Proceedings of the PolEval 2024 Workshop

Maciej Ogrodniczuk, Łukasz Kobyliński (eds.)

Institute of Computer Science, Polish Academy of Sciences
Warszawa, 2024

ISBN 978-83-63159-33-7

Jana Kazimierza 5
01-248 Warszawa

Warszawa 2024

Contents

PolEval 2024

Łukasz Kobyliński, Maciej Ogrodniczuk
(Institute of Computer Science, Polish Academy of Sciences)

PolEval is an evaluation campaign focused on Natural Language Processing tasks for Polish, intended to promote research on language and speech technologies, create objective evaluation procedures and improve state-of-the-art.

In 2024 the systems competed in the following tasks:

- Task 1: Reading comprehension
- Task 2: Emotion and sentiment recognition
- Task 3: Polish Automatic Speech Recognition Challenge

This year, we received over 300 submissions from 11 different teams. The high submissions-to-teams ratio suggests we should consider introducing an upper limit on submissions in next year's challenge to prevent it from being used as a strategy by participants.

This volume consists of proceedings of the online workshop session organized during the Natural Language Processing seminar[1] on December 2, 2024, presenting the results of the 2024 edition of the shared task.[2]

The main part, as previously, contains three sections dedicated to evaluation tasks. Each section starts with a paper by task organizers, describing the task, its dataset and evaluation procedures, and summarising the submissions and the results. Then selected papers by authors of the solutions are presented.

Thank you for being with us again!

[1] https://zil.ipipan.waw.pl/seminar
[2] http://2024.poleval.pl

Organizers

Concept and administrative issues

Maciej Ogrodniczuk (Institute of Computer Science, Polish Academy of Sciences)
Łukasz Kobyliński (Institute of Computer Science, Polish Academy of Sciences / Sages)

Evaluation platform

Filip Graliński (Adam Mickiewicz University in Poznań)
Karol Saputa (Institute of Computer Science, Polish Academy of Sciences)

Task 1: Reading comprehension

Ryszard Tuora (Institute of Computer Science, Polish Academy of Sciences)

Task 2: Emotion and sentiment recognition

Jan Kocoń (Wrocław University of Science and Technology)
Bartłomiej Koptyra (Wrocław University of Science and Technology)

Task 3: Polish Automatic Speech Recognition Challenge

Michał Junczyk (Adam Mickiewicz University in Poznań)
Iwona Christop (Adam Mickiewicz University in Poznań)

PolEval 2024 Task 1: Reading Comprehension

Ryszard Tuora
(Institute of Computer Science, Polish Academy of Sciences)

Abstract

Automatic question answering is an important facet of contemporary Natural Language Processing. This text describes this year's Reading Comprehension challenge, based on the Polish Question Answering Dataset (PoQuAD).

Keywords

question answering, reading comprehension, retrieval augmented generation

1. Introduction

Question Answering is an important and broad field of research within Natural Language Processing. Previous edition of PolEval (Kobyliński et al. 2023) included the passage retrieval task, crucial in narrowing down the range of documents relevant to a human question, but only after including a reading comprehension system, can the whole process of answering a question be fully automated.

Classical systems for text comprehension relied on span extraction, but this is a limiting technique: it does not work well for morphologically rich languages (such as Polish) and does not fit well with answering yes-no questions. More importantly, it limits the complexity of reasoning that might be used to answer the question (e.g. excluding comparative queries). Recent advances in large, generative language models show that free-form answer generation is feasible even in closed-book, zero-shot scenarios. However these solutions often suffer from a tendency for hallucination, and so recent years showed an explosion of interest in Retrieval Augmented Generation (RAG — Lewis et al. (2020)) systems, where the answer is generated on the basis of an independently retrieved context.

Nevertheless, aligning these models with more precise task definitions is still challenging[1], and traditional supervised learning paradigms continue to outperform few-shot systems[2]. Moreover, the problem of hallucinations is still common[3]. For these reasons working with classical machine comprehension datasets (whether for evaluation, or supervised fine-tuning) is still an important pursuit.

2. Task definition

The goal of the task was to develop a system for open-domain machine reading comprehension for the purposes of question answering. A system was given a question with a paired passage. Some of the questions were "impossible", i.e. they were relevant to the passage, but the passage contained no answer. Others could be answered based on the passage and had gold answers listed. The gold answers *did not* have to be identical with any span from the text, though it usually was the case, and in a large majority of cases, were very close to some fragment from the text (with the notable exception of yes/no questions).

The participant was given:

1. The training set consisting of 11 624 passages, each associated with a list of questions. In total, there were 56 618 questions in the train subset.
2. The development set which could be used to evaluate the system, or as additional training data. It contained 1 453 paragraphs with 7 060 questions.

Each system was tested on two separate test sets (A and B). For each test pair (context + question), the system was supposed to generate an answer based on the information given in the context. Each model was scored on two separate metrics:

1. Text similarity as assessed by Levenshtein edit distance[4], calculated for lowercased strings, normalized by length of the longer sequence, measured only on the answerable subset of the questions. This score measured the power of the system to answer questions.
2. Binary F1 score on the classification with respect to answerability. This score measured the capacity to recognise when then information is sufficient, and when to abstain from answering.

[1] One of the common problems are the tendency for LLMs to be verbose, and frequent difficulties in subsequent automatic processing of their outputs.

[2] Compare the 81.8 performance on SQuAD 2.0 reported by Llama 3 70B creators (Dubey et al. 2024), with low 90's SOTA achieved by fine-tuned models.

[3] Producing answers inconsistent with the provided context is a pervasive problem sometimes referred to under the name of *faithfulness hallucinations* (Huang et al. 2023).

[4] This is a fairly simplistic metric, which does not take into account any semantic factors in evaluating answer correctness. Contemporary systems are often evaluated by LLM's (e.g. in the popular **RAGAS** framework `https://github.com/explodinggradients/ragas`) which allows to take semantics into account, but is much less interpretable and consistent.

The two scores were averaged with equal weights to generate the final score, which was used to select the winner. The scores for each individual test set were aggregated with weights equal to their proportion of the total number of examples.

Participants were free to use any publicly available datasets to develop their systems. It was forbidden to manually label the test examples.

3. Dataset

All the subsets came from the same source (Polish Wikipedia), which was annotated as part of the CLARIN-BIZ initiative, to form the PoQuAD dataset (Tuora et al. 2022, 2023). The original inspiration for the PoQuAD dataset stems from the — SQuAD (Rajpurkar et al. 2016, 2018) and it's subsequent incarnations in other languages (e.g. Heinrich et al. (2022)). All subsets used in the competition exhibited a similar distribution of data.

The training data largely followed the SQuAD JSON format and were available at `https://huggingface.co/datasets/clarin-pl/poquad/tree/main`. The test dataset was made available separately and followed the same format. The dataset was divided into articles, and for each article there was **at least one** and up to two annotated paragraphs. Each paragraph could contain up to five questions.

The following listing shows a sample paragraph with one question:

```
{
 "id": 9773,
 "title": "Miszna",
 "summary": "Miszna (hebr.  miszna „nauczać", „ustnie przekazywać",
    „studiować", „badać", od  szana „powtarzać", „różnić się", „być
    odmiennym"; jid. Miszne) - w judaizmie uporządkowany zbiór tekstów
    ustnego prawa uzupełniający Torę (Prawo pisane)...",
 "url": "https://pl.wikipedia.org/wiki/Miszna",
 "paragraphs": [
   {
     "context": "Pisma rabiniczne - w tym Miszna - stanowią kompilację
        poglądów różnych rabinów na określony temat...",
     "qas": [
       {
         "question": "W jakich formach występowała Tora przekazana
           Mojżeszowi?",
         "answers": [
           {
             "text": "pisanej, a drugą część w formie ustnej",
             "answer_start": 210,
             "answer_end": 248,
             "generative_answer": "pisanej, ustnej"
           }
         ],
         "is_impossible": false
       }
     ]
   }
 }
```

The gold answer for the question was given by the “generative_answer” key. The “answer” key corresponded to the extractive answer and was **not** taken into account during the evaluation process. However, we decided to leave this as an additional layer of information which can be exploited for improving the systems (although the extractive answer was not available for the system in inference time).

In cases where the question was impossible, the annotation took the form of e.g.:

```
{
  "question": "Kto napisał Torę?",
  "plausible_answers": [
    {
      "text": "Boga",
      "answer_start": 150,
      "answer_end": 154,
      "generative_answer": "Bóg"
    }
  ],
  "is_impossible": true
}
```

The answers listed under the “plausible_answers” key were to be treated as distractors, **not** as gold answers. The train and development sections of the dataset were made available at https://huggingface.co/datasets/clarin-pl/poquad/tree/main.

4. Evaluation

The predictions were evaluated using the evaluation script, which calculated normalized Levenshtein and F1 and averaged them.

The in.tsv file contained one question identifier per line, with each question identified by a unique ID in the following format:

```
<article_id>_<paragraph_number>_<question_number>
```

The paragraph and question numbers were 0-indexed.

Submissions were supposed to be written to the two out.tsv files (one in test-A, second in test-B), with each answer in a new line. The order of answers had to match the order of questions in the in.tsv file. For example, the answer to the question on the fifth line of the in.tsv file should be on the fifth line of the out.tsv file.

A perfect solution should exactly match the contents of the expected.tsv file. Examples of the input and expected output formats were included in the `in.tsv` and `expected.tsv` files in the `train` and `dev-0` folders.

5. Baseline

We propose two baseline systems. One is a few-shot GPT 3.5, with a paragraph from the development set listed as example. The other is a fine-tuned plT5 encoder-decoder architecture. Outputs from both are parsed to obtain both the answerability of a given question in the context, and the answer itself (if it is deemed to be possible). When evaluated on the test set, they achieved the scores listed in Table 1.

Table 1: Scores of the systems evaluated on the test set

	Normalized Levenshtein	**F1**	**Score**
GPT 3.5 few shot	67.25	48.20	57.73
plT5 baseline	83.25	57.67	70.46

6. Submission and results

The task received only a single submission by Krzysztof Wróbel, which leaves little room for the comparative approach to understanding the challenges involved in the Question Answering Task. The scores obtained by the winner (on the combined test set) are listed in Table 2.

Table 2: The scores of the winning solution

	Normalized Levenshtein	**F1**	**Score**
Krzysztof Wróbel	84.78	81.89	83.34

Krzysztof Wróbel's solution leverages LLM's (specifically one of the Bielik family of models) finetuned with the parameter efficient LoRA adapters. They are significantly larger than the plT5 model used as a baseline, although the number of parameters used for fine-tuning was significantly smaller. Still it is interesting to observe that the increase in the actual power of answering questions (as measured by the Levenshtein metric) is not big. Instead the main improvement comes in the task of evaluating whether the question is answerable with the supplied context, where an increase of over 23 pp. is observed. This latter gain is particularly commendable, considering how important the ability to refrain from answering is, in modern RAG systems.

Acknowledgements

Hereby I would like to extend my gratitude to Aleksandra Zwierzchowska, Natalia Zawadzka-Paluektau, Cezary Klamra and Łukasz Kobyliński, for their work on preparing, and experimenting with the PoQuAD dataset, Maciej Ogrodniczuk for his help in organizing the project, Piotr Rybak for his insights and help in preparing the GPT baseline, as well as the annotators whose diligent work made it possible to obtain the data. Also, I would like to congratulate the winner of the task: Krzysztof Wróbel.

References

Dubey A., Jauhri A., Pandey A., Kadian A., Al-Dahle A. et al. (2024). *The Llama 3 Herd of Models*. arXiv:2407.21783.

Heinrich Q., Viaud G. and Belblidia W. (2022). *FQuAD2.0: French Question Answering and Learning When You Don't Know*. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pp. 2205–2214, Marseille, France. European Language Resources Association.

Huang L., Yu W., Ma W., Zhong W., Feng Z., Wang H., Chen Q., Peng W., Feng X., Qin B. and Liu T. (2023). *A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions*. arXiv:2311.05232.

Kobyliński Ł., Ogrodniczuk M., Rybak P., Przybyła P., Pęzik P., Mikołajczyk A., Janowski W., Marcińczuk M. and Smywiński-Pohl A. (2023). *PolEval 2022/23 Challenge Tasks and Results*. In Ganzha M., Maciaszek L., Paprzycki M. and Ślęzak D. (eds.), *Proceedings of the 18th Conference on Computer Science and Intelligence Systems*, vol. 35 of *Annals of Computer Science and Information Systems*, pp. 1237–1244.

Lewis P., Perez E., Piktus A., Petroni F., Karpukhin V., Goyal N., Küttler H., Lewis M., Yih W.-t., Rocktäschel T., Riedel S. and Kiela D. (2020). *Retrieval-augmented Generation for Knowledge-intensive NLP Tasks*. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Rajpurkar P., Zhang J., Lopyrev K. and Liang P. (2016). *SQuAD: 100,000+ Questions for Machine Comprehension of Text*. In Su J., Duh K. and Carreras X. (eds.), *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392, Austin, Texas. Association for Computational Linguistics.

Rajpurkar P., Jia R. and Liang P. (2018). *Know What You Don't Know: Unanswerable Questions for SQuAD*. In Gurevych I. and Miyao Y. (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 784–789, Melbourne, Australia. Association for Computational Linguistics.

Tuora R., Zawadzka-Paluektau N., Klamra C., Zwierzchowska A. and Kobyliński Ł. (2022). *Towards a Polish Question Answering Dataset (PoQuAD)*. In Tseng Y.-H., Katsurai M. and Nguyen

H. N. (eds.), *From Born-Physical to Born-Virtual: Augmenting Intelligence in Digital Libraries*, pp. 194–203, Cham. Springer International Publishing.

Tuora R., Zwierzchowska A., Zawadzka-Paluektau N., Klamra C. and Kobyliński L. (2023). *PoQuAD – The Polish Question Answering Dataset – Description and Analysis*. In *Proceedings of the 12th Knowledge Capture Conference 2023*, K-CAP ’23, pp. 105–113, New York, NY, USA. Association for Computing Machinery.

Optimizing LLMs for Polish Reading Comprehension: A Comparative Study of Ensemble and Unified Approaches

Krzysztof Wróbel
(Enelpol, Jagiellonian University, SpeakLeash)

Abstract

This paper presents our approach to the PolEval 2024 Task 1 on Polish language reading comprehension, utilizing state-of-the-art Large Language Models (LLMs). We developed a system that effectively handles both answer generation and answerability classification by leveraging decoder-only models. Our solution addresses key challenges including processing long contexts exceeding typical model limitations and identifying questions that cannot be answered from the given text. The system achieves strong performance on both the Levenshtein score for answer quality and the F1 score for answerability classification.

Keywords

question answering, reading comprehension, Large Language Models, Polish, natural language processing

1. Introduction

Reading comprehension is a fundamental natural language processing task that evaluates a system's ability to understand text passages and answer questions about them. This paper presents our approach to the PolEval 2024 Task 1, which focuses on Polish language reading comprehension using state-of-the-art Large Language Models (LLMs). The task presents unique challenges, including handling long contexts and identifying questions that cannot be answered from the given text.

Our solution leverages recent advances in LLM architectures and efficient fine-tuning techniques to develop a robust reading comprehension system for Polish text. We demonstrate how decoder-only models, combined with Low-Rank Adaptation (LoRA) and Supervised Fine-Tuning (SFT), can effectively process lengthy contexts while maintaining high performance on both answer generation and answerability classification.

To facilitate reproducibility and further research in Polish language reading comprehension, we have made our source code and models publicly available[1].

2. Data

The dataset utilized in this study was derived from Polish Wikipedia articles and annotated through the CLARIN-BIZ initiative, resulting in the creation of the PoQuAD (Polish Question Answering Dataset; Tuora et al. 2022). The data distribution remains consistent across all subsets.

The dataset architecture follows the established SQuAD (Rajpurkar et al. 2016, 2018) format, with training data publicly accessible and test data distributed separately. The corpus is structured hierarchically, with articles containing one to two annotated paragraphs. Each paragraph is associated with a maximum of five questions, maintaining a consistent annotation schema throughout the dataset.

Table 1: Dataset statistics showing total number of examples and number of non-answerable questions per split. The average and maximum lengths are measured in tokens.

Split	**Total**	**Non-answerable**	**Average token length**	**Max token length**
train	56 618	10 431 (18.42%)	409.24	3 427
dev	7 060	1 296 (18.36%)	412.40	1 598
test-A	3 501	–	399.26	2 036
test-B	3 585	–	410.20	2 237

As shown in Table 1, the dataset contains a significant portion of non-answerable questions, comprising approximately 18% of both training and development sets. The distribution of non-answerable questions in test sets is unknown. The contexts are relatively long, with an average length of around 410 tokens (using Bielik-11B tokenizer) across all splits. Some examples contain contexts of over 3 000 tokens, which significantly exceeds the typical context window limitations of transformer models (512–1024 tokens). This poses a particular challenge for model architecture selection and implementation.

3. Evaluation

The evaluation of the reading comprehension system is based on two key metrics that assess different aspects of model performance:

[1] `https://github.com/enelpol/poleval2024-task1`

- **Answerability Score**: This metric evaluates the model's ability to determine whether a question can be answered based on the given context. It is calculated as a binary F1 score, where the positive class represents questions that are not answerable. This measures the model's capability to identify questions that cannot be answered from the provided context.
- **Levenshtein Score**: For questions that are determined to be answerable, this metric assesses the quality of the generated answers. It is computed using the Levenshtein edit distance between the predicted and ground truth answers, after converting both to lowercase. The distance is normalized by the length of the longer sequence to provide a score between 0 and 1, where higher values indicate better performance.

The final performance metric is calculated as the arithmetic mean of the Answerability and Levenshtein scores, providing a balanced assessment of the system's ability to both identify answerable questions and generate accurate responses.

4. Methods

In addressing the reading comprehension task, we carefully considered several model architectures commonly used for Polish language processing. Encoder-only models such as HerBERT (Mroczkowski et al. 2021), Polish RoBERTa (Dadas et al. 2020), and XLM-RoBERTa (Conneau et al. 2020) were initially taken into consideration. However, these models have a context length limitation of 512 tokens, making them unsuitable for processing longer passages in our dataset. Additionally, encoder-only architectures would only be capable of handling the Answerability classification subtask, not the full question-answering requirement.

We also examined encoder-decoder architectures like plT5 (Chrabrowa et al. 2022) and Polish BART (Dadas 2019). While these models can generate text responses, they still face context length constraints of 512 and 1024 tokens, which is insufficient for many examples in our dataset.

Given these limitations, we opted for a decoder-only Large Language Model (LLM) architecture, which can process the full length of our queries and contexts. To efficiently train the model while maintaining performance, we employed two key techniques:

- Low-Rank Adaptation (LoRA) (Hu et al. 2022), which reduces the number of trainable parameters by decomposing weight updates into low-rank matrices, enabling efficient fine-tuning while preserving model quality.
- Supervised Fine-Tuning (SFT), where we train the model on carefully curated question-answer pairs to improve its performance on both the answerability classification and answer generation tasks.

Our implementation addresses both the answerability classification and answer generation through two approaches:

- Using a CausalLM head for joint prediction of both answerability and answer generation through fine-tuning. The CausalLM head is a language modeling head that predicts the next token in a sequence, allowing the model to generate free-form text responses that can include both the answerability classification and the answer itself in a natural language format.

- Implementing a dedicated SequenceClassification head specifically for the answerability task, which can be used in conjunction with or separately from the answer generation component. The SequenceClassification head adds a linear layer on top of the model's final hidden state to output classification probabilities, making it well-suited for the binary answerability prediction task.

This architectural choice, combined with our training approach using LoRA and SFT, provides the flexibility to handle both aspects of the task while accommodating the full context length requirements of our dataset, all while maintaining computational efficiency during training.

5. Experiments

Our experimental approach focused on evaluating various Large Language Models (LLMs) for the reading comprehension task. The models were selected based on their 5-shot performance metrics from the Open PL LLM Leaderboard (Wróbel et al. 2024), particularly considering their capabilities in Retrieval-Augmented Generation (RAG) tasks (Tuora et al. 2022, Rybak et al. 2024).

We designed three distinct prompting strategies to evaluate different aspects of the models' performance:

1. A binary classification prompt to determine if the context contains an answer to the question:

   ```
   Tytuł: {title}\n
   Kontekst: {context}\n
   Pytanie: {question}\n
   Czy kontekst jest relewantny dla pytania?\n
   Odpowiedź:
   ```

2. A direct question-answering prompt that generates an answer based on the provided context:

   ```
   Kontekst: {context}n
   Pytanie: {question}\n
   Odpowiedz krótko i zwięźle na powyższe pytanie.\n
   Odpowiedź:
   ```

3. A conditional answering prompt that generates an answer only if the context contains relevant information:

   ```
   Tytuł: {title}\n
   Kontekst: {context}\n
   Pytanie: {question}\n
   Jeśli kontekst zawiera odpowiedź na powyższe pytanie
   ```

```
to odpowiedz krótko i zwięźle, a jeśli kontekst nie zawiera
odpowiedzi to napisz: "Brak informacji".\n
Odpowiedź:
```

During our initial experiments, we observed that most models tended to generate verbose responses in complete sentences, rather than providing concise answers. To address this, we implemented a simple post-processing step that truncates responses beginning with "yes" or "no" to improve answer precision and maintain consistency with the task requirements.

Table 2 presents the performance of various LLM models on the Answerability task using prompts 1 and 3. The results demonstrate that larger models generally achieved superior performance, with models like Llama-3.1-405B-Instruct-FP8 and Mistral-Large-Instruct-2407 showing particularly strong results. The prompt engineering proved to be a crucial factor in model performance, highlighting the importance of careful prompt design in extracting optimal results from these models.

Table 2: Performance comparison of different LLM models for Answerability task. Results show Precision, Recall and F1 scores for prompt 1 and prompt 3.

Model	**Prompt 1**			**Prompt 3**		
	Precision	**Recall**	**F1**	**Precision**	**Recall**	**F1**
Llama-3.1-405B-Instruct-FP8	87.08	50.46	63.90	78.26	64.74	70.86
Mistral-Large-Instruct-2407	80.67	52.16	63.36	70.42	67.05	68.70
Qwen2.5-72B-Instruct	77.58	50.46	61.15	75.93	60.11	67.10
Qwen2-72B-Instruct	59.74	57.72	58.71	64.96	56.94	60.69
Mixtral-8x22B-Instruct-v0.1	60.13	49.23	54.14	78.53	48.53	59.99
Llama-3-70B-Instruct	80.56	44.14	57.03	81.47	43.75	56.93
Qwen2-72B	56.51	58.26	57.37	64.00	43.75	51.97
Openchat-3.5-0106-gemma	79.02	25.00	37.98	65.87	42.75	51.85
Llama-3.1-70B-Instruct	75.48	51.77	61.42	83.99	34.41	48.82
Bielik-11B-v2.0-Instruct	56.84	53.86	55.31	71.65	21.45	33.02
Llama-3-70B	72.17	36.42	48.41	59.50	14.74	23.62
Bielik-7B-Instruct-v0.1	35.00	24.85	29.06	28.10	9.80	14.53

A notable pattern emerged across most models where precision consistently exceeded recall scores. This indicates that the models were more likely to incorrectly claim that a context contained an answer than to incorrectly state that a context lacked an answer. This bias towards positive predictions suggests that the models may be overly optimistic in their assessment of answer presence, which could have important implications for real-world applications where high confidence in answer availability is crucial.

