Text2Arch: A Dataset For Generating Scientific Architecture Diagrams From Natural Language Descriptions

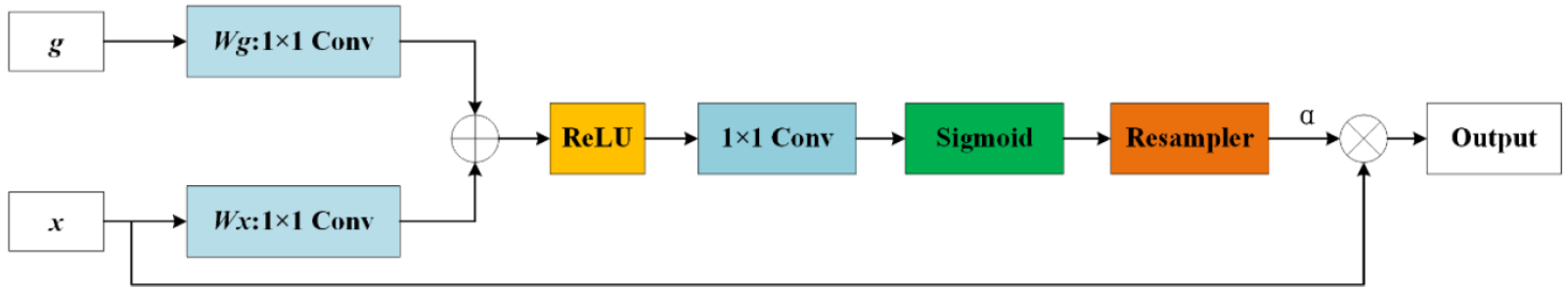
Shivank Garg
IIT Roorkee, India
shivank_g@mfs.iitr.ac.in

Sankalp Mittal
Google, India
sankalpmittal123@gmail.com

Manish Gupta
Microsoft, India
gmanish@microsoft.com

ICLR 2026, Rio de Janeiro, Brazil

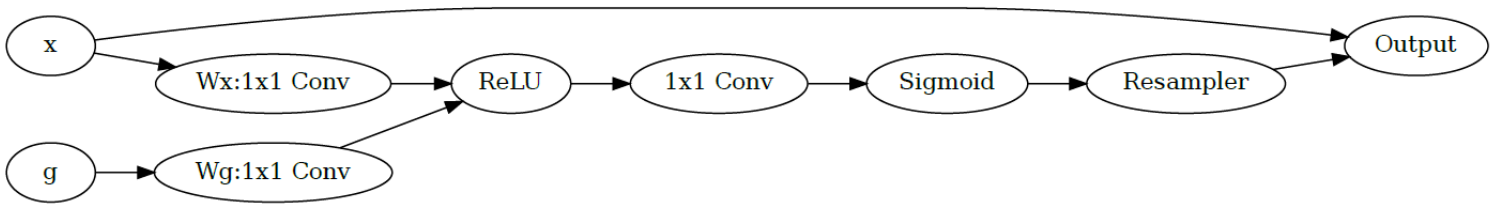
Text2Arch



(a) **Original figure**: Fig. 5 from https://arxiv.org/pdf/2110.01014v1

(b) **Figure description**: The image is an architecture diagram depicting an attention gate mechanism. The diagram starts with two inputs: g (decoding matrix) and x (encoding matrix). Both inputs undergo a 1×1 convolution operation separately, denoted as Wg:1×1 Conv and Wx:1×1 Conv respectively. The outputs of these convolutions are then added together at corresponding points. This sum is then passed through a ReLU activation function, followed by another 1×1 convolution operation. The resulting output is then processed through a Sigmoid function to generate an attention coefficient, denoted as α. This attention coefficient is used to resample the input encoding matrix x through a Resampler module. Finally, the resampled output is multiplied by the attention coefficient α to produce the final output of the attention gate.

(c) **DOT code**: digraph {
0 [label="Sigmoid"]; 1 [label="Output"]; 2 [label="1x1 Conv"]; 3 [label="Resampler"]; 4 [label="Wx:1x1 Conv"]; 5 [label="g"]; 6 [label="ReLU"]; 7 [label="x"]; 8 [label="Wg:1x1 Conv"];
5 → 8; 7 → 4; 8 → 6; 4 → 6; 6 → 2; 2 → 0; 0 → 3; 3 → 1; 7 → 1;
}



(d) **Generated figure using TEXT2ARCH**.

