📌 Guidelines used for Annotating ArnaDataset Layouts with LabelImg

📄 Version:

v1.0 – August 2025

1. Objective

This document provides a comprehensive guideline for annotating Persian document images into seven semantic layout categories using the LabelImg tool. It aims to ensure consistency, reproducibility, and quality throughout the annotation process. A notable feature of this dataset is the inclusion of XML files containing all content present in the document images. This resource can be applied to a wide range of intelligent document processing tasks.

2. Annotation Tool

Tool Used: LabelImg (https://github.com/tzutalin/labelImg)

Annotation Format: Pascal VOC (XML)

Bounding Box Type: Axis-aligned rectangles

Image Format: JPG (grayscale or color)

3. Label Classes and Definitions

Each bounding box must be assigned exactly one of the seven predefined class labels. Definitions and examples are provided below:

Label Name	Description	Visual Cues	Contains Text?
Text	Main body text including paragraphs and narrative	content Regular font size, continuous blocks	Yes ✓
Title	Title Document or section title	Head line Large font, bold, usually at top of page	Yes ✓
Figure	Photographs, illustrations, or scanned	Figures Non-text, raster regions	No ×
Logo	a graphic symbol used to identify a brand or product	Unique typography, Symbolism related to the brand's mission ,	No×
List	Bulleted or numbered lists	Items starting with •, -, or numbers or other symbols	Yes ✓
Table	Structured data arranged in rows and columns	Grid-like structure, often with borders	Yes ✓
Equation	a mathematical statement that shows the equality of two expressions	Variables (such as x,y), and mathematical operations(+, −, ×, ÷, =)	No×

4. Annotation Instructions

4.1. General Rules

- All visible layout elements must be annotated if they fall under the defined classes.
- Bounding boxes should tightly enclose the content with minimal padding.
- No overlapping boxes within the same class.
- Avoid labeling decorative elements (e.g., lines, watermarks).

4.2. Class-specific Notes

Text: Split text blocks only if they are clearly separated (e.g., multi-column layout).

Title: Mark only the most prominent heading(s), not subheadings unless visually distinct.

Figure: Crop tightly around image content. Ignore any overlaid text inside the image.

Table: Include full structure; headers + body. Subsequently, each row and each individual cell should be explicitly labeled.

List: Mark the entire list block (not individual items).

5. Adding Text Content (Post-Annotation Phase)

In a second pass, textual content from each box (if applicable) was manually extracted and inserted into the XML file under a custom field (<content>). This was done for the following classes only:

- Text
- Title
- List
- Table

Boxes of class Figure, Logo, Equation do not include textual content.

6. Annotation Review & Quality Assurance

Each annotated file was manually reviewed twice:

First during initial annotation

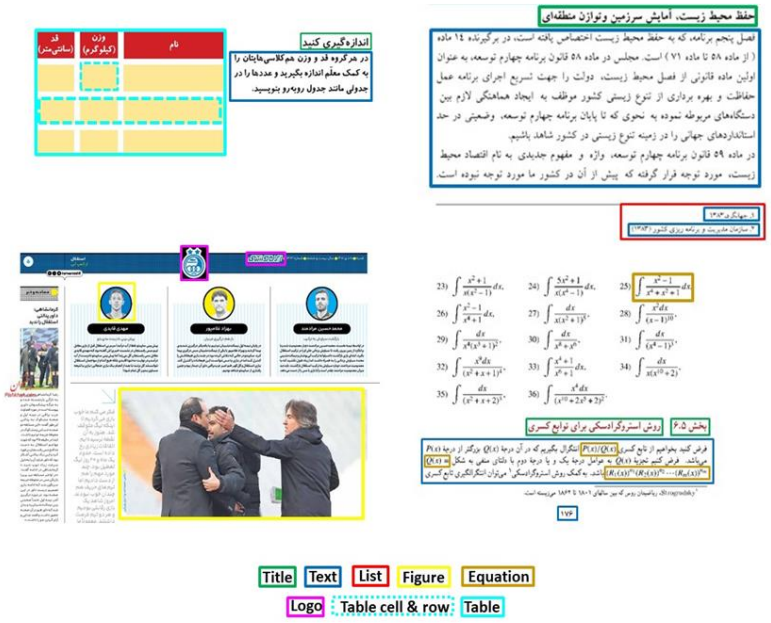
Second during text insertion phase

Overlaps, mislabels, or missing labels were corrected during this process.

Custom validation scripts were used to:

- Ensure each box has a valid label
- Validate non-overlapping structure
- Match textual content with visible region
- Empty File Detection
- Identifying XML files with no <object> elements
- Bounding Box Validity
- Ensuring coordinates were non-negative within image dimensions, and with non-zero width and height
- Same-Class Overlap Analysis
- Flagging bounding boxes of the same class with Intersection over Union (IoU) > 0.5
- The presence of <content> was checked for all text, title and table cell tags.
- Ensured that labels for table cells and table rows were present for all table labels.

