Adaptability of NLP Annotation Tools for Linguistics

Changbing Yang[1], Patricia Anderson[2], Godfred Agyapong[3], Sarah Moeller[3]
[1]University of British Columbia, [2]Revitalization Technology, [3]University of Florida
cyang33@student.ubc.ca, smoeller@ufl.edu

As speech communities shrink (Krauss, 1992), linguists prioritize the documentation of these languages. A key part of documentation involves enriching texts with detailed linguistic annotations. Tools like ELAN (Auer et al. 2010) and FLEx (Rogers 2010) that are widely used for annotation tasks such as time-aligned transcription, free translation, or morpheme analysis, have two significant drawbacks: they are expensive to update and do not incorporate modern Natural Language Processing (NLP). Given that labor-intensive annotation is vital not only for linguistic documentation but also for NLP, many language annotation tools have emerged in recent years. The sheer abundance of these tools now presents an additional complexity for linguists wishing to take advantage of new annotation software.

Our work aims to provide insights that help linguists and community language workers make informed choices and strengthen connections between their efforts and helpful AI. We systematically evaluate over 100 annotation tools, focusing on their features, sustainability, and graphical interface design. We established a list of criteria broken into specific questions[1] to guide the evaluation. This abstract presents preliminary findings. A central question driving our research is whether tools designed for NLP are adaptable for endangered language documentation.

Criteria of NLP Tools Adaptable to Linguistic Research

To select the ideal annotation tool, users must navigate diverse specifications and purposes. Although the work of NLP and linguistics overlap, they differ in the exact subtasks, workflows, and priorities, particularly in ways that align with linguistic or community-based goals. Not all NLP annotation tools support the tasks or data needed by academic or community linguists. The following criteria address the challenge of identifying new tools that may supplement linguistic annotation software.

1. Cost

A tool's financial model plays a critical role in its adaptability, as academic or community users often have limited funding. On the other hand, paid tools may provide better technical support and longevity.

2. Sustainability and Longevity

A tool's long-term viability depends on active maintenance and the nature of the entity maintaining it. Proprietary tools tend to have more consistent maintenance, but some open-source projects thrive thanks to dedicated developer communities.

[1] Tools we evaluated are listed here: https://link/to/our/repo. We will make it publicly available in the final draft.

3. Portability

Given the varied needs of linguistic projects, it is unlikely that a single tool will meet all needs. Therefore, data portability ensures seamless workflows across different apps.

4. User Friendliness

When we consider the varying technical expertise involved in language documentation, usability is critical. The installation process should not require advanced technical expertise. The graphical interface should allow users who are familiar with the tool’s purpose to get started without needing detailed instructions.

5. Sensitivities

Working with endangered languages involves unique ethical considerations related to privacy, access rights, and data ownership. Software specifications should be clear where uploaded data is stored (if not on the user’s computer) and who has access to the data. These considerations are especially important when working with data collected from minority communities, where ethical and privacy standards may differ from those in commercial settings.

6. Linguistic Annotation Capabilities

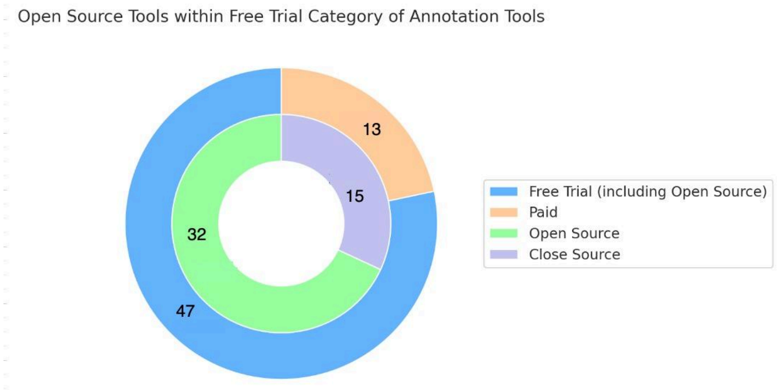
Rather than focusing on the specific tasks (e.g. named entity recognition or dependency parsing) that a tool was designed for, we assess its capacity to adapt to common linguistic tasks such as word-by-word glossing, morpheme segmentation and glossing, as well as tasks that are common in NLP and linguistic such as part-of-speech (POS) tagging and translation.

7. Active Learning

Producing annotated data can be costly in terms of time and resources. There’s a common goal in NLP and linguistics to minimize costs. One strategy that minimizes human labor and maximizes the utility of computer--annotated labels is Active Learning (AL). In the AL paradigm, the machine learning model actively selects data points from which to learn, rather than being passively trained on a fixed dataset. This prioritizes the most informative examples for human annotation, reducing the overall cost and effort. In anticipation that AL can be integrated as AI assistance to language documentation, we assess whether a tool offers active learning functionalities. This includes whether users are provided with feedback on computer predictions or can add their own pre-annotations. For instance, does the tool allow the import of annotations with confidence scores from a machine learning model or does it have functionalities to train and test models?

