Helping Humans Align Efficiently

Jan Frederik Schaefer[1], Michael Kohlhase[2]

Both formal and informal mathematical knowledge is fragmented across many different systems and formats. Aligning the concepts across these resources is a key contribution towards making the data FAIR (Findable, Accessible, Interoperable, Reusable). An alignment links a concept from one system to a concept in another system, indicating that they represent the same object or idea. We ignore the added complexity that concepts may only be similar, but not a precise match (see [1] for a discussion of this issue). Authors and curators often try to generate alignments from existing datasets automatically (via rule-based matching, machine learning, or LLM techniques), but few parallel corpora exist for mathematics, and unsupervised approaches seem difficult for the subtle distinctions crucial in mathematical knowledge. We claim that human-created alignments are necessary – to ensure high quality, or at least to create datasets for training and evaluation.

Example Alignments can be created between diverse resources, including knowledge graphs, libraries for automated theorem provers, and documents. As a running example, consider the simple sentence

> Let n be a natural number.

where we want to align the concept reference "natural number" with the corresponding WikiData concept. Such an alignment could, for example, provide additional information to the reader, or determine the prerequisite knowledge for understanding the document. It could also be a first step towards full formalization: if we also align e.g. the type of natural numbers in Lean's mathlib with Wikidata, we have an indirect alignment between the document and mathlib.

In our example there are, in fact, three possible alignment targets:

- `Q21199` (natural numbers, possibly including 0)
- `Q28920044` (positive integers, i.e. natural numbers excluding 0)
- `Q28920052` (non-negative integers, i.e. natural numbers including 0)

This kind of ambiguity is a major challenge for automated alignment. While a human aligner, especially one familiar with the document, may well be able to correctly disambiguate, they face the additional challenge of remembering what concepts exist or searching for them, which takes significant time and effort.

Contribution In this paper, we propose a workflow for human alignment supported by a simple tool that iterates over the data to be aligned and presents the user with a list of alignment candidates to select from (see Figure 1). For an example implementation, we build on the snify architecture for selection-based, corpus-based semantic annotation [2] and extend it, prototypically, to the alignment problem.

Proposed Architecture To support the workflow in Figure 1, we propose an architecture centered around a **concept glossary** (see Figure 2). The concept glossary is extracted from the concept collection that we want to align with – in our example, WikiData – using a custom **harvester**. The concept glossary contains for each concept a unique **identifier**, a human-readable **description**, and a set of **verbalizations** (natural language renderings of the concept). For example, a concept glossary extracted from WikiData could have the following entry:

- *identifier*: `https://www.wikidata.org/wiki/Q28920044`
- *description*: "*positive integer* (integer greater than zero; natural number explicitly excluding zero)"

Workshop on Math Datasets: alignments and comparisons (ALIGN2025), October 10, 2025, Brasilia, Brazil

© 2025 This work is licensed under a "CC BY 4.0" license.

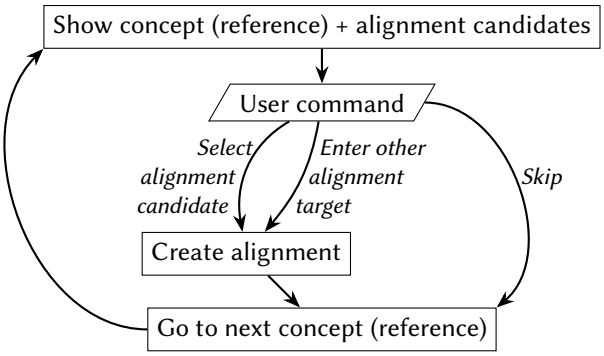

Figure 1: Proposed flowchart for tool support for human aligners.

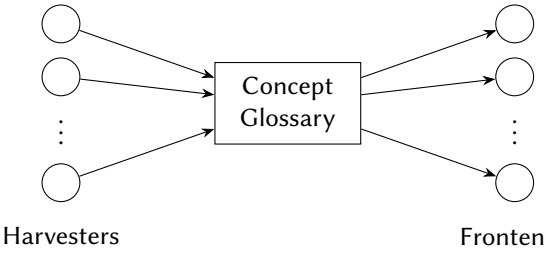

Figure 2: Proposed architecture: Custom harvesters extract a concept glossary from concept collections. The concept glossary can be used by frontends customized to the data we want to align.

- *verbalizations*: "positive integer", "integer greater than zero", "natural number", ...

The description is necessary to help the user identify the correct concept as the identifiers are not human-readable. The (language-specific) verbalizations can be used to find alignment candidates. In practice, not every concept collection has verbalizations, but alignments can mitigate this: if concept A is aligned with concept B and B has the verbalization v, then we can also use v as a verbalization for A.

The concept glossary is used by **frontends** that enable the alignment workflow sketched in Figure 1. The details depend on what we want to align: aligning phrases in a document is, for example, very different from aligning concepts in an ontology. Note that harvesters and frontends are largely independent as they are only connected via the concept glossary. This facilitates reusing harvesters for new frontends or, vice versa, reusing frontends for harvesters for new different concept collections.

Building on snify The snify tool [2] supports semantic annotation with a workflow similar to the one shown in Figure 1 for semantic annotation. Both the harvester and the frontend act on LaTeX documents that use the sTeX package for semantic markup. Using the snify tool has significantly increased the efficiency of annotators, especially for novices who are not deeply familiar with the domain model (i.e. who do not know the identifiers by heart).

In this paper, we build on the snify architecture to support alignment outside of the sTeX ecosystem. Concretely, we have added a harvester for mathematical concepts from WikiData, rudimentary support for aligning formulae, and a frontend for aligning HTML documents along with a browser-based user interface.

