NeoN: A Tool for Automated Detection, Linguistic and LLM-Driven Analysis of Neologisms in Polish

Aleksandra Tomaszewska[0000−0001−6379−3034], Dariusz Czerski[0000−0002−3013−3483], Bartosz Żuk[0009−0008−8473−7718], and Maciej Ogrodniczuk[0000−0002−3467−9424]

Institute of Computer Science, Polish Academy of Sciences
firstname.lastname@ipipan.waw.pl

Abstract. We introduce NeoN, a tool for detecting and analyzing Polish neologisms. Unlike traditional dictionary-based methods requiring extensive manual review, NeoN combines reference corpora, Polish-specific linguistic filters, an LLM-driven precision-boosting filter, and daily RSS monitoring in a multi-layered pipeline. The system uses context-aware lemmatization, frequency analysis, and orthographic normalization to extract candidate neologisms while consolidating inflectional variants. Researchers can verify candidates through an intuitive interface with visualizations and filtering controls. An integrated LLM module automatically generates definitions. Evaluations show NeoN maintains high accuracy while significantly reducing manual effort, providing an accessible solution for tracking lexical innovation in Polish.[1]

Keywords: neologism detection and filtering · new words · lexical innovation · Large Language Models

1 Introduction

Neologism detection in Polish remains challenging due to the reliance on dictionary methods and manual curation. Traditional approaches, while informative, are time-consuming and prone to human bias [8,11,14]. Semi-automatic systems [2,8] provide interactive interfaces for candidate review but their dependence on basic filtering mechanisms limits their ability to capture the full spectrum of emerging lexical phenomena in Polish. In addition, although recent advances in Large Language Models (LLMs) have transformed many areas of natural language processing, no existing tool has yet used LLMs for analysis of new words in Polish. In this paper, we present NeoN, a tool designed for Polish neologism detection, monitoring, and analysis.

Rather than relying only on dictionary lookups, NeoN continuously processes RSS feeds through a multi-layered filtering pipeline. In NeoN, we focus on

[1] The prompt templates and a number of NeoN interface screenshots are available at https://drive.google.com/file/d/1THipys62nlU7panPnIVdAUKzIkOQGlUx/view

neologisms that have entered usage after 2020 to monitor ongoing changes in the language. The filters tailored for Polish help extract candidate neologisms and consolidate variants through frequency analysis, structural constraints, context-aware lemmatization, and the final LLM-driven precision-boosting filter. Its integrated LLM module automatically generates definitions. By combining corpus-based filtering with LLM-driven analysis, NeoN provides a scalable framework for tracking lexical innovation in Polish.

2 Related Work

Earlier studies [14] employed discriminant-based approaches to identify markers flagging potential neologisms. In the Polish context, the dictionary developed by the Language Observatory of the University of Warsaw (UWLO) [11], serves as a recognized but manually curated resource. Semi-automated systems like NeoCrawler [8] and Neoveille [2] have also emerged, using dictionaries and rule-based filters to extract new words from sources such as websites, blogs, and press releases. Although such tools reduce manual workload, they still require significant expert oversight and are seldom tailored to the nuances of Polish. Simultaneously, statistical and machine-learning approaches have advanced the field. For instance, Falk et al. [6] proposed a framework integrating form-related, morphological, and thematic features to detect neologisms in French newspapers. More recent efforts have incorporated LLMs for e.g., definition generation and translation [12,17] and unsupervised techniques that normalize variant forms via embedding-space mapping [16].

3 Functionalities

Our system processes daily RSS feed data through a unified pipeline integrating candidate extraction, variant grouping, and multi-layered filtering for removing noise candidates. All NeoN functionalities are integrated into a web-based interface allowing users to customize filter settings, review candidate lists, and export the resulting data in CSV format.

3.1 Form Filtering

In neologism detection, various filters are applied to determine the likelihood of a word being new or non-standard. The filters in NeoN are (1) **Frequency and occurrence**: Document frequency, Term frequency, Unique domain frequency, Domain distribution; (2) **Structural constraints**: Word length constraints, Invalid character check, Presence of digits, Triple repeated characters, (3) **Lexical validation**: Common Polish corpus check, English loanword detection; (4) **Spelling and typographical errors**: Polish word matching, Edit distance with diacritics, Adjacent character swap detection; (5) **Contextual analysis**: English context detection, Name Entity heuristic, Capitalization pattern; (6) **Other**: Compound word detection, Filtering using a few-shot LLM prompt.

