA shared tasks, where a sentence, a paragraph, or even a whole document from a knowledge base are considered an acceptable result.

In QAC the answers can, when appropriate:

- contain prepositions:

  Q: *W którym mieście trasa drogi krzyżowej przebiega ulicą Via Dolorosa?* [In which city a traditional Good Friday procession is held on the Via Dolorosa street?]

  A: *W Jerozolimie* [In Jerusalem]
- be inflected:

  Q: *Symbolem którego pierwiastka jest Cr*? ['Cr' is the symbol of which element?]

  A: *Chromu* [Of chrome[6]]
- contain punctuation:

  Q: *W jakim filmie mężczyzna w białym garniturze zrywa lilie ze stawu?* [In which film a man in white suit is seen in a pond, picking lilies?]

  A: *„Noce i dnie”*

[6] In Polish this meaning is achieved by inflecting the word instead of adding a preposition.

Table 1: Summary of the question collection

			Development	Test A	Test B	Total
Number of questions			1000	2500	2500	6000
Average question size in tokens			8.43	8.41	9.49	8.87
Median question size in tokens			8	8	9	8
Min question size in tokens			3	3	3	3
Max question size in tokens			21	22	32	32
Number of questions with	1	**answer variants**	890	2178	1890	4958
	2		104	307	570	981
	3		5	13	29	47
	4		0	1	11	12
	5		1	1	0	2

— for person names, include the first name and surname:

Q: *Który brytyjski pisarz wprowadził do literatury pojęcie Wielkiego Brata?* [Which British writer introduced the concept of 'Big Brother'?]

A: *George Orwell*

In some cases, more than one answer text is associated with a question, e.g. *Richard I*, *Richard Cœur de Lion* and *Richard the Lionheart*. Their meaning is identical and this redundancy is necessary to make sure the automatic evaluation procedure accepts different synonyms of the right answer. Also, when two entities are requested, both *A and B* and *B and A* are recorded as the possible answers.

4.4. Data format and characteristics

The whole QAC question collection (see summary in Table 1) was split into the following subsets provided to the participants:

— a small development set, including question and answers,

— test set A, including question and answers,

— test set B, without answers.

Test set B and the answers to test set A were provided in the last week of the competition, which allowed the participants to include test set A in their training data for the final submission. The collections are published in the GitHub repository of the task[7].

The development dataset includes two UTF-8-encoded text files: `in.tsv` with questions and `expected.tsv` with tab-separated answers. The submission file is supposed to contain just answers in separate lines.

Questions contain 8–9 tokens on average and are mostly simple, asking for a single entity. The longest answer has 7 tokens and is associated with the question about two labours of

[7] `https://github.com/poleval/2021-question-answering`

Heracles related to horses (stealing the Mares of Diomedes and cleaning the Augean stables). The data contain various numbers of accepted answers – usually corresponding to different formulations (e.g. full name of a person or just their surname), but sometimes referring to entirely different names for a certain entity, e.g. *trembita*, *trombita*, *bazuna* or *ligawka* for the name of an alpine horn made of wood.

5. Evaluation method

Submissions were evaluated by comparing the known answer (gold standard) to the one provided by the participating systems (predictions). The number of matching answers divided by the number of questions in the test set was considered the **accuracy** of a given approach.

Checking if the two answers match depends on the question type:

- For non-numerical questions, we assessed textual similarity. To that end, a character-wise Levenshtein distance was computed between the two (lowercased) strings and if the obtained value was less than ½ of the length of the gold standard answer, we accepted the candidate answer.
- For numerical questions (e.g. *In which year...*), we assessed numerical similarity. Specifically, we used a regular expression to extract a sequence of characters that could be interpreted as a number (Arabic or Roman numeral). If such sequences could be found in both answers and represented the same number, we accepted the prediction.

For questions, where more than one answer text was available, the answer that had the best match with the candidate was used.

6. Baseline

The `WIKI_SEARCH` baseline solution used the question as a query to the Wikipedia search service and treated the title of the first returned article as an answer, as long as it didn't overlap with the question.