The snify tool was implemented as a simple command line application. It steps through the phrases in a document that match verbalizations from the concept glossary and presents the user with a list of annotation candidates. To recognize inflected forms – e.g. "continuities" instead of "continuity" – snify uses an off-the-shelf stemmer. Figure 3 shows the tool in action. The user can select the correct entry from the list by entering its number, which then creates the alignment by inserting a macro into the document (here e.g. `\wdalign{Q28920044}{natural number}`). If the user does not want to create an alignment, there are many other commands to skip phrases, undo annotations, modify the selection, etc.

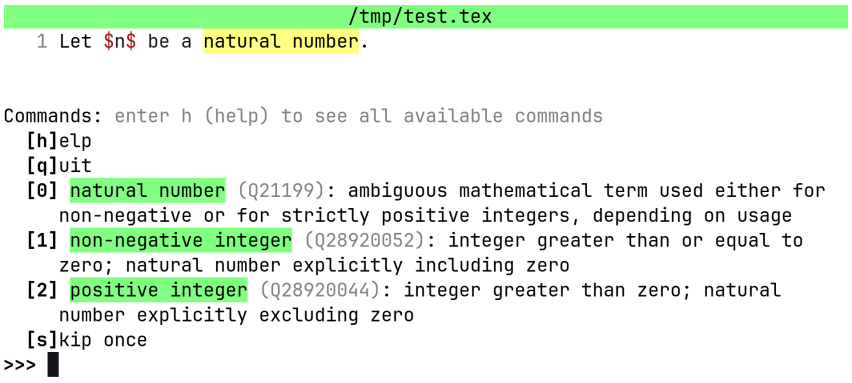

Figure 3: Screenshot of aligning LaTeX documents with WikiData concepts using snify.

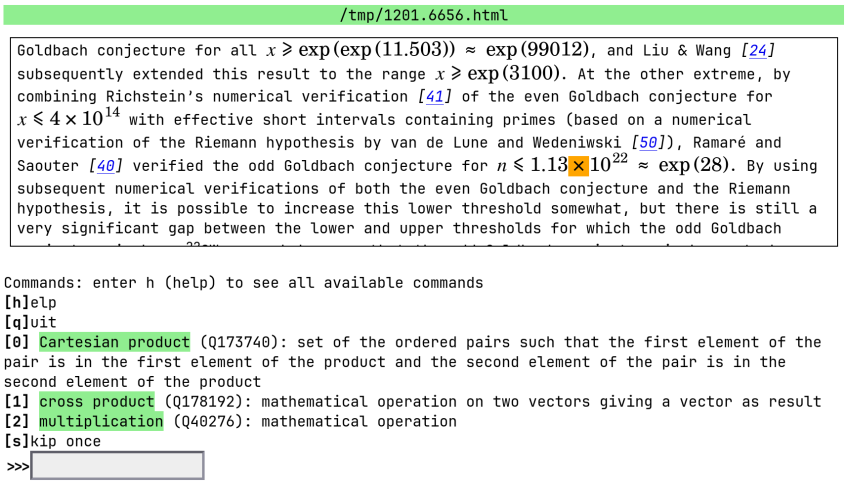

Figure 4: Screenshot of the browser-based snify interface. Here, a MathML operator is being annotated. The document excerpt is from [4].

Supporting Formulae and HTML Aside from verbalizations, WikiData also has formula notations for many concepts, including LaTeX notations like `\mathbb{N}` (in the concept glossary, we could treat them as verbalizations in another language). This allows us to use the same workflow to align formula fragments, but there are significant limitations: operators can be aligned, but their arguments are not marked, and some operators, like the invisible multiplication in "ax", are not represented by a glyph at all. In sTeX, the solution is essentially to write down a semantic representation of the formula and generate the presentation from it. A grammar-based approach to generate the semantic representation for "conventional" LaTeX formulae in collaboration with a human annotator was presented in [3]. For alignments, this approach could be considered too invasive.

To explore other formats, we have added support for HTML documents, where the created alignments are stored in HTML attributes. Reading plain HTML in the command line interface is not ideal, especially for complex MathML formulae. To improve this, we have created a browser-based interface that closely imitates the command line interface but can render HTML instead of showing the raw source (see fig. 4).

Conclusion In summary, we have described a workflow for efficient human alignment based on a concept glossary and presented a prototypical implementation based on the snify tool that explores this workflow for the alignment of text and formulae in LaTeX and HTML documents with WikiData concepts. While the formula alignment is rather limited, it was relatively easy to implement and could be improved in the future. On a higher level, this shows how relatively simple tools can enable much more efficient alignment workflows, and we hope to inspire the community to think about other tools that could help alignment authors and maintainers.

References

[1] D. Müller, T. Gauthier, C. Kaliszyk, M. Kohlhase, F. Rabe, Classification of alignments between concepts of formal mathematical systems, in: H. Geuvers, M. England, O. Hasan, F. Rabe, O. Teschke (Eds.), Intelligent Computer Mathematics, Springer International Publishing, Cham, 2017, pp. 83–98.

[2] M. Kohlhase, J. F. Schaefer, Semantic authoring in a flexiformal context – bulk annotation of rigorous documents, 2025. URL: https://kwarc.info/kohlhase/submit/cicm25-snify.pdf, submitted.

[3] L. Vrečar, J. Wells, F. Kamareddine, Towards semantic markup of mathematical documents via user interaction, in: A. Kohlhase, L. Kovacz (Eds.), Intelligent Computer Mathematics (CICM) 2024, volume 14960 of *LNAI*, Springer, 2024, pp. 223–240. doi:10.1007/978-3-031-66997-2.

[4] T. Tao, Every odd number greater than 1 is the sum of at most five primes, 2012. URL: https://ar5iv.labs.arxiv.org/html/1201.6656. arXiv:1201.6656.