**Syllabus for LLM, RAG, AI-Agent principle and development**

Theory 40%. Practice 45%. Project 15% with Evalution(Quiz 20%, Report 30%, Project 50%).

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | 주제 | 소주제 | ID | dm | 학습 및 실습 자료 | 레퍼런스 및 설명 | 비고 |
| 1 | 목적주제설명  개발환경준비 | Boot | D1 | 6 | [LLM-RAG-Agent-Tutorial](https://github.com/mac999/LLM-RAG-Agent-Tutorial) | 각자 소개. 목적 및 주제설명  프로젝트 및 과제 설명. 프로젝트 진행 방식 설명.  향후 절차. 개발환경설명. | 작업환경 확인 준비 |
|  | 트랜스포머 | 인코더 | T1 | 6 | [트랜스포머 인코더 핵심 코드 구현을 통한 동작 메커니즘 이해](https://daddynkidsmakers.blogspot.com/2023/12/blog-post.html) | 개념 소개(PPT)  <https://pytorch.org/tutorials> | 이론 및 실습 |
|  |  | 디코더 | T2 | 3 | [트랜스포머 디코더 핵심 코드 구현을 통한 동작 메커니즘 이해](https://daddynkidsmakers.blogspot.com/2024/06/nlp.html) | <https://github.com/pytorch/pytorch>  /blob/main/torch/nn/modules/transformer.py |  |
|  |  | LLAMA | T3 | 3 | [오픈소스 기반 LLAMA 논문 분석 및 기술 요약](https://daddynkidsmakers.blogspot.com/2024/02/llama-2.html) | <https://github.com/rania-hossam/LLAMA_FROM_SCRATCH_PYTORCH/tree/master/pytorch-llama-main> |  |
|  |  | 시계열예측 | T4 | 3 | [시계열 데이터 예측을 위한 트랜스포머 모델 개발](https://daddynkidsmakers.blogspot.com/2024/04/blog-post.html) |  |  |
|  |  | 퀴즈 및 과제 | Q | 3 |  | 과제 풀이 제출방법 |  |
| 2 | 토큰 임베딩 | 토큰 임베딩 | E1 | 3 | [NLP 토큰, 임베딩 모델 파인튜닝](https://daddynkidsmakers.blogspot.com/2024/06/nlp.html) |  |  |
|  |  | 허깅페이스 | E2 | 3 | [허깅페이스 라이브러리 사용법 총정리](https://daddynkidsmakers.blogspot.com/2024/03/blog-post_21.html) | 예. 번역, 감정, 요약 등 |  |
|  | 생성AI 프롬프트 | 스테이블디퓨전 이론 | P1 | 6 | [멀티모달 생성AI 모델, Stable Diffusion 아키텍처 분석과 동작 원리](https://daddynkidsmakers.blogspot.com/2024/02/stable-diffusion.html) | 개념 소개(PPT)  [멀티모달 생성AI 모델, Stable Diffusion 아키텍처](https://daddynkidsmakers.blogspot.com/2024/02/ai-stable-diffusion.html) 코드 설명 |  |
|  |  | 이론 | P2 | 6 | [생성AI 프롬프트 엔지니어링 사례 및 실습](https://github.com/mac999/LLM-RAG-Agent-Tutorial/blob/main/2-2.genai-prompt/gen-ai-prompt-engineering.pdf) | 도구 소개(word). KlingAI 등 생성AI 도구 설명 |  |
|  |  |  |  |  |  |  |  |
|  |  | 사례 | P3 | 4 | [프롬프트 기반 그래픽스 모델 생성](https://medium.com/@laputa99999/creating-3d-data-visualizations-with-p5-python-7933d5b3a59f) | <https://github.com/mac999/gen_ai_gpt> OpenAI API key 사용  <https://github.com/mac999/llm-media-art-demo> |  |
|  |  | 실습 | P4 | 1 | ChatGPT 기반 이미지 생성 프롬프트 실습 |  |  |
|  |  | 실습과제 | P5 | 8 | [Kling AI 등 생성AI 프롬프트 실습](https://github.com/mac999/LLM-RAG-Agent-Tutorial/tree/main/2-2.genai-prompt/prompt-practice) | 영상, 음악, 음성 등 프롬프트 엔지니어링. 미디어 작품 개발  개별 작품 소개. 영상편집 시 Davinci Resolve 등 사용. |  |
