OTH Amberg-Weiden

Project Work Natural Language Processing Winter 2025

Prof. Dr. Patrick Levi

November 26, 2025

Start: 08.12.2025 12:00 (via Moodle)
Deadline: 22.12.2025 12:00 (via Moodle)

OTH Amberg-Weiden

Grading Criteria

General

- Your solution solves the task and has sufficient quality.
- Your solution is well founded and well justified. Explain your solution in the documentation.
- Do not limit yourself just to techniques from the lecture but also research other possible approaches to find the best way to solve the project. Include current knowledge in the field and the current literature
- Your solution is efficient and effective (do the right things, do the things right).
- Your solution exceeds the quality obtained by AI tools when they are asked to solve the task.
- Your solution demonstrates a deep understanding of the problem.

Code

- Code must be written in Python (except otherwise specified)
- Your code is well structured (packages, classes, methods, ...), easy to read, understandable, and there are sufficient comments in the code. Uncommented code will be down-graded.
- In addition to comments in the code, every function must contain an appropriate docstring. You can follow the NumPy docstring guide: `https://numpydoc.readthedocs.io/en/latest/format.html`. Notice, a Sphinx documentation is not required.
- As a rule of thumb: The more complex the function, the more comments are required in the code.
- Your code is efficient, understandable, and written in a way that is not error-prone.
- Wherever possible, use available Python packages. Restrictions might be specified in the project description.
- Your code must run on the computers in the GPU lab (DC 1.07).

Documentation

- Your documentation must be a PDF. The use of LaTeX is recommended, but not obligatory. However, the documentation shall be in a proper report format (see grading). Markdown files or Jupyter notebooks hardly fulfill good report criteria.
- Your documentation presents your solution. Avoid unnecessary information in the documentation.
- It must be written in a way that another AI master student, who is not an expert in the field of the project task, could follow what you did and why you did it.
- It must be well-structured and written in proper language.
- Tables and figures shall be on point, clear, and concise.
- Each step in your solution must be well justified in the documentation.
- List all your references, use a proper scientific citation standard.

OTH Amberg-Weiden

Project – Image to Text Dataset for Quantum Computing

Image to text models describe images and produce a short description of what can be seen in that image. Typically, these models are trained with datasets consisting of photographs and short textual descriptions or captions.

On schematic images, they do not work accurately, since these schematics are usually not part of their training data. If you want to specialize an image-to-text model, you need to fine-tune it. To this end, you need a dataset specific for this task.

In this project, you will assess whether compiling such a dataset is possible with reasonable effort. You will collect a small prototypical dataset for a specialized use case.

Detailed Task Description

You are required to compile a dataset consisting of images, descriptive text and some additional data as outlined below. Your dataset shall only consist of schematic images showing **quantum circuits** as they are used in quantum computing.
Main focus of your work is the development of a method for compiling such a dataset, evaluating and improving its quality as far as possible. To this end, you compile a prototypical dataset with your method.

You collect images from scientific publications on the arXiv platform (`arxiv.org`). You will work on the publications in category ”quant-ph” from recent years. Notice, not all quant-ph publications are about quantum computing.
Every exam participant is required to select an exam ID in Moodle. For every exam ID you will find a list of allowed papers in Moodle. The list is in the file named ”paper_list_<YOUR EXAM ID>.txt”. *You are strictly required to evaluate the papers from the list in Moodle corresponding to your exam ID in the order given in that list. Refusing to do so will lead to a grade 5.0 without any further evaluation of your work.*

Go through *your* list of papers in the given order starting from the first one and extract all relevant images from each paper. As soon as you have found 250 images with quantum circuits, you can neglect all further papers in the list. Use as few papers as possible, i.e. find all relevant images. Describe your information retrieval and selecton process for the images briefly in the documentation. Put the corresponding source code in a dedicated Python file. To verify and demonstrate the successful identifcation of relevant images, add a second column to your paper list, stating how many images you extracted from each paper. For the papers in your list you did not look into, leave the value blank. If you did not find an image in a paper you analyzed, set the value to zero. Attach this list as ”paper_list_counts_<exam ID>.csv” to your final submission.

Save every valid image you find in PNG format exclusively in a folder ”images_<exam ID>”. Extract the following information per image in your collection as json dictionary. Main key is your filename for the image. The corresponding item is a dictionary containing the following data:

- arxiv number of the paper the image was found in (type: string)
- page number where the image is found (type: integer)
- figure number of the image in that paper (type: integer)

- quantum gates: A list of all quantum gates appearing in the image (type: list of strings)
- quantum problem: Which quantum problem, algorithm, ... is solved or realized with that quantum gate, e.g. Shor's algorithm (type: string)
- descriptions: A list of descriptive text parts from the paper (type: list of strings)
- text positions: Indicate a beginning and an end position of the texts found in "descriptions". Store them as a tuple (beginning, end) in a list. (type: list of tuples) Describe the meaning of these positions in the documentation.

Ensure that your dataset is correct, consistent and well formatted. Improve your dataset quality as far as possible.

Assess errors and quality issues that occur in your dataset, find solutions and describe them in the documentation.

Your method must be generalizable to collect a considerably bigger dataset from all available and new papers. Therefore, your dataset must not be hand-crafted. Your methods must apply generally.

All your methods must be reproducible, i.e. when they are re-run, they must yield the same results.

Your solution must run on the DC1.07 GPU machines. The use of external computation resources or APIs (except arXiv) is not permitted.

Documentation

Your documentation shall briefly describe any issues and challenge you found during compilation of the dataset, how you solved it, and how your dataset quality improved. Please also provide reference to your source code where you implemented that solution (e.g. "see method clean_gate_name() in file cleaning_methods.py").

Your documentation shall contain all relevant methods to compile the dataset. Though, limit your documentation to 5-7 pages of pure text, 10-15 pages in total. Cite references you use following a scientific standard. Your documentation does not require thesis structure, but it must be understandable for someone who has basic knowledge in machine learning and language processing.

Based on your results, conclude on the feasibility of collecting such a dataset on a large scale.

Hint: To perform this project, you need to acquire a very basic knowledge about quantum circuits and quantum gates. You will find lots of resources on the internet to quickly read into this topic. Focus on the relevant knowledge and avoid loosing time on unnecessary details here.

The following deliverables for this project must be included in your submission:

- The dataset in json format
- A folder called "images_<exam ID>" with all your images in PNG format

- The list of papers with the number of extracted images as CSV (”paper_list_counts_<exam ID>.csv”)
- Your documentation as PDF.
- Your source code in a separate folder.