Preliminary Findings of NLP Tool Assessment

Given that most linguistic software tools are free and that the resources of many linguists and community members are constrained, we prioritized free tools for these preliminary findings. Out of the 100 tools initially considered, 40 were removed for being defunct or irrelevant. We focused on the remaining 60, then narrowed to the 47 tools that offer at least a free trial (Results of this narrowing process are shown in the cycle chart below). We evaluated how these 47 tools meet the criteria above.



Open Source Tools within Free Trial Category of Annotation Tools

Finding 1: No Single Tool Meets All Linguistic Needs
No one tool provides a comprehensive solution for all linguistic tasks but many tools offer complementary functionalities to linguistic software like ELAN (Auer et al. 2010) and FLEx (Rogers 2010). Our work reduces the decision space by eliminating clearly irrelevant choices. The ideal choice depends on the user's priorities—whether it is cost, ease of use, sustainability, or AI support. Recognizing which criteria are essential for one's project will enable a more focused selection and further reduce the decision space.

Finding 2: Trade-offs in Tool Selection
Each criterion introduces trade-offs with other criteria, reinforcing the lack of a one-size-fits-all solution. Researchers must weigh which trade-offs align with their priorities and projects. Below are two examples of trade-offs.

Trade-off Example 1: Cost, Sustainability and Longevity
Cost (criterion 1) and sustainability and longevity (criterion 2) are tightly intertwined. Sustainability hinges on active maintenance, which is costly. Open-source tools offer full functionality without licensing fees, making them ideal for projects with limited funding. Proprietary tools like Labelbox[2] or LightTag[3] can become expensive when large projects scale beyond the freemium version or when using advanced features like active learning. For-profit companies tend to offer consistent tool updates, ensuring long-term usability. In contrast, open-source tools depend on community involvement for maintenance, which leads to variability in update frequency. However, there are open-source tools, like INCEpTION (Klie et al., 2018) and Label Studio (Tkachenko et al., 2020-2024), which benefit from strong developer communities and see consistent improvements. At the same time, proprietary tools may risk discontinuation if a startup fails or loss of access if subscriptions lapse, which could undermine their long-term reliability.

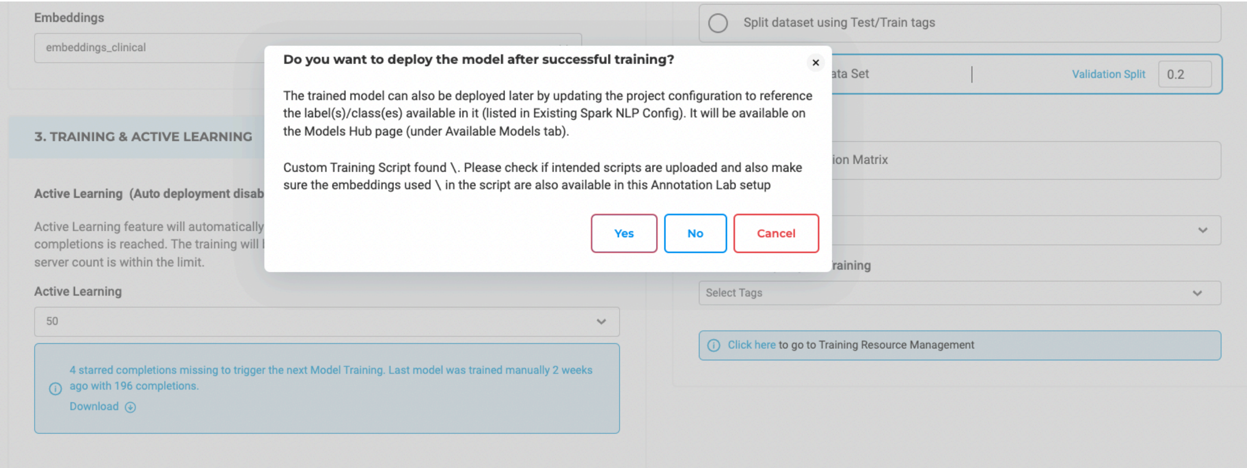
Trade-off Example 2: User-Friendliness and Customizability for Linguistic Projects
Ease of use (criterion 4) is critical for adaptability, particularly for non-technical users, but some tools prioritize advanced functionality (criterion 6) over simplicity. Among the 47 free tools, 15

[2] https://labelbox.com/
[3] https://primer.ai/

are web-based, requiring minimal installation. For example, Doccano's[4] web-based interface appeals to non-technical users but its limited flexibility makes it less suitable for handling complex linguistic tasks. In contrast, tools like Piaf[5] offer broader functionality and customization but require an involved setup, including configuring virtual environments, Docker (Merkel, 2014), and administrative access. The choice of tool will depend on the user's technical skills and their need for advanced features. Projects with limited IT resources might favor user-friendly, web-based tools, while more technically complex projects could benefit from tools that, although harder to set up, support rich and customized annotation workflows.