|  | 트랜스포머 심화 | 코드분석 | T6 | 6 | [개발 예시 - transformer](https://github.com/mac999/LLM-RAG-Agent-Tutorial/tree/main/2-3.deep-tranformer) | [상세 코드 - transformer](https://github.com/mac999/transformer) |  |
|  |  | 딥시크 분석 | T7 | 5 | [딥시크(deep seek) 오픈소스 코드 및 구조 분석](https://daddynkidsmakers.blogspot.com/2025/03/deep-seek.html) | GITHUB 코드 분석 실습 |  |
|  | NLP 처리 | 마이닝 실습 | N1 | 3 | [파이썬 기반 PDF 보고서 생성 및 텍스트 마이닝](https://daddynkidsmakers.blogspot.com/2023/04/pdf.html) |  |  |
|  |  | 퀴즈 및 과제 | Q | 3 |  |  |  |
| 3 | NLP 처리 | 마이닝 실습 | N2 | 3 | 교안. NLP 부분 학습. 정규식 실습. | [자연어처리 NLP 오픈소스 기반 도구를 활용한 텍스트 데이터 마이닝 방법](https://daddynkidsmakers.blogspot.com/2023/04/blog-post.html) |  |
|  |  | 번역 | N3 | 3 | [트랜스포머 생성AI로 다국어 번역기, 문장 분류, 이미지 설명 텍스트 생성 프로그램 개발](https://daddynkidsmakers.blogspot.com/2023/10/ai.html) | [파이토치 트랜스포머 모델 사용자화를 통한 간단한 자연어 번역기 개발](https://daddynkidsmakers.blogspot.com/2024/02/blog-post_6.html) |  |
|  | LLM | 소개 | L1 | 3 | [LLM 의 통계적 패턴 예측성과 한계에 대한 연구](https://daddynkidsmakers.blogspot.com/2024/06/llm_16.html) | 논문 리뷰. 현장 과제. 소논문 쓰기. |  |
|  | 파인튜닝 | 파인튜닝 | F1 | 6 | [도메인 모델 성능개선을 위한 Lora, LLAMA3 기반 LLM 개발](https://daddynkidsmakers.blogspot.com/2024/06/lora-llama3.html) | 개념 소개(PPT) |  |
|  |  | 데이터개발 | F2 | 3 | [LLM 학습 데이터 개발 및 준비 방법](https://daddynkidsmakers.blogspot.com/2024/07/llm.html) |  |  |
|  |  |  |  |  |  |  |  |
|  | 파인튜닝 | 데이터개발 | F3 | 3 | [LLM 기반 BIM 지식 모델 학습 데이터셋 자동 생성 및 LLM 파인튜닝](https://daddynkidsmakers.blogspot.com/2025/01/llm-bim-llm.html) |  |  |
|  |  | 공학용데이터개발 | F4 | 3 | [파인튜닝 예시 - 컴퓨터그래픽스 모델 분류 model](https://huggingface.co/mac999/earthwork-net-model) |  |  |
|  |  | 실습 | F5 | 9 | [Gemma 기반 파인튜닝 코드 실습](https://github.com/mac999/LLM-RAG-Agent-Tutorial/tree/main/3-2.finetuning) | https://huggingface.co/google/gemma-2b-it |  |
|  |  | 성능지표 | L2 | 1 | 교안 NLP 성능지표 | [LLM Evaluation metrics explained. ROUGE score, BLEU, Perplexity, MRR](https://medium.com/data-science-in-your-pocket/llm-evaluation-metrics-explained-af14f26536d2) |  |
|  | 멀티모달 | CLIP | M1 | 3 | [OpenAI의 CLIP모델 이해, 코드 분석, 개발, 사용](https://daddynkidsmakers.blogspot.com/2024/02/clip.html) |  |  |
|  |  | CLIP 실습 | M2 | 6 | CLIP 코드 설치 및 실습 | GITHUB, HUGGINGFACE 사용 |  |
|  |  | LLaVA | M3 | 3 | [오픈소스 LLaVA 기반 ChatGPT 4.0 유사한 멀티모달 생성AI 개발](https://daddynkidsmakers.blogspot.com/2024/02/llava-ai.html) |  |  |
|  |  | 퀴즈 및 과제 | Q | 2 |  |  |  |
| 4 | LangChain | RAG | R1 | 6 | [LLM 모델 통합과 다양한 데이터소스를 지원하는 LangChain 아키텍처 및 동작 메커니즘 분석](https://daddynkidsmakers.blogspot.com/2024/04/blog-post_21.html) | 개념 소개(PPT). 이론(교안). [LangChain 작동 메커니즘 분석](https://daddynkidsmakers.blogspot.com/2025/01/langchain_3.html) |  |
