Preproceedings of the of the 18th Conference on Computer Science and Intelligence Systems pp. 1249–1258

# Abbreviation Disambiguation in Polish Press News Using Encoder-Decoder Models

Krzysztof Wróbel
0000-0002-3485-7825
Jagiellonian University, Enelpol
Krakow, Poland
Email: krzysztof@wrobel.pro

Jakub Karbowski
0009-0009-3051-7354
AGH University
of Science and Technology
Krakow, Poland
Email: carbon225@proton.me

Paweł Lewkowicz
0009-0000-5752-7610
AGH University
of Science and Technology,
Krakow, Poland
Email: pawlew@agh.edu.pl

***Abstract*—The disambiguation of abbreviations and acronyms is a longstanding problem in Natural Language Processing (NLP) that has garnered significant attention from researchers. Previous approaches have employed statistical methods, semantic similarity metrics, and machine learning algorithms. Various languages and document types have been explored, with English being the most commonly studied language. Recent advances have been driven by the application of pre-trained transformer models. Standardization and addressing the challenges of multilingual and multi-document type disambiguation remain ongoing goals in the field of NLP. This paper presents an in-depth exploration of abbreviation disambiguation using state-of-the-art neural Encoder-Decoder models, specifically the ByT5 and plT5 architectures. Advanced synthetic data generation techniques are introduced and their effect on model performance is analysed. The methods are evaluated in the context of the PolEval abbreviation disambiguation competition, where the authors achieve top ranking.**

## I. Introduction

THE problem of disambiguation of both acronyms and abbreviations has been the subject of interest for many researchers in the field of Natural Language Processing (NLP) for many years. Even before the era of widely used machine learning algorithms and text recognition using Deep Neural Networks (DNN), methods based solely on statistics were used. An example of such work is the paper [1] from 2004, which used a semantic similarity metric. The author determines the adequacy of abbreviation expansion candidates based on the similarity between the context of the target abbreviation and that of its expansion candidate.

The motivation for recognizing abbreviations often stemmed from the need to understand passages in documents such as provisions in law or medical notes. However, due to the richness, diversity, and uniqueness of languages, it is difficult to generalize the solution for expanding abbreviations or acronyms. Articles [2], [3] examine Jewish Law documents written in Hebrew, while [4], [5] present research on clinical papers using methods such as Support Vector Machines (SVM). Scientific papers are usually dedicated to only one language for which the datasets were prepared and the most widely used language in datasets is English. But another example of a different language is the research analysis [6] in Chinese, where the authors present their unconventional method based on Integer Linear Programming (ILP) and decode abbreviations from the generated candidates. Meanwhile, [7] analyze the Russian language by comparing methods such as SVM, Random Forest (RF), and Gradient Boosting (GB). Research has also been conducted in the Polish language, for example, in the paper [8], which utilized the bidirectional long short-term memory (LSTM) neural network architecture and compared two methods: automatically selecting all words in a text and using clustering of abbreviation occurrences.

Another aspect worth noting is the diversity of approaches to solving the disambiguation problem. As it turns out, supervised methods such as Convolutional Neural Networks (CNN), which are primarily used for image analysis, can also be used for this purpose in NLP, as evidenced by the article [9]. A diffrent example is the utilization and combination of pre-trained models such as RoBERTa and SciBERT, based on the transformer architecture, to create their own model named hdBERT, as presented in the research [10]. In this study, the authors also compared many state-of-the-art non-deep and deep learning methods up to 2017.

In 2020, the article [11] presented Google's T5 model as the Unified Text-to-Text Transformer. Since then, models with this architecture have found applications not only in text translation but also in expanding abbreviations, as shown in the publication [12]. In another article [13], expansions of acronyms were presented using pre-trained language models such as BERT and T5 for datasets consisting of four categories: Legal English, Scientific English, French, and Spanish.

The authors of the article [14] built their end-to-end acronym expander system named AcX and compared various existing methods such as Cosine Similarity (Cossim), RF, Logistic Regression (LR), and SVM. Based on this, they also prepared a benchmark on various types of datasets, including those from biomedical document, scientific papers and Wikipedia.

Based on the above considerations, research on abbreviation disambiguation over the years can be divided into three main categories:

- the type of documents from which the dataset was created, e.g. medicine, law, articles, or news
- the language in which the datasets were prepared



**Technical track:** Challenges for Natural Language Processing (1st Symposium CNLPS 2023)

- the method used, which is almost always related to various machine learning algorithms

Former articles mainly focused on one type of document, one language, and one or two methods. However, today we increasingly encounter works related to multilingual models such as mT5 (multilingual T5), trained over 101 languages. However, these models undoubtedly require more memory (T5-base model - 220M parameters, mT5-base model - 580M parameters). The area of recognizing abbreviations is moving towards standardization and dealing with multiple languages and document types at once, but achieving satisfactory results in this area still poses a challenge for the field of NLP. Currently, promising models for this task appear to be encoder-decoder models like T5, pre-trained on a multi-task mixture of unsupervised and supervised tasks.

This article presents an attempt to standardize and formalize different aspects of abbreviation disambiguation, methods, challenges and limitations. State-of-the-art methods are evaluated on the PolEval 2022/23 competition, specifically in Task 2: Abbreviation disambiguation [1]. Additional dataset augmentation techniques are described, such as dictionary-lookup and algorithmic generation of arbitrary abbreviations. A unified training framework for abbreviation disambiguation is provided in a public online code repository. The additional created datasets are also provided. By combining the above methods, the authors achieve number one ranking on the PolEval contest.

