

DS 6050: Deep Learning Course Project Plan

Heman Shakeri, PhD

School of Data Science
University of Virginia

Spring 2026

Agenda

- 1 Project Overview
- 2 Dataset Requirements
- 3 Project Timeline
- 4 Milestone I: Literature Review & Proposal
- 5 Milestone II: Architecture & Experiments
- 6 Final Project
- 7 Project Ideas
- 8 Best Practices
- 9 Resources & Support
- 10 latex
- 11 Q&A

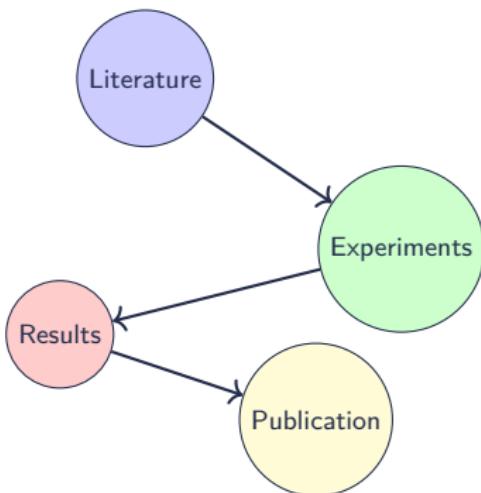
Course Project: Grand Challenge

Objective:

- Apply deep learning concepts to solve real-world problems
- Conduct rigorous research with ablation studies
- Produce publication-quality work

Team Size:

- **4 members** per team (3 allowed)



Critical: Dataset Requirements

Public Dataset Requirement

! Your dataset MUST be publicly available!

Why Public Datasets?

- ✓ Enable paper publication
- ✓ Allow result reproduction
- ✓ Foster academic transparency
- ✓ Build your portfolio

Not Acceptable:

- ✗ NDA-protected data
- ✗ Proprietary datasets
- ✗ Private company data
- ✗ Restricted access data

Recommendation

Choose well-documented benchmark datasets or create your own public dataset with proper documentation.

Recommended Dataset Sources

Computer Vision:

- ImageNet, COCO, CIFAR
- Medical: NIH Chest X-ray, ISIC
- Kaggle Vision Competitions
- Hugging Face Datasets

NLP & Text:

- GLUE, SuperGLUE
- Common Crawl
- WikiText, BookCorpus
- arXiv papers dataset

Time Series & Audio:

- UCI ML Repository
- Google Speech Commands
- LibriSpeech, AudioSet
- Financial data (Yahoo Finance)

Multimodal:

- MS-COCO Captions
- Visual Question Answering
- CLIP datasets
- Video datasets (UCF101, Kinetics)

Include dataset URL in your proposal!

Why Milestones?

Milestones are **not just deadlines**—they are structured checkpoints designed to help you succeed.

For you:

- Keep your team on track with incremental progress
- Avoid the “start everything the last week” trap
- Build your paper *iteratively*, not all at once

For me:

- An opportunity to **review your work early** and give feedback
- Catch scope issues, dataset problems, or methodology gaps *before* the final deadline
- Ensure every team is making meaningful progress

Think of each milestone as a mini peer-review: you submit, I give feedback, you improve.

Project Timeline & Milestones



Milestone	Due	Weight
I: Literature Review & Proposal	Module 6–7	5%
II: Architecture & Preliminary Results	Module 9–10	10%
III: Final Report & Presentation	After Module 12	25%

Each milestone builds on the previous one → start early!

Milestone I: Literature Review & Proposal

Due: Module 6–7 (RNN / Attention)

Deliverables:

- ① **Motivation:** Problem statement
- ② **Dataset:** Public URL required
- ③ **Literature Review:**
 - Minimum 3 papers
 - Prior methodologies
 - Identify gaps
- ④ **Proposed Method:** Initial approach
- ⑤ **Experiments:** Evaluation plan

Format Requirements:

- Use arXiv template in Overleaf
- 2–3 pages (excluding references)
- Proper citations
- Clear research questions

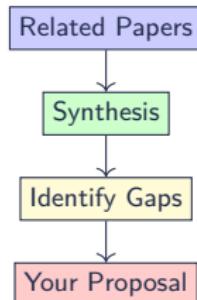
Submission:

- Format: GroupID_proposal.pdf
- Submit via Canvas

Literature Review Best Practices

What Makes a Good Literature Review?