|  |  | RAG 기본 실습 | R2 | 6 | [LLM-RAG-Agent-Tutorial/4-1.langchain at main · mac999/LLM-RAG-Agent-Tutorial](https://github.com/mac999/LLM-RAG-Agent-Tutorial/tree/main/4-1.langchain) |  |  |
|  |  | RAG | R3 | 3 | [Langchain 기반 개인화된 ChatGPT 대화, 코딩, QA 챗봇 자체 서비스 개발](https://daddynkidsmakers.blogspot.com/2023/10/chatgpt-langchain.html) |  |  |
|  |  | RAG | D1 | 3 | [대형언어모델 검색증강생성의 핵심기술, 벡터 데이터베이스 Chroma 분석](https://daddynkidsmakers.blogspot.com/2024/06/llm-rag-chroma.html) |  |  |
|  |  |  |  |  |  |  |  |
|  | 데이터베이스 | Graph DB | D2 | 6 | [그래프 데이터베이스 Neo4J 기반 데이터 질의 서비스 개발](https://daddynkidsmakers.blogspot.com/2024/05/neo4j.html) | [LLM 기반 그래프 RAG 기술 구현](https://daddynkidsmakers.blogspot.com/2024/05/rag.html) |  |
|  |  | Function call 실습(RAG) | R4 | 6 | [LLM function call 개념 및 메커니즘](https://medium.com/@laputa99999/effective-and-simplified-llm-function-call-for-ai-agent-development-6ccd6e44a191) 구현 | https://platform.openai.com/docs/guides/function-calling/function-calling |  |
|  | 챗봇 UI | Gradio, Streamlit | U1 | 6 | [Gradio 와 LangChain 을 이용한 간단한 챗봇 개발](https://daddynkidsmakers.blogspot.com/2025/01/gradio-langchain.html) | [Daddy Makers: Gradio HTML Javascript 렌더링 방법](https://daddynkidsmakers.blogspot.com/2025/04/gradio-html.html) |  |
|  | 챗봇 호스팅 | 호스팅 서버 | U2 | 2 | [웹 앱 서비스 호스팅 Fly.IO](https://daddynkidsmakers.blogspot.com/2024/10/flyio.html) | 구매 결재 [Stripe](https://stripe.com/), 도메인 호스팅 [Namecheap](https://www.namecheap.com/) 등 |  |
|  | Ollama | 설명 및 실습 | O1 | 3 | [Ollama 도구 및 실습](https://daddynkidsmakers.blogspot.com/2024/09/webui-ollama.html) | Ollama 설치. Chatbot 실습 |  |
|  |  | Chatbot 실습 | O2 | 3 | [로컬 호스트 LLM 오픈소스 OLLAMA 기반 PDF 지식 챗봇 서비스 개발](https://daddynkidsmakers.blogspot.com/2024/02/github-copilot-ai.html) |  |  |
|  |  | 퀴즈 및 과제 | Q | 4 |  |  |  |
| 5 | 에이전트 | 개요 | A1 | 3 | [벡터DB, 도구, 메모리 기반 간단한 PDF, Web 검색 에이전트 개발과 랭체인 라이브러리 설명](https://daddynkidsmakers.blogspot.com/2024/12/pdf-web.html) |  |  |
|  |  | 이론 | A2 | 1 | [Why we no longer use LangChain for building our AI agents (Hacker News](https://news.ycombinator.com/item?id=40739982)) | [RAG LLM 기반 멀티 에이전트 시스템 개발 이론과 한계](https://daddynkidsmakers.blogspot.com/2024/11/llm.html) |  |
|  |  | 이론 | A3 | 2 | Google AI agents, Open AI agent 백서  (Reasoning, ReAct, CoT, Human-in-the-Loop) | google-ai-agents-whitepaper, practical-guide-to-building-agents.pdf | [arXiv.org e-Print archive](https://arxiv.org/) |
|  |  | MCP | A3 | 6 | [인공지능 AI 에이전트 표준 프로토콜 MCP 분석 및 사용](https://daddynkidsmakers.blogspot.com/2025/03/ai-mcp.html) | <https://modelcontextprotocol.io/quickstart/user> |  |