Specifically, the following procedure was followed:

1. Split the question into tokens using spaCy (model `pl_core_news_sm`) and ignore the one-character tokens,
2. Send the space-separated tokens as a query to the Search API of the Polish Wikipedia[8],
3. For each of the returned articles:
   (a) Split its title into tokens with spaCy,
   (b) If none of the tokens of the title has at least 50% overlap (measured as in Section 5) with any of the tokens of the question:

[8] `https://www.mediawiki.org/wiki/API:Search`

        i. remove the part of the title starting from '(', if found
        ii. return the title as an answer,
    (c) Otherwise, continue to the next result,

4. If no answer is found at this point, remove the first of the question tokens and jump back to (2).

7. Submissions and results

The competition attracted 50 submissions in total. In its initial phase (with submissions evaluated on test-A dataset) 22 solutions (including the official baseline from Section 6) were submitted by only 2 external participants. In the final phase 28 submissions were made by 7 participants.

Table 2 reports on the accuracies of individual highest-scoring submissions for each participant. The winning system by Mateusz Piotrowski scored 71.68%, surpassing other competitors by more than 20 points. Two independent second-best submissions by Aleksander Smywiński-Pohl and Piotr Rybak scored 50.96%. WIKI_SEARCH entry corresponds to the official baseline and the two other lowest-scoring submissions were marked as baselines by their authors.

Table 2: Accuracy scores at PolEval 2021 Task 4

Submission	**test-B accuracy**
Mateusz Piotrowski	71.68%
Aleksander Smywiński-Pohl et al.	50.96%
Piotr Rybak	50.96%
Darek Kłeczek	46.44%
Karol Gawron	36.12%
WIKI_SEARCH	9.60%
Filip Graliński	4.16%
BI Insight	0.96%

The authors of the four top-scoring submissions agreed to describe their systems and present them at the PolEval workshop. You can find short summaries of each solution in Sections 7.1–7.4 and relevant papers later in this volume.

7.1. Search-augmented question answering system using multilingual transformer model

The solution by Piotrowski (2021) used a combination of Wikipedia knowledge retrieval and multilingual transformer model. First, questions were used as search queries to retrieve a set of relevant paragraphs from the Polish Wikipedia. Then the results ranked with a BM25 scoring function, along with a question, were fed to a neural network, which is responsible for generating the final answer. The mT5 model (Xue et al. 2021) was used to generate answers.

7.2. Dense Passage Retriever

The solution by Smywiński-Pohl et al. (2021) used a combination of approaches dependent on question type. For extractive question answering the authors used Dense Passage Retrieval (Karpukhin et al. 2020) trained on "Czy wiesz" dataset (Marcińczuk et al. 2013). Boolean questions were answered according the Natural Language Inference model, trained on CDSCorpus (Wróblewska and Krasnowska-Kieraś 2017, Krasnowska-Kieraś and Wróblewska 2019) and a dump of Wikipedia (Smywiński-Pohl 2019), utilizing HerBERT large (Mroczkowski et al. 2021).

7.3. Retrieve and refine system for Polish question answering

The solution by Rybak (2021) consisted of two modules. The first one searched for Wikipedia paragraphs containing the answer to a question. For training such a model, 10 000 question-paragraph pairs were manually annotated and merged with the "Czy wiesz" collection (Marcińczuk et al. 2013, Rybak et al. 2020). The encoder was trained with the HerBERT Large model (Mroczkowski et al. 2021). Then it was used to find the 10 most relevant paragraphs for each question. In the second step a generative model was trained on plT5 Base[9] which takes the question and the top 10 paragraphs as input and generates the answer as output. The second model was trained on the test-A validation set, a thousand additional questions from "Jeden z dziesięciu" quiz and a subset of Multi-lingual Knowledge Questions & Answers (Longpre et al. 2020).

7.4. Retriever-reader approach with data-driven improvements

The solution by Kłeczek (2021) was based on retriever-reader architecture using Deepset Haystack framework (Deepset 2021) with ElasticSearch document store. The dataset was further supplemented with Polish Wikipedia dump, word definitions from Słowosieć (Maziarz et al. 2016) and the Polish subset of mC4 dataset (Xue et al. 2021) filtered for URLs of popular educational websites. BM25 was used as a retriever and reader was XLM-Roberta Large pre-trained on original English SQUAD (Chan et al. 2021). Finally the results were postprocessed by converting numbers to numeric form and answering all yes/no questions with *yes*.

Acknowledgements

This work was supported by the *Polish National Agency for Academic Exchange* through a *Polish Returns* grant number PPN/PPO/2018/1/00006.

[9] https://hf.co/allegro/plt5-base

We would like to thank Aleksander Smywiński-Pohl for spotting a few errors in our dataset.

