Final Report

Leonora Goble

CSPB 3112 - Professional Development

Causal Discovery API

Introduction and Background

My project was inspired by my interest in historical AI techniques as well as my introduction to Bayesian networks via the Artificial Intelligence class. I wanted to use the opportunity of this class to focus my (initially vague) research interest into a concrete project that would allow me to learn practical development skills, modern Python library usage and project management best practices, and explore the current state of the field and its relationship to the broader AI research field.

I consider this project important to my professional development as an AI researcher and AI-enabled software developer because it is related to explainable/interpretable AI, one of my main research interests. I want to be able to quickly pick up an understanding of a technical subfield and explore and evaluate whether it contains potential for new and unexpected developments, not just follow popular research trends. At the very least, I would like a precise understanding of the limitations of currently-considered "dead ends" in AI research, both to avoid recreating them and to spur alternative approaches.

Original Project Goals

- Implement a basic causal inference pipeline with example data using causal inference python packages. Document experience and reasoning for choices.
- Outcome: gain knowledge of the state of existing causal inference packages (egs. DoWhy, CausalNex, EconML) in python, including tradeoffs/benefits of packages not used for the implementation of the project.
- Create a basic API for the pipeline using `fastapi`. Verify that it works with tests. Keep it updated as the pipeline and data change.
- Outcome: gain experience with API implementation and maintenence.
- Develop a more advanced causal inference pipeline with a real world (or synthetic approximation) dataset and additional functionality.
- Outcome: gain knowledge of causal inference techniques and results, research experience, and development experience.

Methodology, Materials and Methods

My methodology was to combine learning the theory, research, and development into a continuous process. I began by looking for general introductory material (textbooks, explainers, etc.), explored the basics using available packages and code (while evaluating my experience with them), and used my new understanding to hone my research approach. I would also use my research to guide my learning and development, if I encountered an interesting idea in a paper that was beyond my current level of understanding I would

determine which knowledge I lacked, what prerequisites I needed to get there, how I could explore it in code, etc.

An important part of my methodology, facilitated by the requirements of this class, was to document my experience and the resources I acquired as I went. I also continuously documented questions and further research ideas I had as I went, so I could maintain a balance of staying focused at the task at hand while being exploratory, rather than constantly going down new rabbit holes.

Results

The main theoretical knowledge gain from this project was simply clarifying what the basic categories of causal reasoning task were and their methods, relationships to other tasks, and limitations. Before doing the project this was all muddy to me; I understood the general ideas of structure learning and inference but wasn't quite sure what the boundaries were, or quite what the significance of their results was. I also gained a better understanding of causal reasoning's relationship to data science and AI, particularly applications to reinforcement learning.

I met all of my basic project goals. I am confident that I have investigated the majority of existing causal inference/discovery packages in Python, and have a good idea of which are the most complete and actively developed, as well as a feeling for their differences. I also got some idea of the ecosystem outside of Python, including R and Java. I implemented causal discovery and inference methods with them and used tests (pytest) to verify the endpoints. The API has Pydtantic verification and typing features. I gained experience implementing an async/background job system with a database, ORM/schema, with FastAPI. The program has a clean architectural separation between the API, discovery logic, and background services. I explored various canonical datasets with my API.

Discussion / Reflection

I met all of my original project goals to an extent, but I would by no means consider myself an expert in the material nor particularly skilled at FastAPI etc. development as a result of this project. The biggest criticism I would have of my own project is that it somewhat lacks a practical identity or purpose, it does not provide much functionality beyond a API/server wrapper with some combination of libraries and streamlined functions. I did not have time to go particularly deep into any research area nor explore any "cutting edge" areas such as implementing novel algorithms described in papers. In the context of this particular class it was a good compromise of practicality and exploring my interests, but I don't think that I would necessarily do something like it again. In the future, I think that I could benefit from a cleaner separation between exploration/research and focused "product" projects. For example, if I had done a more general/directed FastAPI project first, I may have gained those skills faster rather than dealing with the particularities of another library I was learning at the same time.

Overall, I think that the experience was good for my development simply for the experience of going through a project cycle and documenting it, I developed a lot of new methods for recording and presenting my progress, self-evaluating, and exploratory research. One of the main things that I learned was both the utility and extreme difficulty of setting concrete and attainable goals in a space that I do not fully understand the

scope of. I do not feel that any of my attempts at prescribing a weekly plan adhered to what happened, but there was still utility in setting them. A couple of the (minor) goals that I set in my project update were not met because simply I did not realize that they were not applicable.

Conclusion

While there is certainly a lot to be added if I want a coherent portfolio piece, I think that I have arrived at a good stopping point for the project in terms of this class and have a good idea about next steps and potential future research ideas if I choose to pursue them or a deeper understanding of this field in the future. I think that my new awareness of causal methods in RL will be a good pathway back into that subject for me. Most importantly, I feel that I have refined my self-guided research process in a useful way.

Code

https://github.com/leonorae/rung

References or Bibliography

https://www.pywhy.org/

https://causalwizard.app/inference/categories/

https://erdogant.github.io/bnlearn/pages/html/Introduction.html

https://www.bnlearn.com/research/sachs05/

https://www.datacamp.com/tutorial/intro-to-causal-ai-using-the-dowhy-library-in-python

Books/Learning

Causal Inference for The Brave and True

What If?

Starting areas for further research

https://crl.causalai.net/ https://blog.gopenai.com/llms-for-causal-discovery-745e2cba0b59

Additional resources https://leonorae.github.io/posts/causal-inference-learning-resources/