The first innovation in our filtering framework is incorporating corpora alongside a dictionary as references for neologism validation. NeoN cross-references

extracted lexemes against the corpora to exclude existing Polish words, improving detection accuracy. General reference corpora are used by default, with an option to include a web corpus for recent data. The corpora include frequency lists from the National Corpus of Polish [1] (up to 2010), the Corpus of Contemporary Polish [9] (2011–2020), the web corpus NEKST [3] (up to 2020), and the latest Polish Wikipedia dump. The second innovation is employing LLMs as the final filtering stage. We use the `Llama-3.3-70B-Instruct` with a few-shot prompt with 3 positive and 3 negative examples to enhance identification. Recent studies show that LLMs can outperform humans in this task [17].

Experimental Setup The experiments involved a corpus of 233,538 web documents (873 RSS) from approx. 2 months. The documents underwent a processing pipeline that involved language detection, main content extraction to isolate the primary textual content from web pages, and NLP analysis to process the text using the `Hydra` [10] for NLP. Following filtering with a Polish language dictionary (Available at: `https://sjp.pl`), we generated a set of 200,696 candidate neologisms. The candidates were then refined through multi-step filtering to distinguish neologisms from noise or established lexemes.

Filtering Pipeline We implemented an iterative sequence of filters, each designed to eliminate specific types of non-neologisms. For consistency evaluation purposes (neologisms that appeared after 2020), only selected filters were used. (1) **Length constraints**: at least 3 characters and no more than 20 characters; (2) **Numerical content**: must not contain digits; (3) **Frequency**: must appear in more than 5 documents; (4) **Case sensitivity**: must appear in lowercase at least 5 times; (5) **Proper noun exclusion**: must not function as proper nouns in at least 5 occurrences; (6) **Edit distance**: the minimum edit distance to any known word in the Polish dictionary must exceed 0.5; (7) **Spelling**: must not be diacritical variations, result from swapping adjacent letters of existing Polish dictionary words and not contain triple repetitions of the same letter; (8) **English dictionary check**: if a word appears in an English dictionary, it must occur in at least 5 Polish-language contexts; (9) **Exclusion from other dictionaries**: must not be present in the corpora or dictionaries. (10) **LLM filtering**: filtering (we used the `Llama-3.3-70B-Instruct` model) based on a few-shot prompt.

For evaluation, we used data added to the UWLO after 2020 [11]. This timeframe aligns with our primary reference corpora. We preprocessed the dataset by removing neologisms already listed in the Polish dictionary, as these typically reflect existing words with new meanings rather than new lexemes. After preprocessing, our set included 610 neologisms. To ensure the corpus was suitable for evaluation, we expanded it by gathering the top 100 Google search results for each neologism in the training set. For a more precise evaluation, we conducted a manual review of 1,740 neologisms identified during the final filtering stage – before applying the LLM filter – excluding those already in the UWLO. This method enables a more effective evaluation of our tool's precision. The results obtained are presented in Table 1 as an additional section labeled 'Including human-annotated data'. The assessment was conducted by 3 individuals: 2 an-

notators and 1 adjudicator who resolved conflicting evaluations. The recall for this set is identical to that of the test set, as the manual annotations did not change the number of detected neologisms. This process only validated the existing candidates without adding or removing any, allowing the focus to shift to precision and F1 scores based on these annotations.

Experimental Procedure The experiment was conducted iteratively, each iteration introducing an additional filter. At each stage, we evaluated filtering performance using precision, recall, and F1 score, comparing filtered candidates to the testing set ground truth. The results are presented in Table 1.

Table 1: Results of incremental filtering in the neologism detection. (*) The recall measure cannot be effectively calculated based only on annotated data.

	Test set				
Conditions	**All**	**Matches**	**Precision**	**Recall**	**F1**
No filter	200 696	610	0.003	0.993	0.006
+ Min Token Len	199 977	610	0.003	0.993	0.006
+ Max Token Len	199 289	609	0.003	0.992	0.006
+ No Digits	186 422	609	0.003	0.992	0.007
+ Freq ≥ 5	33 801	607	0.018	0.989	0.035
+ Non-Uppercase Freq ≥ 5	5 116	603	0.118	0.982	0.210
+ Non-NE Freq ≥ 5	4 198	597	0.142	0.972	0.248
+ Min Edit Distance	3 130	552	0.176	0.899	0.295
+ Spelling	2 726	549	0.201	0.894	0.329
+ Non-Eng Freq ≥ 5	2 657	549	0.207	0.894	0.336
+ Not in NKJP	1 784	538	0.302	0.876	0.449
+ Not in KWJP100	1 740	536	0.308	0.873	0.455
+ LLM filtering	1 056	536	0.508	0.873	**0.642**
	Including human-annotated data				
+ Not in KWJP100	1 740	1 385	0.796	(*)	—
+ LLM filtering	1 056	968	**0.917**	0.699	**0.793**

