Preproceedings of the of the 18th Conference on Computer Science and Intelligence Systems pp. 1237–1244

PolEval 2022/23 Challenge Tasks and Results

Łukasz Kobyliński*, Maciej Ogrodniczuk*, Piotr Rybak*, Piotr Przybyła*‡, Piotr Pęzik¶‖, Agnieszka Mikołajczyk‖, Wojciech Janowski‖, Michał Marcińczuk† and Aleksander Smywiński-Pohl§
*Institute of Computer Science, Polish Academy of Sciences
Jana Kazimierza 5, 01-248 Warszawa, Poland
E-mail: {lukasz.kobylinski, maciej.ogrodniczuk, piotr.rybak, piotr.przybyla}@ipipan.waw.pl
†Department of Artificial Intelligence, Wrocław University of Science and Technology,
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
E-mail: michal.marcinczuk@pwr.edu.pl
‡Universitat Pompeu Fabra
Barcelona, Spain
E-mail: piotr.przybyla@upf.edu
§AGH University of Krakow
Kraków, Poland
E-mail: apohllo@agh.edu.pl
¶University of Łódź, Poland
E-mail: piotr.pezik@uni.lodz.pl
‖VoiceLab.AI, Gdańsk, Poland
E-mail: agnieszka.mikolajczyk@voicelab.ai

***Abstract*—This paper summarizes the 2022/2023 edition of PolEval — an evaluation campaign for natural language processing tools for Polish. We describe the tasks organized in this edition, which are: Punctuation prediction from conversational language, Abbreviation disambiguation and Passage Retrieval. We also discuss the datasets prepared for each of the tasks, evaluation metrics chosen to rank the submissions and also sum up the approaches chosen by the participants to tackle the tasks.**

I. Introduction

POLEVAL[1] [14] is a SemEval-inspired evaluation campaign for natural language processing tools for Polish. Submitted tools compete against one another within certain tasks selected by organizers, using available data and are evaluated according to pre-established procedures.

The 2022/2023 edition of Poleval was the sixth event in a series of challenges organized since 2017. During this edition three tasks have been proposed:

1) Punctuation prediction from conversational language,
2) Abbreviation disambiguation,
3) Passage Retrieval.

The participants of this edition have been very active, as we have received more than 400 submissions from 23 teams. The submissions were made through our evaluation platform[2], which has been introduced last year.

In the following part of the paper we describe each of the tasks in detail, present the datasets created for the particular challenges, discuss the evaluation metrics and we give the overview of submissions made by the participants.

[1]http://poleval.pl
[2]https://beta.poleval.pl

II. Task 1: Punctuation prediction from conversational language

A. Problem statement

Speech transcripts generated by Automatic Speech Recognition (ASR) systems typically do not contain any punctuation or capitalization. In longer stretches of automatically recognized speech, lack of punctuation affects the general clarity of the output text [24]. The primary purpose of punctuation restoration (PR), punctuation prediction (PP), and capitalization restoration (CR) as a distinct natural language processing (NLP) task is to improve the legibility of ASR-generated text and possibly other types of texts without punctuation. For the purposes of this task, we define PR as restoration of originally available punctuation from read speech transcripts (which was the goal of a separate task in the PolEval 2021 competition) [10] and PP as prediction of possible punctuation in transcripts of spoken/ conversational language. Aside from their intrinsic value, PR, PP, and CR may improve the performance of other NLP aspects such as Named Entity Recognition (NER), part-of-speech (POS), and semantic parsing or spoken dialog segmentation [5], [12].

One of the challenges of developing PP models for conversational language is the availability of consistently annotated datasets. The very nature of naturally-occurring spoken language makes it difficult to identify exact phrase and sentence boundaries [21], [23], which means that dedicated guidelines are required to train and evaluate punctuation models.

The goal of the present task is to provide a solution for predicting punctuation in the test set collated for this task.

B. Task description

The workflow of this task is illustrated in Figure 1 below. Given raw ASR output, the task is to predict punctuation in annotated ASR transcripts of conversational speech.

C. Dataset

The test set consisted of time-aligned ASR dialogue transcriptions from three sources:

1) CBIZ, a subset of DiaBiz [17], a corpus of phone-based customer support line dialogs[3]
2) VC, a subset of transcribed video-communicator recordings, which are included in the SpokesBiz Corpus[4]
3) SPOKES, a subset of the SpokesMix corpus [16].