## II. Data

### A. Training, validation and test datasets

The training, validation, and test datasets have been provided by the organizers of PolEval. As part of the PolEval competition, a training set called `train` and a validation set called `dev-0` with expected output were created, along with two test sets: `test-A` and `test-B` with implicit output. The collection and preparation of the datasets were carried out by:

- Michał Marcińczuk (Wrocław University of Science and Technology)
- Łukasz Kobyliński (Institute of Computer Science, Polish Academy of Sciences / Sages)

*1) Assumptions:* During the preparation, the authors of the reference corpus based their work on three assumptions:

- focus on abbreviations of common words or phrases ending with a dot (excluding initials, acronyms, and proper names)
- the context and common knowledge should be sufficient to expand the abbreviation (excluding incomplete or confusing examples)
- the base forms should follow the guidelines of phrase lemmatization from PolEval 2019 Task 2 [2] with some exceptions, such as abbreviations joined with other abbreviations or phrases

[1] http://poleval.pl/tasks/task2
[2] http://2019.poleval.pl/index.php/tasks/task2

TABLE I
Examples of abbreviation disambiguation

| Abbr | Context | Inflected form | Base form |
|---|---|---|---|
| t. | a na Dolnym Śląsku - 530-540 zł/**<mask>** | tonę | tona |
| gat. | czystą miedź (gat. M1) i mosiądz niklowy (**<mask>** M55N, CuZn35Ni6Mn4Si0,2) | gatunek | gatunek |
| j. ukr. | Kier. Natalia Szelest (nagroda burmistrza Węgorzewa) Wertep punktu naucz. **<mask>** w Baniach Mazurskich | języka ukraińskiego | język ukraiński |
| ład. | Żyjemy w okresie przejściowym, międzyepoce poprzedzającej nowy **<mask>** | ład. | ład. |

Table I presents several examples of abbreviations disambiguation found in these datasets.

*2) Input and expected output:* In the system, the input data in the `in.tsv` file consists of two columns as we can see in Table II: the first column contains a phrase to be analyzed, and the second contains the context in which the phrase appears. If a masked token in an input is not an abbreviation then both columns in expected file are the same as masked token.

TABLE II
Input and expected output format

| in.tsv | expected.tsv |
|---|---|
| l. ciągników serii 1523 i 1221 otrzymały sześciocylindrowe silniki o pojemności 7,2 **<mask>** Także te modele nie były wysilone mocowo i osiągały 158 KM w przypadku Belarusa | litra litr |

The occurrence of the phrase is marked by the keyword `<mask>`. The output file `expected.tsv` is also composed of two columns: the first column contains the inflected form, and the second contains the base form.

*3) Data processing:* The corpus was created based on the following four steps:

1. The datasets were built based on a collection of press news.
2. Regular expressions were used to collect potential abbreviations.
3. Each candidate was represented as a matched phrase and the context with several words before and after the match.
4. Candidates were selected and manually annotated.

*4) Dataset cleanup:* During the review, the authors removed any examples where the text was somehow corrupted, making it difficult to analyze.

**1.** Chi lij ski gi gant mie dzio wy (16,1 mld do **<mask>** war to ści) roz pro szył do brym zy skiem oba wy inwesto rów o je go wzrost mi mo

**2.** l.,ulica,ulica,A) 9. U**l. Kościuszki** na odcinku od Lelewela do al. Krasińskiego 10. U**<mask> Zwierzyniecka** od al. Krasińskiego do ul. Retoryka 11. Ul**. Bieżanowska** na

Above two examples of corrupted texts: the first with words divided into syllables, and the second with "**" characters within the abbreviation.

*5) Challenges:* The task poses several challenges that need to be addressed in order to solve it.

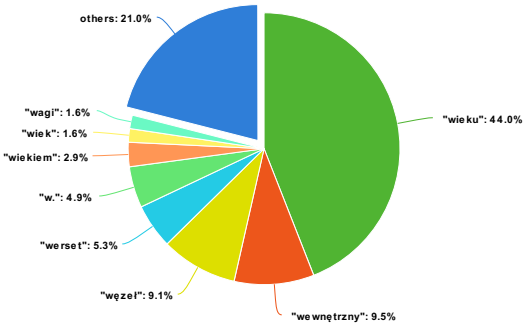
*a) Challenge – ambiguous forms:* When looking at one-word abbreviations, there is a large pool of words to which these abbreviations can be expanded.

TABLE III
AMBIGUOUS ONE-WORD ABBREVIATION FORMS EXPANDED TO THE LARGEST VARIETY OF WORDS (IN THE TRAIN SET)

| Abbreviation | Base forms | Inflected forms |
|---|---|---|
| p. | 38 | 57 |
| w. | 21 | 41 |
| s. | 20 | 36 |
| m. | 14 | 30 |

It can be observed in Table III that practically every word can be shortened to a single letter, and based on the context, one can infer the intended word. However, the challenge lies in correctly identifying this word.

Fig. 1. Distribution of abbreviation "w." extensions




The Figure 1 shows that there are multiple potential forms for each abbreviation, and the distribution is not such that one form constitutes 90% of expansions. What makes this task interesting is that there are many expansions for each abbreviation. In each case, there is a dominant word, for example, for the abbreviation `w.`, the most frequent expansion is `wieku` which accounts for 44.0% of expansions, while the remaining cases below 1.6% each sum up to 21.0%.

*b) Challenge – unbounded space of abbreviations:* The second element that affects the complexity of this task is the practically unbounded set of phrases that are subject to analysis. This task needs to be approached creatively because many cases not present in the training set may appear in the test set and should be handled correctly.

Based on Table IV, it can be assumed that about half of the phrases are non-abbreviated elements, while the other half are actual abbreviations that require expansion.

TABLE IV
THE NUMBER OF DISTINCT PHRASES AND ABBREVIATIONS TO BE EXPANDED

| Set | Distinct phrases | Distinct abbreviations |
|---|---|---|
| train | ∼ 1013 | ∼ 500 |
| full | >1600 | >900 |

*c) Challenge – mixed abbreviations and non-abbreviations:* Another challenge is distinguishing whether a given phrase is an abbreviation or not. There are also cases where a phrase can be both an abbreviation and a regular word that does not require expansion, which introduces an additional level of complexity to the task.

TABLE V
OCCURRENCES OF DIFFERENT ABBREVIATION FORMS

| Set | Train set |
|---|---|
| Distinct phrases | ∼ 1013 |
| Non-abbreviations | ∼ 480 |
| Abbreviations | ∼ 540 |
| Both | 26 |

Furthermore, certain phrases are a combination of abbreviations and non-abbreviations, marked as `others` in Table V, for example, replacing `mln. ton.` with `miliona ton.`

*6) Special cases:* There are several specific cases that have appeared to some extent in the corpus and go beyond the previous assumptions.