Summary of Results The detection pipeline, integrating rule-based filters and a LLM, effectively identified new Polish lexemes in a noisy web corpus, achieving F1 scores of 0.642 on the test set and 0.793 on the annotated set. At the final filtering stage (before applying the LLM filter), recall cannot be computed because the process does not retain information about false negatives. In earlier stages, only the candidates that passed the filter are tracked, so any items mistakenly removed (false negatives) are lost, making it impossible to determine recall accurately. Starting with 200,696 candidates, the pipeline reduced this to 1,056 highly probable neologisms, with precision rising to 0.508 and recall settling at 0.873. Rule-based filters drastically reduce non-neologistic candidates with minimal recall loss, while the LLM filter increases precision from 0.308 to 0.508 (test set) and from 0.796 to 0.917 (annotated set) by leveraging contextual and semantic cues. This demonstrates that LLMs today can substantially

enhance the neologism detection process. Each filter contributed to a stepwise improvement, making this approach highly effective.

3.2 Form Grouping

NeoN detects alternative spellings, inflectional forms, and syntactic variants of neologisms, including multi-word forms (e.g., *tusko-bus*, *tuskobus*). Post-processing groups related forms (e.g., hyphenated, spaced), aggregates frequencies, and lemmatizes variants—a key step for Polish's rich morphology. We evaluated four tools: `Stanza` [15] and `spaCy` [5] (general NLP toolkits), `Hydra` [10] (Polish-specific), LLMs `GPT4o` [13] and `DeepSeek-R1` [4] with custom prompts.

Using a UWLO dataset with 978 neologisms ($\geq$3 forms each; 3,659 total), we assessed lemmatization quality. Standard accuracy fails to capture consistency across inflectional groups. We propose 2 group-based metrics over neologism groups G: **Group Accuracy** $A_{gr} = \frac{S}{G}$, where S = groups with all forms mapped to the same lemma. **Strict Group Accuracy** $A_{strict} = \frac{K}{G}$, where K = groups all correctly mapped.

Experimental Setup We tested lemmatization in 2 setups: isolated words (e.g., *NFTs* $\rightarrow$ *NFT*) and contextualized sentences (e.g., *They NFTs gained popularity* $\rightarrow$ *NFT*).

Table 2: Neologism lemmatization results.

Experiment	Model	Accuracy	Group Accuracy	Strict Group Accuracy
Without context	`SpaCy`	50.18%	14.52%	13.50%
	`Stanza`	73.41%	**53.58%**	**50.41%**
	`Hydra`	72.01%	49.08%	46.22%
	`GPT4o`	72.81%	53.07%	49.90%
	`DeepSeek-R1`	**75.13%**	51.53%	49.80%
With context	`SpaCy`	52.94%	16.26%	15.44%
	`Stanza`	73.35%	51.94%	48.77%
	`Hydra`	**79.31%**	62.47%	**60.22%**
	`GPT4o`	78.57%	**62.99%**	59.41%
	`DeepSeek-R1`	77.51%	57.16%	55.32%

Summary of Results Experiments on neologism lemmatization show large performance gaps across models. Basic tools like SpaCy perform poorly ($\approx$50%), while Stanza reaches $\approx$73% but surprisingly drops slightly with context. Hydra, optimized for Polish, performs best with 79.31% accuracy and 60.22% strict group accuracy. LLMs like `GPT4o` and `DeepSeek-R1` also perform well, especially without context, and remain competitive with it. These results underline the limits of basic tools, the contextual strength of Hydra, and the promise of LLMs. Future research should focus on fine-tuning a specialized LLM that integrates Hydra's contextual strengths with LLMs' robustness.

3.3 Definition Generation

We conducted experiments to evaluate LLMs' capability to automatically generate neologism definitions, focusing on the most recent lexemes. We only selected neologisms registered in 2024 in UWLO. For each lexeme, we obtained definitions and usage examples from the UWLO website. We filtered out entries with fewer than 5 examples, resulting in a final dataset of 81 neologisms. Our experiments used `Llama-3.3-70B-Instruct` [7], with a knowledge cutoff date of December 2023, and `DeepSeek-R1` [4], no known cutoff date as of February 28, 2025. We chose `DeepSeek-R1` to compare newer reasoning-focused models against traditional LLMs like `Llama-70B`.