Table 3 presents the performance comparison of various LLM models using two different prompting approaches. The "Prompt 2 with Oracle" column represents an idealized scenario where we leverage perfect answerability detection (an oracle) to only request answers for questions that are known to be answerable from the given context. This serves as an upper bound for model performance by eliminating errors that arise from incorrect answerability

Table 3: Performance comparison of LLM models using different prompting strategies. "Prompt 2 with Oracle" represents an idealized scenario where the model is only asked to generate answers for questions that are known to be answerable based on perfect answerability detection. Prompt 3 shows results for the conditional answering approach where the model must both determine answerability and generate answers. Scores are Levenshtein-based similarity metrics between generated and reference answers.

Model	**Parameters**	**Prompt 2 with Oracle**	**Prompt 3**
Llama-3.1-405B-Instruct-FP8	405B	83.87	81.26
Qwen2-72B	72B	83.59	78.69
Mistral-Large-Instruct-2407	123B	82.04	76.84
Qwen2-72B-Instruct	72B	81.83	75.50
Mixtral-8x22B-Instruct-v0.1	141B	81.64	79.25
Llama-3-70B	70B	81.44	80.29
Bielik-11B-v2.0-Instruct	11B	81.16	78.77
Qwen2.5-72B-Instruct	72B	80.91	77.88
Llama-3-70B-Instruct	70B	80.83	79.33
Llama-3.1-70B-Instruct	70B	80.51	79.89
Openchat-3.5-0106-gemma	7B	78.98	75.37
Bielik-7B-Instruct-v0.1	7B	70.62	63.07

assessment. The "Prompt 3" column shows results for the more realistic conditional answering approach, where models must both determine if a question is answerable and generate an appropriate response.

For our experiments, we selected Bielik-11B (Ociepa et al. 2024a,c,b) as the most promising Polish language model that could be fine-tuned on a single GPU. This model represents a good balance between performance and computational requirements for the reading comprehension task.

In our initial experiments, we found that using approximately one-third of the available training data for CausalLM head was sufficient to achieve optimal performance. Additional training beyond this point did not yield improved results. Based on this observation, we decided not to pursue further training with additional datasets (e.g. PolQA (Rybak et al. 2024)), as the model appeared to reach its learning capacity with the existing data. During the fine-tuning process, we deliberately chose not to use any chat template formatting, focusing instead on different parameters that could improve the model's performance.

Through extensive experimentation, we identified several key factors that influenced model performance. The inclusion of document titles in the input context generally yielded modest improvements in model scores. However, incorporating additional context elements such as summaries or their first sentences consistently led to performance degradation. This suggests that concise, relevant context is more beneficial than additional potentially noisy information.

For models addressing the answerability task, we found that dataset balancing had varying effects depending on the model architecture. Specifically, balancing was crucial for models using the CausalLM head but showed minimal impact on those employing the SequenceClassification head.

Table 4: Comparison of training parameters for the three model variants. Model 1 focuses on answer generation for answerable questions only, Model 2 specializes in answerability classification, and Model 3 represents a balanced approach handling both tasks.

Parameter	**Model 1**	**Model 2**	**Model 3**
Learning rate	2e-4	2e-4	1e-4
Scheduler type	Linear	Linear	Linear
Warmup	None	None	None
Batch size	16	16	16
Max steps	1 000	7 000	3 500
Eval steps	100	500	100
LoRA r	16	16	16
LoRA alpha	16	16	16
Max sequence length	1 024	1 024	1 024
Head type	CausalLM	SequenceClassification	CausalLM
Data subset	Answerable only	Original	Original
Title included	No	Yes	Yes

We conducted comprehensive testing across multiple versions of Bielik-11B (v2.0-Instruct, v2.1-Instruct, v2.2-Instruct, v2.3-Instruct). Notably, version v2.0-Instruct consistently demonstrated superior performance compared to later versions, suggesting that certain architectural or training modifications in subsequent versions may have impacted the model's capability for this specific task.

Our hyperparameter optimization efforts explored several key dimensions:

- Learning rate: 1e-5 to 5e-4
- LoRA rank (r): 8, 16, 32, 64
- NEFTune activation
- Effective batch size: 16, 32
- Context enrichment strategies (titles and summaries)
- Training duration through max steps adjustment

However, we observed significant performance variability across different random seeds and training runs, making it challenging to draw definitive conclusions about optimal hyperparameter settings. This variability suggests that model performance may be sensitive to initialization conditions and training dynamics, highlighting the need for robust evaluation across multiple training runs. Given that each training run took between 4 to 10 hours on our available hardware, our ability to conduct extensive hyperparameter optimization was constrained by computational resources.

6. PolEval submissions

For the PolEval competition, we submitted two distinct approaches to evaluate their relative effectiveness:

- An ensemble system combining two specialized models: one optimized for answerability classification and another for answer generation. This approach leverages the complementary strengths of both models to achieve more robust performance.
- A single unified model trained to handle both answerability classification and answer generation tasks simultaneously, offering a more streamlined solution with reduced computational overhead during inference.

7. Results

Table 5 presents a comprehensive comparison of model performance across three key metrics: Answerability classification accuracy, Levenshtein similarity score for answer generation, and the final combined score. The ensemble approach consistently demonstrates superior performance across all evaluation sets, achieving the highest scores in both development and test scenarios. Notably, while the single model shows competitive performance, particularly in Levenshtein scores, it falls short in answerability classification. The plT5 baseline and GPT 3.5 few-shot results, provided by competition organizers, show interesting patterns - while plT5 exhibits strong performance in answer generation (Levenshtein score: 83.25), comparable to our best models, its answerability classification capabilities are significantly lower. This suggests potential opportunities for hybrid approaches combining the strengths of different architectures.

Table 5: Detailed evaluation results across different model configurations and test sets. The metrics include Answerability accuracy (Ans.), Levenshtein similarity score (Lev.), and the final combined Score. The ensemble approach (2 models) consistently outperforms other configurations across all evaluation sets, with best scores highlighted in bold. GPT-3.5 few-shot and plT5 baseline results are shown for dev-0 set comparison.

	dev-0			**test-A**			**test-B**		
	Ans.	**Lev.**	**Score**	**Ans.**	**Lev.**	**Score**	**Ans.**	**Lev.**	**Score**
2 models	81.73	84.42	**83.07**	81.44	86.08	**83.76**	82.33	83.52	**82.92**
1 model	77.44	83.42	80.43	79.09	85.64	82.36	77.94	82.24	80.09
GPT 3.5 few shot	48.20	67.25	57.73	–	–	–	–	–	–
plT5 baseline	57.67	83.25	70.46	–	–	–	–	–	–

Beyond the overall accuracy metrics, it is crucial to examine the precision and recall of the answerability classification, as these metrics have cascading effects on the system's performance.

Precision in answerability classification is particularly important because false positives (incorrectly classifying an answerable question as unanswerable) lead to the model not generating answers for questions that should be answered, which inevitably results in lower Levenshtein scores. Our ensemble approach achieved a precision of 84.03 and recall of 79.60 on the dev-0 set. When using an oracle classifier for answerability (perfect classification), our answer generation model achieves a Levenshtein score of 87.15, indicating significant headroom for improvement through better answerability classification.

The ensemble approach's superior performance can be attributed to three primary advantages:

1. **Task Specialization**: Each model optimizes for a single task - either answerability classification or answer generation - without parameter sharing compromises.
2. **Independent Optimization**: Separate training allows for task-specific hyperparameter tuning and optimization strategies.
3. **Complementary Architectures**: Each model can utilize architectures and loss functions specifically designed for its task, rather than compromising with a one-size-fits-all approach.

This specialization particularly benefits the answerability classification task, as evidenced by the wider performance gap in classification metrics (81.73 vs 77.44 on dev-0) compared to generation metrics.

While the ensemble approach achieves superior performance, it comes with notable trade-offs:

- **Computational Cost**: Running two separate models requires approximately twice the computational resources during inference compared to the unified approach.
- **Storage Requirements**: Maintaining separate models increases storage requirements, though this can be mitigated by using LoRA adapters which require only a small fraction of the full model's parameter space.
- **Deployment Complexity**: The ensemble system requires coordinating two separate models and managing their interactions, increasing operational complexity.

The unified model, while showing slightly lower performance (2-3% decrease in overall score), offers advantages in resource efficiency and deployment simplicity. This trade-off becomes particularly relevant in resource-constrained environments or high-throughput applications where computational efficiency is paramount.

8. Error analysis

To better understand the limitations and failure modes of our models, we conducted a manual evaluation of 20 errors made by the system, separately for Answerability and Levenshtein, by randomly choosing 20 errors. For the Answerability task, out of these, 10 errors were deemed valid, while the remaining 10 contained various issues:

- **Question Error**: In some cases, the question itself contained errors. For example, the question "Czy kiedykolwiek ORP Zwinny przeszedł remont grawitacyjny?" should have been "gwarancyjny" instead of "grawitacyjny".
- **Unclear Question**: Some questions were found to be unclear or ambiguous. An example is "Jakie wyłącznie gazety funkcjonują na terenie Szczecina?", which was difficult to interpret.
- **Imprecise Question**: There were instances where the question lacked necessary context or specificity. For example, "Jak nazywał się prezydent Syrii, z którym się przyjaźnił?" did not provide enough information about who or what was being referred to.
- **Different Assessment**: In some cases, our assessment differed from the reference evaluation. For example, "Po publikacji 11 maja 2019 filmu Tylko nie mów nikomu, pod adresem księdza Makulskiego padły oskarżenia o kontakty seksualne z osobami małoletnimi. Bohaterem jakiego filmu został Eugeniusz Makulski?".

This analysis highlights the importance of question clarity and precision in reading comprehension tasks. Addressing these issues could further improve the performance and reliability of our models.

For the answer generation task, our manual analysis of 20 randomly selected errors revealed that most responses were semantically correct but contained minor linguistic variations from the reference answers, as shown in Table 6. The most common differences included:

- Grammatical inflection variations in Polish words
- Presence or absence of prepositions
- Different number formats (numerical vs written form)
- Use of Roman vs Arabic numerals
- Incomplete personal names (surnames without given first names)
- Presence or absence of quotation marks and other punctuation

These variations, while affecting the Levenshtein score, did not impact the factual correctness of the answers, suggesting that our evaluation metric may be overly sensitive to superficial linguistic differences rather than semantic accuracy.

In a detailed analysis of 20 examples with Levenshtein ratio lower than 0.5, we found that 16 of them (80%) were semantically correct despite their low similarity scores. This finding suggests that the Levenshtein similarity metric, while useful for standardized comparison, may not fully capture the nuanced nature of Polish language variations and answer correctness.

9. Conclusions

Our work demonstrates the effectiveness of both ensemble and unified approaches in tackling the reading comprehension task. The ensemble system, combining specialized models for answerability classification and answer generation, consistently outperformed other configurations across all evaluation metrics. However, the single unified model showed promising results while offering reduced computational overhead during inference.

Several promising directions for future work emerge from our findings. First, we plan to modify the output format of the unified model to explicitly separate the answerability classification and answer generation steps, potentially improving the model's decision-making process. Second, we aim to explore joint training of CausalLM and SequenceClassification outputs, seeking to match the performance of separately trained models while maintaining the efficiency advantage of single inference.

Additionally, we see potential in enhancing the few-shot learning capabilities of our models by developing more sophisticated methods for identifying and utilizing relevant examples from the training set. This could improve the model's ability to handle novel questions and contexts while maintaining computational efficiency.

Based on our error analysis findings, we recommend several improvements to enhance dataset quality in future iterations. First, questions should undergo rigorous quality control to eliminate linguistic errors and ambiguity. Second, questions lacking sufficient context should be revised to include necessary references. Third, the answer evaluation criteria should be standardized to handle acceptable variations in Polish inflections, number formats, and name completeness. Finally, we suggest implementing a semantic similarity metric alongside the Levenshtein distance to better capture answer correctness. These refinements would lead to more reliable model evaluation and improved training data quality.

These future directions aim to bridge the performance gap between ensemble and unified approaches while optimizing for both accuracy and computational efficiency.

Acknowledgements

We gratefully acknowledge Poland's high-performance Infrastructure PLGrid ACK Cyfronet AGH for providing computer facilities and support within computational grant no PLG/2024/016951.

References

Chrabrowa A., Dragan Ł., Grzegorczyk K., Kajtoch D., Koszowski M., Mroczkowski R. and Rybak P. (2022). *Evaluation of Transfer Learning for Polish with a Text-to-Text Model*. In Calzolari N., Béchet F., Blache P., Choukri K., Cieri C., Declerck T., Goggi S., Isahara H., Maegaard B., Mariani J., Mazo H., Odijk J. and Piperidis S. (eds.), *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pp. 4374–4394, Marseille, France. European Language Resources Association.

Conneau A., Khandelwal K., Goyal N., Chaudhary V., Wenzek G., Guzmán F., Grave E., Ott M., Zettlemoyer L. and Stoyanov V. (2020). *Unsupervised Cross-lingual Representation Learning at*

Scale. In Jurafsky D., Chai J., Schluter N. and Tetreault J. (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8440–8451, Online. Association for Computational Linguistics.

Dadas S. (2019). *A Repository of Polish NLP Resources*. Github.

Dadas S., Perełkiewicz M. and Poświata R. (2020). *Pre-training Polish Transformer-based Language Models at Scale*. In *Artificial Intelligence and Soft Computing*, pp. 301–314. Springer International Publishing.

Hu E. J., Shen Y., Wallis P., Allen-Zhu Z., Li Y., Wang S., Wang L. and Chen W. (2022). *LoRA: Low-Rank Adaptation of Large Language Models*. In *Proceedings of the 10th International Conference on Learning Representations (ICLR 2022)*.

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.

Ociepa K., Flis Ł., Kinas R., Gwoździej A., Wróbel K., SpeakLeash Team and Cyfronet Team (2024a). *Bielik-11B-v2.0-Instruct model card*.

Ociepa K., Flis Ł., Kinas R., Gwoździej A. and Wróbel K. (2024b). *Bielik 2: A Family of Large Language Models for the Polish Language — Development, Insights, and Evaluation*. Manuscript in preparation.

Ociepa K., Flis Ł., Wróbel K., Gwoździej A. and Kinas R. (2024c). *Bielik 7B v0.1: A Polish Language Model – Development, Insights, and Evaluation*. arXiv:2410.18565.

Rajpurkar P., Zhang J., Lopyrev K. and Liang P. (2016). *SQuAD: 100,000+ Questions for Machine Comprehension of Text*. In Su J., Duh K. and Carreras X. (eds.), *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392, Austin, Texas. Association for Computational Linguistics.

Rajpurkar P., Jia R. and Liang P. (2018). *Know What You Don't Know: Unanswerable Questions for SQuAD*. In Gurevych I. and Miyao Y. (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 784–789, Melbourne, Australia. Association for Computational Linguistics.

Rybak P., Przybyła P. and Ogrodniczuk M. (2024). *PolQA: Polish Question Answering Dataset*. In Calzolari N., Kan M.-Y., Hoste V., Lenci A., Sakti S. and Xue N. (eds.), *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pp. 12846–12855, Torino, Italia. ELRA and ICCL.

Tuora R., Zawadzka-Paluektau N., Klamra C., Zwierzchowska A. and Kobyliński Ł. (2022). *Towards a Polish Question Answering Dataset (PoQuAD)*. In Tseng Y.-H., Katsurai M. and Nguyen H. N. (eds.), *From Born-Physical to Born-Virtual: Augmenting Intelligence in Digital Libraries*, pp. 194–203, Cham. Springer International Publishing.

Wróbel K., SpeakLeash Team and Cyfronet Team (2024). *Open PL LLM Leaderboard*.

Appendix

Table 6 presents a detailed comparison between model predictions and reference answers, highlighting various types of linguistic variations that affected the Levenshtein similarity scores.

Table 6: Examples of model predictions with their Levenshtein similarity scores (Lev. column, ranging from 0 to 1 where higher values indicate greater similarity) compared to reference answers. The examples illustrate various types of differences including preposition usage, inflection variations, and answer completeness.

Lev.	Question	Reference answer	Predicted answer
0.98	Dlaczego młode tego gatunku ...?	gdyż **mają** za słabo wykształconą warstwę tłuszczową, by przeżyć w wodach Arktyki lub Antarktyki	gdyż **ma** za słabo wykształconą warstwę tłuszczową, by przeżyć w wodach Arktyki lub Antarktyki
0.97	Na co wskazuje ... kryzys religii ...?	religia nie przetrwa jeśli nie współgra z elementarnym rozumnym postrzeganiem świata	**że** religia nie przetrwa jeśli nie współgra z elementarnym rozumnym postrzeganiem świata
0.96	W jakich krajach ...?	**w** Austrii, Finlandii, Izraelu, Holandii i Hiszpanii	Austrii, Finlandii, Izraelu, Holandii i Hiszpanii
0.95	W jakich miejscach ...?	**w** organizacjach, instytucjach i firmach	organizacjach, instytucjach i firmach
0.94	Jaką teorię na temat tkanki ... posiadał ...?	powstaje ona z metaplazji nabłonka otrzewnej	**że** powstaje ona z metaplazji nabłonka otrzewnej
0.91	Czego wyrazem było ...?	**podkreślić rolę** Alego w swoich wierzeniach	**podkreślenia roli** Alego w swoich wierzeniach
0.91	Jakie przebudowy ... miały miejsce ...?	obmurowano krużganki i zmieniono wnętrza zamku, rozebrano część budynków (m.in. kościoły św. Jerzego i św. Michała)	obmurowano krużganki i zmieniono wnętrza zamku, rozebrano część budynków (m.in. kościoły św. Jerzego i św **[reached max tokens]**)
0.90	W meczu z jakim klubem ...?	**z** Rakowem Częstochowa	Rakowem Częstochowa
0.90	W jakim kraju ...?	**W** Jordanii	**w** Jordanii
0.80	Jaki statek...?	Civilian	„Civilian"
0.87	Od czego zależy ...?	**od** występowania deszczy	występowania deszczy
0.80	Z jakiego powodu ...?	**Powodem wojny miała być** polityka Ozeasza, króla Izraela, który odmówił płacenia trybutu Asyrii i sprzymierzył się z Egiptem	polityka Ozeasza, króla Izraela, który odmówił płacenia trybutu Asyrii i sprzymierzył się z Egiptem
0.71	O ile zmniejsza się ...?	**o** 18,4%	18,40%
0.66	Jakim typem państw były ...?	teokratyczny**ch** islamski**ch państw**	teokratyczny**mi** islamski**mi**
0.57	Kto kierował ...?	**lejtnant** W. Moczulski	W. Moczulski
0.55	Kiedy Polska ...?	30 września 1938 **o godz 23:45**	30 września 1938

Lev.	Question	Reference answer	Predicted answer
0.53	W jakim celu ...?	**Na cele** wystawowe	wystawowe
0.48	Ile razy dziennie ...?	dwa lub trzy **razy na dobę**	dwa lub trzy
0.29	W jaki sposób ...?	rozpad budynku nie postąpił dalej **i reszta wieżowca, choć też uszkodzona, „osiadła" na gruzach zmiażdżonych pięter**	rozpad budynku nie postąpił dalej
0.11	W której lidze ...?	**w** I **lidze**	I

Figure 1 shows the distribution of Levenshtein similarity scores between predicted and reference answers across our test set.

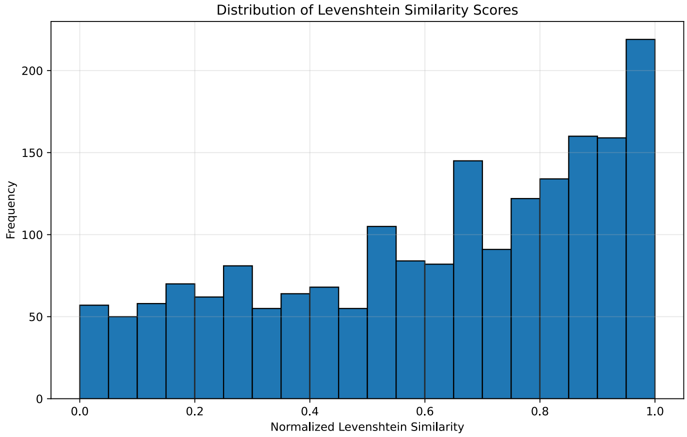

Figure 1: Distribution of Levenshtein similarity scores between predicted and reference answers.

PolEval 2024 Task 2: Emotion and Sentiment Recognition

Jan Kocoń, Bartłomiej Koptyra
(Wrocław University of Science and Technology)

Abstract

This paper presents an overview of the emotion and sentiment recognition task in PolEval 2024. The task aimed to develop systems capable of identifying emotions from Plutchik's wheel of emotions (e.g., joy, trust, anticipation, etc.) and sentiments (positive, negative, neutral, or ambivalent) in Polish consumer reviews across four domains: hotels, medicine, products, and school. Annotated by six independent annotators, the dataset captured the emotional and sentiment nuances at both the sentence and review levels. Participants were challenged to classify emotions and sentiments using pre-defined metrics, focusing on macro-averaged F1 scores for both sentence-level and review-level annotations. This paper describes the dataset structure, annotation process, task requirements, evaluation methodology, and results, highlighting the complexity and significance of emotion recognition in Polish NLP.

Keywords

emotion recognition, sentiment analysis, natural language processing, Polish, evaluation

1. Introduction

Understanding human emotions presents a significant challenge in natural language processing (NLP). Emotions are inherently subjective, varying greatly from person to person, and are often difficult to capture accurately in written language. This complexity is further compounded by the fact that individuals frequently struggle to articulate their feelings through words fully. Additionally, interpreting emotions often demands understanding the broader context in which they are expressed. This context may include cultural, social, or situational nuances and sometimes requires external knowledge beyond what is explicitly stated in the text. Consequently, effectively analyzing and processing emotional content in text remains a nuanced and multifaceted task for NLP researchers (Miłkowski et al. 2021, Kazienko et al. 2023, Kocoń and Maziarz 2021, Janz et al. 2017, Ngo et al. 2022, Miłkowski et al. 2022, Kocoń 2023, Wierzba et al. 2021, Miłkowski et al. 2023).

Today, the challenges of understanding the structure and subtleties of language and the need for knowledge beyond the immediate context of a text are often addressed through large pre-trained models. These models are trained on vast amounts of unlabeled data and provide associative knowledge that significantly enhances performance in tasks like emotion recognition. However, this approach is far from perfect. While these models offer a strong foundation, they frequently require additional fine-tuning to adapt to the specific requirements of a given task. Despite these limitations, the gains achieved by leveraging large-scale pre-trained models represent a major step in addressing the complexities of language and emotion understanding in natural language processing.