*a) Special case – ambiguous forms:* There are instances where certain abbreviations can be expanded into multiple words, as in the example presented in Table VI, where `m.` can be expanded into both `miejscowości` and `miasta`.

TABLE VI
AN EXAMPLE OF A CASE WITH A POSSIBLE DUAL OUTPUT

| in.tsv | expected.tsv | |
|---|---|---|
| m. które przyszło na świat, gdy już był w więzieniu. Sam miał lat 40 i pochodził z <mask> Luboml pow. Kowel. Był urzędnikiem w składnicy Monop. Spirytusowego. (...) Nadmieniam, | miejscowości; miasta | miejscowość; miasto |

Wherever this ambiguity can be resolved through context, such as certain signals indicating that it refers to one specific form, we expect only one form to appear in the output. However, in cases where there is no ambiguity, we assume that both forms should appear in the output, and they will be compared in this way.

*b) Special case – non-abbreviation:* In situations exposed in Table VII where a given phrase is not an abbreviation, we expect the output to repeat the phrase, which will be recognized as a non-abbreviation. Therefore, in this case the second part of answer is not a base form.

Both the base form and any inflected forms should be inflected in the same way as in the input, including the dot.

TABLE VII
AN EXAMPLE OF A CASE WITH A NON-ABBREVIATION

| in.tsv | expected.tsv | |
|---|---|---|
| **Goi.** Fascynowały ją obrazy **<mask>** Czytywała klasyków. Dowodzą tego jej, właśnie odnalezione, zapiski - pisze Bartosz | Goi. | Goi. |

*c) Special case – abbr and non-abbr:* When there is a combination of an abbreviation and a non-abbreviated element, the abbreviated fragment should be expanded, while the non-abbreviated one should be preserved in the same form as in the input phrase.

TABLE VIII
AN EXAMPLE OF A CASE WITH A MIXED-ABBREVIATION

| in.tsv | expected.tsv | |
|---|---|---|
| **ws. T.** też: "Będziemy sprawdzać. czy nie fałszowano dowodów". Prokuratura bada śledztwo **<mask>** Komendy | w sprawie T. | w sprawie T. |

In the example shown in Table VIII, these are initials, which we also do not want to expand.

*d) Special case – lemmatization of joined abbreviations:* In yet another case, there may be a combination of abbreviations that affects lemmatization. In the example below, two abbreviations appear consecutively, and each of these abbreviations should be expanded separately.

TABLE IX
AN EXAMPLE OF A CASE WITH A JOINED ABBREVIATIONS

| in.tsv | expected.tsv | |
|---|---|---|
| **mm. Temp. maks.** opady deszczu lub burze. Na zachodzie i południu prognozowana wysokość opadu do 25 **<mask>** od 19 do 23 st., nad morzem od 13 do 18 st. Wiatr północno-wschodni, slaby, na wybrzeżu | milimetrów<br>Temperatura<br>maksymalna | milimetrów<br>temperatura<br>maksymalna |

In Table IX, the one-element abbreviation `mm.` is expanded to `millimeters` (it is an annotation error, it should be in singular: `millimeter`), and the two-element abbreviation `Temp. maks` is expanded to `Maximum temperature` which are then joined together.

## B. Dictionary-based additional data

From dictionaries we extracted abbreviations with expanded forms, e.g. `bdb. -> bardzo dobry`. Then from Polish corpus CC100 [15] we extracted text fragments with inflected expanded forms of abbreviations and replaced them with abbreviations. An example is in Table X.

Only samples with unique inflected forms were taken into consideration giving 1982 new data points. The process creates also incorrect samples, e.g. `najlepszym` is abbreviated to `db.`. This dataset lacks non-abbreviation examples.

TABLE X
DICTIONARY-BASED ADDITIONAL EXAMPLE

| in.tsv | expected.tsv | |
|---|---|---|
| **bdb.** pełno dyspozycyjny, zaangażowany, pracowity, komunikatywny, bardzo dobrze znam budowe komputera jak i **<mask>** obsługa komputera jak i programów biurowych Microsoft Office, Open Office itd. Prosze o kontakt | bardzo dobra | bardzo dobry |

*1) Morfeusz:* Morfeusz [16] is a morphological analysis tool for the Polish language. With its help, all the abbreviations with their expanded form were filtered out from the dictionary. The pairs were selected on the basis of the morphosyntactic tags `brev:pun` or `brev:npun`. The `brev` feature indicates the base form of an abbreviation expansion, while `pun` and `npun` denote the presence or absence of a dot after the abbreviation. All extracted abbreviations are unambiguous, meaning that there is no more than one expansion for each abbreviation in the data.

TABLE XI
NUMBER OF ABBREVIATIONS FOUND IN THE MORFEUSZ DICTIONARY ACCORDING TO THE MORPHOSYNTACTIC TAGS

| tag | | |
|---|---|---|
| **brev:pun** | **brev:npun** | total |
| 279 | 154 | 433 |

Table XI shows that there are more abbreviations without a dot at the end. Due to the problem posed in the task, `pun` abbreviations may be more useful since they can occur anywhere in a sentence, whereas `npun` abbreviations only appear at the end of a sentence, to be followed by a dot.

*2) Wiktionary:* Based on the free, multilingual dictionary Wiktionary [3], 554 different abbreviations were extracted with their meaning or meanings serving as an extension of the abbreviation. An example of multiple meanings in this dataset is, for example, `wyd.`, which can mean: `wydanie`, `wydawca`, `wydawnictwo`, `wydawniczy`.

As we can see in Table XII, most of the found abbreviations have only one meaning, while in some cases, there are as many as 14 meanings for `n.` or 19 for `a.`.

In addition to the meanings, examples were also extracted from the Wiktionary, serving as context; however, this information was not utilized in solving the task.