- **Organized:** Directly related to your research question
- **Synthesized:** Summary of known vs unknown
- **Critical:** Identify controversies and gaps
- **Forward-looking:** Formulate new research questions



Resources

- Google Scholar, arXiv, Papers with Code
- Conference proceedings: NeurIPS, ICML, CVPR, ACL
- Focus on recent work (2020–2025)

Milestone II: Model Architecture & Experiments

Due: Module 9–10 (Transformers / Vision Transformers)

Key Components

- ① **Abstract:** Concise project summary
- ② **Introduction:** Problem context and motivation
- ③ **Literature Survey:** Expanded from Milestone I
- ④ **Method:** Detailed architecture description
- ⑤ **Preliminary Experiments:** Initial results
- ⑥ **Next Steps:** Planned improvements
- ⑦ **Member Contributions:** Individual responsibilities

Required Elements:

- At least one baseline model
- Training curves & metrics
- Error analysis
- Ablation study plan

Submission Format:

- Compact, up to 3 pages
- Include code repository URL
- Jupyter notebooks welcome
- GroupID_checkpoint.pdf

Experimental Design Guidelines

Baseline Selection:

- Start simple (e.g., ResNet for vision)
- Use pre-trained when applicable
- Document hyperparameters
- Ensure reproducibility

Ablation Studies:

- Vary one component at a time
- Document all changes
- Use consistent seeds
- Track with W&B or TensorBoard

Metrics to Report:

- Task-specific metrics
- Training/validation curves
- Computational efficiency
- Parameter count
- Inference time

Final Project Requirements

Due: After Module 12 (Generative Models)

Deliverables:

① Final Report (6–8 pages)

- Full methodology
- Comprehensive results
- Thorough analysis
- Future work

② Code Repository

- Clean, documented code
- README with instructions
- Requirements file

③ Presentation (5 minutes)

- Key findings
- Live demo if applicable
- Q&A session

Evaluation Criteria:

- Technical depth
- Innovation
- Experimental rigor
- Result interpretation
- Presentation quality
- Reproducibility

Peer Review:

- Review 2 other projects
- Provide constructive feedback
- Vote for “Best Insight”

Project Ideas by Module Coverage

Computer Vision (Modules 4–5, 10):

- Medical image diagnosis
- Object detection improvements
- ViT vs CNN comparison
- Domain adaptation

NLP (Modules 6–9):

- Text summarization
- Question answering
- Code generation
- Sentiment analysis

Multimodal (Module 11):

- Image captioning
- Visual question answering
- CLIP-style training
- Video understanding

Generative (Module 12):

- VAE for data generation
- GAN improvements
- Diffusion models
- Style transfer

Innovation Opportunities

- Combine architectures. Reproduce advanced SoTA
- Apply models to new domains
- Improve efficiency/speed
- Address ethical considerations

Best Practices for Success

Team Collaboration:

- Use Overleaf for LaTeX
- GitHub for version control
- Regular team meetings
- Clear task division

Experiment Tracking:

- Use W&B or TensorBoard
- Document all hyperparameters
- Save model checkpoints
- Keep experiment logs

Code Quality:

- Modular design
- Clear documentation
- Unit tests for key functions
- Requirements.txt file

Time Management:

- Start literature review early
- Allocate time for debugging
- Plan for compute resources
- Buffer for write-up

Common Pitfalls to Avoid

- Choosing overly complex problems
- Ignoring baseline comparisons
- Poor reproducibility documentation
- Last-minute dataset changes

Resources & Support

Technical Resources:

- UVA Rivanna HPC cluster
- Google Colab (free GPU)
- PyTorch documentation
- D2L textbook examples

Office Hours:

- Dedicated project hours
- TA support sessions
- Discussion on Teams
- Peer collaboration

Writing Resources:

- Overleaf templates
- arXiv paper format
- Citation management (BibTeX)
- Academic writing guides

Mentorship:

- Primary: Instructor & TAs
- Based on project topic
- Regular check-ins
- Feedback on milestones

Note: This project is your opportunity to create portfolio-worthy work!