2. Task definition

This task aimed to develop a system capable of identifying emotions based on Plutchik's wheel of emotions, along with the corresponding sentiments expressed in consumer reviews. The system was designed to function on two levels: analyzing the overall emotional tone of the entire review and detecting specific emotions and sentiments within individual sentences. This dual-level analysis ensures a comprehensive understanding of the general mood and the nuanced emotional shifts throughout the text. A key restriction on the task was the prohibition of manual labeling for the test examples.

3. Dataset

The dataset consisted of consumer reviews written in Polish, spanning four domains: hotels, medicine, products, and university. In addition to opinion-based reviews, the dataset included non-opinion, informative texts from the same domains, which were predominantly neutral in tone. Each review, along with its individual sentences, was annotated with emotions from Plutchik's wheel of emotions: joy, trust, anticipation, surprise, fear, sadness, disgust, and anger. Furthermore, perceived sentiment was labeled as positive, negative, or neutral, with ambivalent sentiment marked using both positive and negative labels.

The annotations were conducted by six independent annotators, who worked without influencing one another's decisions. A consensus-based approach was used to aggregate these annotations, wherein any label selected by at least two out of the six annotators was retained. This method allowed controversial texts and sentences to be annotated with conflicting emotions, reflecting emotion recognition's inherent complexity and subjectivity. Importantly, while each sentence received its own specific annotations, these were determined within the broader context of the entire review, ensuring that sentence-level emotions aligned with the overarching themes and sentiments of the text.

The dataset is now accessible at: `https://huggingface.co/datasets/clarin-knext/CLARIN-Emo`.

For more information about this dataset, see (Koptyra et al. 2023, Kocoń et al. 2023).

3.1. Training set

The training dataset comprised 776 consumer reviews, containing a total of 6 393 sentences. These reviews were randomly selected from the entire dataset to create a representative training sample. Importantly, the data split was performed at the review level, ensuring that no individual review was divided between the training and other subsets. This approach preserved the contextual integrity of each review, enabling the model to better learn patterns and relationships within complete texts rather than fragments. The training set served as the foundation for developing and fine-tuning the emotion and sentiment recognition system.

3.2. Test sets

The evaluation process utilized two separate test sets, each comprising 167 reviews. These reviews contained 1 234 and 1 264 annotated sentences, respectively. The test sets were carefully constructed to maintain the same level of integrity as the training set, with complete reviews included in each set rather than fragmented portions. This ensured that the system's performance could be assessed in a manner consistent with how it was trained, providing a reliable measure of its ability to recognize emotions and sentiments across both full texts and individual sentences.

3.3. Dataset format

The datasets were organized into three directories: one for the training set and two for the test sets. All datasets followed a consistent format to ensure compatibility and ease of processing.

Each input row represented an ordered sentence from a review. To mark the end of a review, a special sentence consisting solely of the symbol # was added. This placeholder did not belong to the original review and was not part of its content. Instead, it served a dual purpose: signaling the conclusion of the current review and acting as a placeholder for the annotation of the entire review. The annotation associated with this special row corresponded to the aggregated sentiment and emotional labels for the full review.

The row immediately following a # symbol marked the beginning of a new review, with its first sentence appearing in the subsequent row. This structure ensured clear boundaries between reviews and allowed for straightforward association of annotations with either individual sentences or entire reviews.

The following fragment of the training input file:

```
Była to pierwsza wizyta ale moze i ostatnia.
Lakarz troche apatyczny, nie wypowiadajacy sie jasno.
Mam zrobic jakies badanie ale nie dardzo wiem jakie.
Nie napisal skierowania/zalecenia, chyba mowil o gastrologii.
Powinnam byla byc bardzej wymagajaca i dopytujaca.
Nie polecam tego lekarza.
```

corresponds to the following annotations:

```
False False True  False False True  False False False True  False
False False False False False True  True  False False True  False
False False False True  False True  False False False True  False
False False False True  False True  False False False True  False
False False False True  False True  False True  False True  False
False False False False False True  False False False True  False
False False False True  False True  False False False True  False
```

which means that sentences were labeled as:

```
"Była to pierwsza wizyta ale moze i ostatnia."
- anticipation, sadness, negative
"Lakarz troche apatyczny, nie wypowiadajacy sie jasno."
- sadness, disgust, negative
"Mam zrobic jakies badanie ale nie dardzo wiem jakie."
- surprise, sadness, negative
"Nie napisal skierowania/zalecenia, chyba mowil o gastrologii."
- surprise, sadness, negative
"Powinnam byla byc bardzej wymagajaca i dopytujaca."
- surprise, sadness, anger, negative
"Nie polecam tego lekarza."
- sadness, negative
```

and the review as a whole, starting from ”Była to pierwsza wizyta ale moze i ostatnia.” and ending at ”Nie polecam tego lekarza.” was labeled as: `surprise,sadness,negative`.

3.4. Submission format

The goal of the task was to classify whether an emotion was contained in a text or not. Each submission was supposed to consist of a single tab-separated file. Each of the eleven columns should contain one boolean value indicating whether a specific emotion is contained or not. Each line should contain annotations relevant to the matching row from the `in.tsv` file, e.g.:

```
True  True  False False False False False False True False False
```

4. Evaluation

The final evaluation metric used was the arithmetic mean of two F1 macro scores, one of which is calculated on only the text annotations, and the other is calculated on only the sentence annotations:

$$Final\,score = \frac{F1_{macro}\,sentences + F1_{macro}\,texts}{2} \tag{1}$$

where each $F1_{macro}$ is calculated by:

$$F1_{macro} = \frac{\sum_{i=1}^{n} F1_i}{n} \tag{2}$$

where n is the number of labels, and $F1$ for each label is given by the equation:

$$F1 = 2 * \frac{precision * recall}{precision + recall} \tag{3}$$

The metric is only properly defined when $precision + recall \neq 0$. If this case is encountered, the calculated metric for that label will be set to 0.

$$precision = \frac{TP}{TP + FP} \tag{4}$$

The metric is only properly defined when $TP + FP \neq 0$ where TP and FP represent the number of true positives and false positives, respectively. If this case is encountered, the calculated metric for that label will be set to 0.

$$recall = \frac{TP}{TP + FN} \tag{5}$$

The metric is only properly defined when $TP + FN \neq 0$ where TP and FN represent the number of true positives and false negatives, respectively. If this case is encountered, the calculated metric for that label will be set to 0.

5. Submission and results

The PolEval 2024 task for Emotion and Sentiment Recognition saw a variety of innovative approaches. Results are presented in Table 1. Participants utilized state-of-the-art large language models (LLMs), ensemble techniques, and novel training methodologies to tackle the challenge. Below, we present a detailed summary of the submissions, highlighting their methodologies, results, and unique contributions to the task.

5.1. The winning submission by Krzysztof Wróbel

Methodology

The winning approach leveraged the Bielik-11B model with Low-Rank Adaptation (LoRA) and ensemble techniques. Specialized models were trained separately for sentence-level and text-level predictions. The "Test-A optimized ensemble" (Model 10) achieved the highest overall performance, combining specialized models for sentence-level and text-level tasks into a robust ensemble.

Table 1: PolEval 2024 competition results showing sentence-level (Sent.), text-level (Text), and final F1 scores for test sets A and B. Source: (Wróbel 2024).

Rank	Submitter	Entries	test-A scores			test-B scores		
			Sent.	Text	Final	Sent.	Text	Final
1	Krzysztof Wróbel	10	81.62	79.40	80.51	81.51	78.48	79.99
2	Tomasz Warzecha	196	78.87	81.54	80.20	79.34	79.28	79.31
3	Cezary Kęsik	25	74.94	76.42	75.68	76.66	79.33	77.99
4	Jakub Pokrywka	15	78.65	75.93	77.29	79.43	75.77	77.60
5	Paweł Lewkowicz	10	74.29	77.73	76.01	77.27	77.20	77.23
6	Katarzyna Baraniak	32	75.94	77.47	76.70	76.11	77.76	76.94
7	Cezary Kęsik	5	73.62	79.12	76.37	75.94	70.43	73.19
8	Jakub Kosterna	4	50.47	28.71	39.59	52.19	28.71	40.45
9	Paweł Cyrta	5	33.04	32.74	32.89	31.86	34.28	33.07

Notable observations

The ensemble effectively balanced predictions across emotion and sentiment categories, maintaining consistency between test sets. The approach showcased the advantage of model specialization and optimization for specific tasks.

5.2. Second place by Tomasz Warzecha

Methodology

This submission utilized an ensemble of three large-scale models: HerBERT, Polish RoBERTa-v2, and XLM-RoBERTa. Each model was fine-tuned with context variations to improve predictions. Separate ensembles were constructed for sentence-level and text-level predictions to address the unique characteristics of each task.

Notable observations

The approach successfully leveraged variance reduction techniques and ensemble methods, resulting in stable performance across tasks.

5.3. Cezary Kęsik's approach

Methodology

Fine-tuning focused on improving the performance of underrepresented labels. The models were initially trained on sentence-level data and extended to whole reviews to capitalize on knowledge transfer. Experiments with loss functions and samplers addressed label imbalance.

Notable observations

Although experiments with focal loss and samplers improved results for rare labels, they required careful tuning to avoid overfitting.

5.4. Jakub Pokrywka's approach

Methodology

The solution utilized the Polish RoBERTa model (`sdadas/polish-roberta-large-v2`) available on HuggingFace. The HuggingFace Transformers library was employed for fine-tuning the model for multi-label classification using the `AutoModelForSequenceClassification` API with `problem_type="multi_label_classification"`. Data was truncated to 521 tokens per input to handle the token limit. The training setup included a batch size of 8, a learning rate 2e-5, and training for up to 10 epochs with a warmup ratio of 0.1. The best checkpoint for submission was selected based on the `f1_micro_average` metric evaluated on a validation set.

The final submission employed an ensemble of three models, each trained on different train/validation splits. A 70/30 split strategy was applied, ensuring distinct full comments were randomly assigned to the validation dataset in each case. To enhance contextual understanding, the input for each target sentence included preceding and succeeding sentences from the same commentary. Tags `[COMMENT_START]` and `[COMMENT_STOP]` were used to mark the boundaries of the target sentence within the commentary.

Notable observations

Combined context inclusion and multi-split training contributed to strong and consistent performance. Using tags for target sentence marking provided a clear input structure, which may have aided in accurate predictions.

5.5. Katarzyna Baraniak's generative approach

Methodology

A generative model (Meta-Llama-3-8B) was used to create synthetic data, which was combined with the original dataset for classification. A fine-tuned Polish RoBERTa model was used for the classification task.

Notable observations

Synthetic data generation improved overall scores. However, quality control of generated samples was identified as a key area for future improvement.

5.6. Jakub Kosterna's ensemble model

Methodology

An ensemble of five traditional machine learning algorithms (e.g., Random Forest, XGBoost) was trained on numerical features derived from text representations. Features were extracted using various pretrained models specialized in Polish language tasks.

Notable observations

While the ensemble approach demonstrated foundational potential, further optimization and the use of advanced preprocessing methods are required for competitive performance.

6. Conclusions

This shared task on emotion and sentiment recognition in Polish text provided an opportunity to evaluate a diverse range of approaches, highlighting the complexities and opportunities in the field. Below, we summarize the key insights and lessons learned from the competition, which also pave the way for future research and development.

Large Language Models (LLMs)

The competition underscored the transformative role of LLMs in emotion and sentiment recognition tasks. Submissions leveraging advanced models like Bielik-11B, Meta-Llama-3-8B, and Polish RoBERTa demonstrated state-of-the-art performance. Task-specific fine-tuning significantly enhanced the models' capabilities. These results confirm the utility of LLMs not only for general text understanding but also for highly nuanced tasks requiring multi-label classification of emotions and sentiments.

Ensemble approaches

Ensemble methods proved to be a powerful strategy, particularly for mitigating class imbalance and improving robustness. By combining diverse models and leveraging their complementary strengths, ensembles consistently outperformed individual models. Techniques such as majority voting and hybrid approaches effectively captured subtle distinctions between emotion categories, improving overall F1 scores. This success highlights the importance of model diversity in ensemble design, especially for tasks with rare labels or conflicting annotations.

Synthetic data generation

The use of generative models to create synthetic training data demonstrated potential as an effective augmentation strategy. While submissions employing this technique showed notable improvements in performance, they also revealed challenges related to the quality and consistency of generated samples. This finding emphasizes the need for better quality control mechanisms in generative data pipelines, as well as further exploration of diverse generative models to maximize their benefits.

Challenges and future directions

The task brought to light key challenges in emotion and sentiment recognition, including:

- **Label imbalance**: Rare emotions like "fear" and "surprise" were underrepresented in the dataset, which adversely affected model performance on these categories. Future datasets should ensure better coverage of such labels to support balanced evaluation.
- **Evaluation metrics**: The reliance on macro-averaged F1 scores provided a fair assessment across all labels but also amplified the impact of errors in rare categories. Exploring alternative evaluation metrics or weight adjustments could provide more nuanced insights.
- **Contextual understanding**: Many submissions highlighted the importance of leveraging context in both sentence-level and text-level predictions. Developing architectures that better integrate contextual cues remains an open research direction.

Broader implications

This competition demonstrated the potential for emotion and sentiment recognition systems to support a wide array of real-world applications, from customer feedback analysis to mental health monitoring. As models and datasets continue to improve, the ability to understand human emotions in text is likely to drive significant advancements in human-computer interaction and automated decision-making.

In conclusion, the PolEval 2024 task showcased the strengths of current state-of-the-art methodologies while identifying critical areas for further improvement. The combination of LLM advancements, ensemble strategies, and innovative data augmentation approaches represents a promising path forward in the quest for robust and reliable emotion and sentiment recognition systems.

Acknowledgements

This work was financed by (1) the statutory funds of the Department of Artificial Intelligence, Wroclaw University of Science and Technology; (2) the National Science Centre, Poland, project no. 2021/41/B/ST6/04471; (3) the Polish Ministry of Education and Science within the programme “International Projects Co-Funded”; (4) the European Regional Development Fund, as a part of the 2014-2020 Smart Growth Operational Programme, projects no. POIR.04.02.00-00C002/19, POIR.01.01.01-00-0923/20, POIR.01.01.01-00-0615/21, and POIR.01.01.01-00-0288/22; (5) CLARIN ERIC – European Research Infrastructure Consortium: Common

Language Resources and Technology Infrastructure (period: 2024-2026) funded by the Polish Minister of Science under the programme: "Support for the participation of Polish scientific teams in international research infrastructure projects", agreement number 2024/WK/01; (6) the European Union under the Horizon Europe, grant no. 101086321 (OMINO). However, the views and opinions expressed are those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Executive Agency. Neither the European Union nor European Research Executive Agency can be held responsible for them.

References

Janz A., Kocon J., Piasecki M. and Zasko-Zielinska M. (2017). *plWordNet as a Basis for Large Emotive Lexicons of Polish*. Proceedings of Human Language Technologies as a Challenge for Computer Science and Linguistics Poznan: Fundacja Uniwersytetu im. Adama Mickiewicza w Poznaniu, pp. 189–193.

Kazienko P., Bielaniewicz J., Gruza M., Kanclerz K., Karanowski K., Miłkowski P. and Kocoń J. (2023). *Human-centered Neural Reasoning for Ssubjective Content Processing: Hate Speech, Emotions, and Humor*. Information Fusion, 94, p. 43–65.

Kocoń J. (2023). *Deep Emotions Across Languages: A Novel Approach for Sentiment Propagation in Multilingual WordNets*. In *2023 IEEE International Conference on Data Mining Workshops (ICDMW)*, pp. 744–749. IEEE.

Kocoń J. and Maziarz M. (2021). *Mapping WordNet onto Human Brain Connectome in Emotion Processing and Semantic Similarity Recognition*. Information Processing & Management, 58(3), p. 102530.

Kocoń J., Cichecki I., Kaszyca O., Kochanek M., Szydło D., Baran J., Bielaniewicz J., Gruza M., Janz A., Kanclerz K., Kocoń A., Koptyra B., Mieleszczenko-Kowszewicz W., Miłkowski P., Oleksy M., Piasecki M., Łukasz Radliński, Wojtasik K., Woźniak S. and Kazienko P. (2023). *ChatGPT: Jack of all trades, master of none*. Information Fusion, 99, p. 101861.

Koptyra B., Ngo A., Radliński Ł. and Kocoń J. (2023). *CLARIN-Emo: Training Emotion Recognition Models Using Human Annotation and ChatGPT*. In Mikyška J., de Mulatier C., Paszynski M., Krzhizhanovskaya V. V., Dongarra J. J. and Sloot P. M. (eds.), *Computational Science – ICCS 2023*, pp. 365–379, Cham. Springer Nature Switzerland.

Miłkowski P., Gruza M., Kanclerz K., Kazienko P., Grimling D. and Kocoń J. (2021). *Personal bias in Prediction of Emotions Elicited by Textual Opinions*. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Student Research Workshop*, pp. 248–259.

Miłkowski P., Saganowski S., Gruza M., Kazienko P., Piasecki M. and Kocoń J. (2022). *Multitask Personalized Recognition of Emotions Evoked by Textual Content*. In *2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, pp. 347–352. IEEE.

Miłkowski P., Karanowski K., Wielopolski P., Kocoń J., Kazienko P. and Zięba M. (2023). *Modeling Uncertainty in Personalized Emotion Prediction with Normalizing Flows*. In *2023 IEEE International Conference on Data Mining Workshops (ICDMW)*, pp. 757–766. IEEE.

Ngo A., Candri A., Ferdinan T., Kocon J. and Korczynski W. (2022). *StudEmo: A Non-aggregated Review Dataset for Personalized Emotion Recognition*. In Abercrombie G., Basile V., Tonelli S., Rieser V. and Uma A. (eds.), *Proceedings of the 1st Workshop on Perspectivist Approaches to NLP @ LREC2022*, pp. 46–55, Marseille, France. European Language Resources Association.

Wierzba M., Riegel M., Kocoń J., Miłkowski P., Janz A., Klessa K., Juszczyk K., Konat B., Grimling D., Piasecki M. et al. (2021). *Emotion Norms for 6000 Polish Word Meanings with a Direct Mapping to the Polish wordnNt*. Behavior Research Methods, 54, p. 1–16.

Wróbel K. (2024). *Emotion and Sentiment Recognition in Polish Texts Using Large Language Models: A Winning Approach to PolEval 2024*. In Ogrodniczuk M. and Łukasz K. (eds.), *Proceedings of the PolEval 2024 Workshop*, pp. 43–55.

Emotion and Sentiment Recognition in Polish Texts Using Large Language Models: The Winning Approach to PolEval 2024

Krzysztof Wróbel
(Enelpol, Jagiellonian University, SpeakLeash)

Abstract

This paper presents a comprehensive approach to emotion and sentiment recognition in Polish texts using Large Language Models (LLMs). We describe our winning solution for the PolEval 2024 shared task, which achieved state-of-the-art performance in classifying both emotions from Plutchik's wheel and sentiment polarity at sentence and text levels. Our methodology leverages the Bielik-11B model with Low-Rank Adaptation (LoRA) and employs various ensemble techniques to handle the challenges of multi-label classification and class imbalance. We conducted extensive experiments exploring different model architectures, training strategies, and ensemble methods, ultimately developing a system that effectively processes texts of varying lengths while maintaining robust performance across all emotion categories. Our best-performing model achieved F1 scores of 80.51% and 79.99% on the two test sets, outperforming other submissions. The study also provides valuable insights into the impact of context length, training parameters, and loss functions on model performance. Our implementation is publicly available to facilitate reproducibility and further research in Polish language emotion recognition.

Keywords

emotion recognition, sentiment analysis, natural language processing, Polish, evaluation

1. Introduction

Emotion and sentiment recognition is a critical area of natural language processing (NLP) that aims to identify and classify emotions and sentiments expressed in text. This study focuses on advancing emotion and sentiment recognition within the Polish language by leveraging large language models (LLMs). The primary objective of our research is to improve the accuracy and robustness of emotion and sentiment detection in Polish texts, thereby contributing to the broader field of NLP.

In this paper, we present a comprehensive approach that utilizes state-of-the-art LLMs to analyze and classify emotions and sentiments in Polish consumer reviews. Our methodology includes fine-tuning pre-trained models on a domain-specific dataset, implementing various ensemble techniques, and evaluating the performance of different model configurations. By making our source code and models publicly accessible[1], we aim to enhance reproducibility and encourage further research in this domain.

The significance of this work lies in its potential to provide valuable insights into the emotional and sentiment dynamics of Polish language texts, which can be applied in various applications such as customer feedback analysis, social media monitoring, and human-computer interaction. Our open-source initiative is intended to serve as a robust foundation for future explorations and innovations in Polish language processing, fostering collaboration and k nowledge sharing within the research community.

2. Data

The competition organizers provided a dataset consisting of Polish consumer reviews spanning four domains: hotels, medicine, products, and school. In addition to opinion-based reviews, the dataset contained informative texts from these domains that did not express opinions. The annotation scheme covered both emotions from Plutchik's wheel (joy, trust, anticipation, surprise, fear, sadness, disgust, anger) and sentiment polarity (positive, negative, neutral), applied at both the sentence and full review levels. Cases of mixed sentiment were annotated with both positive and negative labels.

Six annotators worked independently to label the data. To establish the final annotations, emotions and sentiments were included if marked by at least two annotators. This methodology permitted multiple, even contradictory emotion labels for contentious content. Although sentences received individual annotations, annotators considered the full review context when making their decisions.

The dataset was randomly split as follows:

- Training set: 776 reviews containing 6,393 sentences
- Two test sets: 167 reviews each, containing 1,234 and 1,264 sentences respectively

Each annotation was represented as a binary vector encoding the presence or absence of the emotion and sentiment categories.

More details about the dataset construction and annotation process can be found in (Koptyra et al. 2023) and (Kocoń et al. 2023).

Table 1 shows the distribution of emotion and sentiment labels in the training set, revealing several interesting patterns. Joy and sadness are the most frequently occurring emotions,

[1] https://github.com/enelpol/poleval2024-task2

Table 1: Distribution of labels in the original training set at sentence and text level.

Label	Sentence (%)	Text (%)
Joy	47.91	57.09
Trust	23.29	26.03
Anticipation	12.97	8.63
Surprise	6.46	6.83
Fear	4.00	3.61
Sadness	42.73	54.51
Disgust	17.63	27.06
Anger	15.34	23.32
Positive	53.26	60.31
Negative	45.61	55.41
Neutral	26.98	15.46

present in nearly half of all sentences and over half of full texts. Trust appears in roughly a quarter of the data, while emotions like fear and surprise are relatively rare. For sentiment polarity, positive labels slightly outweigh negative ones at both sentence and text levels. The lower frequency of neutral sentiment in full texts (15.46%) compared to sentences (26.98%) suggests that while individual sentences may be neutral, complete reviews tend to express more definitive sentiment.

The low prevalence of surprise (6.83%) and fear (3.61%) emotions in the dataset raises important considerations for evaluation. Given these percentages, we can expect approximately 11 and 6 texts with these emotions respectively in each test set of 167 reviews. This limited representation of certain emotion categories in the test data could potentially lead to high variance in evaluation metrics. Furthermore, if conventional random splitting was used to create the test sets, there is a risk that the class distributions could differ significantly from the training set, potentially making any validation dataset unrepresentative of the actual test conditions. This is particularly concerning for emotion recognition tasks where class imbalance can significantly impact model performance.