Why Convert Text → Architecture Diagrams?

- Scientific systems are often described only in text, which leads to:
  - Hard to understand complex pipelines
  - Ambiguous or incomplete visual representation
  - Manual diagram creation is time-consuming
- Goal
  - Automatically generate accurate architecture diagrams from natural language descriptions
- Potential Applications
  - Scientific papers
  - Software architecture documentation
  - AI system design
  - Education and learning material

Text2Arch Problem Definition

- Input: Natural language description of a system architecture
- Output: A structured architecture diagram
- Our approach
  - Text description
  - Generate DOT graph code
  - Render diagram using Graphviz
- Advantages of intermediate DOT code
  - Structured
  - Editable
  - Deterministic rendering
  - Easier evaluation

Challenges for Text2Arch

- Unlike natural image generation
  - Strict semantic alignment required
  - Precise structure of nodes and edges
  - Correct labels and flow direction
  - Long technical descriptions
- Existing text-to-image models fail because:
  - limited input context
  - poor text rendering
  - weak structural reasoning

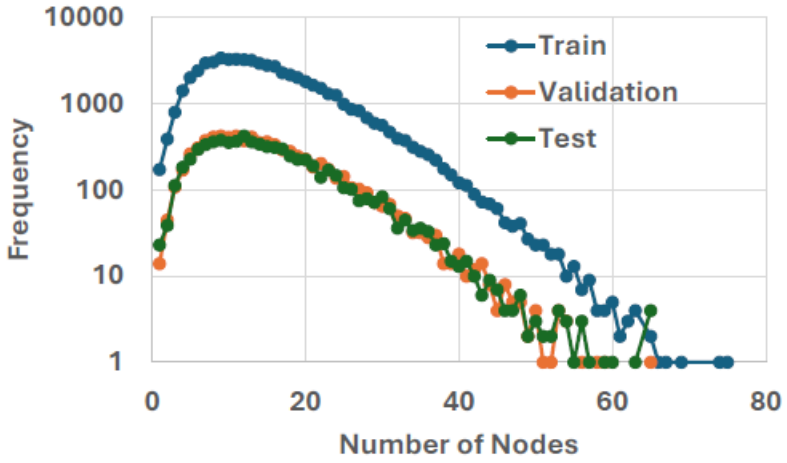
Key Contributions

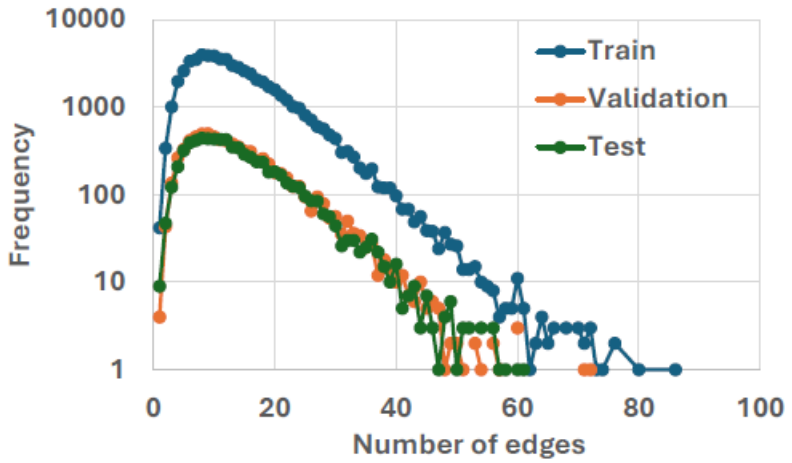
- Introduce Text2Arch task: Generate architecture diagrams from text.
- Release Text2Arch dataset
  - 75K samples
  - text description
  - architecture image
  - DOT code
- Fine-tune open-source LLMs: Llama-3, Qwen-2, DeepSeek
- Propose graph-based evaluation metrics to measure structural correctness.