LATEX crash course

with LATEX typesetting!

1. Bibliography Management with BibTeX

Step 1: Create references.bib file

In Overleaf: New File → references.bib

Step 2: Get BibTeX from Google Scholar

- ① Search for paper on scholar.google.com
- ② Click the quote icon ("") below paper
- ③ Select “BibTeX” at bottom
- ④ Copy entire entry
- ⑤ Paste into references.bib

Step 3: Cite in your paper

```
1 \cite{vaswani2017attention}
2 \citet{he2016deep} % with natbib
3 \citet{lecun2015deep} % with natbib
```

Example BibTeX entry:

```
1 @article{vaswani2017attention,
2   title={Attention is all you need},
3   author={Vaswani, Ashish and ...},
4   journal={NeurIPS},
5   volume={30},
6   year={2017}
7 }
```

In your main .tex file:

```
1 \documentclass{article}
2 \usepackage[natbib] % better citations
3
4 \begin{document}
5 Transformers \cite{vaswani2017attention}
6 are powerful models.
7
8 \bibliographystyle{plainnat}
9 \bibliography{references}
10 \end{document}
```

Tip: Use consistent cite keys like lastname2023keyword

2. Creating Diagrams with TikZ + LLM chatbot!

Why TikZ?

- Vector graphics (scales perfectly)
- Integrated with LaTeX math
- Version control friendly (text-based)
- Professional appearance

Workflow with LLM, e.g. Claude:

① Describe your diagram:

"Create a TikZ neural network with 3 layers: input (4 nodes), hidden (6 nodes), output (2 nodes)"

② Or sketch it on paper and upload:

"Convert this sketch to TikZ"

③ Iterate:

"Make the arrows thicker ..."

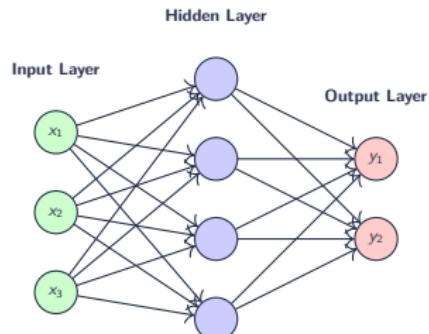
④ Copy to Overleaf and compile

Tip: Start simple, then refine.

Example: Simple Neural Network

```
1 \begin{tikzpicture}[scale =0.3, transform shape,
2   node distance=1.5cm and 2.5cm,
3   neuron/.style={circle, draw, minimum size=0.8cm, fill
4     =blue!20},
5   input/.style={circle, draw, minimum size=0.8cm, fill=
6     green!20},
7   output/.style={circle, draw, minimum size=0.8cm, fill
8     =red!20},
9   ]
10
11 % Input layer
12 \node[input] (i1) at (0,2) {$x_{-1}$};
13 \node[input] (i2) at (0,0.5) {$x_{-2}$};
14 \node[input] (i3) at (0,-1) {$x_{-3}$};
15 ...
16
17 % Hidden layer
18 \node[neuron] (h1) at (2,2) {};
19 \node[neuron] (h2) at (2,0.5) {};
20 \node[neuron] (h3) at (2,-1) {};
21
22 % Output layer
23 \node[output] (o1) at (4,2) {$y_1$};
24 \node[output] (o2) at (4,0.5) {$y_2$};
25
26 % Connections
27 \draw (i1) -- (h1);
28 \draw (i1) -- (h2);
29 \draw (i1) -- (h3);
30 \draw (i2) -- (h1);
31 \draw (i2) -- (h2);
32 \draw (i2) -- (h3);
33 \draw (i3) -- (h1);
34 \draw (i3) -- (h2);
35 \draw (i3) -- (h3);
36 \draw (h1) -- (o1);
37 \draw (h1) -- (o2);
38 \draw (h2) -- (o1);
39 \draw (h2) -- (o2);
40 \draw (h3) -- (o1);
41 \draw (h3) -- (o2);
```