Table I below summarizes the size of the three subsets in terms of dialogs, words and duration of recordings.

TABLE I
OVERALL STATISTICS OF THE CORPUS

Subset	Corpus and license	Files	Words	Audio (s)	Speakers
CBIZ	DiaBiz (CC-BY-SA-NC-ND)	69	36 250	16 916	14
VC	Video conversations (CC-BY-NC)	8	44 656	17 123	20
Spokes	Casual conversations (CC-BY-NC)	13	42 730	20 583	19

The full dataset has been split into three subsets as summarized in Table II below.

TABLE II
TRAINING / DEVELOPMENT / TEST SET STATISTICS

Set	Files	Words	Audio (s)	License
Train	69	98 095	44 030	CC-BY-SA-NC-ND
Dev	11	12 563	4 718	CC-BY-NC
Test	10	12 978	5 874	CC-BY-NC

The punctuation annotation guidelines were developed in the CLARIN-BIZ project by Karasińska et al. [20].

Participants are encouraged to use both text-based and speech-derived features to identify punctuation symbols (e.g. multimodal framework [22] or to predict casing along with punctuation [15]. We allow using the punctuation dataset available at http://2021.poleval.pl/tasks/task1 [10].

The punctuation marks evaluated as part of the task are listed in Table III below. Blanks are marked as spaces. The distribution of explicit punctuation symbols in the training and development portion of the dataset provided is shown in Tables III–VI.

1) Data format: We provide two types of data: text and audio data. Text data is provided in the TSV format. For Audio data we provide audio files encoded in WAV and transcripts with force-aligned timestamps. The audio files can be downloaded separately from the website of PolEval.

[3]https://clarin-pl.eu/dspace/handle/11321/887
[4]http://docs.pelcra.pl/doku.php?id=spokesbiz

TABLE III
PUNCTUATION FOR RAW TEXT (ALL SUBCORPORA)

	Symbol	Mean	Median	Max	Sum
fullstop	.	111.15	59	1 157	8 892
comma	,	161.51	69	1 738	12 921
question_mark	?	24.36	11	229	1 949
exclamation_mark	!	3.46	4	45	277
hyphen	-	0.64	25	50	51
ellipsis	...	63.28	11	1 833	5 062
words		1 383.23	569	16 528	110 658

TABLE IV
PUNCTUATION FOR RAW TEXT (CBIZ)

	Symbol	Mean	Median	Max	Sum
fullstop	.	58.06	54	213	3 600
comma	,	70.61	59	388	4 378
question_mark	?	11.26	10	35	698
exclamation_mark	!	0.34	1	5	21
hyphen	-	0.02	1	1	1
ellipsis	...	12.29	9	54	762
words		528.74	483	2 180	32 782

TABLE V
PUNCTUATION FOR RAW TEXT (VC)

	Symbol	Mean	Median	Max	Sum
fullstop	.	411.86	384	1 157	2 883
comma	,	737.86	577	1 738	5 165
question_mark	?	85.29	41	229	597
exclamation_mark	!	10.43	5	43	73
hyphen	-	/	/	/	/
ellipsis	...	514.00	365	1 833	3 598
words		5 704.14	4 398	9 469	39 929

TABLE VI
PUNCTUATION FOR RAW TEXT (SPOKES)

	Symbol	Mean	Median	Max	Sum
fullstop	.	219.00	193	607	2 409
comma	,	307.09	313	614	3 378
question_mark	?	59.45	39	150	654
exclamation_mark	!	16.64	10	45	183
hyphen	-	4.55	50	50	50
ellipsis	...	63.82	45	186	702
words		3 449.73	1 966	16 528	37 947

2) Transcriptions and metadata: The datasets are encoded in the TSV format.

Field descriptions:

- column 1: name of the audio file
- column 2: unique segment id
- column 3: segment text, where each word is separated by a single space

The segment text (column 3) format is:

