Annotation Tools and Data Formats I

Hands-on with Label Studio

Jin Zhao

Brandeis University

March 9, 2025

Today's Agenda

1. Overview of annotation tools
2. Label Studio deep dive
3. Configuring annotation interfaces
4. Setting up projects
5. LLM backend integration
6. Workflow management
7. Hands-on practice

Project: Draft 1 guidelines + pilot annotations due!
Assignment: HW 2 assigned

Annotation Tool Landscape

Traditional tools:

- **brat:** Web-based, standoff format, linguistic annotation
- **MAE:** Multi-document annotation, DTD-based
- **WebAnno:** Collaborative, multi-layer
- **GATE:** Integration with NLP pipelines

Modern tools:

- **Label Studio:** Flexible, ML backends, multi-modal
- **Argilla:** RLHF focus, Hugging Face integration
- **Prodigy:** Active learning, spaCy integration
- **Doccano:** Simple, open-source

Why Label Studio?

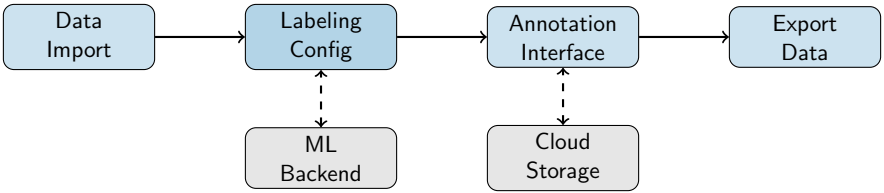
Key advantages:

1. **Flexibility:** Text, images, audio, video
2. **Customizable:** XML-based configuration
3. **ML integration:** Pre-annotation with models
4. **Collaboration:** Multi-user, role-based access
5. **Export options:** JSON, CSV, COCO, many formats
6. **Free tier:** Open-source community edition

Use cases:

- NER and text classification
- Image labeling and object detection
- Audio transcription
- Preference annotation

Label Studio Architecture



Components:

- Data import (JSON, CSV, files)
- Labeling configuration (XML template)
- Annotation interface (web UI)
- Export (multiple formats)
- Optional: ML backend, cloud storage

Installation and Setup

Installation options:

1. **pip install:**
   ```
   pip install label-studio
   label-studio start
   ```
2. **Docker:**
   ```
   docker run -p 8080:8080 heartexlabs/label-studio
   ```
3. **Cloud:** Label Studio Teams (managed)

Default access: `http://localhost:8080`

Create account, then create project

Labeling Configuration: Text Classification

XML template for sentiment classification:

```
<View>
  <Text name="text" value="$text"/>
  <Choices name="sentiment" toName="text"
           choice="single">
    <Choice value="Positive"/>
    <Choice value="Negative"/>
    <Choice value="Neutral"/>
  </Choices>
</View>
```

Components:

- `<Text>`: Display text from data
- `<Choices>`: Classification options
- `choice="single"`: One selection only

Labeling Configuration: NER

XML template for named entity recognition:

```
<View>
  <Labels name="ner" toName="text">
    <Label value="PER" background="red"/>
    <Label value="ORG" background="blue"/>
    <Label value="LOC" background="green"/>
    <Label value="DATE" background="orange"/>
  </Labels>
  <Text name="text" value="$text"/>
</View>
```

Usage:

- Click label, then highlight text
- Or highlight text, then select label
- Keyboard shortcuts available

Labeling Configuration: Relations

XML template for relation annotation:

```
<View>
  <Labels name="ner" toName="text">
    <Label value="PER"/>
    <Label value="ORG"/>
  </Labels>
  <Text name="text" value="$text"/>
  <Relations>
    <Relation value="works_for"/>
    <Relation value="founded"/>
    <Relation value="CEO_of"/>
  </Relations>
</View>
```

First annotate entities, then draw relations between them

Data Import Format

JSON format for import:

```
[
  {
    "data": {
      "text": "Apple announced a new iPhone today."
    },
    "predictions": [{
      "result": [
        {"from_name": "ner", "to_name": "text",
         "type": "labels", "value": {
           "start": 0, "end": 5, "labels": ["ORG"]
         }}
      ]
    }]
  }
]
```

ML Backend Integration

Use ML models for pre-annotation:

Options:

1. **Built-in models:** Label Studio ML backends
2. **Custom models:** Python SDK integration
3. **LLM APIs:** OpenAI, Anthropic integration
4. **Hugging Face:** Transformers models

Benefits:

- Pre-populate annotations
- Annotators correct rather than create
- Faster annotation workflow
- Active learning (prioritize uncertain examples)

Workflow Management

Project organization:

1. **Projects:** Separate by task or dataset
2. **Members:** Add annotators with roles
3. **Data manager:** Filter, sort, assign tasks
4. **Queue:** Control annotation order

Quality control:

- Multiple annotators per task
- Review mode for adjudication
- Agreement metrics dashboard
- Skip/flag problematic items

Export Options

Label Studio supports multiple export formats:

- **JSON:** Full annotation data
- **JSON-MIN:** Minimal format
- **CSV:** Tabular format for classification
- **TSV:** Tab-separated values
- **CONLL:** BIO format for sequence labeling
- **COCO:** For image annotation
- **spaCy:** Direct spaCy integration

Export via:

- Web UI (Settings $\rightarrow$ Export)
- API (programmatic access)
- Python SDK

Best Practices

Project setup:

1. Start with clear labeling config
2. Test on small sample first
3. Include instructions in interface
4. Set up keyboard shortcuts

Annotation workflow:

1. Pre-annotate when possible
2. Use consistent naming conventions
3. Regular export/backup
4. Track annotation progress

Quality control:

1. Multiple annotators for overlap
2. Regular IAA checks
3. Adjudication for disagreements

Hands-On: Setting Up Your Project

Today's exercise:

1. Install Label Studio (or use existing)
2. Create new project for your task
3. Configure labeling interface
4. Import sample data
5. Annotate a few examples
6. Export and verify format

Goal: Have your project configured for pilot annotation

Next Class: Annotation Tools II

Lecture 15 (Mar 11): Annotation Tools and Data Formats II

Topics:

- Data format conversion
- Export formats and pipelines
- Batch processing annotations
- Quality control setup
- Working with BIO format

Project: Draft 1 guidelines due!

Key Takeaways

1. **Label Studio** is flexible and supports many annotation types
2. **XML configuration** defines your annotation interface
3. **ML backends** enable pre-annotation workflows
4. **Export formats** vary – choose based on your needs
5. **Workflow management** includes roles, queues, and QC
6. **Test early** – configure and test before full annotation

Questions?

Office Hours: Wednesdays 1-3pm, Volen 109

jinzhao@brandeis.edu