*3) SJP:* More abbreviations than in previous dictionaries, a total of 1199, were found in the Polish language dictionary SJP. Just like in Wiktionary, some abbreviations had multiple meanings, for example, `woj.` could stand for `województwo`, `wojewoda`, `wojewódzki`, `wojenny`, `wojskowy`. The abbreviations were selected by reviewing all the words or phrases in the dictionary and then filtering out those that had periods at the end.

[3]https://www.wiktionary.org

TABLE XII
DISTRIBUTION OF THE NUMBER OF MEANINGS FOR ABBREVIATIONS IN THE DATASET WIKTIONARY

| Number of meanings for the abbreviation | Number of abbreviations with a given number of meanings | |
|---|---|---|
| 1 | 419 | |
| 2 | 83 | |
| 3 | 29 | |
| 4 | 13 | |
| 5 | 5 | |
| 6 | 1 | |
| 7 | 1 | |
| 8 | 1 | |
| 14 | 1 | |
| 19 | 1 | |
| **Number of meanings** | **Number of abbreviations** | |
| 803 | 554 | **total** |

TABLE XIII
DISTRIBUTION OF THE NUMBER OF MEANINGS FOR ABBREVIATIONS IN THE DATASET SJP

| Number of meanings for the abbreviation | Number of abbreviations with a given number of meanings | |
|---|---|---|
| 1 | 1051 | |
| 2 | 97 | |
| 3 | 30 | |
| 4 | 10 | |
| 5 | 11 | |
| **Number of meanings** | **Number of abbreviations** | |
| 1430 | 1199 | **total** |

Table XIII shows that there are no abbreviations with as many meanings as sometimes found in Wiktionary, but the number 5 can be considered the maximum number of meanings that almost always appeared in SJP and Wiktionary.

### C. Synthetic additional data

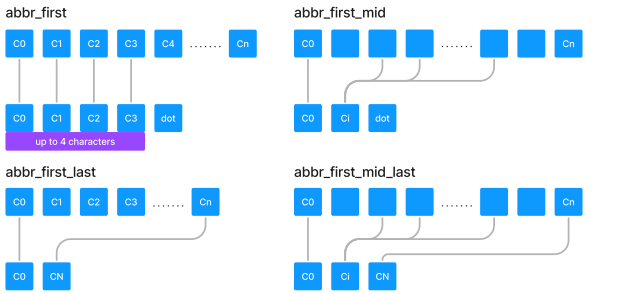
Collecting a sufficiently large and diverse dataset with accurately annotated abbreviations can be a challenging and time-consuming task. In this section, we describe the methodology used to generate synthetic data.

*1) Data collection and preprocessing:* The source corpus was the Polish Wikipedia. A sliding window is used to randomly select a context of 140 to 200 characters. This context length is representative of the PolEval abbreviation disambiguation dataset. Within each such context, a continuous span of words is randomly selected. These words are then processed with algorithmic abbreviation.

*2) Algorithmic abbreviation:* The custom abbreviation algorithm operates with four different strategies, as seen in Figure 2:

1. `abbr_first`: `profesor` → `prof.`
   Choose 1 to 4 of the first characters.
2. `abbr_first_last`: `profesor` → `pr`
   Choose the first and last characters.
3. `abbr_first_mid`: `profesor` → `pf.`
   Choose the first and one middle character.

Fig. 2. Algorithmic Abbreviation Schemes




4. `abbr_first_mid_last`: `profesor` → `pfr`
   Choose the first, one middle and last characters.

The algorithm applies a random strategy to each word in the span. It must be noted that the generated abbreviations are not guaranteed to be grammatically correct.

*3) Base form prediction:* The base forms of all words from the span before abbreviation are generated with the `spaCy pl_core_news_lg` model [17].

TABLE XIV
SYNTHETIC WIKIPEDIA EXAMPLE

| in.tsv | expected.tsv |
|---|---|
| **bs.** jest zgodny ze światem, w którym istnieje problem zła i cierpienie, a **<mask>** miłość jest ukryta przed wieloma osobami. Podobną argumentację | boska boski |

Each such context containing an abbreviated span is used as a dataset sample. Table XIV shows one generated example. The process repeats until reaching the end of the Wikipedia corpus.

*4) Considerations:* This synthetic dataset applies to a broader task of corrupted text restoration, where abbreviation disambiguation can be seen as a sub-task. In the context of abbreviation disambiguation, it is a low-quality dataset. This disadvantage is countered by its vast size of 14 million examples, which is 3375 times larger than the PolEval training dataset.

The exact version of the created dataset cannot be deterministically reproduced because of multi-process random number generation used during processing. A snapshot of the dataset is provided online[4].

## III. EVALUATION

The process of abbreviation disambiguation i.e. replacing abbreviations with their appropriate expansions in text can be divided into two stages.

The first stage is to find the abbreviation in a dictionary and replace it with an appropriate word or phrase that is its base forms.

[4]https://huggingface.co/datasets/carbon225/poleval-abbreviation-disambiguation-wiki

The second stage is to transform the result from the first stage into its correctly inflected grammatical form based on the context in which the abbreviation appears in the text.

It is worth noting that sometimes there is more than one meaning for a given abbreviation, so in the first stage it is also important to take the context into account in order to better predict which meaning is intended in a specified example.

Two metrics were used to objectively evaluate the replacement of abbreviation with their appropriate expansion in text:

- $Af$ - the accuracy of provided expanded forms of abbreviations
- $Ab$ - the accuracy of provided base forms of abbreviations

The matching check for both metrics was case-insensitive.

Based on the above metrics, the ultimate formula was defined to determine the final score:

$$Acc = 0.25 \cdot Af + 0.75 \cdot Ab \tag{1}$$

Therefore, the task of finding the appropriate base form is three times more important than the task of finding its appropriate expansion.

## IV. Methods

The solutions are based on a sequence to sequence model using the T5 [18] architecture. Krzysztof Wróbel's submission used the ByT5 [19] model, while Jakub Karbowski used the plT5 [20] model.

Both submissions used a similar workflow. The input to the transformer encoder is the context with the abbreviation. The transformer decoder generates both the base and inflected forms. Multiple methods of encoding the input and output of the model were used. They are described in detail in their corresponding sections.