- single word text:word start timestamp in ms-word end timestamp in ms

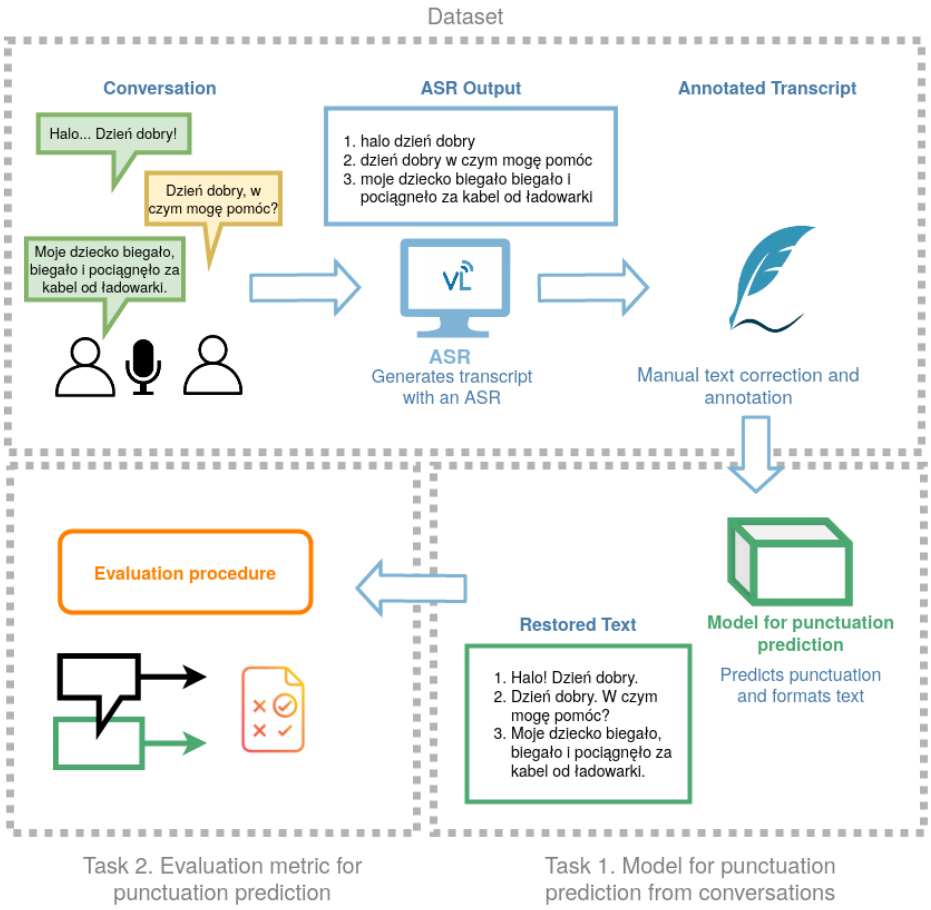

Fig. 1. Overview of the punctuation prediction task

D. Evaluation

a) Submission format: Results were to be submitted as plain text file, where each line corresponds to a single segment. The text should include the predicted punctuation marks.

1) Metrics: The final results were evaluated in terms of precision, recall, and F1 scores for predicting each punctuation mark separately. Submissions were compared with respect to the weighted average of F1 scores for each punctuation sign. The method of evaluation was similar to the one used in a PolEval 2021 task named "Punctuation restoration from read text"[5] [10].

2) Per-document score::

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$F_1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{3}$$

[5]http://2021.poleval.pl/tasks/task1

3) Global score per punctuation sign p::

$$P_p = avg_{micro} Precision(p) = \frac{\sum_{d \in Documents} TP}{\sum_{d \in Documents} TP + FP}$$

$$R_p = avg_{micro} Recall(p) = \frac{\sum_{d \in Documents} TP}{\sum_{d \in Documents} TP + FN}$$

The final scoring metric was calculated as the weighted average of global scores:

$$\frac{1}{N} \sum_{p \in Punctuation} support(p) * avg_{micro} F_1(p)$$

TABLE VII
SUBMISSIONS TO THE PUNCTUATION PREDICTION TASK

Submission	Weighted-F1 score	
	Test-A	Test-B
Oskar Bujacz	79.24	**83.30**
Michał Pogoda	80.47	**82.33**
Jakub Pokrywka	67.30	**71.44**
Filip Graliński	30.88	**35.30**

E. Results

The winning solution submitted for Task 1 by Oskar Bujacz achieved a weighted F-measure of 83.3 (see Table VII). The author used a token classifier based on the largest variant of the HerBERT model[11] with customized output postprocessing rules.

III. Task 2: Abbreviation disambiguation

A. Problem statement

Abbreviations are often overlooked in many NLP pipelines. However, they are still an important point to tackle, especially in such applications as machine translation, named entity recognition, or text-to-speech systems.