Since no development set was provided by the competition organizers, we created our own by splitting the training data. To ensure representative sampling across all emotion and sentiment categories, we employed iterative stratification (Sechidis et al. 2011, Szymański and Kajdanowicz 2017), a technique specifically designed for multi-label data. This approach allocated 20% of the training data (156 texts) to a development set while maintaining similar label distributions between the resulting splits, as evidenced by the statistics in Table 2.

Table 2 presents the distribution of emotion and sentiment labels across the train and development sets at both sentence and text levels. The iterative stratification approach effectively preserved the label distributions, with most emotions showing minimal variations between splits. Notably, the relative frequencies of major emotions like Joy (48-49% for sentences, 57% for texts) and Sadness (42-43% for sentences, 54% for texts) remain highly consistent. The sentiment polarities (Positive, Negative, Neutral) also maintain similar proportions,

Table 2: Distribution of labels in train (620 texts) and dev (156 texts) sets at sentence and text level.

	Sentence level		**Text level**	
Label	**Train (%)**	**Dev (%)**	**Train (%)**	**Dev (%)**
Joy	47.57	49.13	57.10	57.05
Trust	22.65	25.61	26.13	25.64
Anticipation	13.40	11.40	8.71	8.33
Surprise	6.49	6.35	6.77	7.05
Fear	4.13	3.54	3.39	4.49
Sadness	42.84	42.35	54.52	54.49
Disgust	17.00	19.91	26.94	27.56
Anger	14.66	17.82	23.23	23.72
Positive	53.41	52.74	60.32	60.26
Negative	45.56	45.82	55.16	56.41
Neutral	27.60	24.75	15.48	15.38

demonstrating the effectiveness of the stratification strategy in creating representative splits for model development and evaluation.

Analysis of the token length (using Bielik-11B tokenizer) distribution in the training set reveals important characteristics about the data. The average text length is 412.27 tokens, indicating that most samples can be processed by standard transformer models. However, there is significant variation in length, with the longest text containing 6,043 tokens. Looking at cumulative statistics:

- 85.44% (663) of samples contain fewer than 512 tokens, making them suitable for processing with standard BERT-style models
- 93.69% (727) of samples are under 1,024 tokens
- 97.42% (756) of samples are under 2,048 tokens
- 99.36% (771) of samples are under 4,096 tokens

This distribution suggests that while most texts can be handled by models with standard context windows, a small but significant portion of samples (14.56%) exceed the 512-token limit of traditional transformer encoders. This motivated our choice of architecture, as discussed in the following section.

3. Evaluation

The evaluation metric for this task was designed to assess both sentence-level and text-level predictions. The final score is calculated as the arithmetic mean of two F1 macro scores - one for sentence annotations and one for text annotations:

$$Final\, score = \frac{F1_{macro}\, sentences + F1_{macro}\, texts}{2} \tag{1}$$

Each F1 macro score is computed by averaging the F1 scores across all emotion and sentiment labels:

$$F1_{macro} = \frac{\sum_{i=1}^{n} F1_i}{n} \tag{2}$$

where n is the number of labels (11 in total - 8 emotions and 3 sentiment categories), and the F1 score for each label is calculated using the standard formula:

$$F1 = 2 * \frac{precision * recall}{precision + recall}, \quad precision = \frac{TP}{TP + FP}, \quad recall = \frac{TP}{TP + FN} \tag{3}$$

where TP, FP, and FN represent true positives, false positives, and false negatives respectively. In cases where the denominator in any of these metrics equals zero, the corresponding score is set to 0 to handle undefined values.

This evaluation approach equally weights the model's ability to detect emotions and sentiment at both the granular sentence level and the broader text level, providing a balanced assessment of system performance. However, given the imbalanced distribution of some emotions in the dataset (particularly fear and surprise), the macro-averaging of F1 scores means that performance on rare categories has equal weight in the final metric as performance on more common categories.

Each level-label pair (sentence/text level combined with emotion/sentiment) contributes 1/22 to the final score, which means that individual model decisions on rare emotions can impact the overall score by several percentage points. This highlights the critical importance of handling rare emotion categories effectively, as poor performance on even a single rare emotion can significantly affect the final evaluation metric. The equal weighting also encourages development of models that perform consistently across all emotion categories rather than focusing solely on the most frequent ones.

4. Methods

In addressing the emotion and sentiment recognition task, we carefully evaluated several model architectures commonly used for Polish language processing. Initially, we considered encoder-only models such as HerBERT (Mroczkowski et al. 2021), Polish RoBERTa (Dadas et al. 2020), and XLM-RoBERTa (Conneau et al. 2020). However, these models have a context length limitation of 512 tokens, making them unsuitable for processing longer text passages in our dataset.

Given these limitations, we opted for a decoder-only Large Language Model (LLM) architecture, which can process longer sequences effectively. To efficiently train the model while maintaining high performance, we employed two key techniques:

— Low-Rank Adaptation (LoRA) (Hu et al. 2022), which reduces the number of trainable parameters by decomposing weight updates into low-rank matrices, enabling efficient fine-tuning while preserving model quality.

- Supervised Fine-Tuning (SFT), where we train the model on carefully curated question-answer pairs to improve its performance on both the answerability classification and answer generation tasks.

For the model architecture, we augmented the base LLM with a custom multilabel classification head consisting of 11 binary classifiers - one for each emotion (8) and sentiment (3) category. This classification head is implemented as a linear layer that takes the final hidden state of the LLM and produces probability scores for each category. The model processes both sentence-level and text-level inputs through the same architecture, with a different prompt indicating the input type to help the model differentiate between the two tasks.

We format our input as a structured prompt that includes:

- left context in case of sentence-level prediction
- text content to be analyzed
- right context in case of sentence-level prediction
- prompt for the classification

Used prompt template in Polish:

```
\texttt{Lewy kontekst: \{left_context\}\n
Tekst do oceny: \{text_to_evaluate\}\nPrawy kontekst:
\{right_context\}\nOznacz tekst do oceny względem emocji
i sentymentu: radość, zaufanie, oczekiwanie, zaskoczenie, strach,
smutek, obrzydzenie, gniew, pozytywny, negatywny, neutralny.}
```

The model is trained end-to-end using binary cross-entropy loss for each emotion and sentiment category.

This architectural choice, combined with our efficient training approach, provides the flexibility to handle both sentence and text-level predictions while accommodating the full context length requirements of our dataset. The shared model architecture for both levels allows for better parameter efficiency and enables the model to learn complementary features across the two tasks.

To enhance model performance, we explored various ensemble techniques for combining predictions from multiple models. One approach involved majority voting across models at the final prediction level, where each model casts a vote for each emotion and sentiment category, and the most common prediction is selected. We also experimented with per-class/level voting, where separate models specialize in different emotion categories or prediction levels (sentence vs. text), and their predictions are combined through voting. This ensemble approach helps mitigate individual model biases and improves robustness, particularly for rare emotion categories where single model performance can be unstable.

5. Experiments

For our experiments, we selected Bielik-11B (Ociepa et al. 2024a,c,b) as the most promising Polish language model that could be fine-tuned on a single GPU according to Open PL LLM Leaderboard (Wróbel et al. 2024). This model represents a good balance between performance and computational requirements for the multilabel classification task.

Considered parameters:

- learning rate: 1e-4, 2e-4
- effective batch size: 16, 32
- number of epochs: 3, 5, 8, 10
- LoRA rank: 16, 64
- for sentence-level examples number of sentences as left and right context: 0, 1, 2, 3, all
- loss: binary cross-entropy, focal loss (Lin et al. 2017)
- class weights: uniform, proportional to inverse of class frequency, proportional to inverse of square root of class frequency
- weight for text loss: 1.0, 5.0, 10.0
- train on sentence level, train on text level or both
- using additional dataset: XED (Öhman et al. 2020)
- 5 weight initialization seeds
- max sequence length: 1024, 2048, 4096

About 100 experiments were conducted to evaluate the impact of different parameters on model performance (see Section 8 for chosen results).

6. PolEval submissions

For the PolEval competition, we prepared several model variants and ensemble approaches to comprehensively evaluate different strategies for emotion and sentiment recognition. The numbers below correspond to the submission names in the competition:

1. Single unified: Best model optimized for overall performance across both prediction levels
2. Sentence specialist: Model specialized for sentence-level emotion and sentiment detection
3. Text specialist: Model focused on text-level analysis (the same model as in 2 but using checkpoint from 1000 training steps)
5. Per-label ensemble: Ensemble combining best models per emotion/sentiment class and prediction level

7. Majority voting: Voting ensemble utilizing predictions from models 1, 2, and 3
9. Hybrid ensemble: Ensemble combining strategy from 5 with the voting approach from 7
10. Test-A optimized: Ensemble of two specialized models selected based on test-A performance on sentence level and text level
11. Full data retrain: Model 1's architecture retrained on the complete dataset including development set

Our primary models (1 and 2) shared these fundamental parameters:

— Base LLM: Bielik-11B-v2.0-Instruct
— Fine-tuning learning rate: 2e-4
— Efficient training with batch size 2 and 8-step gradient accumulation
— LoRA adaptation (rank 16) for parameter-efficient training
— Focal loss with inverse frequency class weighting
— Training regime: 10 epochs with validation every 100 steps
— Extended context handling with 4096 token maximum sequence length
— Scheduler type: Linear
— Warmup: None

Key architectural differences between models 1 and 2:

— Contextual window: Model 1 used single sentence context while Model 2 incorporated 3 surrounding sentences
— Text-level emphasis: Model 1 applied stronger weighting (10.0) compared to Model 2 (5.0) for text-level predictions

For ensemble approaches (models 5, 7, 9), we explored various combination strategies:

— Emotion-specific specialization: Training dedicated models for each emotion/sentiment category
— Democratic voting: Aggregating predictions from top-performing models
— Hybrid methods: Combining specialized emotion detectors with voting mechanisms

Submission 10 represents a strategic ensemble approach that combines two specialized models selected based on their performance on the test-A dataset. Specifically, for sentence-level predictions, we utilized the model configuration that showed the best performance on test-A sentences. For text-level predictions, we employed the model variant that demonstrated superior performance on test-A full texts.

Model 11 replicated the architecture of Model 1 but leveraged the complete dataset including development samples, aiming to maximize the available training data for final predictions.

7. Results

We present a comprehensive analysis of our models' performance across development and test datasets. Table 3 shows detailed evaluation results for all model variants, while Table 4 presents the final competition standings.

Table 3: Comprehensive evaluation results across development and test sets. Performance is measured using F1 scores for sentence-level (Sent.), text-level (Text) predictions and their average (Avg). Model variants include specialized architectures, ensemble approaches, and different training strategies. Best scores for each metric and dataset are highlighted in bold.

Model	**Dev**			**Test-A**			**Test-B**		
	Sent.	**Text**	**Avg**	**Sent.**	**Text**	**Avg**	**Sent.**	**Text**	**Avg**
1 Single unified	81.19	78.81	80.00	80.78	77.86	79.32	80.72	74.27	77.49
2 Sentence specialist	82.31	76.78	79.54	80.68	76.81	78.74	81.06	74.40	77.73
3 Text specialist	79.24	80.17	79.71	78.36	**79.40**	78.88	77.82	**78.48**	78.15
5 Per-label ensemble	83.99	**85.67**	**84.83**	80.84	78.19	79.51	80.80	73.92	77.36
7 Majority voting	82.84	81.06	81.95	**81.62**	77.77	79.70	81.51	74.95	78.23
9 Hybrid ensemble	**84.21**	84.75	84.48	80.71	78.32	79.51	**81.72**	76.07	78.90
10 Test-A optimized	82.84	80.17	81.51	**81.62**	**79.40**	**80.51**	81.51	**78.48**	**79.99**
11 Full data retrain	–	–	–	80.83	77.82	79.32	81.50	76.12	78.81

Our experimental results reveal several key findings:

- The Test-A optimized ensemble (Model 10) achieved the best overall performance on both test sets, with F1 scores of 80.51% on Test-A and 79.99% on Test-B, demonstrating the effectiveness of specialized model selection.
- Ensemble approaches (Models 5 and 9) showed remarkable performance on the development set but experienced some performance degradation on test sets, suggesting potential overfitting despite their sophisticated combination strategies.
- The Text specialist model (Model 3) consistently achieved the highest text-level scores across both test sets, indicating the importance of focused training for specific prediction levels.
- The Full data retrain (Model 11) showed competitive performance, particularly on Test-B with an F1 score of 78.81%, achieving the best result among single models and validating the benefit of utilizing all available training data.
- The correlation between Test-A and Test-B average scores was relatively weak (Pearson's r = 0.20), suggesting that performance on Test-A was not strongly predictive of Test-B results, despite our model selection strategy based on Test-A performance proving effective.

Table 4: PolEval 2024 competition results showing sentence-level (Sent.), text-level (Text) and final F1 scores for test sets A and B. Our winning submission achieved consistent performance across both test sets, maintaining a clear margin over other participants.

Rank	Submitter	Entries	test-A scores			test-B scores		
			Sent.	Text	Final	Sent.	Text	Final
1	Krzysztof Wróbel	10	81.62	79.40	80.51	81.51	78.48	79.99
2	Tomasz Warzecha	196	78.87	81.54	80.20	79.34	79.28	79.31
3	Cezary Kęsik	25	74.94	76.42	75.68	76.66	79.33	77.99
4	Jakub Pokrywka	15	78.65	75.93	77.29	79.43	75.77	77.60
5	Paweł Lewkowicz	10	74.29	77.73	76.01	77.27	77.20	77.23
6	Katarzyna Baraniak	32	75.94	77.47	76.70	76.11	77.76	76.94
7	Cezary Kęsik	5	73.62	79.12	76.37	75.94	70.43	73.19
8	Jakub Kosterna	4	50.47	28.71	39.59	52.19	28.71	40.45
9	Paweł Cyrta	5	33.04	32.74	32.89	31.86	34.28	33.07

In the final competition standings (Table 4), our approach secured the first place with a significant margin, demonstrating robust performance across both sentence-level and text-level predictions. The consistency between Test-A and Test-B results validates the generalization capability of our models, particularly the Test-A optimized ensemble which maintained its superior performance across both evaluation sets.

8. Conclusions

In this study, we have demonstrated the efficacy of leveraging Large Language Models (LLMs) for emotion and sentiment recognition tasks. Our findings underscore the critical importance of selecting appropriate evaluation metrics to accurately assess model performance. Specifically, we highlight the necessity of ensuring a minimal count of rare emotions in future test datasets. This is crucial to reduce the range of errors and enhance the robustness of the evaluation process.

Moreover, we encountered significant challenges in creating a correct validation set for the competition. The variability and imbalance in emotion categories posed difficulties in achieving a representative and unbiased validation set. Addressing these challenges is essential for the advancement of emotion recognition systems and for fostering fair and reliable competition environments.

Future research directions may include testing other models, particularly smaller ones, to evaluate the impact of model size on performance. Another potential direction is to create 22 outputs for each pair of sentence and text-level predictions, which could provide more granular insights into model performance.

Acknowledgements

We gratefully acknowledge Poland's high-performance Infrastructure PLGrid ACK Cyfronet AGH for providing computer facilities and support within computational grant no PLG/2024/017168.

References

Conneau A., Khandelwal K., Goyal N., Chaudhary V., Wenzek G., Guzmán F., Grave E., Ott M., Zettlemoyer L. and Stoyanov V. (2020). *Unsupervised Cross-lingual Representation Learning at Scale*. In Jurafsky D., Chai J., Schluter N. and Tetreault J. (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8440–8451, Online. Association for Computational Linguistics.

Dadas S., Perełkiewicz M. and Poświata R. (2020). *Pre-training Polish Transformer-based Language Models at Scale*. In *Artificial Intelligence and Soft Computing*, pp. 301–314. Springer International Publishing.

Hu E. J., Shen Y., Wallis P., Allen-Zhu Z., Li Y., Wang S., Wang L. and Chen W. (2022). *LoRA: Low-Rank Adaptation of Large Language Models*. In *Proceedings of the 10th International Conference on Learning Representations (ICLR 2022)*.

Kocoń J., Cichecki I., Kaszyca O., Kochanek M., Szydło D., Baran J., Bielaniewicz J., Gruza M., Janz A., Kanclerz K., Kocoń A., Koptyra B., Mieleszczenko-Kowszewicz W., Miłkowski P., Oleksy M., Piasecki M., Łukasz Radliński, Wojtasik K., Woźniak S. and Kazienko P. (2023). *ChatGPT: Jack of all trades, master of none*. Information Fusion, 99, p. 101861.

Koptyra B., Ngo A., Radliński Ł. and Kocoń J. (2023). *CLARIN-Emo: Training Emotion Recognition Models Using Human Annotation and ChatGPT*. In Mikyška J., de Mulatier C., Paszynski M., Krzhizhanovskaya V. V., Dongarra J. J. and Sloot P. M. (eds.), *Computational Science – ICCS 2023*, pp. 365–379, Cham. Springer Nature Switzerland.

Lin T.-Y., Goyal P., Girshick R., He K. and Dollar P. (2017). *Focal Loss for Dense Object Detection*. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2980–2988.

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.

Ociepa K., Flis Ł., Kinas R., Gwoździej A., Wróbel K., SpeakLeash Team and Cyfronet Team (2024a). *Bielik-11B-v2.0-Instruct model card*.

Ociepa K., Flis Ł., Kinas R., Gwoździej A. and Wróbel K. (2024b). *Bielik 2: A Family of Large Language Models for the Polish Language — Development, Insights, and Evaluation*. Manuscript in preparation.

Ociepa K., Flis Ł., Wróbel K., Gwoździej A. and Kinas R. (2024c). *Bielik 7B v0.1: A Polish Language Model – Development, Insights, and Evaluation*. arXiv:2410.18565.

Öhman E., Pàmies M., Kajava K. and Tiedemann J. (2020). *XED: A Multilingual Dataset for Sentiment Analysis and Emotion Detection*. In Scott D., Bel N. and Zong C. (eds.), *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6542–6552, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Sechidis K., Tsoumakas G. and Vlahavas I. (2011). *On the Stratification of multi-label Data*. In Gunopulos D., Hofmann T., Malerba D. and Vazirgiannis M. (eds.), *Machine Learning and Knowledge Discovery in Databases*, pp. 145–158. Springer Berlin Heidelberg.

Szymański P. and Kajdanowicz T. (2017). *A Network Perspective on Stratification of Multi-Label Data*. In Torgo L., Krawczyk B., Branco P. and Moniz N. (eds.), *Proceedings of the First International Workshop on Learning with Imbalanced Domains: Theory and Applications*, vol. 74 of *Proceedings of Machine Learning Research*, pp. 22–35, ECML-PKDD, Skopje, Macedonia. PMLR.

Wróbel K., SpeakLeash Team and Cyfronet Team (2024). *Open PL LLM Leaderboard*.

Appendix

Detailed Model Performance

To provide a comprehensive understanding of our model's performance, we present additional detailed results and analyses. These include various metrics and visualizations that highlight the strengths and weaknesses of our approach, as shown in Figure 1.

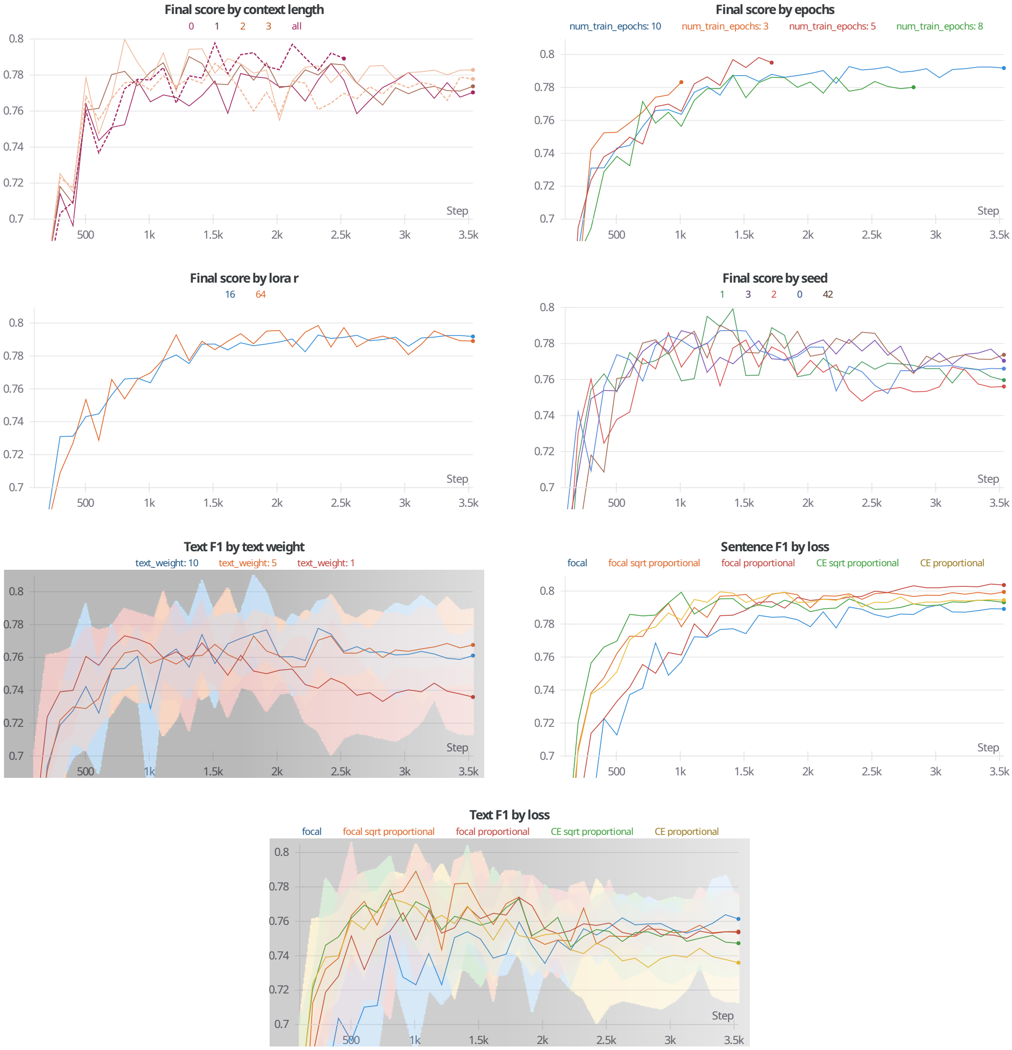

Figure 1: Visual representation of model performance metrics and training progress.

Ensemble as a Variance Reduction Method for Emotion and Sentiment Recognition

Tomasz Warzecha (Independent researcher)

Abstract

This paper presents how an ensemble of different models can be used for the emotion and sentiment recognition for textual data. The work touches challenges in emotion recognition problem including these related to data amount and imbalance as well as differences in difficulty for recognizing a particular emotion. The problem which is being solved is stated for Polish language and the presented solution is based on three popular models that are used in Polish NLP tasks – HerBERT, Polish RoBERTa-v2, and XLM-RoBERTa. It is shown that the surrounding context is beneficial for emotion recognition. Different methods of gluing the context to the data are evaluated. Results of individual models and ensembles of different sizes are presented. The paper shows how to overcome the variance using an ensemble and that the variance can be beneficial when combining models. Presented solution was ranked second in PolEval 2024 Emotion and Sentiment Recognition competition.