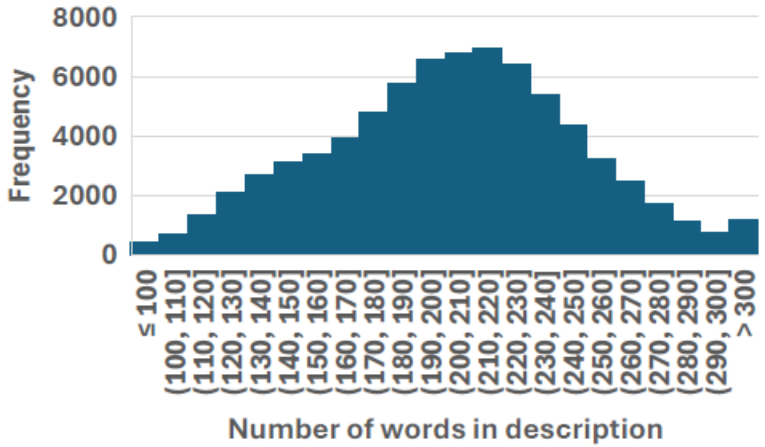
Text2Arch Dataset

- Total samples: 75,127
  - Train: 60,519
  - Validation: 7,565
  - Test: 7,043
- Manual annotation set: 99 samples.
- Each sample contains
  - Architecture diagram image
  - Clean textual description
  - DOT graph representation

- Average statistics:
  - 15.2 nodes; 13.9 edges
  - ~203 words per description
  - Around 54.3% samples are easy, 32.4% are medium and the rest are of hard complexity.