References

Brown T. B., Mann B., Ryder N., Subbiah M., Kaplan J., Dhariwal P., Neelakantan A., Shyam P., Sastry G., Askell A., Agarwal S., Herbert-Voss A., Krueger G., Henighan T., Child R., Ramesh A., Ziegler D. M., Wu J., Winter C., Hesse C., Chen M., Sigler E., Litwin M., Gray S., Chess B., Clark J., Berner C., Mccandlish S., Radford A., Sutskever I. and Amodei D. (2020). *Language Models are Few-Shot Learners*. In *Advances in Neural Information Processing Systems*, vol. 33, pp. 1877–1901.

Chan B., Möller T., Pietsch M. and Soni T. (2021). *Multilingual XLM-RoBERTa large for QA on Various Languages*. `https://huggingface.co/deepset/xlm-roberta-large-squad2`.

Dang H. T., Kelly D. and Lin J. (2007). *Overview of the TREC 2007 Question Answering Track*. In *Proceedings of The Sixteenth Text REtrieval Conference (TREC 2007)*. NIST.

Deepset (2021). *Haystack*. `https://github.com/deepset-ai/haystack`. GitHub repository.

Dhingra B., Mazaitis K. and Cohen W. W. (2017). *Quasar: Datasets for Question Answering by Search and Reading*. arXiv:1707.03904.

Duclaye F., Sitko J., Filoche P. and Collin O. (2002). *A Polish Question-Answering System for Business Information*. In *Proceedings of the 5th International Conference on Business Information Systems (BIS 2002)*, pp. 209–212. Department of Information Systems, Poznań University of Economics.

Ferrucci D. A., Brown E. W., Chu-Carroll J., Fan J., Gondek D., Kalyanpur A., Lally A., Murdock J. W., Nyberg E., Prager J. M., Schlaefer N. and Welty C. A. (2010). *Building Watson: An Overview of the DeepQA Project*. „AI Magazine", 31(3), p. 59–79.

Green B. F., Wolf A. K., Chomsky C. and Laughery K. (1961). *BASEBALL: an Automatic Question Answerer*. In *Proceedings of Western Joint IRE-AIEE-ACM '61 Computer Conference*, pp. 219–224. ACM Press.

He W., Liu K., Liu J., Lyu Y., Zhao S., Xiao X., Liu Y., Wang Y., Wu H., She Q., Liu X., Wu T. and Wang H. (2018). *DuReader: a Chinese Machine Reading Comprehension Dataset from Real-world Applications*. In *Proceedings of the Workshop on Machine Reading for Question Answering*, pp. 37–46, Melbourne, Australia. Association for Computational Linguistics.

Karpukhin V., Oguz B., Min S., Lewis P., Wu L., Edunov S., Chen D. and Yih W.-t. (2020). *Dense Passage Retrieval for Open-Domain Question Answering*. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781, Online. Association for Computational Linguistics.

Karzewski M. (1997). *Jeden z dziesięciu — pytania i odpowiedzi*. Muza SA.

Katz B. (1997). *Annotating the World Wide Web Using Natural Language*. In *Proceedings of the 5th RIAO Conference on Computer Assisted Information Searching on the Internet (RIAO 97)*, pp. 136–155.

Kłeczek D. (2021). *Simple Recipes for Question Answering*. In Ogrodniczuk and Kobyliński (2021), pp. 159–161.

Krasnowska-Kieraś K. and Wróblewska A. (2019). *Empirical Linguistic Study of Sentence Embeddings*. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 5729–5739, Florence, Italy. Association for Computational Linguistics.

Longpre S., Lu Y. and Daiber J. (2020). *MKQA: A Linguistically Diverse Benchmark for Multilingual Open Domain Question Answering*. arXiv:2007.15207.

Marcińczuk M., Ptak M., Radziszewski A. and Piasecki M. (2013). *Open Dataset for Development of Polish Question Answering Systems*. In Vetulani Z. and Uszkoreit H. (eds.), *Proceedings of the 6th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*, pp. 479–483. Wydawnictwo Poznańskie, Fundacja Uniwersytetu im. Adama Mickiewicza.

Marcińczuk M., Radziszewski A., Piasecki M., Piasecki D. and Ptak M. (2013). *Evaluation of a Baseline Information Retrieval for a Polish Open-domain Question Answering System*. In *Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP 2013)*, pp. 428–435. Association for Computational Linguistics.

Maziarz M., Piasecki M., Rudnicka E., Szpakowicz S. and Kędzia P. (2016). *plWordNet 3.0 – a Comprehensive Lexical-Semantic Resource*. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 2259–2268, Osaka, Japan. The COLING 2016 Organizing Committee.