Result:



3. Labels, References & Appendix

Equations, Figures, Tables

```
1 % Equation with label
2 \begin{equation} \label{eq:loss}
3 L = \frac{1}{n} \sum_{i=1}^n \ell(y_i, \hat{y}_i)
4 \end{equation}
5
6 % Reference it
7 As shown in Equation~\ref{eq:loss}, ...
8
9 % Figure with label
10 \begin{figure}
11   \includegraphics[width=0.5\linewidth]{fig.png}
12   \caption{Model architecture.}
13   \label{fig:architecture}
14 \end{figure}
15
16 See Figure~\ref{fig:architecture} for details.
17
18 % Table with label
19 \begin{table}
20   \caption{Results on CIFAR-10.}
21   \label{tab:results}
22   \centering
23   \begin{tabular}{lcc}
24     ... table content ...
25   \end{tabular}
26 \end{table}
27
28 Table~\ref{tab:results} shows ...
```

Key Principles:

- Always use `\label{...}` after `\caption`
- Use descriptive labels: `eq:loss`, `fig:architecture`, `tab:results`
- Use `\sim` for non-breaking space: `Figure~\ref{...}`
- Compile twice for references to update!

Creating an Appendix:

```
1 \section{Conclusion}
2 Our method achieves state-of-the-art...
3
4 % Start appendix
5 \appendix
6
7 \section{Additional Experiments}
8 \label{app:experiments}
9 We provide additional results...
10
11 \section{Proof of Theorem 1}
12 \label{app:proof}
13 The proof follows from...
14
15 % Reference appendix in main text:
16 % \SeeAppendix{\ref{app:experimental}}
```

Bonus: Quick Reference Commands

Math Environments:

```
1 % Inline math
2 The loss  $L = \sum_i \ell_i$  is minimized.
3
4 % Display math (unnumbered)
5 [
6   \nabla_\theta L = \frac{\partial L}{\partial \theta} \\
7   \]
8
9 % Numbered equation
10 \begin{equation}
11   E = mc^2
12 \end{equation}
13
14 % Aligned equations
15 \begin{align}
16   x &= \sigma(Wx + b) \\
17   y &= \text{softmax}(Ux)
18 \end{align}
19
20 % Split long equation (single number)
21 \begin{equation}
22 \begin{split}
23   L &= \frac{1}{n} \sum_{i=1}^n \big[ \ell(y_i, \\
24   &\quad \hat{y}_i) \\
25   &\quad + \lambda \|\theta\|^2 \big]
26 \end{split}
27 \end{equation}
```

Common Packages for ML Papers:

```
1 \usepackage{amsmath}          % Math
2 \usepackage{amssymb}          % Math symbols
3 \usepackage{algorithm}        % Algorithms
4 \usepackage{algorithmic}      % Algorithm
5   \fbox{formatting}
6 \usepackage{booktabs}         % Better tables
7 \usepackage{graphicx}         % Figures
8 \usepackage{subcaption}       % Subfigures
9 \usepackage{hyperref}         % Clickable refs
10 \usepackage{cleveref}        % Smart refs
11 \usepackage{natbib}          % Better citations
12 \usepackage{tikz}             % Diagrams
```

Using cleveref (smart references):

```
1 \usepackage{cleveref}
2
3 % Automatically adds ``Figure'', ``Table'', etc.
4 \cref{fig:architecture}    % produces ``Figure 1''
5 \cref{eq:loss}              % produces ``Equation
6 \cref{tab:results}          % produces ``Table 1''
                           (capitalized)
```

We use this template!

Questions & Discussion

Next Steps:

- ① Form teams (3–4 members)
- ② Explore dataset options
- ③ Start literature review

Register your team in this [Google Sheet!](#)

Key Takeaways

Public Dataset

! No NDA data!

Follow Timeline

3 milestones

Research Quality

Ablations required

Reproducibility

Clean code

Good luck with your projects!