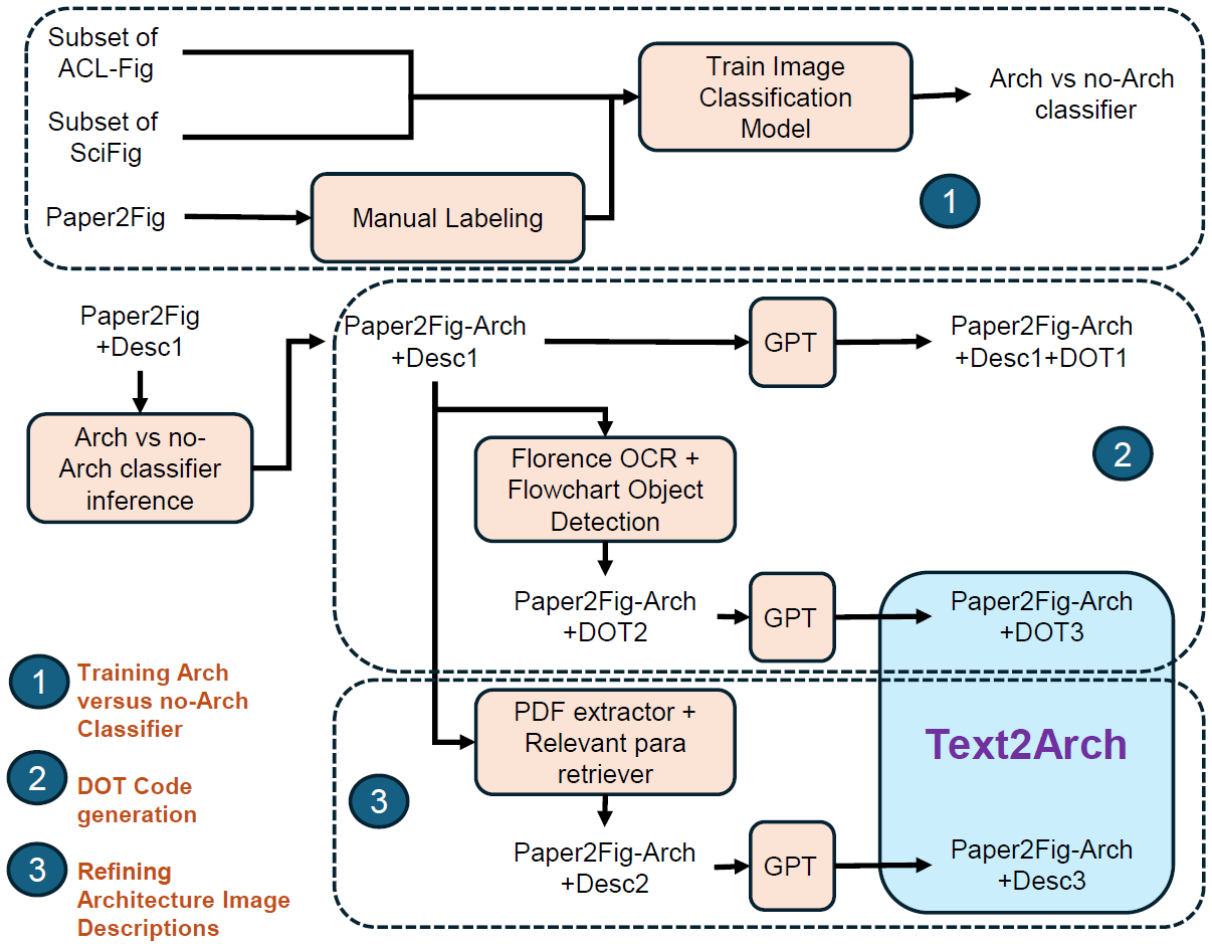






Dataset Curation

- Architecture Diagram Detection
  - Train classifier: Arch vs Non-Arch diagrams
  - Models tested: CLIP, ViT, BEiT, ResNet
  - Best model: CLIP (83.45%)
- DOT Code Generation to obtain structured graph representation
  - DOT1: Generated by GPT-4o
  - DOT2: Extract nodes and edges using OCR (Florence2) and Object detection
  - DOT3: GPT refinement of DOT2
  - Result: significantly higher graph accuracy
- Description Refinement
  - Original dataset contained noisy text.
  - We improve descriptions via:
    - Extract relevant paragraphs from paper
    - TF-IDF similarity filtering
    - GPT refinement
  - Final output: Desc3
  - Human evaluation: Desc3 preferred >90% of the time.



Models evaluated

- Baselines
  - DiagramAgent
  - GPT-4o zero-shot
- Few-shot ICL
  - Llama-3-8B
  - Qwen-2-7B
  - DeepSeek-7B
- Fine-tuned models
- Same models trained on Text2Arch dataset.

Evaluation Metrics

- Text Similarity Metrics
  - ROUGE-L
  - CodeBLEU
  - Edit Distance
  - chrF
- Graph Structure Metrics
  - Node Precision / Recall / F1
  - Edge Precision / Recall
  - Jaccard similarity
  - Node PR-AUC
- These measure semantic + structural fidelity.

Results

		Text Metrics				Graph Metrics								
		ROUGE-L	Code BLEU	Edit Distance	chrF	Node Prec	Node Recall	Node F1	Node PR-AUC	Edge Prec	Edge Recall	Edge F1	Edge PR-AUC	Jaccard Sim.
	DiagramAgent	42.2	31.0	680	48.2	60.7	56.5	55.1	20.9	31.7	22.6	24.8	18.0	17.8
	GPT	30.8	17.7	730	42.9	**71.6**	56.5	60.7	**27.4**	**56.3**	**39.5**	**44.6**	**31.6**	**36.1**
Few-shot ICL	Llama-3-8B	34.9	21.5	709	41.0	69.6	56.8	59.7	24.2	41.5	29.0	32.5	22.0	24.6
Few-shot ICL	Qwen2-7B	27.4	19.4	811	30.4	64.9	48.1	52.0	19.8	32.8	21.2	24.3	15.9	17.2
Few-shot ICL	DeepSeek-7B	30.4	21.4	1079	31.4	54.1	41.3	43.5	17.4	25.0	13.6	16.6	13.0	11.5
Fine-tuned	Llama-3-8B	28.2	27.8	956	39.8	27.8	45.3	31.9	7.0	22.9	10.2	13.2	9.1	8.4
Fine-tuned	Qwen2-7B	35.0	30.7	975	45.3	33.3	49.8	36.8	8.1	21.8	10.8	13.7	8.8	8.8
Fine-tuned	DeepSeek-7B	**46.8**	**34.5**	**608**	**55.7**	66.2	**69.6**	**65.7**	21.5	46.4	34.2	38.0	23.7	28.6

