PLeW-NLG: Interactive Exploration of Human Evaluation Data across NLG Tasks

Fahime Same[1], Leonard Sprague[2], Francesco Cangemi[2]

[1]University of Cologne, [2]The University of Tokyo

Correspondence: fahimeh.same@gmail.com, lsprague@g.ecc.u-tokyo.ac.jp, cangemi@g.ecc.u-tokyo.ac.jp

Abstract

Human evaluation is the gold standard in NLG, yet results are usually reported as aggregate tables detached from the underlying judgments. We present PLeW-NLG, a tool for interactive exploration of judgment-level evaluation data: dimensions like system, criterion, and annotator map onto a grid of small-multiple panels, and any point can be inspected or edited. A case study based on the SimpEval dataset shows how, because the judgments are directly accessible, the analysis can follow the data rather than a pre-planned path.

1 Introduction

Human evaluation is standard practice in Natural Language Generation (NLG) (van der Lee et al., 2019), yet the way these evaluations are conducted and communicated suffers from two persistent issues. First, **practices are heterogeneous**: criteria, scales (Likert, sliders, pairwise, spans), and terminology vary across subfields, and standardization efforts (Belz et al., 2021) are only partially adopted. Second, results are **detached from the data**: readers see mean scores per system and a few inline examples, with no easy, intuitive way to explore the underlying judgments or surface new insights. Judgment-level files, if released at all, sit in CSV/JSON that require scripting to explore further. This detachment makes it hard to ask basic questions of the data.

PLeW-NLG addresses the detachment problem via interactive access to judgment-level data. It builds on PLeW (Cangemi et al., in press), a browser-based multidimensional visualization tool for linguistic studies, available at https://www.francescocangemi.com/plew, which we adapt and augment for NLG human evaluation. PLeW-NLG lets researchers (1) *visualize* judgment-level data across dimensions (e.g., system, criterion), (2) *inspect* any point down to its full text and media, and (3) *edit* annotations in place with an exportable change log. To handle heterogeneous files, we pair the tool with a reusable transformation skill and demonstrate on a curated set of public human-evaluation datasets across NLG tasks.

Code, processed datasets, and the transformation skill are at https://github.com/fsame/PLeW-NLG; the live editor and example gallery are at https://fsame.github.io/PLeW-NLG/; and a short screencast is at https://tinyurl.com/yyjubnv3.

2 Interface

PLeW-NLG is a static, install-free web app that ingests flat CSV/JSON whose columns carry role prefixes: *dim::* for categorical dimensions, *desc::* for long text shown on demand, *med::* for media, and *res::* for metadata.

Six dimensions map to grid X/Y, color, panel X/Y, and shape, rendering the data as a lattice of small-multiple point panels. For a typical (item $\times$ system $\times$ criterion $\times$ annotator) file, mapping system $\times$ criterion to the grid, score to color, and annotator to panels exposes disagreement and per-category weaknesses at a glance. Dimensions are re-mapped by drag-and-drop, so switching perspective takes seconds rather than a script edit, and any value can be toggled off from the dimension list or legend to drill down, e.g. on a system pair or the lowest scores. Clicking a point opens its full record, linking the aggregate view back to individual records. Points can be dragged between cells to reclassify them, e.g. to correct an erroneous label; every edit is written to a change log, and both the log and the revised dataset can be exported for auditable re-annotation. For reproducibility, the complete setup exports as JSON and can be re-imported later or by another user; figures save as PNG, and a report view provides per-dimension statistics and crosstabs.

Figure 1 shows the main PLeW-NLG grid editor

Proceedings of the 19th International Natural Language Generation Conference: System Demonstrations, pages 4–7
October 17-21, 2026 ©2026 Association for Computational Linguistics

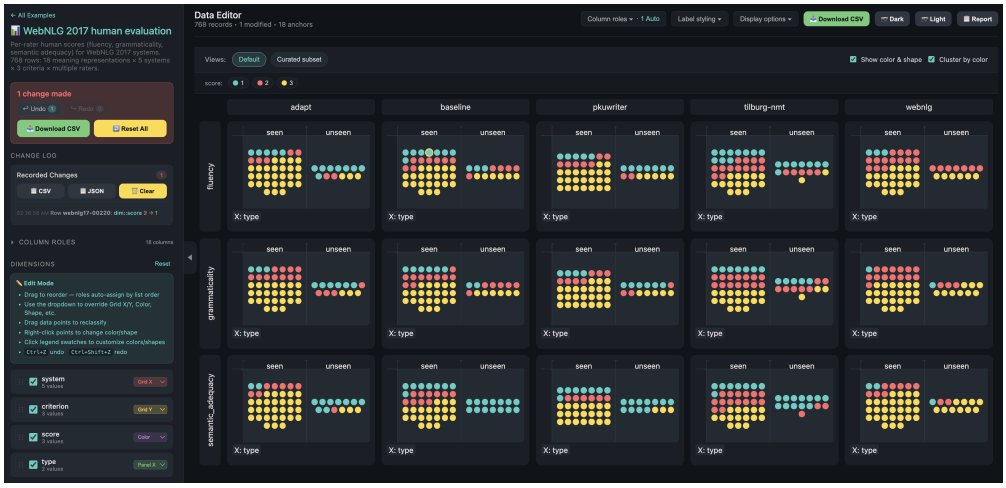


Figure 1: The PLeW-NLG grid editor on WebNLG 2017 human-evaluation judgments.