Evaluation Protocol We test 3 prompting setups: (1) the 0-shot setup where we do not provide any examples of neologism usage, (2) 3-shot, and (3) 5-shot where we provide 3 and 5 examples of their usage, respectively. For all the experiments, we sampled the models using the recommended temperature of 0.6 and top-p value of 0.95. We evaluated the generated definitions using the *LLM-as-a-judge* approach [18], which employs LLMs to score, rank, or select from candidate options. For our experiments, we used the `GPT4o` model [13] (knowledge cutoff: October 2023) as the judge, performing pointwise evaluations - the judging LLM compared the generated definition against a human-made reference, outputting `CORRECT` or `INCORRECT`. This setup largely follows [17] and focuses exclusively on the definition correctness. To increase quality, we included all 5 usage examples in the prompt.

Summary of Results Figure 1 shows the accuracy of models in the pointwise evaluation. Performance improved monotonically with additional usage examples. `DeepSeek-R1` outperformed `Llama-70B` across all setups, achieving the max. 96% accuracy in the 5-shot setup compared to `Llama-70B`'s 88%. Table 3 presents results for each setup.

Meta Evaluation Upon manual inspection, we conducted a meta evaluation verifying `GPT4o`'s effectiveness as a judge. Using 3 human annotators, we evaluate the generated definitions. We focus only on the 5-shot setup as it produces the best results across both models. The evaluation followed our previously described protocol. The results of the meta evaluation are presented in Figure 2. Human annotators showed high agreement with `GPT4o`'s judgments, consistently rating `Llama-70B` lower than `DeepSeek-R1`, which aligns with the results in Figure 1. Annotators varied in their strictness: Annotators 2 and 3 marked more definitions as incorrect compared to `GPT4o`, while Annotator 1 was more lenient, marking only 2 `Llama-70B` definitions as incorrect and none for `DeepSeek-R1`.

4 Conclusions and Future Work

We presented NeoN, a web-based system that integrates corpus-driven filtering, context-aware lemmatization, and LLM-based validation and definition generation to automate Polish neologism detection. Our multi-stage pipeline reduced an initial set of 200 696 candidate tokens to 1 056 high-confidence neologisms,

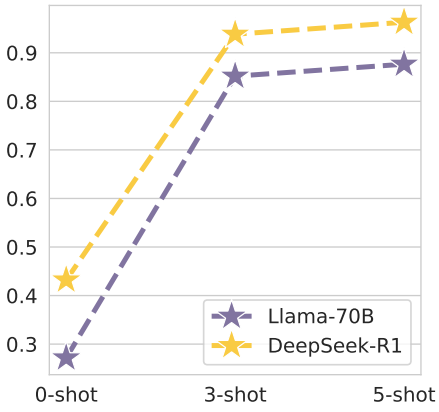

Fig. 1: Accuracy of `DeepSeek-R1` and `Llama-70B` in pointwise evaluation across three prompting setups.

Table 3: Results for pointwise evaluation of `DeepSeek-R1` and `Llama-70B` across 3 prompting setups.

	Verdict	
	Correct	Incorrect
`Llama-70B` 0-shot	22	59
`Llama-70B` 3-shot	69	12
`Llama-70B` 5-shot	71	10
`DeepSeek-R1` 0-shot	35	46
`DeepSeek-R1` 3-shot	76	5
`DeepSeek-R1` 5-shot	78	3

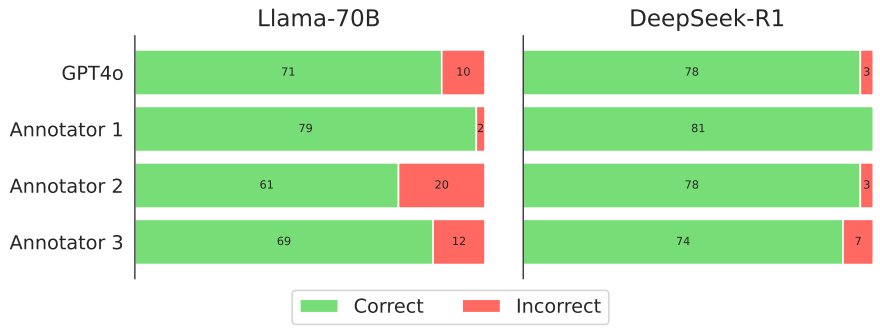

Fig. 2: Results of pointwise meta evaluation shown for 3 human annotators and `GPT4o` (LLM judge) across 2 judged models: `Llama-70B` and `DeepSeek-R1`.