Finding 3: Limited Support for Specialized Linguistic Formats and Morpheme Analysis
Linguistic data often follows the formatting conventions of interlinear glossed text (IGT) (criterion 6). Only 23 of 47 free annotation tools provide support for word-level annotation, let alone subword level that might accommodate morpheme analysis. Tools such as BRAT (Stenetorp et al., 2012) and INCEpTION (Klie et al. 2018) offer strong support for these types of annotations, but many general-purpose NLP annotation tools are not equipped to handle common linguistics formats without significant customization (often limited to open-source tools).

Finding 4: Challenges and Potential of Active Learning for Low-Resource Language Annotation
Active learning (criterion 7) is available in only 17 of the free tools, with varying degrees of allowed customization that would accommodate linguistic tasks. For example, NLP Lab (John Snow Labs)[6] offers pre-annotation and model training (a screenshot example is shown below) that is designed for tasks like named entity recognition. Adapting this functionality for morpheme segmentation or glossing requires additional configuration.



Embeddings: embeddings_clinical

Do you want to deploy the model after successful training?

The trained model can also be deployed later by updating the project configuration to reference the label(s)/class(es) available in it (listed in Existing Spark NLP Config). It will be available on the Models Hub page (under Available Models tab).

Custom Training Script found \. Please check if intended scripts are uploaded and also make sure the embeddings used \ in the script are also available in this Annotation Lab setup

Active Learning (Auto deployment disab

Active Learning feature will automatically
completions is reached. The training will
server count is within the limit.

Active Learning: 50

4 starred completions missing to trigger the next Model Training. Last model was trained manually 2 weeks ago with 196 completions.

Validation Split: 0.2

Click here to go to Training Resource Management

3. TRAINING & ACTIVE LEARNING
Download
Yes
No
Cancel
○ Split dataset using Test/Train tags
ata Set
ion Matrix
raining
Select Tags

NLP software developers seem to assume that active learning is ineffective for small datasets. Many tools are optimized for very large-scale data. This highlights that, although linguists can benefit from the plethora of annotation tools designed to accommodate data-hungry AI, there is still a need to fund research software designed specifically for low-resource language work, where active learning could enhance annotation efficiency despite data scarcity.

[4] https://github.com/doccano/doccano
[5] https://github.com/etalab/piaf
[6] https://nlp.johnsnowlabs.com/docs/en/alab/quickstart

Next Steps

Moving forward, we will identify tools that are most adaptable for language documentation, focusing on transcription, translation, or morpheme analysis. Additionally, we will evaluate tools based on data portability, and whether they ensure smooth transfer of annotations to ELAN, FLEx, and other linguistic software. We aim to explore further tools that offer computer-assisted workflows, enhancing efficiency even with small datasets which are common in low-resource language projects.

References

Michael Krauss. 1992. The world’s languages in crisis. Language, 68(1):4–10.

Auer, Eric, Albert Russel, Han Sloetjes, Peter Wittenburg, Oliver Schreer, S. Masnieri, Daniel Schneider & Sebastian Tschöpel. 2010. ELAN as Flexible Annotation Framework for Sound and Image Processing Detectors. In Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner & Daniel Tapias (eds.), European Language Resources Association LREC 2010: Proceedings of the 7th International Language Resources and Evaluation, 890–893. Paris: ELRA: European Language Resources Association. http://dblp.uni-trier.de/db/conf/lrec/lrec2010.html#AuerRSWSMST10.

Rogers, Chris. 2010. Review of Fieldworks Language Explorer (FLEx) 3.0. Language Documentation & Conservation 4. 78–84.

Klie, Jan-Christoph, et al. The inception platform: Machine-assisted and knowledge-oriented interactive annotation. Proceedings of the 27th international conference on computational linguistics: System demonstrations. 2018.

Tkachenko, M., Malyuk, M., Holmanyuk, A., & Liubimov, N. (2020-2024). Label Studio: Data labeling software. Available from https://github.com/HumanSignal/label-studio.

Merkel, D. (2014). Docker: lightweight linux containers for consistent development and deployment. Linux Journal, 2014(239), 2.

Stenetorp, Pontus, et al. BRAT: a web-based tool for NLP-assisted text annotation. Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics. 2012.