test set

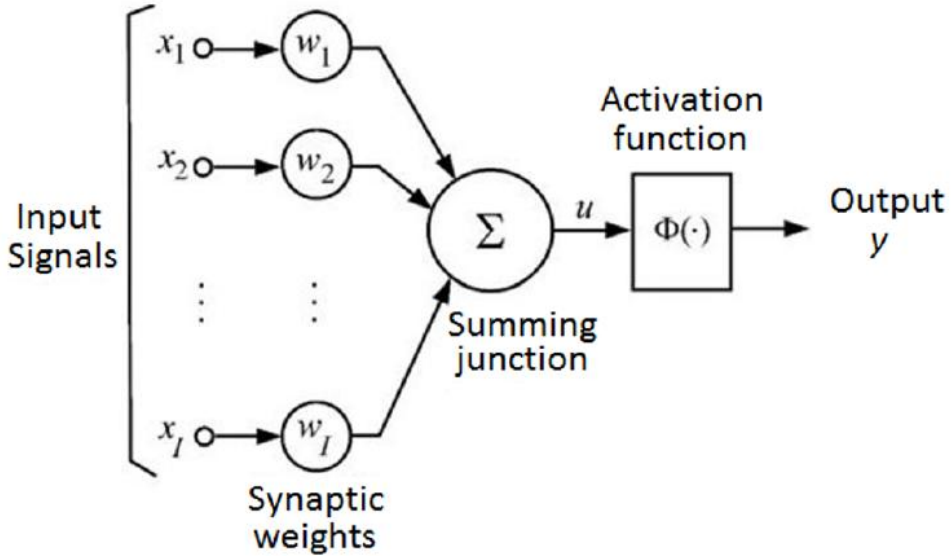
		Text Metrics				Graph Metrics								
		ROUGE-L	Code BLEU	Edit Distance	chrF	Node Prec	Node Recall	Node F1	Node PR-AUC	Edge Prec	Edge Recall	Edge F1	Edge PR-AUC	Jaccard Sim.
	DiagramAgent	49.1	40.9	959	55.3	55.0	59.8	54.3	12.4	32.0	22.8	25.3	16.9	17.8
	GPT	28.2	16.3	790	43.4	**69.7**	60.1	63.0	**28.0**	55.9	40.4	46.2	30.9	37.8
Few-shot ICL	Llama-3-8B	37.3	23.1	474	43.6	62.4	52.9	54.1	17.0	42.4	28.5	32.5	17.4	25.7
Few-shot ICL	Qwen2-7B	30.1	22.0	562	32.0	54.9	50.7	48.9	13.2	37.1	21.7	25.5	14.9	18.1
Few-shot ICL	DeepSeek-7B	36.9	28.6	872	35.8	52.1	42.2	42.9	13.8	27.5	17.5	20.4	13.9	15.4
Fine-tuned	Llama-3-8B	30.9	46.0	1024	44.0	21.3	46.4	27.5	4.9	27.7	11.9	15.2	14.3	10.1
Fine-tuned	Qwen2-7B	40.9	46.3	891	53.1	35.3	57.1	40.2	9.4	19.4	11.5	14.0	11.8	9.3
Fine-tuned	DeepSeek-7B	**55.2**	**49.3**	**407**	**66.6**	66.1	**78.1**	**69.4**	27.4	**59.4**	**44.6**	**49.1**	**35.1**	**39.8**

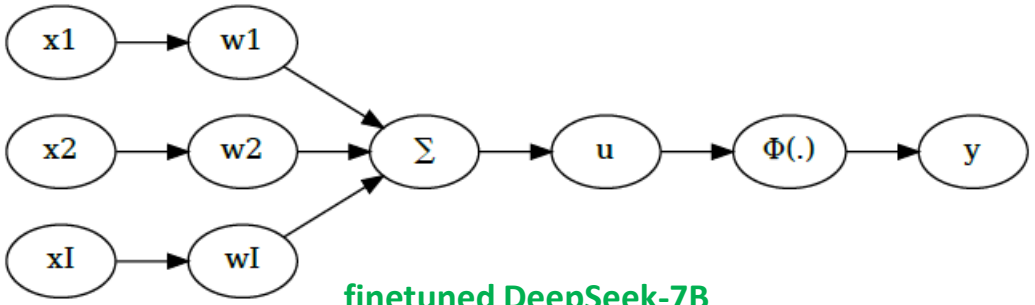
manual annotation set

- Fine-tuned models outperform few-shot models and GPT-4o.
- Best model: Fine-tuned DeepSeek-7B
- Close to GPT-4o perf while being open and lightweight.