Mroczkowski R., Rybak P., Wróblewska A. and Gawlik I. (2021). *HerBERT: Efficiently Pretrained Transformer-based Language Model for Polish*. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*, pp. 1–10, Kiyv, Ukraine. Association for Computational Linguistics.

Nguyen T., Rosenberg M., Song X., Gao J., Tiwary S., Majumder R. and Deng L. (2016). *MS MARCO: A Human Generated MAchine Reading COmprehension Dataset*. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches (CoCo 2016)*, Barcelona, Spain. CEUR Workshop Proceedings.

Ogrodniczuk M. and Kobyliński Ł., editors (2021). *Proceedings of the PolEval 2021 Workshop*, Warsaw, Poland. Institute of Computer Science, Polish Academy of Sciences.

Piotrowski M. (2021). *Search-Augmented Question Answering System Using Multilingual Transformer Model*. In Ogrodniczuk and Kobyliński (2021), pp. 137–140.

Przybyła P. (2015). *Gathering Knowledge for Question Answering Beyond Named Entities*. In Biemann C., Handschuh S., Freitas A., Meziane F. and Métais E. (eds.), *Proceedings of the 20th International Conference on Applications of Natural Language to Information Systems (NLDB 2015)*, pp. 412–417, Passau, Germany. Springer-Verlag.

Przybyła P. (2016). *Boosting Question Answering by Deep Entity Recognition*. arXiv:1605.08675.

Rajpurkar P., Zhang J., Lopyrev K. and Liang P. (2016). *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, Austin, Texas. Association for Computational Linguistics.

Rybak P. (2021). *Retrieve and Refine System for Polish Question Answering*. In Ogrodniczuk and Kobyliński (2021), pp. 151–157.

Rybak P., Mroczkowski R., Tracz J. and Gawlik I. (2020). *KLEJ: Comprehensive Benchmark for Polish Language Understanding*. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1191–1201, Online. Association for Computational Linguistics.

Simmons R. F. (1970). *Natural Language Question-Answering Systems: 1969*. „Communications of the ACM", 13(1), p. 15–30.

Smywiński-Pohl A. (2019). *Results of the PolEval 2019 Shared Task 3: Entity Linking*. In Ogrodniczuk M. and Łukasz Kobyliński (eds.), *Proceedings of the PolEval 2019 Workshop*. Institute of Computer Science, Polish Academy of Sciences.

Smywiński-Pohl A., Zhylko D., Wróbel K. and Król M. (2021). *Answering Polish Trivia Questions with the Help of Dense Passage Retriever*. In Ogrodniczuk and Kobyliński (2021), pp. 141–150.

Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser Ł. and Polosukhin I. (2017). *Attention Is All You Need*. In *Advances in Neural Information Processing Systems 30*, pp. 5998–6008. Curran Associates, Inc.

Vetulani Z. (1988). *PROLOG Implementation of an Access in Polish to a Data Base*. In *Studia z automatyki XII*, pp. 5–23. PWN.

Vetulani Z., Marciniak J., Vetulani G., Dąbrowski A., Kubis M., Osiński J., Walkowska J., Kubacki P. and Witalewski K. (2010). *Zasoby językowe i technologia przetwarzania tekstu POLINT-112-SMS jako przykład aplikacji z zakresu bezpieczeństwa publicznego*. Wydawnictwo Naukowe UAM.

Voorhees E. M. (1999). *The TREC-8 Question Answering Track Report*. In *Proceedings of The Eight Text REtrieval Conference (TREC 2000)*, vol. 7. National Institute of Standards and Technology, NIST.

Walas M. and Jassem K. (2010). *Named Entity Recognition in a Polish Question Answering System*. In Kłopotek M. A., Marciniak M., Mykowiecka A., Penczek W. and Wierzchoń S. T. (eds.), *Intelligent Information Systems*, pp. 181–191. Publishing House of University of Podlasie.

Wróblewska A. and Krasnowska-Kieraś K. (2017). *Polish Evaluation Dataset for Compositional Distributional Semantics Models*. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 784–792, Vancouver, Canada. Association for Computational Linguistics.

Xue L., Constant N., Roberts A., Kale M., Al-Rfou R., Siddhant A., Barua A. and Raffel C. (2021). *mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer*. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 483–498, Online. Association for Computational Linguistics.

Zhu F., Lei W., Wang C., Zheng J., Poria S. and Chua T.-S. (2021). *Retrieving and Reading: A Comprehensive Survey on Open-domain Question Answering*. arXiv:2101.00774.