There are at least two practical challenges in processing abbreviations. The first is the ability to find the full, expanded dictionary form of an abbreviation. In many cases, this may be done by a simple dictionary lookup, but: - the use of abbreviations is often unconventional and there is no complete list of all possible abbreviation uses, - many of the abbreviations are ambiguous. That is, the same abbreviation may have more than one meaning, translating to possibly different expanded forms.

As in many other NLP tasks, the disambiguation of abbreviations needs to include context and additional language knowledge to be feasible.

The second challenge, which is specific to languages with rich morphology, such as Polish, is the necessity to produce the expanded form of an abbreviation in correct grammatical form, in concordance with the rest of the sentence.

B. Task description

The task aimed to propose a method of disambiguating Polish abbreviations. The method should recognize if a given phrase is an abbreviation and, if so, produce its expanded form, both base, and inflected ones.

C. Dataset

1) Training data: In this task a (relatively small) training dataset was provided (see example in Figure 2), which included:

- the abbreviation
- an expanded form of the abbreviation
- a base form of the abbreviation
- context of the abbreviation, with the `''` placeholder marking the place where the abbreviation appeared.

The participants were encouraged to collect and use additional training and dictionary data and to publish it after the competition.

2) Test data: The test data consists of only the abbreviation and the context. The system aims to provide the expanded and base forms of the abbreviation.

D. Evaluation

We will calculate two measures of accuracy for each provided submission:

- Af — the accuracy of provided expanded forms of abbreviations (case insensitive string match)
- Ab — the accuracy of provided base forms of abbreviations (case insensitive string match).

Based on these measures, the final score will be calculated using a weighted average:

$$Acc = 0.25 * Af + 0.75 * Ab \tag{4}$$

E. Results

We received five submissions (see Table VIII). The final ranking was calculated based on the weighted accuracy of the Test-B dataset. The scores ranged from 92.01 to 19.09. Krzysztof Wróbel obtained the highest score of 92.01.

TABLE VIII
Submissions to the abbreviation disambiguation task

Submission	Weighted accuracy	
	Test-A	Test-B
Krzysztof Wróbel	92.76	**92.01**
Jakub Karbowski	91.75	**91.27**
Marek Kozlowski	89.00	**88.73**
Jakub Pokrywka	65.48	**66.25**
Rafał Prońko	n/a	**19.09**

Krzysztof Wróbel utilized an ensemble of three models, each based on the byt5-base model[6], trained on different seeds, and employing a majority voting. The training of these models incorporated both the train and dev datasets, as well as a small dataset automatically generated from abbreviations sourced from various dictionaries such as Morfeusz [9], sjp.pl, and Wiktionary [7].

Jakub Karbowski (2nd place submission) trained a sequence-to-sequence model based on the plt5-base model[8]. The input to the model consisted of a context with a masked abbreviation, a target base form, and inflected forms of the expanded abbreviation. The initial training was performed on a synthetic dataset generated from the Polish Wikipedia. The dataset was created by randomly selecting contexts of varying lengths and shortening consecutive words using one of several strategies, such as using the first few letters, the first and last letters, or the first, middle, and last letters. The base form was generated using Spacy[9]. Then, the model was fine-tuned on the PolEval dataset.

IV. Task 3: Passage retrieval

A. Problem statement

Passage Retrieval is a crucial part of modern open-domain question-answering systems that rely on precise and efficient

[6]https://huggingface.co/google/byt5-base
[7]https://www.wiktionary.org
[8]https://huggingface.co/allegro/plt5-base
[9]https://spacy.io

Abbr	Expanded form	Base form	Context
s.	sobota	sobota	Karpaty Siepraw (n. 16). IV liga, grupa wschodnia: Olimpia Wojnicz - Grybovia (16), Orkan Szczyrzyc - Wolania Wola Rzędzińska (s. 16), Sandecja II Nowy Sącz -
d.	dawniej	dawniej	poinformowała w piątek na swej stronie internetowej rosyjska korporacja państwowa Rostech (Rostechnologii). Nie podano daty przechwycenia amerykańskiego drona. „Dron MQ-5B,"
n.	niedziela	niedziela	11) Gościbia - Piast (s. 16) Wiślanka - Sęp (s. 16); Skawinka - pauza Sęp - Gościbia (16.30) Piast - Hejnał (n. 17) Orzeł - Jordan (n. 17) Czarni - Nadwiślanka (s.
pkt. proc.	punktu procentowego	punkt procentowy	proc. Kolejne 0,12 pkt. proc. wynika ze spadku popytu na polski eksport, a 0,08 z zaburzeń na rynku wewnętrznym" - oszacowali.
rp.pl.	rp.pl.	rp.pl.	Jutro rozpocznie się proces posła ruchu Palikota - dowiedziała się Biedroń został oskarżony o naruszenie nietykalności cielesnej funkcjonariusza policji

Fig. 2. Examples from the training dataset

retrieval components to find passages containing correct answers. Traditionally, lexical methods, such as TF-IDF or BM25 [18], have been used to power retrieval systems. They are fast, interpretable, and don't require training (and therefore a training set). However, they can only return a document if it contains a keyword present in a query. In addition, their understanding of text is limited because they ignore word order.