on WebNLG 2017 human-evaluation judgments. The left sidebar lists the active dimensions (*system*, *criterion*, *score*, *type*), the change log, and undo/redo controls. The grid maps *system* to the X-axis (columns) and *criterion* to the Y-axis (rows), with each cell's judgments drawn as points colored by *score*; *type* (*seen* / *unseen*) splits each cell into panels. The screenshot also shows a re-annotation in progress: one point is highlighted (baseline × fluency × seen), the sidebar reports a single change, and the change log records the edit (`dim::score` 2 → 1 for row `webnlg17-00220`).

3 Datasets and Transformation Skill

Dataset selection. We focus on human-evaluation data for two reasons: such judgments remain the gold standard of NLG evaluation yet are rarely explored beyond the aggregate tables of the original paper, and their structure suits PLeW, with a few low-cardinality dimensions (system, criterion, annotator) wrapped around long, unique text. To sample NLG subfields systematically, we used the GEM benchmark data cards (Gehrmann et al., 2021), and searched the ACL Anthology for papers that both evaluate a GEM-listed dataset with human judges and release the judgments. We supplement these with datasets beyond GEM. The catalog is listed in Table 1, Appendix A.

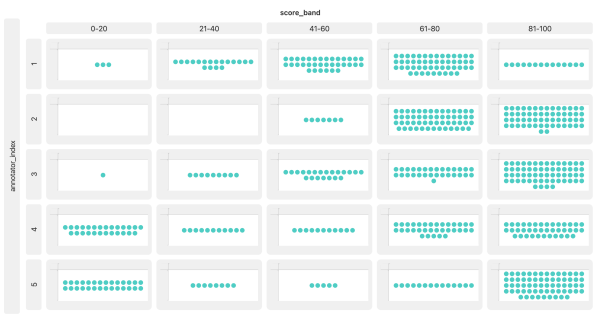
Transformation skill. Since released judgment files are heterogeneous (nested JSON, wide multi-criterion CSVs, blinded pairwise records), we package conversion as a reusable agent skill, `plew-prepare-dataset`. Given a raw dataset, it examines the data, checks its suitability for PLeW, reshapes it via a small pattern library (melting criteria, exploding annotator arrays, resolving pairwise winners), assigns prefixes by inferred role, and subsets to demo scale with a fixed seed. The output is a ready-to-load dataset, a reproducible transformation script, and documentation. Since the script ships with the data, inferred column roles stay open: to move a column onto the grid or into the record window, change its prefix in the script and re-run.

Skill coverage. We ran the skill on a set of public judgment datasets and picked nine of them for the gallery (Table 1, Appendix A) so the examples cover a range of NLG subfields. In only one case could the skill not produce a PLeW-ready dataset: HEAP (Mehri and Shwartz, 2023) keeps only good/bad text pairs, with no annotator, system, or item identifier to place on a grid. All the others were reshaped into a format the grid can load. The current version of PLeW builds its panels from categorical fields, so continuous judgments stay with the record and reach an axis only once binned, which the skill can do during preparation (e.g., score-band column in Figure 2).

4 Demonstration

This section shows how judgment-level exploration makes the data accessible in ways aggregate score tables do not. Researchers can interrogate the data along whatever dimension matters to them, discover unexpected patterns, and drill down to the individual judgments behind them. Consider annotator behavior in the SimpEval dataset (Maddela et al., 2023)[1]: five annotators rate each system output for overall simplification quality on a 0–100

[1]These observations are illustrative and, being based on a subsample, not definitive.



(a) Score band × annotator.



(b) Same layout with system as color.

Figure 2: SimpEval in PLeW-NLG: left, annotators use the 0–100 scale very differently; right, color by system shows low scores concentrate on `scramble_words`, except for annotator 2.

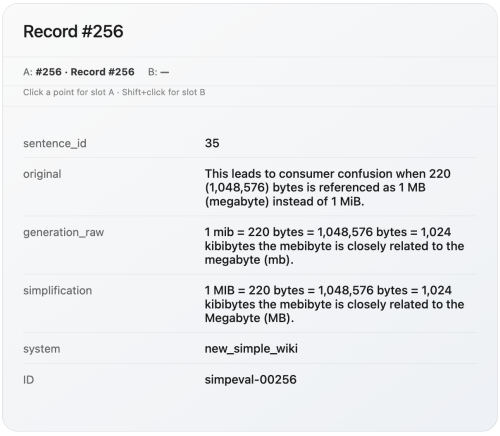


Figure 3: Record window for sentence 35.

slider, which we bin into five 20-point bands. Our subset covers 20 source sentences and 6 systems. Figures in this section, including the record window, are exported directly from the interface.