achieving an F1 score of 0.642 on held-out data (0.793 on expert-annotated data) and exceeding 0.90 precision after the LLM filter. In 5-shot prompting, the LLM module produced definitions with up to 96% accuracy, as confirmed by three linguists. This end-to-end framework markedly lowers manual intervention, consolidates inflectional and orthographic variants, and offers visualizations for tracking lexical innovation, all without requiring programming expertise. Future work will enable researchers to upload custom corpora, extending beyond RSS feeds, and will explore fine-tuning open LLMs on manually and semi-automatically annotated neologism datasets, supplemented with synthetic examples to improve base-form detection. We also plan to develop fully LLM-driven detection workflows and to release benchmark datasets and standardized

evaluation protocols to foster reproducible research in automated neology and lexical innovation analysis.

References

1. Narodowy Korpus Języka Polskiego. Wydawnictwo Naukowe PWN, Warsaw (2012)
2. Cartier, E.: Neoveille, a Web Platform for Neologism Tracking. In: Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the ACL. pp. 95–98. ACL (2017)
3. Czerski, D., Ciesielski, K., Dramiński, M., Kłopotek, M., Łoziński, P., Wierzchoń, S.: What NEKST? — Semantic Search Engine for Polish Internet. In: Challenging Problems and Solutions in Intelligent Systems, pp. 335–347. Springer International Publishing, Cham (2016)
4. DeepSeek-AI, Guo, D., Yang, D., et al.: DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning (2025)
5. Explosion: spaCy: Industrial-Strength Natural Language Processing in Python (2020)
6. Falk, I., Bernhard, D., Gérard, C.: The Logoscope: a Semi-Automatic Tool for Detecting and Documenting French New Words (2018)
7. Grattafiori, A., Dubey, A., Jauhri, A., et al.: The Llama 3 Herd of Models (2024)
8. Kerremans, D., Stegmayr, S., Schmid, H.J.: The NeoCrawler: identifying and retrieving neologisms from the internet and monitoring ongoing change, pp. 59–96. De Gruyter Mouton, Berlin, Boston
9. Kieraś, W., Marciniak, M., Łaziński, M., Woliński, M., Bojałkowska, K., Eźlakowski, W., Kobyliński, L., Komosińska, D., Krasnowska-Kieraś, K., Rudolf, M., Tomaszewska, A., Wołoszyn, J., Zawadzka-Paluektau, N.: Korpus Współczesnego Języka Polskiego. Dekada 2011–2020. Język Polski (2024)
10. Krasnowska-Kieraś, K., Woliński, M.: Parsing Headed Constituencies. In: Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation. pp. 12633–12643. ELRA and ICCL (2024)
11. Kłosińska, K.: Mikro- i makrostruktura słownika neologizmów Obserwatorium Językowego Uniwersytetu Warszawskiego. LingVaria **19**(2(38)), 47–61 (2024)
12. Lerner, P., Yvon, F.: Towards the Machine Translation of Scientific Neologisms. In: Proceedings of the 31st International Conference on Computational Linguistics. pp. 947–963. ACL (2025)
13. OpenAI, Achiam, J., Adler, S., et al.: GPT-4 Technical Report (2024)
14. Paryzek, P.: Comparison of selected methods for the retrieval of neologisms. Investigationes linguisticae **16**, 163–181 (2008)
15. Qi, P., Zhang, Y., Zhang, Y., Bolton, J., Manning, C.D.: Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. In: Proceedings of the 58th Annual Meeting of the ACL: System Demonstrations. pp. 101–108 (2020)
16. Zalmout, N., Thadani, K., Pappu, A.: Unsupervised Neologism Normalization Using Embedding Space Mapping. In: Proceedings of the 5th Workshop on Noisy User-generated Text. pp. 425–430. ACL, Hong Kong, China (2019)
17. Zheng, J., Ritter, A., Xu, W.: NEO-BENCH: Evaluating Robustness of Large Language Models with Neologisms. In: Proceedings of the 62nd Annual Meeting of the ACL (Volume 1: Long Papers). pp. 13885–13906. ACL (2024)
18. Zheng, L., Chiang, W.L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E.P., Zhang, H., Gonzalez, J.E., Stoica, I.: Judging LLM-as-a-judge with MT-bench and Chatbot Arena. In: Proceedings of the 37th International Conference on Neural Information Processing Systems. pp. 46595–46623 (2023)