Recently, neural retrieval systems (e.g. Dense Passage Retrieval [8]) have surpassed these traditional methods by fine-tuning pre-trained language models on a large number of (query, document) pairs. They solve the aforementioned problems of lexical methods but at the cost of the need to label training sets and poor generalisation to other domains. As a result, in a zero-shot setup (i.e. no training set), lexical methods are still competitive or even better than neural models.

B. Task description

The aim of the *passage retrieval* task was to develop a system for cross-domain question-answering retrieval. For each test question, the system should retrieve an ordered list of the ten most relevant passages (i.e. containing the answer) from the given corpus. The system is evaluated on the basis of its performance on test examples from three different domains, namely trivia, law, and customer support.

C. Dataset

1) Training set: The training set consisted of 5,000 trivia questions from the PolQA dataset [19]. Each question was accompanied by up to five passages from Polish Wikipedia containing the answer to the question. In total, the training set consisted of 16,389 question-passage pairs. In addition, we provided a Wikipedia corpus of 7,097,322 passages. The raw Wikipedia dump was parsed with WIKIEXTRACTOR[10] and split into passages at the end of paragraphs or if the passage was longer than 500 characters.

2) Test sets: The systems were evaluated on three test sets with questions from different domains. The first dataset consisted of 1,291 trivia questions similar to those in the training set.

The second dataset consisted of 900 questions and 921 passages related to the large Polish e-commerce platform - Allegro[11]. The dataset was created based on help articles and lists of frequently asked questions available on the Allegro website. Each question-passage pair was manually checked and edited where necessary.

The third dataset contained over 700 legal questions. It was created by randomly selecting the passage and manually writing a question. We also provided a corpus of approximately 26,000 passages extracted from over a thousand acts of laws published between 1993 and 2004.

D. Evaluation

The submitted systems were evaluated using Normalised Discounted Cumulative Gain for the top 10 most relevant passages [7, NDCG@10], where the score of each relevant passage depends on its position in descending order:

$$\mathrm{DCG}_p = \sum_{i=1}^{p} \frac{rel_i}{\log_2(i+1)} \tag{5}$$

$$\mathrm{IDCG}_p = \sum_{i=1}^{|REL_p|} \frac{rel_i}{\log_2(i+1)} \tag{6}$$

$$\mathrm{NDCG}_p = \frac{\mathrm{DCG}_p}{\mathrm{IDCG}_p} \tag{7}$$

where rel_i is the relevance of the i-th passage and REL_p is the list of relevant passages ordered by their relevance.

E. Results

Seven teams submitted a final solution to the task (see Table IX). All systems followed a similar architecture. First, the retriever was used to find the top N most relevant passages, and then the ranker scored these passages in order of importance to select the final 10 most relevant passages. Below are brief descriptions of the submitted systems, starting with the highest scoring ones.

Jakub Pokrywka implemented a retriever using the BM25 algorithm with text stemming using Polimorf.[12] To improve

[10]https://github.com/attardi/wikiextractor

[11]https://allegro.pl/

[12]https://github.com/dzieciou/pystempel

TABLE IX
COMPARISON OF PASSAGE RETRIEVAL TASK SUBMISSIONS

Submission	Retriever	Ranker	External Datasets	Model per domain	NDCG@10
Jakub Pokrywka	BM25	mt5-3B, mt5-13B, custom	No	Yes	69.36
Marek Kozlowski	Hybrid	mt5-13B	Yes	No	68.19
Konrad Wojtasik	Hybrid	mt5-13B, custom	Yes	No	67.44
Norbert Ropiak	Hybrid	MiniLM-L12, mDeBERTa	No	Yes	63.27
Anna Pacanowska	BM25	MiniLM-L6, custom	No	No	54.23
Maciej Kazuła	BM25	MiniLM-L6	Yes	No	51.78
Daniel Karaś	Hybrid	mBERT	No	No	51.71

the ranking of answers, separate rankers were used for each domain. For the Allegro and legal domains, an ensemble of mt5-3B[13] and mt5-13B[14] models was used, considering a pool of 1,500 candidates. Conversely, for trivia domain, Jakub Pokrywka also used the mt5-3B model but it was supplemented by a custom-trained cross-encoder models, mDeBERTa[15], and mmarco-mMiniLMv2-L12-H384-v1[16]. For trivia domain, the system included 3,000 candidate passages for effective ranking.