Keywords

emotion recognition, sentiment analysis, ensemble, BERT, Polish, natural language processing, PolEval

1. Introduction

Emotion recognition is a well known but still challenging task. While some of the emotions are relatively easy to recognize even for simple solutions (like Joy and Sadness) - other are connected to the subtle differences of word usage, nuances in meaning and larger surrounding context. Those hard to distinguish emotions (like Fear and Surprise) are often underrepresented in training datasets and PolEval 2024 competition dataset is no different here. Besides that, subtle nuances can also be a subject of disagreement during annotators' classification. Having all this - we need to add that the problem we are trying to solve is for the Polish language. Recent LLM developments gave us highly multilingual models, however this work is focused on smaller BERT-like models (Devlin et al. 2019) which can be easily trained and run on consumer grade GPUs. Among smaller models performance for English is often superior than for other languages and we have not that many baseline models that produce decent results for Polish.

2. Experiment setup

2.1. Problem statement

The problem we were trying to solve was to recognize emotion and sentiment in textual data in Polish. There were 8 classes reflecting basic emotions from Plutchik's Wheel of Emotions to be recognized (Joy, Trust, Anticipation, Surprise, Fear, Sadness, Disgust, Anger) and 3 sentiment classes to be predicted (Positive, Negative, Neutral). The textual data contained reviews of hotels, medicine, products, and school. Annotators manually classified each review sentence as well as each review as a whole with 11 labels. For more information refer to (Koptyra et al. 2023) and (Kocoń et al. 2023). That resulted in a multilabel problem as a particular sentence or a particular review could belong to multiple different classes. Two subtasks were given - to predict emotions and sentiment for individual sentences and to predict emotions and sentiment for whole review texts. Evaluation of the solutions was based on the macro average of F1 scores. Final score was constructed as an average of two subtask scores. Emotion and sentiment recognition was part of the PolEval 2024 competition which is more widely described in Task 2 description (Kobyliński et al. 2023).

2.2. Data preparation

Despite two competition subtasks (recognizing emotions from individual sentences taken from reviews and recognizing emotions from whole review texts), a single dataset was used for most of the training runs. As target column contained expected labels for both individual sentences and whole review texts and given textual data contained only review sentences, the dataset rows were adjusted to also contain whole review texts to match the target column in appropriate rows. Adjusted dataset looked like this:

```
<sentence 1 of the first review>
<sentence 2 of the first review>
...
<sentence N of the first review>
<sentence 1 + sentence 2 + ... + sentence N of the first review>
<sentence 1 of the second review>
...
```

In many cases the correct assessment of the emotions is not possible without having a surrounding context. For example a sentence:

```
It was unique.
```

could be recognized differently based on its larger context:

```
It was unique. It was extraordinary and definitely worth seeing!

The show? Well... Let's say... It was unique.
```

Early experiments confirmed that providing context produces better results. That's why the belonging review context was added to each dataset row in the vast majority of experiments. Because the evaluated models had input limited to 512 tokens, only that amount of context was taken so the concatenation of the sentence and its context fitted within the model size limit. Context was provided in various ways to get the most out of the final ensemble and that is how each model was trained. No further preprocessing nor cleanup was applied to the dataset.

2.3. Variance as an ally

The given dataset was imbalanced, not only because some classes were underrepresented, but also because some classes were more difficult to learn. Because we did not want to lose too much positive examples, competition dataset was split into 0.9/0.1 parts for train and validation respectively. Such small validation set was not a very reliable source of feedback on model's quality. Moreover, multilabel training using imbalanced data resulted in a variance not only between experiments but also between checkpoints within a single experiment. That led to an idea of ensemble which was to reduce the variance. The interesting observation was that there were a lot of differences between individual models' outputs, which made the idea of ensemble even more promising.

Table 1: Emotions frequencies showing imbalance in the competition dataset

Joy	**Trust**	**Anticipation**	**Surprise**	**Fear**	**Sadness**	**Disgust**	**Anger**	**Positive**	**Negative**	**Neutral**
3005	1378	767	433	265	2933	1256	1098	3348	3115	1699

2.4. Training methods

Multiple experiments were executed, with different ways of gluing surrounding context to the review sentence which was to be predicted. In first two experiments (A, B) the whole review context was concatenated to the sentence to predict, so in fact the sentence was repeated twice. Example of context providing:

```
I love this product! I'm so happy. 5 stars! [SEP] I'm so happy.
```

Third experiment (C) used a previous sentence as a context.

```
I love this product! [SEP] I'm so happy.
```

Fourth experiment (D) used whole review text as a context and sentence to be predicted was tagged with separation tokens.

```
I love this product! [SEP] I'm so happy. [SEP] 5 stars!
```

Next three (marked further as O, OS, OT) did not use any context.

```
I'm so happy.
```

In some cases the training was specific to the competition subtasks - predicting emotions for individual sentences ("S" suffix to method name) and recognizing emotions in whole review texts ("T" suffix).

A) <WHOLE_REVIEW_TEXT> + <SENTENCE_TO_PREDICT> (with left trim)[1]
B) <SENTENCE_TO_PREDICT> + <WHOLE_REVIEW_TEXT> (with right trim)
C) <PREVIOUS_SENTENCE> + <SENTENCE_TO_PREDICT>
D) <WHOLE_REVIEW_TEXT_WITH_SENTENCE_TAGGED> (both side trim)

A few experiments were done with fine-tuning using data without context:

O) fine-tuning without context
OS) fine-tuning without context only with sentences
OT) fine-tuning without context only with whole review texts

A few experiments were executed with final fine-tuning of previously fine-tuned models which was focused on sentences or whole reviews:

AS) fine-tuning A models with review sentences only
AT) fine-tuning A models with whole review texts only
CT) fine-tuning C models with whole review texts only

2.5. Evaluated models

This work was focused on utilizing BERT-like models that are suitable for Polish NLP. For initial evaluation a couple of multilingual models were taken: XLM-RoBERTa (Conneau et al. 2020), RemBERT (Chung et al. 2021), mDeBERTa (He et al. 2021) and specifically pretrained for Polish: HerBERT (Mroczkowski et al. 2021) and Polish RoBERTa-v2 (Dadas et al. 2020). After first experiments, XLM-RoBERTa, HerBERT and Polish RoBERTa-v2 showed superior performance over others and these three were used in further evaluation. Early experiments showed that large model variants were significantly better than their base counterparts, thus large variants were used in most of the experiments. Also some limited experiments were conducted with XLM-RoBERTa-XL (Goyal et al. 2021).

[1] left or right trimming effectively equals taking that much of the context that fits the model input size

2.6. Important training parameters

- batch size: 32
- learning rate: 2e-5
- optimizer: AdamW (Hugginface, with default parameters)
- loss: Binary Cross Entropy with weights inversely proportional to class frequency
- learning rate schedule: linear with warmup (0.05)

Most of the experiments were conducted for 20 epochs and best checkpoints were selected for the final ensemble.

For XLM-RoBERTa-XL training - first 24 layers were frozen, AdaFactor optimizer and higher learning rates were used.

2.7. The ensemble

Each model (XLM-RoBERTa, HerBERT, Polish RoBERTa-v2) was fine-tuned using different methods of concatenating context (described in section 2.4). Additionally, limited experiments were conducted with XLM-RoBERTa-XL with a subset of above methods (A, AT, OS). However, the training was unstable. These experiments were marked with the ":XL" suffix.

Because some of the training methods (AS, AT, OS, OT, CT) were specific to the particular competition subtask (single sentence or whole review text), two subtask-specific ensembles were built:

- 14 models were used for "single sentence" subtask (methods: A, B, C, D, A:XL, OS:XL),
- 16 were used for "whole review text" subtask (methods: AT, B, CT, D, OT, AT:XL)

Ensembles used logits sum for "single sentence" subtask and majority voting for "whole review text" subtask.

3. Results

3.1. Individual emotion scores

The results how well models could recognize particular emotions is shown in Tables 2 and 3. Scores come from evaluation with validation dataset for models trained with A and B methods.

Due to a very small validation dataset, the results have a big uncertainty, especially for underrepresented classes like Fear and Surprise. Rare emotions achieve significantly lower scores. Some of the emotions are easier to recognize than others - e.g. Anticipation is underrepresented but achieved a better F1 score than Anger and Disgust (80.63 vs 76.71, 76.54). On contrary, the Neutral sentiment which has a decent representation in the dataset was much harder to recognize than less represented Trust (74.97 vs 86.61).

Table 2: F1 scores per emotion, methods A and B, validation ds, part 1

	Joy	**Trust**	**Anticipation**	**Surprise**	**Fear**	**Sadness**
Row count (train)	3005	1378	767	433	265	2933
A) Polish RoBERTa v2	94.78	85.98	82.22	55.31	55.17	90.72
A) HerBERT	93.72	86.41	80.31	46.51	**64.28**	91.23
A) XLM-RoBERTa	**94.81**	**88.36**	81.27	57.14	47.05	91.90
A) XLM-RoBERTa-XL	94.61	85.66	77.17	55.17	46.66	**92.00**
B) Polish RoBERTa v2	94.13	86.92	80.29	57.14	48.48	91.75
B) HerBERT	93.69	85.55	**82.98**	50.00	38.46	91.66
B) XLM-RoBERTa	93.01	87.38	80.16	**60.86**	44.44	91.57
min	93.01	85.55	77.17	46.51	38.46	90.72
max	94.81	88.36	82.98	60.86	64.28	92.00
avg	94.11	86.61	80.63	54.59	49.22	91.55

Table 3: F1 scores per emotion, methods A and B, validation ds, part 2

	Disgust	**Anger**	**Positive**	**Negative**	**Neutral**	**Average**
Row count (train)	1256	1098	3348	3115	1699	1754
A) Polish RoBERTa v2	77.00	75.24	95.63	90.73	75.00	79.80
A) HerBERT	75.91	76.00	95.68	**92.75**	**76.72**	79.96
A) XLM-RoBERTa	77.61	74.22	95.91	91.58	72.94	79.34
A) XLM-RoBERTa-XL	**80.00**	**80.32**	**96.35**	92.60	75.00	79.59
B) Polish RoBERTa v2	74.79	79.24	96.18	91.04	74.28	79.48
B) HerBERT	74.01	74.76	96.11	91.48	76.47	77.74
B) XLM-RoBERTa	76.47	77.22	95.68	92.19	74.37	79.40
min	74.01	74.22	95.63	90.73	72.94	76.27
max	80.00	80.32	96.35	92.75	76.72	82.68
avg	76.54	76.71	95.93	91.77	74.97	79.33

3.2. Models trained with different methods and their ensembles

All presented results are taken from competition Test-A. The F1 scores achieved by models and their ensembles are shown in Tables 4 and 5. Additionally, XL models results are presented in Table 6.

When drawing conclusions about individual models it needs to be taken into consideration that the results come from checkpoints chosen based on a validation dataset of limited size. There are a couple of findings we can point out.

XLM-RoBERTa, HerBERT and Polish RoBERTa-v2 produce similar results for this problem. We could observe that providing a whole review context acted better than concatenating just a previous sentence. However, differences between different methods of gluing context were diminishing for ensembles.

Table 4: F1 scores for individual sentences subtask, Test-A

	A	B	C	D	**avg**
XLM-RoBERTa$_{\text{large}}$	77.30	75.79	74.10	77.08	**76.07**
HerBERT$_{\text{large}}$	77.51	76.68	74.50	77.48	**76.54**
Polish RoBERTa-v2$_{\text{large}}$	77.69	78.38	75.80	75.21	**76.77**
3 models average	77.50	76.95	74.80	76.59	**76.46**
3 models ensemble	78.13	78.46	77.72	77.95	**78.07**

Table 5: F1 scores for whole review texts subtask, Test-A

	AT	B	CT	D	O	**avg**
XLM-RoBERTa$_{\text{large}}$	76.55	74.73	75.78	78.81	77.18	**76.61**
HerBERT$_{\text{large}}$	77.25	77.36	77.71	78.65	77.71	**77.74**
Polish RoBERTa-v2$_{\text{large}}$	78.05	74.04	76.32	78.20	76.32	**76.59**
3 models average	77.28	75.38	76.60	78.55	77.07	**76,98**
3 models ensemble	79.28	79.04	77.84	77.86	79.14	**78.63**

Ensembles showed superior performance over individual models - combining together 3 models (HerBERT, Polish RoBERTa v2 and XLM-RoBERTa trained with the same method) produced 2% better results on average. In 8 out of 9 cases three models ensemble gave better results than the average of individual models. In 7 out of 9 cases such three models ensemble achieved better results than the best of the individual models in the group.

Table 6: XL model evaluation on sentences (left) and whole review subtask (right), F1 scores compared to large models, Test-A

	F1
best large model	78.38
A) XLM-RoBERTa$_{\text{large}}$	77.30
A) XLM-RoBERTa$_{\text{XL}}$	77.30
OS) XLM-RoBERTa$_{\text{XL}}$	77.77

	F1
best large model	78.81
AT) XLM-RoBERTa$_{\text{large}}$	76.55
AT) XLM-RoBERTa$_{\text{XL}}$	78.65

Although just based on competition Test-A results we cannot conclude that XLM-RoBERTa-XL was superior than large models, validation dataset results suggested[2] that it was better by around 0.5pp. We need to stress out that the XL models were much harder to train and limited experiments may have not undercovered their full potential.

[2] Comparison without Fear and Surprise classes that were producing high variance

3.3. Results for the final ensemble

The final solution consisted of two ensembles - 14 models targeting single sentence subtask and 16 models for whole review text subtask (refer section 2.7 for details).

Table 7: Ensemble F1 scores for competition Test-A

	Sentences	Whole texts	**Test-A score (avg)**
Average of individual models	76.61	77.08	**76.85**
Ensemble of individual models	79.95	81.54	**80.75**

Proposed ensemble resulted in 5% gain over individual models' average and 2.7% over best performing ones. We need to be critical here and take into consideration some possible overfitting. The more cautious estimate of the gain would be 3-4% over average model and around 2% over best ones, which matches the final scores on the competition hidden data (Test-B). Presented solution achieved the highest score in the competition Test-A and was ranked second in the final competition score (Test-B).

3.4. The study of extended ensembles

Having results for epoch checkpoints, the experimental extended ensembles were built using all checkpoints meeting certain F1 thresholds. Multiple thresholds were checked, best were for F1s between 76.5 and 77.5. Such ensembles combined even up to 180 individual checkpoint results. Experiments showed that for bigger ensembles it was better to use majority voting instead of logits sum.

Big ensembles did not show measurable improvement. Likely combining checkpoints from the same training runs were not that beneficial.

4. Conclusion

This work shows how efficient an ensemble of relatively small language models can be for the emotion and sentiment recognition task, especially for well represented classes. Models working as a group can successfully be used to overcome the training variance and imbalanced data, which was the case in the PolEval 2024 competition. Although combining tens of models may not be practical, combining even just a few models can give gain which in some cases may be hard to achieve using a single but bigger model. Sometimes using a single but larger model may also not be possible due to hardware constraints. However, for achieving good results more important from all described methods would be to gather a dataset that better covers the underrepresented and hard to learn classes. Further work could also explore LLM capabilities, especially for such underrepresented cases.

References

Chung H. W., Févry T., Tsai H., Johnson M. and Ruder S. (2021). *Rethinking Embedding Coupling in Pre-trained Language Models*. In *Proceedings of the 9th International Conference on Learning Representations (ICLR 2021)*.

Conneau A., Khandelwal K., Goyal N., Chaudhary V., Wenzek G., Guzmán F., Grave E., Ott M., Zettlemoyer L. and Stoyanov V. (2020). *Unsupervised Cross-lingual Representation Learning at Scale*. In Jurafsky D., Chai J., Schluter N. and Tetreault J. (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8440–8451, Online. Association for Computational Linguistics.

Dadas S., Perełkiewicz M. and Poświata R. (2020). *Pre-training Polish Transformer-based Language Models at Scale*. In *Artificial Intelligence and Soft Computing*, pp. 301–314. Springer International Publishing.

Devlin J., Chang M.-W., Lee K. and Toutanova K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. In Burstein J., Doran C. and Solorio T. (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Goyal N., Du J., Ott M., Anantharaman G. and Conneau A. (2021). *Larger-Scale Transformers for Multilingual Masked Language Modeling*. In Rogers A., Calixto I., Vulić I., Saphra N., Kassner N., Camburu O.-M., Bansal T. and Shwartz V. (eds.), *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pp. 29–33, Online. Association for Computational Linguistics.

He P., Liu X., Gao J. and Chen W. (2021). *DeBERTa: Decoding-enhanced BERT with Disentangled Attention*. In *Proceedings of the 9th International Conference on Learning Representations (ICLR 2021)*.

Kobyliński Ł., Ogrodniczuk M., Rybak P., Przybyła P., Pęzik P., Mikołajczyk A., Janowski W., Marcińczuk M. and Smywiński-Pohl A. (2023). *PolEval 2022/23 Challenge Tasks and Results*. In Ganzha M., Maciaszek L., Paprzycki M. and Ślęzak D. (eds.), *Proceedings of the 18th Conference on Computer Science and Intelligence Systems*, vol. 35 of *Annals of Computer Science and Information Systems*, pp. 1237–1244.

Kocoń J., Cichecki I., Kaszyca O., Kochanek M., Szydło D., Baran J., Bielaniewicz J., Gruza M., Janz A., Kanclerz K., Kocoń A., Koptyra B., Mieleszczenko-Kowszewicz W., Miłkowski P., Oleksy M., Piasecki M., Łukasz Radliński, Wojtasik K., Woźniak S. and Kazienko P. (2023). *ChatGPT: Jack of all trades, master of none*. Information Fusion, 99, p. 101861.

Koptyra B., Ngo A., Radliński Ł. and Kocoń J. (2023). *CLARIN-Emo: Training Emotion Recognition Models Using Human Annotation and ChatGPT*. In Mikyška J., de Mulatier C., Paszynski M., Krzhizhanovskaya V. V., Dongarra J. J. and Sloot P. M. (eds.), *Computational Science – ICCS 2023*, pp. 365–379, Cham. Springer Nature Switzerland.

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.

Transferring Knowledge and Tackling Imbalance

Cezary Kęsik
(University of Warsaw)

Abstract

This paper presents a contribution to PolEval 2024 Task 2: Emotion and sentiment recognition. Simple approaches based on existing models are compared with the aim of finding a solution that improves predictions on underrepresented labels. The overall approach to the task involves testing these solutions on sentences, and then reusing those models for further training on the level of whole reviews in order to use knowledge gained from the first stage of training.

Keywords

natural language processing, multilabel classification, emotion classification, sentiment classification, imbalanced dataset

1. Introduction

Emotion recognition is a widely recognized task in the field of NLP. As data-labeling conventions differ, much of the previous data that is available was based on sentences with a single label, which would express the dominant emotion. It is a strategy that has been endorsed in order to create systems that are confident in identifying primary emotions from text, which would have high agreement across annotators (Islam et al. 2022).

While it is a valid approach, a single sentence can often convey multiple emotions. Human perception of emotion from a given text might be influenced not only by a sentence itself, but also by the context in which it is embedded, which presents another challenge for the correct classification of an emotion. This calls for datasets and methods capable of detecting the main emotion expressed in the text, but also other emotions, which might come from the sentence-level understanding or from the influence of context.

2. The task and the dataset

The goal of this task is to classify emotion and sentiment in a way that maximizes balance between categories, which means an evaluation based on F1 macro score. Due to this, extra focus should be placed on labels that are challenging, either as a result of conflicting emotions or a minority label appearing in a sentence.

The dataset consists of 8 emotion labels and 3 sentiment labels, marked as true or false depending on whether the emotion or sentiment was detected. As the authors of the task note, a label was selected if at least 2 out of 6 annotators agreed on it, which means that there is a good chance that opposing labels will appear in the dataset. A brief analysis of the data revealed:

- 529 occurrences of joy and sadness in the same sentence
- 955 occurrences where positive or negative sentiment appeared with neutral in the same sentence

Out of these 955 occurrences, as many as 80 sentences were marked as positive, negative, and neutral at the same time.

Additionally, some labels rarely appeared in the dataset, such as “fear” (about 4% of sentences) or surprise (about 6.5% of sentences). This sharply contrasts with labels that are well represented, such as “joy” (about 48% of sentences) or “sadness” (about 43% of sentences).

Based on this insight, strategies that were aimed to increase F1 scores on underrepresented labels were chosen for the main experiment.

3. Initial experiments

In order to simulate the task, the data provided was first split into train, validation and test datasets. The reviews were randomly shuffled and split in a 70–15–15 proportion, which recreated the conditions of final evaluation, where two test of 167 reviews each were given. It has to be noted here, that as the result of the split, the number of sentences was quite similar, while the number of reviews differed from that of the final evaluation (116 in the validation split and 117 in the test split). That could mean that the results from reviews based classifier could be more prone to variation in the experiment.

As the main aim of the task is to provide a solution that will successfully classify emotion and sentiment in sentences and on the level of whole reviews, 4 approaches were selected for comparison and evaluated on their utility in dealing with underrepresented labels.

The general approach was to train a single model for prediction of emotion and sentiment on the level of sentences, and a single model on the level of whole reviews. A baseline model was prepared and then adjusted by employing strategies addressing low performance on underrepresented labels.

3.1. Sentence models

Taking advantage of pretrained models has been an established practice that leverages the knowledge gained from general training (Singh et al. 2024). For this reason, 2 pretrained models were selected for fine-tuning:

1. KARTONBERT-USE-BASE-V1 – a small model with robust performance, used for initial exploration and comparison on the sentence level predictions between 4 approaches described later.
2. POLISH-ROBERTA-LARGE-V2 – a significantly larger model used for comparison on the sentence level as well as the whole review level.

Apart from this, several approaches were considered for the initial experiment. Literature suggests that resampling methods might be beneficial for multilabel classification tasks (Tarekegn et al. 2021) as well as methods that are meant to focus on examples that are hard to classify by reducing loss for examples that are easily classified (Luo et al. 2024). Additionally, a single data augmentation technique was selected to see if it improves the general classification.

Consequently, these approaches were considered for the sentence level data:

1. baseline model – a pretrained model fine-tuned on training data with a standard binary cross-entropy loss function.
2. model with a sampler for imbalanced datasets – a model with a sampler to address label imbalance by oversampling rare labels and undersampling frequent ones.
3. model with focal loss – a model utilizing a specialized loss function to target examples that are hard to classify.
4. model with back translated data – a model trained on augmented data created by translating the training set to English and back to Polish.

These approaches were initially tested on the KARTONBERT model in order to determine optimal training parameters based on the validation data. The following parameters were selected for the first three models:

- optimizer: AdamW
- batch size: 8
- learning rate: 2e-5
- number of epochs: 8
- weight decay: 0.01
- warmup steps: 100

The same parameters were applied to the model with back translated data, the only difference was that the number of epochs was reduced to 5.