In order to improve the results, majority voting with multiple models has been applied. The final decision is determined by the majority vote, where each model's prediction contributes one vote, and the outcome with the most votes is selected as the final prediction. This approach leverages the collective knowledge and expertise of multiple models to improve the accuracy and robustness of predictions in scientific studies. For this task, majority voting has been applied separately for inflected and base form.

## V. Experiments

### A. PolEval submissions

Initial experiments were carried out with limited time because of the competition deadlines. They were the base for further post-competition research. First, the exact methods used to produce the competition submissions are described.

*1) Krzysztof Wróbel submissions:* Proper validation is very important in every competition. The original validation (`dev-0`) dataset has only 300 samples which is insufficient for tracking scores with a precision of 0.1 percentage points. Therefore, 1000 samples from the training data were moved to the validation set.

The input data was prepared as follows: an abbreviation in the sentence was surrounded by `<abbrev>` and `</abbrev>`, e.g. `Komunistyczny deputowany, <abbrev>b.</abbrev> śledczy Prokuratury Generalnej`. The output data is structured as follows:

- for abbreviations: inflected form, separator `<sep>`, and base form, e.g. `były <sep> być`
- for non-abbreviations: the form, e.g. `b.`

The tokens `<abbrev>`, `</abbrev>`, and `<sep>` were added to the model vocabulary.

Initial experiments using Adafactor as an optimizer showed that the plT5 models performed slightly worse than the ByT5 models.

The training dataset was augmented by extracting abbreviations from dictionaries and applying them into sentences sourced from a corpus.

The final submission was created using majority voting on 3 models:

- trained on the training data and dictionary-based additional data using the development data for selecting the best model
- trained on the training data, development data, and dictionary-based additional data with two different seeds

The training parameters were as follows:

- model: byt5-base
- max input length: 250
- max output length: 100
- batch size: 16
- gradient accumulation: 16
- epochs: 24
- learning rate: 0.001
- scheduler: linear with warmup 0.1
- optimizer: Adafactor

TABLE XV
KRZYSZTOF WRÓBEL'S SUBMISSIONS TO POLEVAL. THE NAME IS THE SAME AS IN OFFICIAL LEADERBOARD.

| Description | Name | test-A | test-B |
|---|---|---|---|
| train | 3 | 90.78 | |
| train + dict | 5 | 91.32 | |
| train + dev + dict, seed 1 | 8 | 92.18 | 91.69 |
| train + dev + dict, seed 2 | 9 | 92.14 | 91.65 |
| voting (final) | 11 | **92.76** | **92.01** |

Table XV presents the results of Krzysztof Wróbel's submissions to the PolEval competition. The table includes different models and their corresponding scores on the `test-A` and `test-B` datasets.

The second model, named as `5` was trained on the training data along with additional dictionary-based data. This model performed better by 0.5 percentage points than model trained only on the training data.

The next two models, named as `8` and `9` were trained on the training data, development data, and dictionary-based additional data, using different random seeds for each. Model `8` achieved a score of 92.18 on the `test-A` dataset and 91.69

on the test-B dataset, while model 9 achieved a score of 92.14 on the test-A dataset and 91.65 on the test-B dataset.

The final model, named as 11 is the result of majority voting on three models: 5, 8, and 9. This model achieved the highest scores among all the submissions, with a score of 92.76 on the test-A dataset and 92.01 on the test-B dataset.

*2) Jakub Karbowski submissions:* Input data was encoded in a similar way to Krzysztof Wróbel's submission. The only difference is that <abbrev> and </abbrev> are not added as special tokens and are tokenized as raw text by the model's tokenizer. Instead of <abbrev> they are called <mask>.

The output format was the same as in Krzysztof Wróbel's submission, except the output of the model does not differ between abbreviations and non-abbreviations. The output format is: inflected form; base form, e.g. były; być.

Although ByT5 was considered because of its high performance on noisy data, plT5 was chosen because of limited training hardware available to the author of the submission. Training was performed on single GTX 1080 GPU with 8 GB of VRAM within a single day. Training ByT5 on this hardware would not be feasible.

First, pre-training was carried out on the Wikipedia dataset with synthetic abbreviations.

Pre-training parameters:

- model: plt5-base
- batch size: 4
- gradient accumulation: 64
- training steps: 3300
- learning rate: 0.0000928
- scheduler: linear with warmup 2000 steps
- optimizer: AdamW
- weight decay: 0.001

The training lasted 6 hours and was terminated after just 6% of the dataset. The pre-trained score achieved on the PolEval dev-0 dataset was 29.18%.

The pre-trained model was then fine-tuned on the PolEval train dataset.

Training parameters:

- model: plt5-base (wiki pre-trained)
- batch size: 8
- gradient accumulation: 32
- epochs: 223
- learning rate: 0.000015
- scheduler: linear with warmup 10%
- optimizer: AdamW
- weight decay: 0.0001

The per-device batch size could be increased because of a decrease in sequence length compared to the pre-training dataset. The score of the final submission with pre-training was 91.75% on test-A and 91.27% on test-B.

### B. Post-competition experiments

After the announcement of the competition results, the top two contestants combined their work to evaluate the performance of their methods, with respect to:

- model architectures
- used datasets and their combinations
- a broad range of hyperparameters
- original optimizations and solutions

*1) Setup:* A unified codebase for training was created[5]. It combines all of the methods and datasets used:

- ByT5 and plT5 models
- PolEval, dictionary-based and synthetic Wikipedia datasets
- majority voting

It also contains the hyperparameters and sweep configurations used during experimentation.

*2) Configurations:* Eight different configurations were chosen for final assessment. All combinations of the following options were used:

- Base model:
  - ByT5
  - plT5
- Pre-training dataset:
  - None
  - Wikipedia
- Fine-tuning dataset:
  - PolEval train
  - PolEval train with additional dictionary-based data

*3) Pre-training:* As pre-training on the large synthetic Wikipedia dataset was computationally expensive, sweeps on this dataset were not conducted. Instead, results from fine-tuning runs and manual experimentation provided the hyperparameters for pre-training.