|  |  | MCP 개발 | A5 | 6 | [Gemini 기반 MCP 서버 및 클라이언트 개발](https://daddynkidsmakers.blogspot.com/2025/04/gemini-mcp.html) | 현장 상황 고려해 간단한 MCP 도구 개발 (날씨, 계산기, 임베디드보드 제어 등) |  |
|  |  |  |  |  | Claude Desktop, vscode, git tool 미리 설치 필요. |  |  |
|  |  | 마누스 | A6 | 6 | [오픈 마누스(manus) AI 에이전트 설치, 사용 및 구조 분석](https://daddynkidsmakers.blogspot.com/2025/03/manus-ai.html) | 설치 실습 |  |
|  | 에이전트 응용 | 예제 | W1 | 4 | [Geo Map AI Agent Dashboard Web App for example](https://github.com/mac999/geo-llm-agent-dashboard) | 코드 분석 실습 |  |
|  |  | 예제 | W2 | 2 | [개발 예시 - 공학용\_LLM\_code\_agent](https://github.com/mac999/BIM_LLM_code_agent) |  |  |
|  |  | 예제 | W3 | 4 | [Blender for AI-Assisted 3D Modeling](https://medium.com/@laputa99999/using-open-source-models-with-blender-for-ai-assisted-3d-modeling-comparative-study-with-openai-9848209f93b8) | 블렌더 설치, 코드 분석 실습 |  |
|  | 바이브코딩 | 설명 | W4 | 2 | Vibe 코딩 개발 도구 설명 |  |  |
|  |  | 실습 | W5 | 6 | 미디어 업무 자동화도구, 에이전트 개발 과제 | vscode 기반 실습 (Github Copilot) |  |
|  |  | 총정리 퀴즈 | Q | 6 | 앞서 정리한 것 총 정리 퀴즈 | embedding, cosine similarity, Linear Mapping, attention, VAE, diffusion, transformer, LoRA, softmax, U-net, n-gram, MMR, MCP 등 |  |
| 6 | 프로젝트 메이커톤 |  |  |  | 팀별 프로젝트 수행, 발표 및 피드백, 평가 | 작성 컨텐츠(소스, 계획서 포함)는 github나 huggingface에 업로드. 역할별 개별 github commit upload 기록 이어야 함.  하위 폴더구조 예시) ./doc, ./src, ./dataset, ./test, ./paper | 현장 개발환경 고려해 진행 |
|  | 진행설명 |  |  | 2 | 피드백 기준 = {미디어 주제성, AI활용, 효율성, 가치, 창의성} | 측정지표={정보리드타임, 편의성 설문, 정확도, 속도, 기타}.  비교방식=AS-IS/TO-BE, % |  |
|  | 주제선정 |  |  | 2 | AI 기술, 도구 활용한 주제. 미디어 관련 주제. 예) 업무 자동화, AI생성, 에이전트 등 컨텐츠 개발 | 대용량 데이터는 google drive 에 업로드하고, 링크를 github README 에 설명해 표시할 것. |  |
|  | 팀원역할 |  |  | 1 |  |  |  |
|  | 목표기획 |  |  | 1 |  |  |  |
|  | 컨텐츠설계 |  |  | 2 |  | GPT, 바이브 코딩, 생성AI도구(https://app.visily.ai/ 등) 사용 가능 |  |
|  | 데이터준비 |  |  | 4 |  |  |  |
|  | 개발 |  |  | 6 |  | 바이브 코딩, |  |
|  |  |  |  |  |  |  |  |
|  | 개발 |  |  | 12 |  | [Push Code to your GitHub Account - Under 3 Minutes](https://www.youtube.com/watch?v=vpRkAoCqX3o) |  |
|  | 테스트/커밋 |  |  | 3 |  | github, huggingface 등 |  |
|  | 발표 |  |  | 10 |  | 팀별 발표, 데모, 코드 및 Paper 설명 |  |
|  | 피드백 |  |  | 5 |  |  |  |

* 학습 목표: 미디어 부분 니즈 고려한 프롬프트 엔지니어링 산업 트랜드 이해. 기술적 작동 메커니즘 이해(트랜스포머, LLM, RAG, AI-Agent, MCP 용어, 개념 및 구현).
* 고려 사항: 현장에서 각 개인 능력 수준, 개발 환경 및 상황을 고려해 주제 순서 및 내용은 일부 변경될 수 있음. 선형대수, 수치해석 사전수강 권장.
* 진행 순서: 개념 용어 소개(PPT, PDF, word 등) 후 주요 구현 메커니즘은 가급적 블로그, 논문 및 소스 코드 등을 사용해 확인 이해.
* 과제: 부족한 부분은 과제로 하여 숙지.
* 환경: 개발 실습 환경은 colab 환경을 기준으로 하되, 좀 더 의미있는 프로젝트 수행을 고려해, vscode / anaconda / nvidia cuda 지원 환경 권장.