Placing *score band* on the grid X-axis and *annotator* on the Y-axis (Figure 2a) reveals sharply different rating behaviors. Annotator 2 never scores below 41 and places only 7 in the 41–60 band, whereas annotators 4 and 5 place 26 and 28 judgments in the 0–20 band. Adding *system* as color explains the split (Figure 2b): the low bands are dominated by `scramble_words` (purple), a deliberately corrupted baseline, so the more extreme raters are accurately discriminating, while annotator 2 compresses the same signal into the 41–80 range. The view also surfaces a shared outlier: all five annotators rate sentence 35 (circled in red) lowest among the outputs of `new_simple_wiki` (blue), an otherwise strong system. Clicking the dot in the platform retrieves the sentence in question, which turns out to involve ambiguous megabyte notation (Figure 3). This exploration proceeded without pre-defined hypotheses. Since the judgments are directly accessible, the analysis could go wherever the data pointed rather than where we had planned to look, and engaged us with the data in a qualitatively different way than the aggregates we normally see.

References

Anya Belz, Anastasia Shimorina, Shubham Agarwal, and Ehud Reiter. 2021. The ReproGen shared task on reproducibility of human evaluations in NLG: Overview and results. In *Proceedings of the 14th International Conference on Natural Language Generation*, Aberdeen, Scotland, UK.

Francesco Cangemi, Leonard Sprague, and Fahime Same. in press. Collaborative exploration of complex data landscapes with PLeW. *Oral Archives Journal*.

Sebastian Gehrmann, Tosin Adewumi, Karmanya Aggarwal, et al. 2021. The GEM benchmark: Natural language generation, its evaluation and metrics. In *Proceedings of the First Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*.

Mounica Maddela, Yao Dou, David Heineman, and Wei Xu. 2023. LENS: A learnable evaluation metric for text simplification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Shuhaib Mehri and Vered Shwartz. 2023. Automatic evaluation of generative models with instruction tuning. In *Proceedings of the Third Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 42–52, Singapore. Association for Computational Linguistics.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, Sander Wubben, and Emiel Krahmer. 2019. Best practices for the human evaluation of automatically generated text. In *Proceedings of the 12th International Conference on Natural Language Generation*.

NLG field	Example	Paper	Download	Judgment type
Data-to-text	WebNLG 2017 human evaluation	WebNLG Challenge (Gardent et al., INLG 2017)	GitLab	Fluency, grammar, semantic adequacy
REG	NeuralREG human evaluation	NeuralREG (Castro Ferreira et al., ACL 2018)	GitHub	1–7 Likert × fluency, grammar, clarity
Summarization	SummEval	SummEval (Fabbri et al., TACL 2021)	GitHub	1–5 Likert × 4 criteria (expert + crowd)
Summarization	SEAHORSE	SEAHORSE (Clark et al., EMNLP 2023)	ZIP	Yes/No/Unsure × 6 criteria × multiple systems
Text simplification	SimpEval (LENS)	LENS / SimpEval (Maddela et al., ACL 2023)	GitHub	0–100 overall quality (Rank & Rate)
Image captioning	Flickr8k	Flickr8k ranking eval (Hodosh et al., JAIR 2013)	ZIP	Expert 1–4 Likert; crowd yes/no match
Image captioning	THumB 1.0	THumB 1.0 (Kasai et al., NAACL 2022)	GitHub	Precision/recall (1–5) + fluency, conciseness, inclusive-language penalties
LLM-as-judge	Book highlights (preference judgment)	Book highlights eval (Same et al., RetroEval 2026)	GitHub	Pairwise A/B preference × criteria + overall (confidence, rationale)
LLM-as-judge	Book highlights (faithfulness)	Book highlights eval (Same et al., RetroEval 2026)	GitHub	Factuality (accurate/inaccurate), divergence type, severity

Table 1: Human-evaluation examples currently shown in PLeW-NLG.

A Evaluation Datasets

Table 1 lists the human-evaluation datasets currently available as interactive examples in PLeW-NLG (https://fsame.github.io/PLeW-NLG/examples/). This compilation is ongoing and the list will be extended over time. One important limitation is that the example releases are typically *subsets* of the underlying corpora rather than the full judgment sets. Patterns visible in an example view should therefore be treated as illustrative rather than as definitive findings about the complete data. For deeper analysis, we recommend loading the full dataset through the PLeW full editor (§2), which supports the same exploration workflow without the demo subsampling applied to the gallery examples. In addition to conventional human-evaluation benchmarks, we include an LLM-as-judge study on book highlights to demonstrate that the platform can accommodate other evaluation paradigms and to broaden the diversity of tasks represented in the gallery.