*4) Fine-tuning:* For each configuration, a hyperparameter sweep was conducted. The sweeps considered: learning rate, weight decay, epochs, optimizer (AdamW or Adafactor). To provide a fair comparison between the two model architectures, each sweep was given 24h of computational time on an A100 GPU. The four sweeps with ByT5 managed to perform 16 training runs each, while plT5 sweeps performed 80 runs each.

*5) Voting:* Experiments involving majority voting were conducted to evaluate the performance of the best plT5 and ByT5 models. For each model, a set of 10 models was trained using identical parameters but different random seeds.

## VI. Results

Table XVI shows scores for experiments using plT5 and ByT5 models trained on different datasets. The highest scores are obtained using pre-training on synthetic data and then fine-tuned on train data with dictionary-based data. The models are shared at Hugging Face[6]. ByT5 consistently achieves higher scores than plT5. The results on the dev dataset do not correlate with the test data due to the small size of the dev dataset.

[5]https://github.com/Carbon225/poleval-2022-abbr

[6]https://huggingface.co/carbon225/plt5-abbreviations-pl, https://huggingface.co/carbon225/byt5-abbreviations-pl

Using the synthetic Wikipedia dataset for pre-training improves the performance of both models. For plT5, the `test-B` score improves by around 1% when considering both the pure PolEval `train` dataset and the additional dictionary data. For ByT5, the improvement is under 1%. Using additional dictionary data improves the scores by around 0.5%.

TABLE XVI
RESULTS FOR PLT5 AND BYT5 MODELS ON DIFFERENT TRAINING DATASETS

| | plT5 | | | ByT5 | | |
|---|---|---|---|---|---|---|
| | **dev** | **test-A** | **test-B** | **dev** | **test-A** | **test-B** |
| train | 91.80 | 90.76 | 90.06 | 94.10 | 92.10 | 91.73 |
| wiki-train | 91.39 | **91.76** | 91.32 | 93.61 | 92.46 | 92.53 |
| train-dict | 91.31 | 91.21 | 90.44 | 94.34 | 92.30 | 92.20 |
| wiki-train-dict | 91.31 | 91.64 | **91.37** | 93.44 | **92.71** | **92.92** |

TABLE XVII
SCORES OBTAINED USING MAJORITY VOTING AMONG 1 TO N MODELS. MODELS ARE SORTED BY TEST-A SCORE AND USED IN THAT ORDER.

| | plT5 | | ByT5 | |
|---|---|---|---|---|
| **Models** | **test-A** | **test-B** | **test-A** | **test-B** |
| 1 | 91.72 | 90.96 | 92.71 | 92.92 |
| 1-2 | 91.74 | 91.22 | 92.65 | 92.80 |
| 1-3 | 91.91 | 91.42 | 93.00 | 93.06 |
| 1-4 | 91.88 | 91.45 | **93.33** | 93.15 |
| 1-5 | 92.03 | 91.57 | 93.25 | **93.27** |
| 1-6 | 92.00 | 91.59 | 93.20 | 93.19 |
| 1-7 | 91.99 | 91.56 | 93.16 | 93.14 |
| 1-8 | 92.05 | 91.57 | 93.12 | 93.11 |
| 1-9 | 92.10 | **91.62** | 93.12 | 93.18 |
| 1-10 | **92.12** | 91.59 | 93.13 | 93.17 |

Table XVII provides the test-A and test-B scores for different combinations of plT5 and ByT5 models. The row labeled `1` represents the score obtained when only the first model is used. Subsequent rows, labeled `1-2`, `1-3`, and so on, indicate the scores obtained when additional models are included in the majority voting process. The maximum improvement observed through this process is approximately 0.5 percentage points.

TABLE XVIII
POLEVAL BEST RESULTS AND SCORES BY DIFFERENT SUBMISSIONS.

| | **test-A** | **test-B** |
|---|---|---|
| Krzysztof Wróbel | **92.76** | **92.01** |
| Jakub Karbowski | 91.75 | 91.27 |
| Marek Kozlowski | 89.00 | 88.73 |
| Jakub Pokrywka | 65.48 | 66.25 |
| Rafał Profiko | | 19.09 |

Table XVIII presents the best results and scores achieved by different submissions in the PolEval competition. The table includes two test metrics: `test-A` and `test-B`.

Krzysztof Wróbel emerged as the highest scorer, surpassing Jakub Karbowski by 0.74 percentage points in the test-B metric. Krzysztof Wróbel's success can be attributed to the implementation of the ByT5 model, majority voting, and the utilization of a larger validation dataset. Incorporating the pretraining step and utilizing the AdamW optimizer, as introduced in Jakub Karbowski's solution, has the potential to yield scores higher by more than 1 percentage point.

### A. Error analysis

The error analysis of 50 randomly selected errors made by the ByT5 model in the `wiki-train-dict` variant revealed that the model correctly predicted the answers for half of them. The dataset annotation needs to be improved.

More technical issues apply to about 1.5% of the examples. Approximately 0.76% of the examples in the dataset are annotated with multiple possible answers separated by a semicolon, such as `przeciw; przeciwko`. These cases were not properly taken into account during evaluation, About 0.72% of the examples in the dataset consist of multiword abbreviations with tokens separated by more than one space.

## VII. CONCLUSIONS

In this paper, we addressed the problem of abbreviation disambiguation in Polish press news using encoder-decoder models. The task involved replacing abbreviations with their appropriate expansions in text, taking into account the context.

Our experiments included submissions to the PolEval competition and post-competition research. In the PolEval competition, we achieved first and second place rankings. In the post-competition experiments, we conducted evaluations using different configurations, including pre-training on synthetic Wikipedia data and fine-tuning on additional data, which achieved a new state-of-the-art on the PolEval competition.

In conclusion, our study contributes valuable insights into the abbreviation disambiguation task in Polish press news. We emphasize the importance of proper validation, the trade-off between optimizer choice and memory usage, importance of pre-training, and the effectiveness of majority voting as a simple technique for improving results. Further research can build upon these findings to explore more advanced architectures, optimizations, and techniques for even better performance in Polish abbreviation disambiguation tasks.

