پروژه دوم درس مبانی داده کاوی

طراحی و پیاده‌سازی دستیار پژوهشی هوشمند مبتنی بر LLM

مقدمه

هدف این پروژه، طراحی و پیاده‌سازی یک دستیار پژوهشی هوشمند (Research Assistant) مبتنی بر Large Language Model است که بتواند مجموعه‌ای از مقالات علمی را تحلیل کرده، دانش موجود در آن‌ها را بازیابی نماید، اطلاعات ساختاریافته استخراج کند، مقایسه میان مقالات انجام دهد و در نهایت به عنوان یک سیستم هوشمند پژوهشی با کاربر تعامل داشته باشد.

بخش اول — انتخاب مقالات علمی

هر گروه موظف است حداقل ۴ مقاله علمی معتبر در یک حوزه تخصصی مشخص انتخاب نماید.

شرایط انتخاب مقالات:

۱. مقالات باید از یک حوزه مشترک باشند.

۲. مقالات باید قابل مقایسه با یکدیگر باشند.

۳. ترجیحا مقالات باید یک مسئله مشابه را حل کرده باشند.

۴. بهتر است مقالات روی دیتاست مشترک یا benchmark مشترک ارزیابی شده باشند.

نمونه موضوعات قابل قبول:

- Object Detection on COCO Dataset
- Image Classification on ImageNet
- Machine Translation using Transformer Models
- Multi Object Tracking on MOT17 Dataset
- Medical Image Segmentation on BraTS Dataset
- Large Language Models for Question Answering

منابع پیشنهادی:

- IEEE Xplore Digital Library
- ACM Digital Library
- arXiv

بخش دوم — ساخت پایگاه دانش مقالات

سیستم باید مراحل زیر را پیاده‌سازی نماید.

استخراج متن از فایل PDF

متن مقالات باید از فایل PDF استخراج گردد.

بخش‌های مهم مقاله شامل:

- Abstract
- Introduction
- Related Work
- Methodology
- Experiments
- Results
- Conclusion
- References

تقسیم متن(Chunking)

متون استخراج شده باید با یک استراتژی مناسب به بخش‌های کوچکتر تقسیم شوند.

استراتژی chunking باید توسط دانشجو طراحی و توجیه شود.

تولیدEmbedding

برای هر chunk باید embedding تولید شود.

استفاده از هر embedding model مجاز است.

ساخت پایگاه داده برداری

دانشجو باید از یک Vector Database استفاده نماید.

نمونه ابزارها:

- FAISS
- ChromaDB
- Pinecone

بخش سوم — طراحی سیستمRAG

سیستم باید از معماری Retrieval Augmented Generation استفاده نماید.

مراحل کلی:

- دریافت پرسش کاربر
- تبدیل پرسش بهembedding
- retrievalازvector database
- انتخابchunk های مرتبط

- ارسال context به مدل زبانی
- تولید پاسخ نهایی

سیستم باید بتواند سوالات پژوهشی کاربر را بر اساس مقالات پاسخ دهد.

نمونه سوالات:

- Which paper achieved the best accuracy?
- What datasets were used in these papers?
- What are the main differences between the proposed methods?
- Which paper uses transformer-based architecture?

بخش چهارم — پیاده‌سازی() Function Calling

سیستم باید قابلیت تصمیم‌گیری برای فراخوانی توابع مختلف را داشته باشد.

مدل زبانی باید به صورت خودکار تشخیص دهد چه زمانی لازم است یک تابع فراخوانی شود.

سه تابع زیر **الزامی** هستند.

تابع اول (الزامی)

extract_experimental_results()

وظیفه:

استخراج نتایج تجربی از مقالات و استخراج حداقل دو معیار ارزیابی مشترک بین مقالات.

نمونه معیارها:

- Accuracy
- Precision
- Recall
- F1 Score
- mAP
- IoU

تابع دوم (الزامی)

compare_results()

وظیفه:

مقایسه عملکرد مقالات و تولید خروجی‌های زیر:

- جدول مقایسه
- نمودار مقایسه عملکرد روش‌ها

تابع سوم (الزامی)

generate_survey()

وظیفه:

سیستم باید بر اساس ۴ مقاله انتخاب شده یک Mini Surveyتولید نماید.

