

Reading List

1. LLMs Basics:

- *Attention Is All You Need*. Vaswani, et al. <https://arxiv.org/abs/1706.03762>
- *OpenAI o1 System Card*. OpenAI. https://cdn.openai.com/o1-system-card-20241205.pdf?utm_source=chatgpt.com.

2. Multimodal LLMs

- *Learning Transferable Visual Models From Natural Language Supervision*. Radford, et al. <https://arxiv.org/abs/2103.00020>.
- *Llava: Visual Instruction Tuning*. Liu, et al. <https://arxiv.org/abs/2304.08485>.

3. Software Engineering Benchmarks for LLMs

- *SWE-bench: Can Language Models Resolve Real-World GitHub Issues?* Jimenez, et al. <https://arxiv.org/abs/2310.06770>.
- *SWE-bench Multimodal: Do AI Systems Generalize to Visual Software Domains?* Yang, et al. <https://arxiv.org/abs/2410.03859>.

4. Multimodal LLMs for Front-End Development

- *Design2Code: Benchmarking Multimodal Code Generation for Automated Front-End Engineering*. Si, et al. <https://arxiv.org/abs/2403.03163>.
- *Sketch2Code: Evaluating Vision-Language Models for Interactive Web Design Prototyping*. Li, et al. <https://arxiv.org/abs/2410.16232>.

5. LLMs as Programming Agents

- *OpenHands: An Open Platform for AI Software Developers as Generalist Agents*. Wang, et al. <https://arxiv.org/abs/2407.16741>.
- *ChatDev: Communicative Agents for Software Development*. Qian, et al. <https://arxiv.org/abs/2307.07924>.

6. LLMs Augmented with Tools

- *Augmented Language Models: a Survey*. Mialon, et al. <https://arxiv.org/abs/2302.07842>.
- *ToolPlanner: A Tool Augmented LLM for Multi Granularity Instructions with Path Planning and Feedback*. Wu et al. <https://arxiv.org/abs/2409.14826>.

7. LLMs for Testing

- *Automated Unit Test Improvement using Large Language Models at Meta*. Alshahwan and et al. <https://arxiv.org/abs/2402.09171>.
- *Teaching Large Language Models to Self-Debug*. Chen, et al. <https://arxiv.org/abs/2304.05128>.

8. Reasoning with LLMs

- *Chain-of-Thought Prompting Elicits Reasoning in Large Language Models*. Wei, et al. <https://arxiv.org/abs/2201.11903>.
- *Chain of Code: Reasoning with a Language Model-Augmented Code Emulator*. Li, et al. <https://arxiv.org/abs/2312.04474>.

9. LLMs for Theorem Proving

- *Generative Language Modeling for Automated Theorem Proving*. Polu and Sutskever. <https://arxiv.org/abs/2009.03393>.
- *Formal Theorem Proving by Rewarding LLMs to Decompose Proofs Hierarchically*. Dong, et al. <https://arxiv.org/abs/2411.01829>.

10. Security and Trustworthiness of LLMs

- *Identifying and Mitigating the Security Risks of Generative AI*. Barrett, et al. <https://arxiv.org/abs/2308.14840>.
- *TRUSTLLM: Trustworthiness in Large Language Models*. Huang et al. <https://arxiv.org/abs/2401.05561>.