Our approach can be easily applied to other languages and various types of texts.

## VIII. APPENDIX

Table XIX presents errors of the ByT5 model in the `wiki-train-dict` variant.

## IX. ACKNOWLEDGMENT

The competition submissions made by the team of Krzysztof Wróbel and Paweł Lewkowicz were completely independent of Jakub Karbowski's submissions. However, after the competition, the authors decided to collaborate and write a joint article due to the similarities in the methods used.

We gratefully acknowledge Polish high-performance computing infrastructure PLGrid (HPC Centers: ACK Cyfronet AGH) for providing computer facilities and support within computational grant no. PLG/2023/016304

The research has been supported by a grant from the Faculty of Management and Social Communication under the Strategic Programme Excellence Initiative at Jagiellonian University.

## REFERENCES


[1] A. Terada, T. Tokunaga, and H. Tanaka, "Automatic expansion of abbreviations by using context and character information," *Information Processing & Management*, vol. 40, no. 1, pp. 31–45, 2004. doi: https://doi.org/10.1016/S0306-4573(02)00080-8. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306457302000808

[2] Y. HaCohen-Kerner, A. Kass, and A. Peretz, "Abbreviation disambiguation: Experiments with various variants of the one sense per discourse hypothesis," in *Natural Language and Information Systems, 13th International Conference on Applications of Natural Language to Information Systems, NLDB 2008, London, UK, June 24-27, 2008, Proceedings*, ser. Lecture Notes in Computer Science, E. Kapetanios, V. Sugumaran, and M. Spiliopoulou, Eds., vol. 5039. Springer, 2008. doi: 10.1007/978-3-540-69858-6_5 pp. 27–39. [Online]. Available: https://doi.org/10.1007/978-3-540-69858-6_5

[3] Y. HaCohen-Kerner, A. Kass, and A. Peretz, "Combined one sense disambiguation of abbreviations," in *Proceedings of ACL-08: HLT, Short Papers*, 2008, pp. 61–64.

[4] Y. Wu, J. Xu, Y. Zhang, and H. Xu, "Clinical abbreviation disambiguation using neural word embeddings," in *Proceedings of BioNLP 15*, 2015, pp. 171–176.

[5] A. M. M. Jaber and P. Martínez Fernández, "Disambiguating clinical abbreviations using pre-trained word embeddings," 2021.

[6] L. Zhang, L. Li, H. Wang, and X. Sun, "Predicting Chinese abbreviations with minimum semantic unit and global constraints," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014. doi: 10.3115/v1/D14-1147 pp. 1405–1414. [Online]. Available: https://aclanthology.org/D14-1147

[7] A. Berdichevskaia, "Atypical lexical abbreviations identification in russian medical texts," *2022 12th International Conference on Pattern Recognition Systems (ICPRS)*, pp. 1–5, 2022.

[8] A. Mykowiecka and M. Marciniak, "Experiments with ad hoc ambiguous abbreviation expansion," in *Proceedings of the Tenth International Workshop on Health Text Mining and Information Analysis (LOUHI 2019)*. Hong Kong: Association for Computational Linguistics, Nov. 2019. doi: 10.18653/v1/D19-6207 pp. 44–53. [Online]. Available: https://aclanthology.org/D19-6207

[9] R. Kai and W. Shi-Wen, "Applying convolutional neural network model and auto-expanded corpus to biomedical abbreviation disambiguation." *Journal of Engineering Science & Technology Review*, vol. 9, no. 6, 2016.

[10] Q. Zhong, G. Zeng, D. Zhu, Y. Zhang, W. Lin, B. Chen, and J. Tang, "Leveraging domain agnostic and specific knowledge for acronym disambiguation," *CoRR*, vol. abs/2107.00316, 2021. [Online]. Available: https://arxiv.org/abs/2107.00316

[11] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 5485–5551, 2020.

[12] A. Rajkomar, E. Loreaux, Y. Liu, J. Kemp, B. Li, M.-J. Chen, Y. Zhang, A. Mohiuddin, and J. Gottweis, "Deciphering clinical abbreviations with a privacy protecting machine learning system," *Nature Communications*, vol. 13, no. 1, p. 7456, 2022.

[13] G. Song, H. Lee, and K. Shim, "T5 encoder based acronym disambiguation with weak supervision," *SDU@ AAAI-22*, 2022.

[14] J. L. Pereira, J. Casanova, H. Galhardas, and D. Shasha, "Acx: system, techniques, and experiments for acronym expansion," *Proceedings of the VLDB Endowment*, vol. 15, no. 11, pp. 2530–2544, 2022.

[15] G. Wenzek, M.-A. Lachaux, A. Conneau, V. Chaudhary, F. Guzmán, A. Joulin, and E. Grave, "CCNet: Extracting high quality monolingual datasets from web crawl data," in *Proceedings of the Twelfth Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, May 2020. ISBN 979-10-95546-34-4 pp. 4003–4012. [Online]. Available: https://aclanthology.org/2020.lrec-1.494

[16] W. Kieraś and M. Woliński, "Morfeusz 2 – analizator i generator fleksyjny dla języka polskiego," *Język Polski*, vol. XCVII, no. 1, pp. 75–83, 2017.