Another crucial aspect of this approach was choosing the right split value for predictions at the last layer with sigmoid activation. While 0.5 is a typical split, lowering this value for all labels was a promising approach to correctly classify more examples from underrepresented

Table 1: Performance results for KartonBERT models on sentences (best scores in bold)

KartonBERT version	Joy	Trust	Anticipation	Surprise	Fear	Sadness
base (0.25)	0.8904	0.7568	0.6372	0.4706	0.5316	0.8763
sampler (0.35)	0.8770	**0.7754**	**0.6580**	0.4068	0.4928	**0.8817**
focal (0.35)	0.8786	0.7670	0.6383	0.4793	**0.5588**	0.8706
translation (0.25)	0.8795	0.7638	0.6341	**0.5072**	0.4878	0.8799
	Disgust	**Anger**	**Positive**	**Negative**	**Neutral**	**F1 macro**
base (0.25)	0.6995	**0.6792**	**0.8861**	0.8762	0.7799	0.7348
sampler (0.35)	0.6950	0.6276	0.8835	**0.8853**	**0.7898**	0.7248
focal (0.35)	**0.7269**	0.6732	0.8800	0.8832	0.7636	**0.7381**
translation (0.25)	0.7086	0.6709	0.8855	0.8770	0.7890	0.7348

Table 2: Performance results for RoBERTa models on sentences (best scores in bold)

RoBERTa version	Joy	Trust	Anticipation	Surprise	Fear	Sadness
base (0.25)	0.9001	0.7717	**0.7048**	**0.5512**	0.5385	0.8977
sampler (0.35)	**0.9057**	0.7720	0.6980	0.5138	0.5263	0.8966
focal (0.35)	0.8864	0.7705	0.6820	0.4957	**0.5833**	0.8887
translation (0.25)	0.8998	**0.7814**	0.6770	0.4783	0.4935	**0.9070**
	Disgust	**Anger**	**Positive**	**Negative**	**Neutral**	**F1 macro**
base (0.25)	0.7470	0.6599	0.9013	0.8989	0.8141	**0.7622**
sampler (0.35)	0.7475	0.6920	**0.9028**	0.9021	0.7969	0.7594
focal (0.35)	0.7494	0.6646	0.8923	0.8972	0.8083	0.7562
translation (0.25)	**0.7617**	**0.7051**	0.9006	**0.9048**	**0.8162**	0.7568

labels. At the same time, lowering the threshold did not significantly affect labels with high counts, such as "joy" or "sadness", as the model was quite confident in predicting these.

Eventually, 2 split values were chosen. For the base and translation model a small value of 0.25 was used and a value of 0.35 for the sampler model and focal loss model. This approach was motivated by experiments on the validation dataset, which showed that without the usage of special sampling or focal loss the models were not confident at predicting underrepresented labels. On the other hand, models that focused on underrepresented labels appeared to have learned more about these labels and a higher split value could be chosen for these.

Following the validation procedure used to estimate parameters, the training and validation sets were combined for training and prediction on the previously mentioned final hold-out test set.

The experiments revealed that the proposed approaches did not bring a substantial change to the final scores. However, training with a special sampler or focal loss demonstrated improvements in F1 scores for underrepresented labels within the scope of this dataset. This is more evident in the case of the KartonBERT models, where f1 scores for labels like "trust", "anticipation", "fear" and "disgust" were increased by approximately 2

These differences were not noticeable in the context of RoBERTa models. A possible caveat was the use of the same parameters for the RoBERTa models as those used for KartonBERT. It is possible that better parameters could have been selected for the bigger model if experiments to determine those parameters were conducted.

Overall, both initial exploration and the test set indicated that using the base model with a low split value for predictions (in the range of 0.25-0.30) was the best approach and it was the one used in the best submitted models, which will be discussed later.

3.2. Text models

The models for this task were derived from previously trained sentence level models. The reasoning behind this experiment was that models trained on sentences would already possess knowledge of emotion and sentiment that could be reused in the context of this task. Initial exploration also confirmed that such approach would produce reliable results, although unstable ones given the fact that the number of reviews in the final test sets was relatively small.

For these models the approach for training was not reused, meaning that models previously trained with a focal loss or a sampler were not retrained using these approaches. This decision was based on findings from earlier experiments, which showed that these methods actually worsened model performance. Each sentence level model was subsequently retrained with the basic approach of using standard binary cross entropy loss.

The following parameters were selected for further training on reviews:

1. Baseline model:
   - optimizer: AdamW
   - batch size: 2
   - learning rate: 2e-5
   - number of epochs: 8
   - weight decay: 0.01
   - warmup steps: 100
2. Other models:
   - optimizer: AdamW
   - batch size: 2
   - learning rate: 2e-5
   - number of epochs: 4
   - weight decay: 0.01
   - warmup steps: 100

Table 3: Performance results for RoBERTa models on whole reviews (best scores in bold)

RoBERTa version	Joy	Trust	Anticipation	Surprise	Fear	Sadness
text base (0.2)	0.9014	**0.8169**	0.4762	0.4211	**0.8000**	0.9062
text sampler (0.2)	**0.9057**	0.7720	**0.6980**	**0.5138**	0.5263	0.8966
text focal (0.2)	0.9041	0.7733	0.4348	0.2857	0.6667	**0.9231**
text translation (0.2)	0.9014	0.7826	0.4545	0.3000	0.6667	0.9077
	Disgust	**Anger**	**Positive**	**Negative**	**Neutral**	**F1 macro**
text base (0.2)	0.8060	**0.7451**	**0.9315**	0.9160	0.9091	**0.7844**
text sampler (0.2)	0.7475	0.6920	0.9028	0.9021	0.7969	0.7594
text focal (0.2)	0.7826	**0.7451**	0.9272	**0.9242**	0.8936	0.7509
text translation (0.2)	**0.8333**	0.7059	0.9262	0.9173	**0.9167**	0.7556

Table 4: Final macro scores on TEST-A and TEST-B on sentences and whole reviews

	TEST-A sentence	TEST-A text	TEST-A final	TEST-B sentence	TEST-B text	TEST-B final
RoBERTa v.1.85	74.94	76.42	75.68	76.66	**79.33**	**77.99**
RoBERTa v.10	**76.03**	74.59	75.31	75.86	78.96	77.41

The choice to reduce training time for other models was influenced by the consideration that models which had more data or were trained more intensively on underrepresented labels might overfit if trained too much.

For these models a uniform split value of 0.2 was used to ensure that underrepresented labels will be included.

All of these models performed quite well, with F1 macro scores similar to the ones achieved by sentence level models, the base model stood out in performance. While this might be attributed to a favorable data split, it is not likely as this model achieved reasonable performance on the two final test sets.

4. Final test results

Below are two best solutions that were based on the fine-tuned RoBERTa model with previously established parameters.

4.1. Text model

As noted earlier, the text model was expected to exhibit variability due to the hard to classify examples. This issue seems to have occurred here, although to a lesser degree in the v.1.85 model. The prediction threshold was set at 0.2, consistent with the initial experiment. This value again proved to be reasonable, potentially improving the classification of underrepresented labels. The v.1.85 model was experimentally trained on 85% of the provided data, while the other model was trained on all of the data.

4.2. Sentence model

The F1 macro scores for the sentence models were consistent with earlier experiments, which indicated expected values around 76% with a prediction threshold in the range of 0.25-0.35, and these scores prove to be a reliable representation of sentence level model performance.

5. Final considerations

The proposed approach, while achieving reasonable results, is far from perfect. There is significant room for improvement and enhancement of stability of the whole review model. One possible approach would be to implement an ensemble method by combining predictions from several models and deciding whether to pass a label as true if the majority of models agree on that label.

References

Islam M. A., Mukta M. S. H., Olivier P. and Rahman M. M. (2022). *Comprehensive Guidelines for Emotion Annotation*. In *Proceedings of the 22nd ACM International Conference on Intelligent Virtual Agents*, IVA '22, New York, NY, USA. Association for Computing Machinery.

Luo J., Yuan Y. and Xu S. (2024). *Improving GBDT Performance on Imbalanced Datasets: An Empirical Study of Class-Balanced Loss Functions*. arXiv:2407.14381.

Singh A., Pandey N., Shirgaonkar A., Manoj P. and Aski V. (2024). *A Study of Optimizations for Fine-tuning Large Language Models*. arXiv:2406.02290.

Tarekegn A. N., Giacobini M. and Michalak K. (2021). *A Review of Methods for Imbalanced Multi-label Classification*. Pattern Recognition, 118, p. 107965.

Generating Opinions for Emotion Recognition in Polish

Katarzyna Baraniak
(Independent researcher)

Abstract

This paper presents a solution for PolEval 2024 competition Task 2: Emotion and sentiment recognition. The solution is based on two language models: one for generation of new samples and second for classification. Our approach presents how to simply improve the performance of classifier by generation of new samples.

Keywords

emotion recognition, large language model, generative llm

1. Introduction

Emotion and sentiment recognition in text is important natural language processing task for several reasons, as it plays a crucial role in enhancing the understanding of human communication, both in text and speech.

2. Goal

The aim of the competition (Kobyliński et al. 2023) is to detect all emotions and sentiments in a given opinion. Emotions to detect are: Joy, Trust, Anticipation, Surprise, Fear, Sadness, Disgust, Anger. Sentiment can be positive negative or neutral.

3. Data

Samples are provided as separate sentences or sequence of # with assigned emotions and sentiments. Sequence of # denoted the whole opinion. We decided to join all the sentences to create the opinion and treat in the same way as single sentences. One sentence or opinion can have multiple emotions and sentiments.

4. Solution

Our solution is based on two language models, one for generating new samples and another one for classification. At the beginning we fine tuned existing language model for classification on raw data and it gives us reasonable good results. We decide to improve the score by generating new samples from existing ones. Details of models, hyperparameters and code may be found at `https://github.com/Katarzynaa/poleval2024`.

4.1. Model for text generation

We decided to fine-tune one of the most common large language model: meta-llama/Meta-Llama-3-8B using original data. We also experimented with translated data but the results were lower, probably due to the quality of translation so we did not continue that direction. Other models than llama based were not tested. We prepared training data with proper prompt:

```
"\#\#\# Instrukcja: Napisz recenzję, która wyraża podane emocje: "  \#
"Napisz recenzję, która wyraża emocje " + sentimentpl +
"\#\#\# Recenzja: "+ headline}
```

where `sentimentpl` was representing original emotions and headline the original opinion. After the fine tuning, using the same prompt template without opinion part, we generated new opinions. We add information that opion is about one of four categories: hotel. restaurant, doctor, school. We randomly pick two emotions and one sentiment. Given those information model has to generate 1000 opinions for each category.

4.2. Quality of generated opinions

Quality of generated opinions varies. Most of the opinions are like those written by humans. Other opinions contain short grades, repeated emotions, dates or other words that does not create opinion. Surprisingly, model quite correctly generated opinions from given categories. Model often adds dates at the beginning, some symbols or grades which should be further investigated to create better quality opinions.

4.3. Classification model

We used pretrained model `sdadas/polish-roberta-large-v2` for multilabel classification in a standard way adding sigmoid function after the last layer. We finetunes the model on the original and generated data and that performed prediction on test sets.

5. Results

Results are presented in Table . We present results that where achieved by model without samples generation, with generated 1 000 new samples from any category, and with generated 4 000 new samples in total containing equal number of samples from each category.

Table 1: F1 Scores for Different Opinion Sets

Setting	**test-A F1 scores**			**test-B F1 scores**		
	Sentence	**Text**	**Final**	**Sentence**	**Text**	**Final**
4 000 generated opinions	75.94	77.47	76.70	76.11	77.76	76.94
1 000 generated opinions	76.69	75.49	76.09	75.65	75.81	75.73
without new opinions	74.75	75.48	75.12	75.65	76.31	75.98

6. Conclusion and future work

We showed that generating new samples can improve emotion and sentiment recognition. Still, our solution needs further improvements. First of all, the quality of generated opinions has to be controlled. What is more, other large language models may be tested. The whole opinion needs different classification method because the text is much longer than possible input to the model used.

References

Kobyliński Ł., Ogrodniczuk M., Rybak P., Przybyła P., Pęzik P., Mikołajczyk A., Janowski W., Marcińczuk M. and Smywiński-Pohl A. (2023). *PolEval 2022/23 Challenge Tasks and Results*. In Ganzha M., Maciaszek L., Paprzycki M. and Ślęzak D. (eds.), *Proceedings of the 18th Conference on Computer Science and Intelligence Systems*, vol. 35 of *Annals of Computer Science and Information Systems*, pp. 1237–1244.

Emotion and Sentiment Recognition using Ensemble Models

Jakub Kosterna (Independent researcher)

Abstract

This paper presents a solution to the emotion and sentiment recognition task within the PolEval 2024 competition. The approach is based on an ensemble model combining five different "classic" machine learning algorithms, trained on numerical representations of text, transformed into 76 features derived from multiple variations of the input data and pretrained models specialized in Polish language tasks. This work adopts an experimental approach using relatively simple, pretrained models without the author's prior familiarity with more complex emotion recognition solutions. Results highlight the ensemble model's foundational potential in addressing multi-label classification tasks for emotions and sentiment, while also identifying valuable areas for future improvement and optimization.

Keywords

emotion recognition, sentiment analysis, ensemble models, natural language processing

1. Introduction

Understanding human emotions and sentiments through text analysis has long been a challenging yet promising avenue in natural language processing (NLP). Emotions are often subtle, context-dependent, and nuanced, making their recognition a complex task that requires sophisticated approaches to capture underlying sentiments accurately. The PolEval 2024 competition presented an opportunity to explore this challenge by focusing on the classification of emotions, as structured by Plutchik's wheel of emotions, and the analysis of sentiment in Polish-language consumer reviews.

In this project, the objective was to design a model capable of synthesizing various text representations and integrating predictions from multiple machine learning models. By leveraging diverse approaches within an ensemble framework, the goal was to capture the rich emotional and sentiment landscape in text, with an eye toward identifying areas for refinement and future advancement in NLP-driven emotion recognition.

2. Model Description

The approach involved building an ensemble model combining five well-known machine learning algorithms. These models were trained on numerical representations of text, transformed into 76 features.

2.1. Text Representation and Feature Engineering

The feature extraction process involved four distinct versions of the original reviews, each providing 19 features. These versions were:

1. **in_baseline**: The original dataset provided by the competition organizers.
2. **in_gpt_corr**: A corrected version of `in_baseline` using the Chat GPT-3.5 Turbo model with the prompt "Correct the following text to proper Polish."
3. **in_prep_bas**: The `in_baseline` dataset after preprocessing steps (converting to lowercase, removing non-word characters, and stripping extra spaces).
4. **in_prep_gpt**: The `in_gpt_corr` dataset after the same preprocessing steps.

Each of these versions provided 19 features, resulting in 76 features per example. These features were obtained using:

- **LSTM Model**: A Long Short-Term Memory network was used to process the text and generate 11 features.
- **Pretrained Models from Hugging Face** (`hfam`):
  - **Herbert** (Mroczkowski et al. 2021): `dkleczek/Polish-Hate-Speech-Detection-Herbert-Large`
  - **XLM-RoBERTa** (Conneau et al. 2021): `cardiffnlp/twitter-xlm-roberta-base-sentiment-multilingual`
  - **Multilingual BERT** (Devlin et al. 2019): `nlptown/bert-base-multilingual-uncased-sentiment`

2.2. Ensemble Approach

The ensemble model consisted of the following classifiers:

1. **Random Forest**
2. **XGBoost**
3. **MLP** (Multi-Layer Perceptron)
4. **CNN** (Convolutional Neural Network)
5. **Naive Bayes**

The final prediction for each category was determined by majority voting. If at least three of the five models agreed on a class, it was selected as the final prediction.

3. Data

3.1. Data Preparation

The data was organized into three directories: `train`, `testA`, and `testB`. Each directory contained the four data versions described earlier and a file with numerical observations (`concated_for_ensemble_final.csv`).

3.2. Numerical Observations

Each `concated_for_ensemble_final.csv` file contained 76 numerical features per example, including both discrete and continuous variables within the range [0, 1]. The features came from the four data versions and included:

- **11 features** from the LSTM model
- **2 features** from the Herbert model
- **4 features** from the XLM-RoBERTa model
- **2 features** from the Multilingual BERT model

4. Ensemble Model Construction

4.1. Model Definitions and Hyperparameter Tuning

Random Forest, XGBoost, and MLP models were tuned using `GridSearchCV`. The CNN model was defined with the appropriate architecture, while the Naive Bayes model was trained separately for each label.

4.2. Model Training

Each model was trained on the training data:

- **Random Forest, XGBoost, MLP**: trained as multi-label models with tuned hyperparameters.
- **CNN**: trained on data transformed to the appropriate input shape.
- **Naive Bayes**: separate models were trained for each label.

4.3. Achieved Accuracies

The accuracies achieved by each model are shown in Table 1.

Table 1: Model accuracies on the validation set

Model	**Accuracy**
Random Forest	92.05%
XGBoost	91.49%
MLP	91.49%
CNN	89.19%
Naive Bayes	90.41%

4.4. Model Evaluation

After training, the models were evaluated on the validation set. An ensemble approach combined the predictions of all models through majority voting. Predictions were saved and used to generate final predictions on the test sets `testA` and `testB`.

5. Input Data Preparation Details

For each of the four data variants, 19 features were generated:

- **11 features** from predictions of the LSTM model, which had the following architecture:
  - An embedding layer with dimensions (5000, 128)
  - An LSTM layer with 64 units
  - A dense layer with 11 units and sigmoid activation
- **2 features** from the Herbert model (`dkleczek/Polish-Hate-Speech-Detection-Herbert-Large`)
- **4 features** from the XLM-RoBERTa model (`cardiffnlp/twitter-xlm-roberta-base-sentiment-multilingual`), including sentiment categories and confidence scores
- **2 features** from the Multilingual BERT model (`nlptown/bert-base-multilingual-uncased-sentiment`)

6. Conclusion

The proposed ensemble approach demonstrates the feasibility of applying multiple models to emotion and sentiment recognition in Polish consumer reviews. Leveraging diverse model architectures and rich text representations provided insightful results, highlighting both strengths and opportunities for further refinement in classification accuracy.

Acknowledgements

I would like to express my gratitude to the PolEval 2024 organizers for the opportunity to participate in this challenge, marking my first experience in such an event. Special thanks to Grzegorz Wojdyga, who, through his course 'Language Engineering' in the Cognitive Science (from the Department of Philosophy at the University of Warsaw) program, introduced me to this competition and encouraged my participation.

Despite the less-than-perfect results, I am deeply grateful to the organizers for inviting me to contribute to this publication. This experience has inspired me to pursue and experiment with more advanced and practical solutions in future projects.

References

Conneau A., Baevski A., Collobert R., Mohamed A. and Auli M. (2021). *Unsupervised Cross-Lingual Representation Learning for Speech Recognition*. In *Proceedings of Interspeech 2021*, pp. 2426–2430.

Devlin J., Chang M.-W., Lee K. and Toutanova K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. In Burstein J., Doran C. and Solorio T. (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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.

Zero-shot Approach Using Bielik LLM for Emotion Recognition in Polish

Paweł Cyrta
(Stenograf.ai, SpeakLeash)

Abstract

Unlike conventional methods that rely on fine-tuning or few-shot learning, our solution explores baseline approach with the zero-shot capabilities of the Bielik 11B v2.3 Instruct model, a large-scale Polish language model, for multi-label emotion classification. The study investigates the model's ability to identify eight distinct emotions through direct instruction-based prompting, without additional training or exemplar-based learning. This straightforward approach demonstrates the potential of leveraging pre-trained Polish language models for complex affective computing tasks while minimizing computational overhead and training requirements.

Keywords

emotion recognition, large language model, Polish, natural language processing,

1. Introduction

This study contributes to the broader research initiative presented by Kobyliński et al. (2023), which focuses on multi-label emotion recognition in Polish language textual data.

The primary objective is to develop and evaluate computational methods for the concurrent identification of eight distinct emotions (Joy, Trust, Anticipation, Surprise, Fear, Sadness, Disgust, and Anger) as defined by Plutchik's wheel of emotions.

This study employs a zero-shot learning paradigm utilizing Bielik 11B v2.3 Instruct, a large-scale Polish language model comprising 11 billion parameters. The methodological approach deliberately eschews traditional fine-tuning techniques and few-shot learning strategies in favor of exploring the model's inherent capabilities through direct prompting. This methodology represents a significant departure from conventional approaches that typically rely on task-specific training or exemplar-based learning.

2. Experiment

The foundation of our approach is the Bielik 11B v2.3 Instruct model (Ociepa et al. (2024)), a state-of-the-art Polish language model continuously pre-trained and large corpora and fine-tuned for instruction-following tasks by SpeakLeash community team. The model's architecture leverages transformer-based technology optimized for Polish language understanding and generation.

The zero-shot approach implemented in this study consists of 4 crafted prompts that directly instruct the model to perform emotion recognition without any additional training or example-based guidance. This methodology tests the model's ability to transfer its pre-trained knowledge to specific emotion recognition tasks without task-specific optimization.

I have used Q4 quantized version `Bielik-11B-v2.3-Instruct-GGUF-IQ-Imatrix-Q4_K_M` hosted locally using Ollama server and query as input following prompts:

A.
```
"Napisz jaka emocja jest w tekście z poniższych: Joy, Trust,
Anticipation, Surprise, Fear, Sadness, Disgust,  Anger
TEKST: <tutaj tekst do oceny>
```

B.
```
"Napisz jaka emocja jest w tekście z poniższych: Joy, Trust,
Anticipation, Surprise, Fear, Sadness, Disgust,  Anger
Odpowiedz tylko w formie wpisu w tabeli np.: dla Joy
1 0 0 0 0 0 0 0
TEKST: <tutaj tekst do oceny>
```

C.
```
"Emocje:  Joy, Trust, Anticipation, Surprise, Fear, Sadness,
Disgust, Anger
Jaką emocję z tej listy reprezentuje ten poniższy tekst:
<tutaj tekst do oceny> "
```

D.
```
"Emocje:  Joy, Trust, Anticipation, Surprise, Fear, Sadness,
Disgust, Anger
Odpowiedz w formie wpisu w tabeli np.: dla Joy
1 0 0 0 0 0 0 0
Jaką emocję reprezentuje ten poniższy tekst:
<tutaj tekst do oceny> "
```

3. Results

	Accuracy train set
prompt A	29.05
prompt B	17.49
prompt C	38.43
prompt D	24.11

4. Conclusion and future work

This study investigated the efficacy of a zero-shot approach to emotion recognition using a commonly used quantized version of the the Bielik 11B v2.3 Instruct model in the context Task2 of the PolEval 2024 challenge.

The zero-shot approach revealed both strengths and limitations of the model. While the system demonstrated capability in identifying basic emotions, the performance is rather low across all emotional categories. This result suggests that pre-trained knowledge in the model have some emotional concepts encoded, potentially reflecting patterns in the model's training data or in very limited emotion recognition instructions.

Acknowledgements

We gratefully acknowledge Poland's high-performance Infrastructure PLGrid ACK Cyfronet AGH for providing computer facilities and support within computational grant no PLG/2024/016951.