Marek Kozlowski used a system consisting of three retrievers: a lexical retriever (BM25) and two neural retrievers based on roberta-base[17] and roberta-large[18] [3]. The BM25 retriever used the ElasticSearch engine with the Morfologik analyser[19] for lemmatisation. For the neural encoders, fine-tuning the Roberta models involved using the MultipleNegativeRankingLoss loss function, large batch sizes and training data consisting of a mixture of Poleval training and translated MSMARCO [13] data sets. After retrieval, a re-ranking step was performed, with the mt5-13B model yielding the best results.

Konrad Wojtasik used an ensemble of several retrieval algorithms, starting with the BM25 algorithm, followed by various multilingual retrievers such as mContriever [6], mDPR [1] and LaBSE [4]. To further reduce the number of passages for reranking, he trained the plT5-large model [2] on the translated MSMARCO dataset. The final ranking was performed with mT5-13B on about 350 candidate passages from different sources.

Norbert Ropiak used both lexical (BM25) and neural retrievers (mContriever) and combined the results of both for further processing. He used ms-marco-MiniLM-L-12-v2[20] and mDeBERTa cross-encoders for ranking.

Anna Pacanowska's solution was a combination of several models. First, BM25 was used on lemmatised text to retrieve 1,000 candidate passages. Various statistics were calculated on these candidates, such as BM25 on unlemmatised data or on bigrams. The retrieved passages were then translated into English using OPUS-MT[21], which allowed the English MiniLM-L6 cross-encoder[22] to be used to calculate various scores, including those on raw question/passage pairs and on pairs with answers generated using GPT-3. Finally, logistic regression was used to combine all the results into a final score.

Maciej Kazuła used the BM25 passage retrieval algorithm together with the word inflection dictionary[23] to normalise the text. He fine-tuned the MiniLM-L6 cross-encoder for the ranking process. The cross-encoder was trained on the translated MSMARCO Polish dataset. A new tokeniser was created on the Poleval dataset, as well as on the translated MSMARCO data, in order to better represent Polish words in terms of word forms.

Daniel Karaś used two retrievers, a lexical search using BM25 and neural search using a slightly fine-tuned MiniLM-v6[24] model. Both retrievers were used to find approximately 1,000 candidates per question, except for Allegro where all passages were selected. In a second step, all candidate passages were fed into the mBERT[25], which was used without any additional training.

F. Summary

All submitted systems used the BM25 algorithm as a retriever, but differed in the way they normalised the text. Many lemmatised the passages, while others favoured stemming or using a dictionary of different word forms. In addition, some teams also used the neural retrievers and combined the candidates from these two approaches.

Given a pool of retrieved candidate passages, the systems used different methods to sort them and select the most relevant ones. The most popular were the cross-encoders, either trained on the multilingual data or fine-tuned by the contestants on the Polish examples. Most teams ensembled several models to achieve better performance.

[13] https://hf.co/unicamp-dl/mt5-3B-mmarco-en-pt
[14] https://hf.co/unicamp-dl/mt5-13b-mmarco-100k
[15] https://hf.co/cross-encoder/mmarco-mdeberta-v3-base-5negs-v1
[16] https://hf.co/cross-encoder/mmarco-mMiniLMv2-L12-H384-v1
[17] https://huggingface.co/sdadas/polish-roberta-base-v2
[18] https://huggingface.co/sdadas/polish-roberta-large-v2
[19] https://github.com/allegro/elasticsearch-analysis-morfologik
[20] https://hf.co/cross-encoder/ms-marco-MiniLM-L-12-v2
[21] https://huggingface.co/Helsinki-NLP/opus-mt-pl-en
[22] https://huggingface.co/cross-encoder/ms-marco-MiniLM-L-6-v2
[23] https://sjp.pl
[24] sentence-transformer/multi-qa-MiniLM-L6-cos-v1
[25] amberoad/bert-multilingual-passage-reranking-msmarcoreranker

Three teams used external datasets to train their models. In all cases, they automatically translated the MSMARCO dataset into Polish.