[17] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd, "spaCy: Industrial-strength Natural Language Processing in Python," 2020. doi: 10.5281/zenodo.1212303

[18] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *CoRR*, vol. abs/1910.10683, 2019. [Online]. Available: http://arxiv.org/abs/1910.10683

[19] L. Xue, A. Barua, N. Constant, R. Al-Rfou, S. Narang, M. Kale, A. Roberts, and C. Raffel, "Byt5: Towards a token-free future with pre-trained byte-to-byte models," *CoRR*, vol. abs/2105.13626, 2021. [Online]. Available: https://arxiv.org/abs/2105.13626

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

TABLE XIX
EXAMPLES OF ERRORS BY THE BEST MODEL

| input | expected | | predicted | |
|---|---|---|---|---|
| **m.** możliwy grad. Przewidywana wysokość opadów w burzach od 10 mm do 15 mm, w górach do 20 **<mask>** Temperatura maksymalna od 15 st.C w rejonie Zatoki Gdańskiej, 21 st.C na Suwalszczyźnie | metrów | metr | milimetrów | milimetr |
| **p.** procentowych (do 100 proc. ) przy dopłatach dla 1 osoby i o 10 punktów procentowych (do 40 **<mask>** proc.) dla każdej kolejnej osoby w gospodarstwie domowym najemcy. Ważne zmiany | pikseli | piksel | punktów | punkt |
| **róż.** Pazdanowi żona, Dominika, z wyraźnym onieśmieleniem przyjęła bukiet biało-czerwonych **<mask>** Piłkarz zrewanżował się własną reprezentacyjną koszulką. Potem nastąpiła seria | róż | róż | róż. | róż. |
| **o.** (są) do indywidualnego uzgodnienia z władzami' uczelni', czyli i tak daje ludziom **<mask>** Rydzyka zupełną dowolność. Wyższa Szkoła Kultury Społecznej i Medialnej w Toruniu | ojciec | ojciec | ojca | ojciec |
| **cm.** 38-letniego Granta. Amerykański pięściarz mierzy 201 cm, natomiast 33-letni Adamek - 187 **<mask>** Polak znacznie przegrywa z Grantem również pod względem zasięgu ramion. Dla Adamka, | centrymetrów | centrymetr | centymetrów | centymetr |
| **p.** C-331/94, Komisja p. Grecji, ECLI:EU:C:1996:211, pkt 10; C-111/05, Aktiebolaget NN **<mask>** Skatteverket, ECLI:EU:C:2007:195, pkt 55-58. [10] Z. Knypl, Polskie obszary morskie, | przeciwko | przeciwko | przeciw; przeciwko | przeciw; przeciwko |
| **p.** również osoby bez obywatelstwa, którzy publicznie znieważają osoby, wymienione w **<mask>** 1 Ustawy, przeszkadzają w realizacji praw osób walczących o niezależność Ukrainy | punkcie; paragrafie | punkt; paragraf | paragrafie | paragraf |
| **r.** zawodników urodzonych w 2001 roku i młodszych. Wcześniej we Włocławku walczyli piłkarze **<mask>** 1997. Tym razem nie będzie to turniej międzynarodowy, ponieważ cztery zaproszone | rocznik | rocznik | rocznika | rocznik |
| **p.** pogwałcenia praw osoby w świetle takich oświadczeń (uchwał polskich samorządów – **<mask>** A.J), ale na razie KE pilnie analizuje sytuację. Nie otrzymała, ale przecież mogła | przypis | przypis | pani | pani |
| **d.** nadzwyczajnej, którą odbyli w dniu wczorajszym, są deputowani Iwano-Frankowska (**<mask>** Stanisławów). „Krwawe stłumienie w sercu Europy pokojowych zgromadzeń przez uzbrojonych | dawny | dawny | dawniej | dawniej |
| **m.** Przemysłowym. Tramwaje linii 3, 23, 33>pl. Jana Pawła II skierowano objazdem przez **<mask>** Sikorskiego, Dubois, Nowy Świat. Tramwaje linii 10 i 20>pl. Jana Pawła II skierowano | most | most | mosty | most |
| **p.** konkursu. Oferty należy składać do 5 grudnia (nie decyduje data stempla pocztowego) w **<mask>** 223 w Starostwie Powiatowym. Obecnie placówkę prowadzi Zgromadzenie Sióstr św. | pokoju | pokój | pokój | pokój |
| **s.** William P. Young, Chata, tłum. A. Reszka, Wydawnictwo Nowa Proza, Warszawa 2009, **<mask>** 281. Ponad 6.000.000 sprzedanych egzemplarzy robi wrażenie na każdym, kto ma styczność | stron | strona | strona | strona |
| **m.** małopolskiego (i na 389. miejscu w Polsce) oraz II LO im. Tytusa Chałubińskiego na 70 **<mask>** (poza pierwszą 500 najlepszych liceów w Polsce). W ubiegłym roku „Kościuszko" był | miejscu | miejsce | metrach | metr |
| **f.** powierzchniową. Jak piszą Allaud L.A. i Martin M. bracia Schlumberger'owie przekonali **<mask>** Royal Dutch Shell, po powtórzeniu pomiarów i sprawdzeniu ich wiarygodności, że ta | firmę | firma | firmie | firma |
| **m.** dąbrowski) - rzeka Wisła o 69 cm m. Szczucin (pow. dąbrowski) - rzeka Szreniawa o 8 cm **<mask>** Biskupice (powiat miechowski) - rzeka Wisła o 145 cm Pustynia (powiat oświęcimski) - | miejscowość | miejscowość | miasto | miasto |
| **ub.roku.** krajowymi. Kupili ich w pierwszym kwartale o ponad 9 mld zł więcej niż na koniec **<mask>** – To pokazuje, że zagranica nie ucieka od naszego długu. Równoległy spadek nierezydentów | ubiegłego roku. | ubiegły roku. | ubiegłego roku | ubiegły rok |
| **zew.** wreszcie zbliża się ten dzień / wielki dreszcz emocji w nas / i wolności poczuj **<mask>** / Euro w barwach szczęścia jest / więc ramiona w górę wznieś / a dopóki piłka w grze | zew. | zew. | zewnętrznie | zewnętrznie |
| **zm.** pierwszym francuskim zwycięzcą Ligi Mistrzów. Zbigniew Pacelt (ur. 26 sierpnia 1951, **<mask>** 4 października 2021) - pływak i pięcioboista, dwukrotny olimpijczyk (1968 i 1972). | zmarły | zmarły | zmarł | zmarł |
| **l.** Parku Wodnego dla 3 l. koni półkrwi, Godz. 13.40 – gonitwa czwarta Puchar D&D dla 3 **<mask>** koni półkrwi, Godz. 14.25 – gonitwa piąta dla 3 l. ogierów i wałachów półkrwi, Godz. | letnich | rok | litrów | litr |