References

Kobyliński Ł., Ogrodniczuk M., Rybak P., Przybyła P., Pęzik P., Mikołajczyk A., Janowski W., Marcińczuk M. and Smywiński-Pohl A. (2023). *PolEval 2022/23 Challenge Tasks and Results*. In Ganzha M., Maciaszek L., Paprzycki M. and Ślęzak D. (eds.), *Proceedings of the 18th Conference on Computer Science and Intelligence Systems*, vol. 35 of *Annals of Computer Science and Information Systems*, pp. 1237–1244.

Ociepa K., Flis Ł., Kinas R., Gwoździej A. and Wróbel K. (2024). *Bielik 2: A Family of Large Language Models for the Polish Language — Development, Insights, and Evaluation*. Manuscript in preparation.

PolEval 2024 Task 3: Polish Automatic Speech Recognition Challenge

Michał Junczyk, Iwona Christop
(Adam Mickiewicz University in Poznań)
Piotr Pęzik
(University of Łódź)

Abstract

The Polish Automatic Speech Recognition (ASR) Challenge was designed to facilitate open, multi-domain evaluations on a variety of Polish speech datasets, similar to the evaluation practices used for English. The goal was to encourage the Polish ASR community to adopt common test sets for benchmarking: BIGOS V2 (containing read speech) and PELCRA for BIGOS (containing conversational speech). The combined datasets comprise the most comprehensive publicly available evaluation of Polish ASR systems, accounting for the number of speakers, devices, and acoustic conditions. The solutions provided by challenge participants were compared to baselines consisting of selected proprietary and freely available ASR systems that support the Polish language.

Keywords

automatic speech recognition, Polish, ASR, Bigos, Pelcra, speech dataset

1. Introduction

Automatic speech recognition (ASR) has made significant progress over the last decade. Accuracy levels are now on a par with human transcription, at least for some languages, domains, and speech characteristics.

The main goal of the Polish ASR Challenge was to promote open, multi-domain evaluation on a wide range of speech datasets, similar to the practices used for English (Gandhi et al. 2022, Srivastav et al. 2023) and German languages (Wirth and Peinl 2022).

Two large corpora (BIGOS and PELCRA) have been made available for this purpose. The *BIGOS V2* corpus comprises curated recordings and metadata from 12 open ASR speech datasets, mostly read speech, e.g. Common Voice, MLS, CLARIN. More details can be found on Hugging Face (Junczyk 2024c). The *PELCRA for BIGOS* corpus contains selected recordings and metadata from the PELCRA repository, curated to be compatible with the BIGOS format. The PELCRA corpus contains mostly spontaneous and conversational speech. It is based on the SpokesMix (Pęzik 2018), SpokesBiz (Pęzik et al. 2023) and DiaBiz (Pęzik et al. 2022) corpora. More details are available on Hugging Face (Pęzik and Junczyk 2024).

Combined, BIGOS V2 and PELCRA for BIGOS provide the most comprehensive and easy-to-use Polish speech corpora in terms of number of speakers, devices and acoustic conditions.

2. Task definition

The objective of this challenge was to benchmark community-provided ASR systems against other available ASR solutions for Polish. At the end of the challenge, baseline results for 25 commercial and open-source systems supporting Polish were published on the Polish ASR Leaderboard (Junczyk 2024a), using the test sets provided.

Participants were provided with training, development, and test sets, from the BIGOS[1] and PELCRA[2] corpora. Both datasets were available on Hugging Face. While the results for `test-A` were visible from the start, the final ranking was based on the performance of the systems on `test-B` set, and was made available after the submissions had been closed.

Participants were allowed to both create their own system, and fine-tune an existing solution. However, they were required to provide a relevant description for the submission. It was forbidden for participants to use any data outside of the provided train and validation sets to develop their systems. It was also prohibited to manually transcribe the test examples.

3. Dataset

The dataset was randomly divided into four splits – `train`, `dev-0`, `test-A`, and `test-B`. The distribution of samples within each split is shown in Table 1.

Table 1: Distribution of samples within each split of the Polish ASR Challenge dataset.

Split	**No. samples from BIGOS**	**No. samples from PELCRA**	**Total**
`train`	82 025	229 150	311 175
`dev-0`	14 254	28 532	42 786
`test-A`	1 002	1 167	2 169
`test-B`	991	1 178	2 169
Total	98 272	260 027	358 299

[1] `https://huggingface.co/datasets/amu-cai/pl-asr-bigos-v2`

[2] `https://huggingface.co/datasets/pelcra/pl-asr-pelcra-for-bigos`

All splits were stored in a separate directory, and the corresponding files followed the same structure. For each split, an `in.tsv` file was provided, containing the input data for the relevant set. The `in.tsv` was in tabular format, comprising four columns:

- `dataset` – the name of the dataset, which could be found on Hugging Face, i.e. `amu-cai/pl-asr-bigos-v2` or `pelcra/pl-asr-pelcra-for-bigos`,
- `subset` – the subset of the given dataset, as listed on Hugging Face,
- `split` – the split of the subset, as listed on Hugging Face,
- `audioname` – the file ID, as listed on Hugging Face.

An example of the `in.tsv` format is provided below for reference.

```
amu-cai/pl-asr-bigos-v2  fair-mls-20  train  fair-mls-20-train-0022-00001
amu-cai/pl-asr-bigos-v2  fair-mls-20  train  fair-mls-20-train-0022-00002
```

While the text data was provided in a tab-separated file, the audio files were to be accessed via Hugging Face.

Additionally, for the `train` and `dev-0` splits, and `expected.tsv` file was provided. This was also tab-separated and contained one column, with each row representing a transcription of the matching audio recording from the `in.tsv` file.

4. Evaluation

4.1. Submission format

The objective of the task was to generate an accurate transcription for each utterance. The submission was required to be in a form of a single tab-separated file, with a single column. Each line in the `out.tsv` file was to contain a hypothesis for the corresponding audio recording from the `in.tsv` file. An example of the `out.tsv` file is shown below.

```
szum mnoży w skałach okolicznych staje się rzeką a w gwałtownym pędzie
pieni się huczy i zżyma w bałwany tym sroższy w biegu im dłużej wstrzymany
lecą sandały i trepki i pasy wrzawa powszechna przeraża i głuszy zdrętwiał
hyacynt na takie hałasy chciałby uniknąć bitwy z całej duszy a przeklinając
nieszczęśliwe czasy resztę kaptura nasadził na uszy
```

4.2. Metrics

Two measures of accuracy were calculated for each provided submission:

- **Word Error Rate** (WER) - number of incorrectly transcribed words divided by the total number of tokens in the reference sentences.

$$\text{WER} = \frac{\text{number of errors}}{\text{reference text length in words}}$$

— **Character Error Rate** (CER) - number of incorrectly transcribed characters divided by the total number of characters in the reference sentences.

$$\text{CER} = \frac{\text{number of errors}}{\text{reference text length in characters}}$$

Both metrics range from 0 to 1, with 0 being the best score.

4.3. Text normalization

As some of the references lacked punctuation and capitalization, the evaluation was performed on the normalized text to reduce the likelihood of false errors. All punctuation marks were removed, and case folding was applied. As the normalization was conducted during the evaluation process, there was no necessity for post-processing on the part of the participants.

4.4. Baseline

Following the conclusion of the challenge, the baseline results for 25 commercial and open-source systems supporting Polish were published on the Polish ASR Leaderboard (Junczyk 2024a), using the test sets provided. Further details on the evaluated systems can be found in the relevant publication (Junczyk 2024b).

5. Submission and results

The submissions were provided by three participants – LIT-MR, ryssta, and Paweł Cyrta. Following consultation with the participants, it was decided that only the results of the LIT-MR team would be reported, as the other participants had only submitted a single result and had not marked it as official.

The results from the winning team, LIT-MR, are presented in Table 2.

Table 2: Submissions on the PolEval results leaderboard. Test results excluded. Source: `https://beta.poleval.pl/challenge/2024-asr-bigos`

Description	test-A CER	test-A WER	test-B CER	test-B WER
whisper-large-v3-mix-01-50k	6.97	11.25	7.28	11.49
whisper-large-v3-mix-00-18k-beam4	**6.85**	**11.07**	6.91	11.15
whisper-large-v3-baseline-13k-beam4	7.15	11.52	7.10	11.23
whisper-large-v3-mix-01-25k-beam4	6.90	11.27	**6.85**	**11.07**
conformer-baseline-500k	8.77	17.48	8.37	16.82
conformer-mix-00-500k	7.60	15.25	7.16	14.33
conformer-mix-01-500k	7.08	13.99	6.90	13.40
whisper-large-v3-mix-01-25k	6.58	11.83	7.34	13.00

The submitted models were fine-tuned Whisper v3 models on both the originally provided and synthetic training data. The best-performing submissions achieved an approximate WER of 11% and CER of 7% for both test sets A and B.

The LIT-MR team achieved significantly superior results compared to the baseline of vanilla Whisper-large-v3 model, which obtained WER of 14.51% and 14.02% for test sets A and B, respectively. These findings are shown in Tables 3 and 4.

Table 3: WER and CER scores for test set A for selected ASR systems supporting Polish language and PolEval best submission. Source: (Junczyk 2024a)

System	WER [%]	CER [%]
poleval_best_lit_mr	11.07	6.85
whisper_large_v3	14.51	8.41
whisper_cloud	15.28	8.72
assembly_best	16.47	9.84
whisper_medium	17.61	9.48
google_v2_long	19.54	12.26
google_long	20.27	13.49
google_short	20.69	13.86
nemo_multilang	22.59	12.02
whisper_small	23.39	11.74
mms_all	25.14	10.48
azure_latest	25.85	19.26
w2v-1b-pl	26.62	9.22
nemo_pl_conformer	27.24	14.43
google_cmd_search	27.81	16.48
google_default	29.13	17.76
google_v2_short	29.64	23.23
mms_1107	30.40	9.99
mms_102	31.20	12.27
w2v-53-pl	37.91	14.45
whisper_base	38.58	18.41
assembly_nano	41.56	29.23
whisper_tiny	55.15	26.24
nemo_pl_quartznet	62.34	23.67

The enhancements were even more notable in comparison to other speech recognition systems that were evaluated on the Open ASR Leaderboard in 2024 (Junczyk 2024a).

The differences in WER and CER between the best-performing Whisper-large-v3 mix models and the baseline Whisper-large-v3 model are as follows:

— **Test Set A:** The best Whisper-mix model achieved a WER of 11.07%, in comparison to the baseline WER of 14.51%, thereby demonstrating an improvement of 3.44 percentage points. The CER improved from 8.41% to 6.85%, representing a 1.56 percentage point increase.

— **Test Set B:** The best Whisper-mix model achieved a WER of 11.07%, in comparison to the baseline WER of 14.02%, resulting in an improvement of 2.95 percentage points. The CER improved from 7.98% to 6.85%, resulting in an improvement of 1.13 percentage points.

A comparison of the LIT-MR team's submissions with other ASR systems for test sets A and B reveals that the fine-tuned Whisper models provided by LIT-MR achieved the highest accuracy among all evaluated systems supporting the Polish language. As shown in Tables 3 and 4, the WER and CER scores achieved by LIT-MR's Whisper-large-v3 fine-tuned models consistently outperformed other systems, including Whisper Cloud, Google ASR variants, and other open-source and commercial ASR solutions. This confirms both the effectiveness of models fine-tuning to in-domain data and synthetic data augmentation methods.

Overall, the fine-tuning conducted by the LIT-MR team resulted in significant improvements over the baseline models, establishing a new standard for Polish ASR performance in 2024.

Table 4: WER and CER scores for test set B for selected ASR systems supporting Polish language and PolEval best submission. Source: (Junczyk 2024a)

System	**WER [%]**	**CER [%]**
poleval_best_lit_mr	11.07	6.85
whisper_large_v3	14.02	7.98
whisper_cloud	14.57	8.19
assembly_best	15.97	9.35
whisper_medium	17.21	9.50
google_short	20.17	13.35
google_v2_long	21.12	14.19
google_long	21.59	15.24
nemo_multilang	22.05	11.72
whisper_small	24.14	12.64
mms_all	24.66	10.11
azure_latest	25.59	19.08
w2v-1b-pl	26.21	9.02
nemo_pl_conformer	26.88	13.86
google_cmd_search	27.98	16.79
google_default	29.39	18.23
mms_1107	29.42	9.43
google_v2_short	29.50	22.75
mms_102	30.69	11.97
whisper_base	38.25	18.10
w2v-53-pl	38.79	14.58
assembly_nano	42.05	29.97
whisper_tiny	57.77	27.71
nemo_pl_quartznet	62.34	23.12

Acknowledgements

The authors thanks the PolEval challenge organizers as well as all participants who submitted solutions. The authors gratefully acknowledge the creators of the original dataset, who either openly shared their work under licenses permitting derivative works or explicitly agreed to the release of curated, derivative content. BibTeX citations for the original authors are available on the *BIGOS* (Junczyk 2024a) and *PELCRA for BIGOS* (Pęzik and Junczyk 2024) Hugging Face pages. We strongly encourage users of the curated BIGOS and PELCRA datasets to reference the original sources and authors of datasets.

References

Gandhi S., Platen P. and Rush A. (2022). *ESB: A Benchmark for Multi-Domain End-to-End Speech Recognition*. arXiv:2210.13352.

Junczyk M. (2024a). *AMU Polish ASR Leaderboard*.

Junczyk M. (2024b). *Framework for Curating Speech Datasets and Evaluating ASR Systems: A Case Study for Polish*. arXiv:2408.00005.

Junczyk M. (2024c). *`pl-asr-bigos-v2` (revision 37cc976)*.

Pęzik P. (2018). *Increasing the Accessibility of Time-Aligned Speech Corpora with Spokes Mix*. In Calzolari N., Choukri K., Cieri C., Declerck T., Goggi S., Hasida K., Isahara H., Maegaard B., Mariani J., Mazo H., Moreno A., Odijk J., Piperidis S. and Tokunaga T. (eds.), *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, pp. 4297–4300, Miyazaki, Japan. European Language Resources Association (ELRA).

Pęzik P. and Junczyk M. (2024). *`pl-asr-pelcra-for-bigos` (revision 4205ec7)*.

Pęzik P., Krawentek G., Karasińska S., Wilk P., Rybińska P., Cichosz A., Peljak-Łapińska A., Deckert M. and Adamczyk M. (2022). *DiaBiz – an Annotated Corpus of Polish Call Center Dialogs*. In Calzolari N., Béchet F., Blache P., Choukri K., Cieri C., Declerck T., Goggi S., Isahara H., Maegaard B., Mariani J., Mazo H., Odijk J. and Piperidis S. (eds.), *Proceedings of the Thirteenth Language Resources and Evaluation Conference (LREC 2002)*, pp. 723–726. European Language Resources Association.

Pęzik P., Karasińska S., Cichosz A., Łukasz Jałowiecki, Kaczyński K., Krawentek M., Walkusz K., Wilk P., Kleć M., Szklanny K. and Marszałkowski S. (2023). *SpokesBiz – an Open Corpus of Conversational Polish*. arXiv:2312.12364.

Srivastav V., Majumdar S., Koluguri N., Moumen A., Gandhi S. et al. (2023). *Open Automatic Speech Recognition Leaderboard*.

Wirth J. and Peinl R. (2022). *ASR in German: A Detailed Error Analysis*. arXiv:2204.05617.

Augmenting Polish Automatic Speech Recognition System With Synthetic Data

Łukasz Bondaruk, Jakub Kubiak (Samsung R&D Institute Poland),
Mateusz Czyżnikiewicz (Samsung AI Center Warsaw)

Abstract

This paper presents a system developed for submission to PolEval 2024, Task 3: Polish Automatic Speech Recognition Challenge[1]. We describe Voicebox-based speech synthesis pipeline and utilize it to augment Conformer and Whisper speech recognition models with synthetic data. We show that addition of synthetic speech to training improves achieved results significantly. We also present final results achieved by our models in the competition.

Keywords

speech recognition, ASR, speech synthesis, TTS, speech processing, natural language processing

1. Introduction

Automatic Speech Recognition (ASR) systems became essential in enabling machines to understand and process human language. However, training these systems, especially for less widely spoken languages like Polish, presents challenges due to the limited availability of high-quality labeled data. To address this issue, and based on recent advancements in speech synthesis systems, we decided to explore the use of synthetic data to augment real datasets. This paper focuses on enhancing Polish ASR by incorporating synthetic speech into training data, aiming to overcome the scarcity of natural speech resources and boost model performance. We test our solution on PolEval 2024, Task 3: Polish Automatic Speech Recognition[1].

[1] `http://poleval.pl/tasks/task3`

2. Related work

Speech recognition for Polish language has been the subject of many scientific articles. Researchers have explored classical approaches, such as HMM (Ziółko et al. 2008), developed custom systems based on kNN classifier and wavelets (Ziółko et al. 2011) and, based on Kaldi toolkit (Povey et al. 2011), trained more robust systems (Ziółko et al. 2015, Koržinek et al. 2016). Topics of adapting models to specific domain (Janicki and Wawer 2011) or to specific acoustic conditions (Koržinek et al. 2019) have also been explored.

In parallel with research on speech recognition systems, an increasing number of speech corpora in Polish have been made available (Grocholewski 1997, Demenko et al. 2008, Marciniak 2010). Some of the most notable ones include AGH Corpus of Polish Speech (Żelasko et al. 2016) with more than 25 hours of data from 166 speakers and CLARIN-PL (Koržinek et al. 2016) with around 56 hours of data from 317 speakers.

With the shift to a multilingual paradigm enabled by the use of deep learning, Polish began to be included in large multilingual speech corpora such as Multilingual LibriSpeech (Pratap et al. 2020) or VoxPopuli (Wang et al. 2021). This led to the development of speech recognition systems where Polish was just one of the available languages. Some of the most notable ones are Wav2Vec 2.0 XLSR-53 (Conneau et al. 2021) and Whisper (Radford et al. 2023). More comprehensive survey of speech corpora for Polish was performed by Junczyk (2023).

Topic of speech recognition for Polish has also been tackled in Task 5: Automatic Speech Recognition of PolEval 2019 (Koržinek 2019)[2]. The goal was to build a system for transcribing sessions of the Polish Parliament. Systems developed by four participating teams achieved results ranging from 41.8% to 11.8% WER.

Prior work demonstrated that using synthetic data can significantly improve speech recognition performance. It was shown that augmentation with synthetic speech can increase the robustness of ASR training, leading to a 38% relative improvement in some systems (Rossenbach et al. 2021). In low-resource languages, adding synthetic data reduced WER by up to 25.5% (Bartelds et al. 2023).

However, augmentation with synthetic speech presented challenges due to the differences between synthetic and real data. This problem was addressed with the development of zero-shot voice-cloning TTS systems such as VALL-E X (Zhang et al. 2023) and Voicebox (Le et al. 2023). Authors of Voicebox compared performance of ASR systems trained on real and synthetic data and observed only small reduction in quality (Le et al. 2023). Topic of augmentation to a specific domain with synthetic speech produced with VALL-E X was also researched (Czyżnikiewicz et al. 2024).

[2] https://2019.poleval.pl/index.php/tasks/task5

3. Data

Data provided by organizers was distributed through Hugging Face and was comprised of two parts: BIGOS dataset[3] (Junczyk 2023) and PELCRA benchmark[4]. BIGOS is a compilation of 12 open datasets whereas PELCRA was built as a compilation of selected datasets from PELCRA repository[5]. Details on sizes of specific dataset splits are available in Table 1. More detailed summary, with information on specific subsets (data sources) was provided by the organizers for both parts and all splits, we do not include this information here.

Table 1: Summary of dataset split sizes.

	Number of samples			**Duration [h]**		
Split	**BIGOS**	**PELCRA**	**Total**	**BIGOS**	**PELCRA**	**Total**
train	82 025	229 150	311 175	236.70	432.26	668.96
dev-0	14 254	28 532	42 786	27.51	49.60	77.11
test-A	1 002	1 167	2 169	2.53	2.14	4.67
test-B	991	1 178	2 169	2.48	2.15	4.63

Utilization of multiple datasets ensures high variability of data. In particular, PELCRA provides spontaneous and conversational speech whereas BIGOS contains audiobook data (Pratap et al. 2020), read speech recorded with many devices and in multiple acoustic conditions (Koržinek et al. 2016, Ardila et al. 2020) and spontaneous speech. Such diversity poses a difficult challenge for ASR systems but also ensures comprehensive evaluation of system robustness.

4. Method

4.1. Speech recognition

As our approach focuses mainly on utilization of synthetic data for augmentation of ASR system, we decided to use two standard speech recognition models without any modifications. Both models utilize BPE text tokenization scheme.

Conformer (Gulati et al. 2020) combines the strengths of transformer and convolutional neural network to capture both global and local dependencies in audio sequences. It does so by modifying transformer block to include additional convolutional module between standard multi-head attention and feed-forward layer. By integrating these two architectures, the Conformer demonstrates competitive performance even with compact models and reaches state-of-the-art accuracy in speech recognition. We utilize a pre-existing RNN-T implementa-

[3] https://huggingface.co/datasets/amu-cai/pl-asr-bigos-v2

[4] https://huggingface.co/datasets/pelcra/pl-asr-pelcra-for-bigos

[5] http://docs.pelcra.pl/

tion[6] with RNNTLoss[7]. Our model has 60M parameters and is trained from scratch for 500k steps with effective batch size of 512 audio samples. For decoding we utilize beam search with beam size of 10.

Whisper (Radford et al. 2023) is a standard encoder-decoder transformer model (Vaswani et al. 2017) with small modifications required for handling audio input. Advantage of using Whisper comes from large-scale supervised pretraining in multitask setting. In this work we perform full fine-tuning of *whisper-large-v3*[8] model which has 1550M parameters. We run the fine-tuning and then choose checkpoint with the best validation loss. Fine-tuning utilized effective batch size of 64 audio samples. For decoding we utilize beam search with beam size of 4.

4.2. Speech synthesis

We adopted a Voicebox-based (Le et al. 2023) strategy for speech synthesis. It offers state-of-the-art voice cloning in the resulting speech samples. Moreover, this approach can effectively utilize audios of suboptimal quality that earlier text-to-speech (TTS) systems could not accommodate. By harnessing these benefits, we aim to produce a synthetic dataset that closely mirrors the original in both quality and variability.

To achieve this, we trained a collection of models that work as one system. Models were trained from scratch using only the data provided by the competition organizers.

Voicebox (Le et al. 2023) is a zero-shot TTS that that leverages flow matching (Lipman et al. 2023). It enables the generation of audio conditioned on specific text and prompt audio. During the denoising process, Voicebox transforms a Gaussian distribution into the target distribution by solving ordinary differential equation (ODE) in a fixed number of steps to produce a mel spectrogram. Its architecture is built on a transformer encoder, enhanced with U-NET-like connections (Ronneberger et al. 2015) and rotary embeddings (Su et al. 2024). Additionally, the model underwent an extra pretraining stage in the manner described by Vyas et al. (2023), this step utilized only audio data. Model was pretrained for 270k steps and then adapted for 200k steps. It has 430M parameters and during training the effective batch size of 256 audio samples was used. For inference, we used 15 steps with midpoint ODE solver.