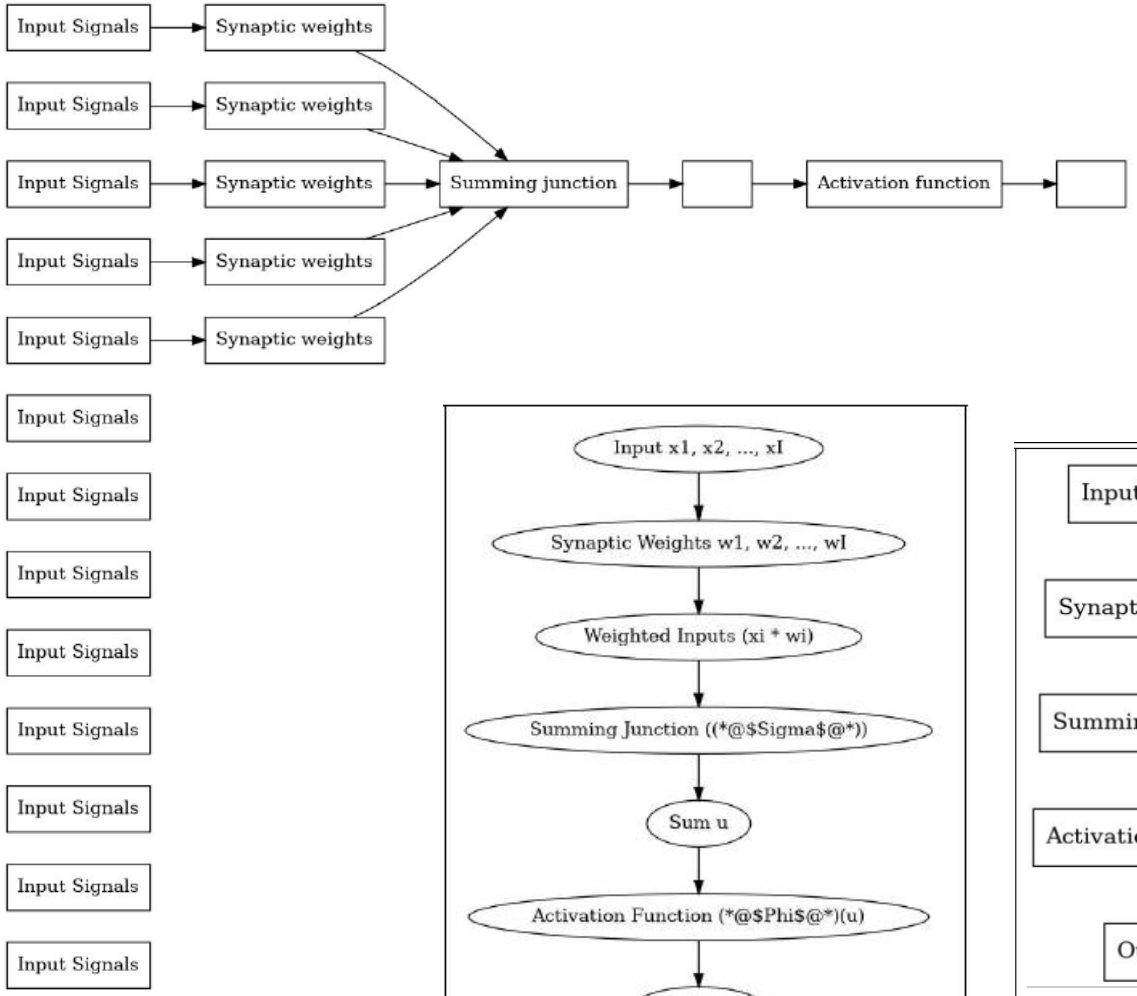
An Example of Text2Arch Results

(a) **Original figure** (Fig. 3 from https://arxiv.org/pdf/1701.07543v1)



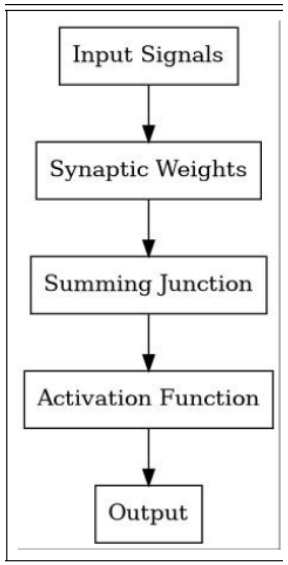


finetuned DeepSeek-7B



DiagramAgent

GPT



fewShot DeepSeek

Takeaways

- Introduced Text2Arch, the first large dataset for architecture diagram generation.
- Demonstrated LLM-based structured diagram synthesis.
- Fine-tuned DeepSeek-7B achieves best performance.
- Graph-based metrics enable rigorous structural evaluation.
- Dataset, models, and code are publicly released.
- Future Work
  - Multimodal training (text + image)
  - Direct diagram generation without intermediate code
  - Interactive diagram editing
  - Application to software engineering tools

Thank you!

- Code: https://github.com/shivank21/text2arch
- Models: https://huggingface.co/shivank21/text2arch-deepseek/
- Data: https://huggingface.co/datasets/shivank21/text2archdata
- Correspondence: gmanish@microsoft.com