7. Example Visualizations



After gathering the document images, the next step involved meticulously labeling the position and content of various components within each image. For this purpose, the image components were categorized into seven different groups: title, text, logo graphic, equation, list, and table. Each of these categories is explained as follows.

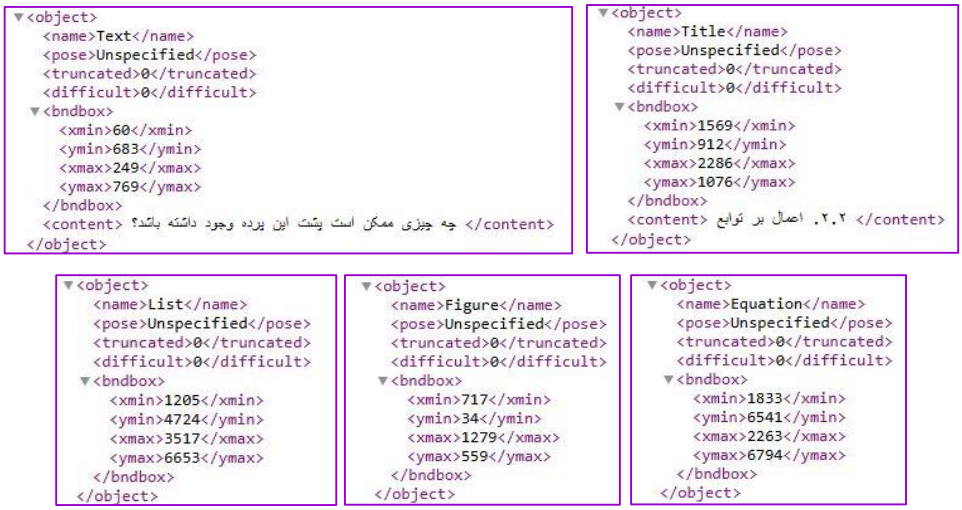
- Title: This typically captures the most visual attention and conveys a significant portion of the subject and meaning of the text. The number of characters and words is limited, and sometimes its color differs from the main text. This label includes all titles and subtitles.
- Figure: This label includes all graphical information, such as charts (excluding logos), which exist in various sizes and qualities.
- Logo: Like figures, logos are categorized under graphical information. Logos can include text, shapes, images, or a combination of these elements.
- Text: This label encompasses all textual information, excluding everything related to titles and logos. There are include all paragraphs, footnotes, appendices, captions for images, tables and charts, ordered list values, page numbers, and website addresses. List: A list refers to a representation of a countable number of ordered values. Each of these ordered values is labeled as "Text".
- Equation: This group includes equations from mathematics and physics. These equations consist not only of mathematical symbols but also include Persian and English letters and numbers.
- Table: Depending on their position and application, tables can vary in type. This variation includes differences in the number of rows, columns, and the presence or absence of a table title. In addition to labeling all tables in the database images, each table cell is labeled as "Cell" each row as "Row" and the table title (if present) is labeled as "Header". The written content of the cells and the table titles are also available in the corresponding XML file.

8. Generation the XML files

XML is a markup language used for storing and transmitting structured data. In XML, information is organized using tags in a way that is easily readable and process able. The current dataset includes an XML file alongside each image in JPG format. After gathering the sources, all files were converted to images while maintaining appropriate quality. Then, using the LabelImg software, the various parts of

each image were labeled according to the groups defined in the previous section. This software saves the specified areas as rectangles with the coordinates of the starting pixel (top-right corner) and the ending pixel (bottom-left corner), along with the corresponding label, in an XML file. In the current dataset, the starting pixel is considered the first pixel at the top-right corner, and the ending pixel is the last pixel at the bottom-left corner. Care was taken to ensure that groups with the same label do not overlap.

The XML follows standard Pascal VOC structure, with an added <content> field under each text, title and table cell <object>.



9. File Naming and Organization

The dataset comprises 5 folders named as: 001A (**A**rticles), 011C (**C**atalogs), 100S (**S**tory), 101AB (**A**cademic **B**ooks), 111TB (**T**ext **B**ooks (school)), and 1001N (**N**ewspapers).

The naming convention conveys content categories. Initially, ten categories were considered; after review and aligned with the dataset's objectives, four categories were removed, leaving the remaining categories as the final folders. The English letters for each folder correspond to the initial letter of the Latin name of the respective group, ensuring consistency and recognition.

Each image has a corresponding xml file with the same name. The XML follows standard Pascal VOC structure, with an added <content> field under each Text, Title and table cell <object>.

10. Version Control and Reproducibility

Annotation history and review steps were tracked using Git.

Any future annotations should strictly follow this guideline.

✍Author

[Saeedeh Salehi]

Document Layout Analysis Researcher

[IDPL lab/ shahid bahonar university of kerman]

Email Address

[saeedeh_salehi@elec.iust.ac.ir]