CTCAligner is a module that aligns audio features with text tokens in a force-aligner manner. This alignment provides information on the duration of each token, allowing for speaker intonation cloning. It shares the same architecture as Voicebox and utilizes CTC loss, enabling it to generate the mapping in an unsupervised manner. It also went through pretraining step in Best-RQ manner (Chiu et al. 2022). During this pretraining step audio features were aligned to codes from random frozen codebook. The model has 36M parameters. It was pretrained for 1M steps and then adapted for another 1M steps. Effective batch size of 512 audio samples was utilized.

[6] `https://pytorch.org/audio/main/generated/torchaudio.prototype.models.conformer_rnnt_model.html`

[7] `https://pytorch.org/audio/stable/generated/torchaudio.transforms.RNNTLoss.html`

[8] `https://huggingface.co/openai/whisper-large-v3`

DurationPredictor (Le et al. 2023) is built in the same way as Voicebox and also takes advantage of flow matching. Its primary function is to estimate the duration of each target token based on the context provided by the results of the CTCAligner. This allows to effectively transfer the intonation from the prompt speech to the target speech. The trained model has 93M parameters and was trained for 50k steps with effective batch size of 8192. For inference, we used 10 steps with midpoint ODE solver, also we calculated average of 10 model runs.

HiFi-GAN (Kong et al. 2020) is a fully convolutional generative adversarial network that functions as a vocoder, converting mel spectrogram features into audio signals. It employs both multi-scale and multi-period discriminators, enabling it to achieve exceptionally high-quality audio output. The model has 14M parameters and the training was run for 1M steps with effective batch size of 512.

4.3. Data preparation

For each model in the speech synthesis pipeline, we utilized the entire *train* split from the data provided by the organizers. The audio files were processed by extracting mel spectrograms using the following parameters: sample rate of 16kHz, hop size of 256, window length of 1024, minimum frequency of 0kHz, maximum frequency of 8kHz, and with 80 mel channels. Text data was lowercased.

For speech recognition models training, in addition to using the entire *train* split of data provided by the organizers, we incorporated two synthetic datasets. These were generated using speech synthesis system applied to randomly selected prompts taken from *train* split that were filtered based on the output of our *conformer-baseline* recognizer and speech rate criteria. Only audio files that achieved a maximum character error rate (CER) of 25% and had a speech rate variation within the range of 0 to 2.5 standard deviations from mean were selected for synthesis. This process resulted in creating two synthetic datasets: *synth-00* (440 hours and 293496 audio samples) and *synth-01* (890 hours and 586992 audio samples). By mixing these datasets with real data we created three training datasets, details on their sizes and composition are provided in Table 2.

Table 2: Summary of datasets used for speech recognition models' training.

Dataset	**Composition**	**Number of samples**	**Duration [h]**
baseline	*train*	311 175	669
mix-00	*train* + *synth-00*	604 671	1 109
mix-01	*train* + *synth-00* + *synth-01*	1 191 663	1 999

As a form of additional augmentation, for speech recognizers, we applied time and frequency masking from SpecAugment (Park et al. 2019). Without time warping, which was also proposed by Park et al. (2019), we were able to precompute all mel spectrograms what made training faster. Moreover authors of SpecAugment suggest that time warping has little to no effect on final results. Based on the findings by Huh et al. (2024), other augmentation techniques, such as adding noise or speech perturbations, were determined to be of questionable benefit, and therefore, were not utilized in our study.

5. Results

Word error rate (WER) and character error rate (CER) were used for comparing submissions. WER is defined as the number of incorrectly transcribed words divided by the total number of words in the reference sentences whereas CER is defined as the number of incorrectly transcribed characters divided by the total number of characters in the reference sentences. For development purposes we only utilized WER metric, calculated on *dev-0* split, for these calculations we lowercased all text and removed all punctuation. All rates are multiplied by 100 for better readability.

Table 3: Mean word error rate (WER) for all evaluated models calculated on *dev-0* splits for both data sources. Mean is weighted based on number of samples in subsets.

Model	**BIGOS**	**PELCRA**	**Total**
whisper-large-v3	6.08	29.04	21.39
whisper-large-v3-baseline	6.16	23.35	17.62
whisper-large-v3-mix-00	5.04	22.58	16.74
whisper-large-v3-mix-01	3.93	20.98	15.30
conformer-baseline	11.22	30.55	24.11
conformer-mix-00	7.85	27.32	20.84
conformer-mix-01	7.26	25.38	19.34

Results of our internal evaluations are shown in Table 3. We provide results for both Conformer and Whisper models trained on all studied training datasets. For comparison, we also evaluated Whisper without any fine-tuning. Results confirm that the addition of synthetic data improves quality of both models. The results are also not clearly saturated even in the case of *mix-01* where total duration of training data was almost tripled. Addition of synthetic data seems to have more impact on the results of Conformer model - between *mix-01* and *baseline,* total WER was reduced by 4.77, whereas in the case of Whisper it was reduced only by 2.32. This can be explained by large-scale pretraining that Whisper did undergo – in its case the *baseline* achieved significantly better results than Conformer. We can also observe that in the case of Whisper, fine-tuning has more impact on the results achieved on PELCRA part of data – between raw model and *mix-01*, WER was reduced by 8.06 on PELCRA and only by 2.15 on BIGOS. This is probably connected to the overall worse results achieved by the models on PELCRA part of the data, which may be caused by PELCRA being more difficult (conversational and spontaneous speech). The best results on *dev-0* split were achieved by *whisper-large-v3-mix-01* what made it the main candidate for our submission.

Results achieved by all our models on *test-A* and *test-B* splits are shown in Table 4. As one would expect, all measured character error rates are lower than corresponding word error rates. Also, results show that addition of synthetic data has smaller impact on Whisper than in the case of *dev-0* split. This may be caused by test splits being balanced with regard to data source being BIGOS or PELCRA and the main gains in results were achieved on PELCRA data. We can also observe that *conformer-mix-01* and *whisper-large-v3-mix-01* have similar quality

Table 4: Character error rate (CER) and word error rate (WER) for all evaluated models calculated on both *test-A* and *test-B* splits.

	test-A		***test-B***	
Model	**CER**	**WER**	**CER**	**WER**
whisper-large-v3-baseline	7.15	11.52	7.10	11.23
whisper-large-v3-mix-00	6.85	11.07	6.91	11.15
whisper-large-v3-mix-01	6.90	11.27	6.85	11.07
conformer-baseline	8.77	17.48	8.37	16.82
conformer-mix-00	7.60	15.25	7.16	14.33
conformer-mix-01	7.08	13.99	6.90	13.40

when measured with CER (difference of 0.18 on *test-A* and 0.05 on *test-B*) but differences in WER are larger (2.72 on *test-A* and 2.33 on *test-B*). It is also not obvious whether *whisper-large-v3-mix-00* or *whisper-large-v3-mix-01* is better as one achieves better results on *test-A* and second one on *test-B*.

6. Discussion

It is possible to present legitimate doubts regarding whether the proposed system, that includes augmentation with synthetic speech, complies with the competition rules. Specifically, with the provision stating: "It is forbidden for the participants to use any data outside of the provided train and validation sets to develop their systems". However, we argue that this system meets the competition requirements, and our reasoning is provided below.

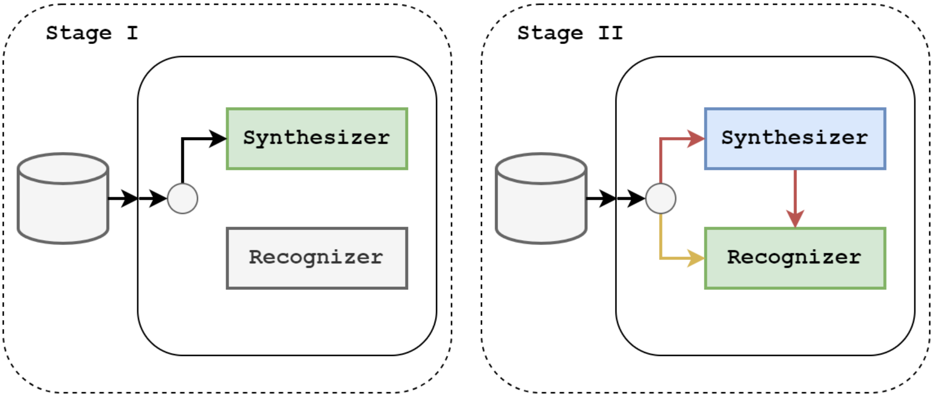

Figure 1: Automatic speech recognition system augmented with synthetic speech presented as a hierarchical system. In Stage I, Synthesizer is trained, its weights are then frozen in Stage II where Recognizer is trained. In Stage II, data is sampled and provided either directly to the Recognizer (yellow data flow) or first is processed with Synthesizer and only then is provided to Recognizer (red data flow).

First, the entire system, consisting of multiple models, can be viewed as a single hierarchical system trained within a multi-stage and multi-task paradigm. This approach is illustrated in Figure 1. Such a system would be trained piece by piece, starting with synthesizer. Then specific weights would be frozen, and further training procedure would utilize sampling that would select appropriate path through the system to accurately reproduce the augmentation process. This is represented in Figure 1 as red and yellow data flow in Stage II. Additionally, certain components of this system could be viewed as interpretable intermediate points (e.g. output of speech synthesis).

Second, in order to both train and to infer from synthesizer we utilize only data provided by the organizers of the competition. No external data was used for that purpose. The gained variability of synthetic speech comes from mixing texts and voices from different samples but we argue that this does comply with the competition rules.

7. Conclusions

We have presented an automatic speech recognition system for Polish augmented with synthetic data generated using state-of-the-art speech synthesizer. We evaluated two speech recognition models, both with and without augmentation. One of the models was trained from scratch and second was only fine-tuned. We presented results we obtained using this approach in PolEval 2024, Task 3: Polish Automatic Speech Recognition. We showed that introducing augmentation with synthetic speech improves the system's results.

There are still some avenues for improvements of the proposed system. In order to introduce more variability in the synthetic data we could utilize language model to generate texts for synthesizer. More careful procedure for choosing audio prompts for voice cloning could also be introduced. In particular, by introducing a more iterative model training procedure, we could select prompts from subsets on which model performs worse. But due to competition time constraints and infrastructure limitations we leave these topics for further research.

References

Ardila R., Branson M., Davis K., Henretty M., Kohler M., Meyer J., Morais R., Saunders L., Tyers F. M. and Weber G. (2020). *Common Voice: A Massively-Multilingual Speech Corpus*. In *Proceedings of the 12th Conference on Language Resources and Evaluation*, pp. 4211–4215.

Bartelds M., San N., McDonnell B., Jurafsky D. and Wieling M. (2023). *Making More of Little Data: Improving Low-Resource Automatic Speech Recognition Using Data Augmentation*. In Rogers A., Boyd-Graber J. and Okazaki N. (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 715–729, Toronto, Canada. Association for Computational Linguistics.

Chiu C.-C., Qin J., Zhang Y., Yu J. and Wu Y. (2022). *Self-supervised Learning with Random-projection Quantizer for Speech Recognition*. In Chaudhuri K., Jegelka S., Song L., Szepesvari

C., Niu G. and Sabato S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, vol. 162 of *Proceedings of Machine Learning Research*, pp. 3915–3924. PMLR.

Conneau A., Baevski A., Collobert R., Mohamed A. and Auli M. (2021). *Unsupervised Cross-Lingual Representation Learning for Speech Recognition*. In *Proceedings of Interspeech 2021*, pp. 2426–2430.

Czyżnikiewicz M., Łukasz Bondaruk, Kubiak J., Wiącek A., Łukasz Degórski, Kubis M. and Skórzewski P. (2024). *Spoken Language Corpora Augmentation with Domain-Specific Voice-Cloned Speech*. arXiv:2406.07090.

Demenko G., Grocholewski S., Klessa K., Ogórkiewicz J., Wagner A., Lange M., Śledziński D. and Cylwik N. (2008). *JURISDIC: Polish Speech Database for Taking Dictation of Legal Texts*. In Calzolari N., Choukri K., Maegaard B., Mariani J., Odijk J., Piperidis S. and Tapias D. (eds.), *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Grocholewski S. (1997). *CORPORA — Speech Database for Polish Diphones*. In *Proceedings of the 5th European Conference on Speech Communication and Technology (Eurospeech 1997)*, pp. 1735–1738.

Gulati A., Qin J., Chiu C.-C., Parmar N., Zhang Y., Yu J., Han W., Wang S., Zhang Z., Wu Y. and Pang R. (2020). *Conformer: Convolution-augmented Transformer for Speech Recognition*. In *Proceedings of Interspeech 2020*, pp. 5036–5040.

Huh M., Ray R. and Karnei C. (2024). *A Comparison of Speech Data Augmentation Methods Using S3PRL Toolkit*. arXiv:2303.00510.

Janicki A. and Wawer D. (2011). *Automatic Speech Recognition for Polish in a Computer Game Interface*. In *2011 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 711–716.

Junczyk M. (2023). *BIGOS — Benchmark Intended Grouping of Open Speech Corpora for Polish Automatic Speech Recognition*. In Ganzha M., Maciaszek L., Paprzycki M. and Ślęzak D. (eds.), *Proceedings of the 18th Conference on Computer Science and Intelligence Systems (FedCSIS 2023)*, vol. 35, pp. 585–590. ACSIS.

Kong J., Kim J. and Bae J. (2020). *HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis*. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA. Curran Associates Inc.

Koržinek D. (2019). *Results of the PolEval 2019 Shared Task 5: Automatic Speech Recognition Task*. pp. 73–78.

Koržinek D., Marasek K., Łukasz Brocki and Wolk K. (2016). *Polish Read Speech Corpus for Speech Tools and services*. In Borin L. (ed.), *Selected papers from the CLARIN Annual Conference 2016*, vol. 136 of *Linköping Electronic Conference Proceedings*, pp. 54–62. Linköping University Electronic Press.

Koržinek D., Wołk K., Łukasz Brocki and Marasek K. (2019). *Automatic Transcription of the Polish Newsreel*. Poznan Studies in Contemporary Linguistics, 55(2), p. 183–209.

Le M., Vyas A., Shi B., Karrer B., Sari L., Moritz R., Williamson M., Manohar V., Adi Y., Mahadeokar J. and Hsu W.-N. (2023). *Voicebox: Text-Guided Multilingual Universal Speech Generation at Scale*. In Oh A., Naumann T., Globerson A., Saenko K., Hardt M. and Levine S. (eds.), *Advances in Neural Information Processing Systems*, vol. 36, pp. 14005–14034. Curran Associates, Inc.

Lipman Y., Chen R. T. Q., Ben-Hamu H., Nickel M. and Le M. (2023). *Flow Matching for Generative Modeling*. In *Proceedings of the 11th International Conference on Learning Representations (ICLR 2023)*.

Marciniak M. (2010). *Anotowany korpus dialogów telefonicznych*. EXIT.

Park D. S., Chan W., Zhang Y., Chiu C.-C., Zoph B., Cubuk E. D. and Le Q. V. (2019). *SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition*. In *Proceedings of Interspeech 2019*, pp. 2613–2617.

Povey D., Ghoshal A., Boulianne G., Burget L., Glembek O., Goel N., Hannemann M., Motlicek P., Qian Y., Schwarz P., Silovsky J., Stemmer G. and Vesely K. (2011). *The Kaldi Speech Recognition Toolkit*. In *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society. IEEE Catalog No.: CFP11SRW-USB.

Pratap V., Xu Q., Sriram A., Synnaeve G. and Collobert R. (2020). *MLS: A Large-Scale Multilingual Dataset for Speech Research*. In *Proceedings of Interspeech 2020*, pp. 2757–2761.

Radford A., Kim J. W., Xu T., Brockman G., McLeavey C. and Sutskever I. (2023). *Robust Speech Recognition via Large-Scale Weak Supervision*. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Ronneberger O., Fischer P. and Brox T. (2015). *U-Net: Convolutional Networks for Biomedical Image Segmentation*. In Navab N., Hornegger J., Wells W. M. and Frangi A. F. (eds.), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pp. 234–241, Cham. Springer International Publishing.

Rossenbach N., Zeineldeen M., Hilmes B., Schlüter R. and Ney H. (2021). *Comparing the Benefit of Synthetic Training Data for Various Automatic Speech Recognition Architectures*. In *Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop*, pp. 788–795.

Su J., Ahmed M., Lu Y., Pan S., Bo W. and Liu Y. (2024). *RoFormer: Enhanced Transformer with Rotary Position Embedding*. Neurocomputing, 568, p. 127063.

Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser L. and Polosukhin I. (2017). *Attention is All you Need*. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Vyas A., Shi B., Le M., Tjandra A., Wu Y.-C., Guo B., Zhang J., Zhang X., Adkins R., Ngan W., Wang J., Cruz I., Akula B., Akinyemi A., Ellis B., Moritz R., Yungster Y., Rakotoarison A., Tan L., Summers C., Wood C., Lane J., Williamson M. and Hsu W.-N. (2023). *Audiobox: Unified Audio Generation with Natural Language Prompts*. arXiv:2312.15821.

Wang C., Riviere M., Lee A., Wu A., Talnikar C., Haziza D., Williamson M., Pino J. and Dupoux E. (2021). *VoxPopuli: A Large-Scale Multilingual Speech Corpus for Representation Learning,* *Semi-Supervised Learning and Interpretation*. In Zong C., Xia F., Li W. and Navigli R. (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 993–1003, Online. Association for Computational Linguistics.

Zhang Z., Zhou L., Wang C., Chen S., Wu Y., Liu S., Chen Z., Liu Y., Wang H., Li J., He L., Zhao S. and Wei F. (2023). *Speak Foreign Languages with Your Own Voice: Cross-Lingual Nural Codec Lnguage Modeling*.

Ziółko B., Manandhar S., Wilson R. C., Ziólko M. and Galka J. (2008). *Application of HTK to the Polish Language*. In *Proceedings of the 2008 International Conference on Audio, Language and Image Processing*, pp. 1759–1764.

Ziółko B., Jadczyk T., Skurzok D., Żelasko P., Gałka J., Pedzimaz T., Gawlik I. and Palka S. (2015). *SARMATA 2.0 Automatic Polish Language Speech Recognition System*. In *Proceedings of Interspeech 2015*, pp. 1062–1063.

Ziółko M., Gałka J., Ziółko B., Jadczyk T., Skurzok D. and Masior M. (2011). *Automatic Speech Recognition System Dedicated for Polish*. In *Proceedings of Interspeech 2011*, pp. 3315–3316.

Żelasko P., Ziółko B., Jadczyk T. and Skurzok D. (2016). *AGH Corpus of Polish Speech*. Language Resources and Evaluation, 50(3), p. 585–601.

Exploration of Training Zipformer and E-Branchformer Models with Polish Language BIGOS Dataset

Paweł Cyrta
(Stenograf.ai, SpeakLeash)

Abstract

We present our work in the the challenge, where we contributed to the development of evaluation data through manual transcription of the test set. Due to our involvement in test set creation, we did not submit official results to maintain evaluation integrity. However, we conducted exploratory experiments using two transformer-based architectures: Zipformer and E-Branchformer, trained on the challenge's training set. This paper describes our preliminary modeling efforts, highlighting the importance of robust evaluation data in advancing Polish ASR research.

Keywords

speech recognition, ASR, automatic speech recognition, Polish, speech-to-text

1. Introduction

Recent advances in transformer-based architectures have shown promising results across various languages, yet their application to Polish requires careful consideration of language-specific characteristics. The development of robust ASR systems heavily relies on the quality of training and evaluation data, where precise with knowledgeable manual transcription is crucial. Professional transcription services Stenograf demonstrate that well-trained annotators following standardized guidelines are essential for creating reliable speech corpora. This meticulous approach to data annotation, combined with clear methodology standards, forms the foundation for meaningful models evaluation and comparison.

The Polish ASR Challenge addresses this gap by providing the BIGOS dataset as part of Task 3, enabling systematic exploration of modern architectures in the Polish language context. In this study, we investigate the potential of Zipformer (Yao et al. (2024)) and E-Branchformer

(Kim et al. (2023)) architectures, two recent developments in transformer-based models, for this specific task.

2. Test set creation contribution

Our team participated in the manual transcription effort for the challenge's test set creation. This decision prevent us from submitting official results to maintain evaluation integrity. The manual transcription process involved: careful annotation of recordings, following standardized transcription guidelines and our good practices that we follow, quality assurance through 2 reviews. This work supported the development of a reliable evaluation benchmark for the challenge, prioritizing the community's need for accurate test data over individual competition participation.

3. Experiments

While our main contribution focused on test set creation, we conducted preliminary experiments using two transformer-based architectures: Zipformer (Icefall K2 framework) and E-Branchformer (ESPnet). Both are well known in speech academic community, and are considered a standard selection when training new models.

We used the training data set given as part of challenge-provided training set. Both models were implemented using standard configurations without extensive hyperparameter optimization, serving as baseline exploration rather than competition entries.

Table 1: Results of fined-tuned models.

	dev-0 WER
Zipformer	21.52
E-Branchformer	19.23

4. Submission

To maintain participation requirements while ensuring our test set knowledge didn't provide unfair advantage, we did not summit our results but we submit distorted text from ASR that was processed in a systematic text transformation approach.

We used a text transformation method that deliberately maximized the Word Error Rate while preserving submission format requirements. Our approach implemented a consistent character substitution where each letter in the Polish alphabet was mapped to a different one, applying this transformation at both character and word levels while preserving common stopwords.

This method achieved our goal of producing nearly 102 % WER (for both test-A and test-B set), effectively submitting to the challenge while ensuring our prior knowledge of the test set could not influence the competition's outcomes. The systematic nature of our transformations maintained the statistical properties of text length and word boundaries.

5. Conclusion

Our work prioritized contributing to the challenge through test set creation. While our model experiments with Zipformer and E-Branchformer architectures demonstrated the feasibility of these approaches for Polish ASR, our main impact lies in helping establish reliable evaluation data for the broader research community.

Acknowledgements

We give our special gratitude to Michał Junczyk, whose Ph.D. research laid the groundwork for this challenge. Our innovative conception of the BIGOS dataset, drawing inspiration from the SpeechSkew (Chan et al. (2021)) dataset for English, has created a valuable resource for the Polish ASR community. The successful completion of his doctoral work, culminating in this comprehensive challenge, represents a significant contribution to Polish speech recognition research. We congratulate him on this remarkable achievement that will undoubtedly influence future developments in the field.

References

Chan W., Park D., Lee C., Zhang Y., Le Q. and Norouzi M. (2021). *SpeechStew: Simply Mix All Available Speech Recognition Data to Train One Large Neural Network*. arXiv:2104.02133.

Kim K., Wu F., Peng Y., Pan J., Sridhar P., Han K. J. and Watanabe S. (2023). *E-Branchformer: Branchformer with Enhanced Merging for Speech Recognition*. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pp. 84–91.

Yao Z., Guo L., Yang X., Kang W., Kuang F., Yang Y., Jin Z., Lin L. and Povey D. (2024). *Zipformer: A Faster and Better Encoder for Automatic Speech Recognition*.