Although the goal of the task was to create a system for cross-domain passage retrieval, it was allowed to submit different systems for different domains. Three participants chose this approach, including the winning system.

Regarding the results, it is observed that the performance of the systems was very much dependent on the ranker. The first three systems that achieved the results in the range of 67-69 NDCG points used a very large mt5-13B model as the reranker. The fourth model which achieved 63 points, used MiniLM-L12 and mDeBERTa. The last three models scoring 51-54 points used only MiniLM-L6 or multilingual BERT (with the exception of Anna Pacanowska's system, which also utilized a custom model). It seems that the retriever did not play an important role in the task, since the best system used only BM25 model. It is also interesting to observe that none of the systems used a learning-to-rank approach. One of the deficiencies of the evaluation is the lack of consideration for the computational heaviness of the approaches, which might be considered in the future incarnations of this task.

V. Conclusions and Future Plans

As each year we observe a growing interest in the PolEval challenge (the number of submissions and participating teams is growing), we plan to continue our efforts to identify new tasks, which are current and interesting in the research area of NLP and Polish language. The next editions will be specifically interesting, considering the current developments in the area of generative AI and language models.

We also plan to organize the datasets created for all the editions of the challenge in a repository to facilitate their distribution and encourage other researchers to use them for their work.

Acknowledgments

This work was supported by the European Regional Development Fund as a part of 2014–2020 Smart Growth Operational Programme, CLARIN — Common Language Resources and Technology Infrastructure, project no. POIR.04.02.00-00C002/19, investment in the CLARIN-PL research infrastructure funded by the Polish Ministry of Science and Higher Education. We gratefully acknowledge Poland's high-performance computing infrastructure PLGrid (HPC Centers: ACK Cyfronet AGH) for providing computer facilities and support within computational grant no. PLG/2023/016304.

References

[1] Asai, A., Yu, X., Kasai, J., Hajishirzi, H.: One question answering model for many languages with cross-lingual dense passage retrieval. In: NeurIPS (2021)

[2] Chrabrowa, A., Dragan, Ł., Grzegorczyk, K., Kajtoch, D., Koszowski, M., Mroczkowski, R., Rybak, P.: Evaluation of transfer learning for Polish with a text-to-text model. In: Proceedings of the Thirteenth Language Resources and Evaluation Conference. pp. 4374–4394. European Language Resources Association, Marseille, France (Jun 2022), https://aclanthology.org/2022.lrec-1.466

[3] Dadas, S., Perełkiewicz, M., Poświata, R.: Pre-training polish transformer-based language models at scale. In: Artificial Intelligence and Soft Computing. pp. 301–314. Springer International Publishing (2020)

[4] Feng, F., Yang, Y., Cer, D., Arivazhagan, N., Wang, W.: Language-agnostic BERT sentence embedding. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 878–891. Association for Computational Linguistics, Dublin, Ireland (May 2022). https://doi.org/10.18653/v1/2022.acl-long.62, https://aclanthology.org/2022.acl-long.62

[5] Hlubík, P., Španěl, M., Boháč, M., Weingartová, L.: Inserting Punctuation to ASR Output in a Real-Time Production Environment. In: Sojka, P., Kopeček, I., Pala, K., Horák, A. (eds.) Text, Speech, and Dialogue. pp. 418–425. Springer International Publishing, Cham (2020)

[6] Izacard, G., Caron, M., Hosseini, L., Riedel, S., Bojanowski, P., Joulin, A., Grave, E.: Unsupervised dense information retrieval with contrastive learning (2021). https://doi.org/10.48550/ARXIV.2112.09118, https://arxiv.org/abs/2112.09118

[7] Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of ir techniques. ACM Trans. Inf. Syst. **20**, 422–446 (10 2002). https://doi.org/10.1145/582415.582418

[8] Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., Yih, W.t.: 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. Association for Computational Linguistics, Online (Nov 2020). https://doi.org/10.18653/v1/2020.emnlp-main.550, https://aclanthology.org/2020.emnlp-main.550

[9] Kieraś, W., Woliński, M.: Morfeusz 2 – analizator i generator fleksyjny dla języka polskiego. Język Polski **XCVII**(1), 75–83 (2017)