خروجی باید شامل:

- معرفی مسئله
- معرفی روش هر مقاله
- مقایسه روش‌ها
- روند پیشرفت روش‌ها
- جمع‌بندی نهایی

بخش پنجم — توابع اختیاری (دارای نمره مثبت)

پیاده‌سازی هر کدام از موارد زیر دارای امتیاز اضافه خواهد بود.

تابع اختیاری اول

find_research_gap()

وظیفه:

تحلیل بخش conclusion و future work و استخراج:

- محدودیت‌ها
- مشکلات حل نشده
- پیشنهاد برای تحقیقات آینده

تابع اختیاری دوم

compare_references()

وظیفه:

تحلیل بخش references و تولید جدول مقایسه:

- تعداد رفرنس‌ها بر حسب سال انتشار
- مقایسه تاریخی منابع مورد استفاده مقالات

نمونه خروجی:

Year	Paper A	Paper B	Paper C	Paper D
2020	12	10	8	15
2021	9	14	11	7

تابع اختیاری سوم

compare_limitations()

وظیفه:

تحلیل limitations هر مقاله و مقایسه نقاط ضعف روش‌های مختلف.

بخش ششم — قابلیت پردازش صوت (امتیاز اضافه)

در صورتی که سیستم بتواند ورودی صوتی دریافت نماید، امتیاز مثبت تعلق خواهد گرفت.

قابلیت‌های قابل قبول:

- دریافت voice انگلیسی
- دریافت voice فارسی
- تبدیل گفتار به متن
- پردازش متن توسط سیستم

نمونه ابزارها:

- OpenAI Whisper
- Whisper GitHub

بخش هفتم — رفتار سیستم در نبود اطلاعات مرتبط در RAG (الزامی)

سیستم باید بتواند تشخیص دهد که آیا اطلاعات کافی در پایگاه دانش وجود دارد یا خیر.

اگر پاسخ پرسش کاربر در مقالات یافت نشود، سیستم نباید صرفا پاسخ را رد کند.

در این شرایط سیستم باید:

۱. اعلام کند پاسخ در مقالات یافت نشده است.
۲. سپس با استفاده از مدل زبانی پاسخ عمومی ارائه دهد.

فرمت پاسخ:

پاسخ شما در مقالات انتخاب‌شده یافت نشد.
اما با استفاده از مدل زبانی، پاسخ عمومی به شرح زیر است:
...

بنابراین سیستم باید صرفا محدود به RAG نباشد.

بخش هشتم — رابط کاربری

حداقل خروجی پروژه باید شامل یک **اپلیکیشن ترمینالی (CLI Application)** باشد.

نمونه:

Ask your question:

> Compare accuracy of all papers

امتیاز اضافه

در صورت طراحی رابط کاربری پیشرفته‌تر، امتیاز مثبت تعلق خواهد گرفت.

نمونه‌ها:

- Web Application
- Desktop Application
- Chat Interface

ابزارهای پیشنهادی:

- Streamlit
- Flask
- FastAPI
- Gradio
- PyQt

گزارش نهایی ویدیویی پروژه

هر گروه باید گزارش ویدیویی پروژه ارائه نماید.

گزارش باید شامل موارد زیر باشد:

- موضوع پروژه
- معرفی ۴ مقاله انتخاب شده
- معماری سیستم
- روش استخراج متن
- روش chunking
- مورد استفاده embedding model
- مورد استفاده vector database
- طراحی function calling
- نمونه پرسش‌ها

- تحلیل عملکرد سیستم
- محدودیت‌ها و نقاط ضعف

نکات:

- فایل ویدیویی توضیح برنامه خود را که شامل توضیح موارد شرح داده شده ، چالش ها و سخت افزار اجرایی و اجرای آن است حداکثر تا روز امتحان ارسال نمایید.
- طول فایل ویدیوی بین ۱۰ تا ۲۰ دقیقه باشد.
- گزارش فقط به صورت ویدیویی مجاز است.
- به ازای هر روز تاخیر نهایتا تا ۲ روز تاخیر در ارسال، کسر نمره تا ۲۰ درصد منظور می گردد.
- به گزارش های مشابه نمره ای تعلق نمی گردد.
- محدودیتی در انتخاب پکیج و مدل و ... برای حل مسئله وجود ندارد. اما در مستندات باید شرح داده شود.
- ارسال پروژه یا به صورت ۲ نفری و یا به صورت ۳ نفری با ارائه قسمتهای اضافی انجام گیرد.

خوشدون باشِد. Drmkiani.ir