[10] Mikołajczyk, A., Wawrzyński, A., Pęzik, P., Adamczyk, M., Kaczmarek, A., Janowski, W.: PolEval 2021 Task 1: Punctuation Restoration from Read Text. In: Ogrodniczuk, M., Kobyliński, Ł. (eds.) Proceedings of the PolEval 2021 Workshop. pp. 21–31. Institute of Computer Science, Polish Academy of Sciences, Warsaw (2021)

[11] 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 (Apr 2021), https://www.aclweb.org/anthology/2021.bsnlp-1.1

[12] Nguyen, T.B., Nguyen, Q.M., Nguyen, T.T.H., Do, Q.T., Luong, C.M.: Improving Vietnamese Named Entity Recognition from Speech Using Word Capitalization and Punctuation Recovery Models. In: Proceedings of Interspeech 2020. pp. 4263–4267 (2020). https://doi.org/10.21437/Interspeech.2020-1896, http://dx.doi.org/10.21437/Interspeech.2020-1896

[13] Nguyen, T., Rosenberg, M., Song, X., Gao, J., Tiwary, S., Majumder, R., Deng, L.: MS MARCO: A Human Generated MAchine Reading COmprehension Dataset (November 2016), https://www.microsoft.com/en-us/research/publication/ms-marco-human-generated-machine-reading-comprehension-dataset/

[14] Ogrodniczuk, M., Kobyliński, Ł. (eds.): Proceedings of the PolEval 2021 Workshop. Institute of Computer Science, Polish Academy of Sciences, Warsaw (2021)

[15] Pappagari, R., Żelasko, P., Mikołajczyk, A., Pęzik, P., Dehak, N.: Joint Prediction of Truecasing and Punctuation for Conversational Speech in Low-Resource Scenarios. In: 2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU). pp. 1185–1191 (2021). https://doi.org/10.1109/ASRU51503.2021.9687976

[16] Pęzik, P.: Spokes – a Search and Exploration Service for Conversational Corpus Data. In: Linköping Electronic Conference Proceedings. Selected Papers from CLARIN 2014. pp. 99–109. Linköping University Electronic Press (2015)

[17] Pęzik, P., Krawentek, G., Karasińska, S., Wilk, P., Rybińska, P., Cichosz, A., Peljak-Łapińska, A., Deckert, M., Adamczyk, M.: DiaBiz – an Annotated Corpus of Polish Call Center Dialogs. In: Proceedings of the Language Resources and Evaluation Conference. pp. 723–726. European Language Resources Association, Marseille, France (Jun 2022), https://aclanthology.org/2022.lrec-1.76

[18] Robertson, S.E., Zaragoza, H.: The probabilistic relevance framework: Bm25 and beyond. Found. Trends Inf. Retr. **3**, 333–389 (2009)

[19] Rybak, P., Przybyła, P., Ogrodniczuk, M.: Improving question answering performance through manual annotation: Costs, benefits and strate-

gies (2022). https://doi.org/10.48550/ARXIV.2212.08897, https://arxiv.org/abs/2212.08897
[20] S., K., Cichosz, A., M., A., P., P.: Evaluating Punctuation Prediction in Conversational Language (2023), Forthcoming
[21] Sirts, K., Peekman, K.: Evaluating Sentence Segmentation and Word Tokenization Systems on Estonian Web Texts. In: Utka, A., Vaičenonienė, J., Kovalevskaitė, J., Kalinauskaitė, D. (eds.) Human Language Technologies – The Baltic Perspective. Frontiers in Artificial Intelligence and Applications, vol. 328, pp. 174–181 (2020). https://doi.org/10.3233/FAIA200620
[22] Sunkara, M., Ronanki, S., Bekal, D., Bodapati, S., Kirchhoff, K.: Multimodal Semi-supervised Learning Framework for Punctuation Prediction in Conversational Speech. In: Proceedings of Interspeech 2020. pp. 4911–4915 (2020). https://doi.org/10.21437/Interspeech.2020-3074, http://dx.doi.org/10.21437/Interspeech.2020-3074
[23] Wang, X.: Analysis of Sentence Boundary of the Host's Spoken Language Based on Semantic Orientation Pointwise Mutual Information Algorithm. In: Proceedings of the 12th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA). pp. 501–506 (2020). https://doi.org/10.1109/ICMTMA50254.2020.00114
[24] Yi, J., Tao, J., Bai, Y., Tian, Z., Fan, C.: Adversarial Transfer Learning for Punctuation Restoration (2020)