

Presentations Creator using LLM

Advanced Techniques for Large Language
Model Applications

GOAL OF PROJECT

The goal of this project is to create a Telegram bot that generates presentations based on course materials (or simply PDF files) and topics for all presentations.

The bot will utilize a combination of technologies:

- Sentence Transformer to embed database items
- Retrieval-Augmented Generation (RAG)
- Google Generative AI, to extract relevant information from course materials and generate presentations.

METHODOLOGY



Database

Qdrant Vector Search using **Sentence Transformers** that turns data into embeddings.



LLM

Google Gemini is used to leverage Google's generative AI capabilities for various text and code problems.



Others

Telebot allows to create and interact with Telegram bots.

python-pptx is used for creation of presentation via python.

Process steps:

1. start bot
2. extract pdf from bot
3. extract topics from pdf page by page and pull it to db using Gemini to summarize and preprocess
4. receive list of topic per course from bot
5. iterate over topics:
 - extract text from db using custom embedding function (Sentence Transformer)
 - checking the similarity of files and topics using Cosine distance
 - ask Gemini to create code for presentation based on this text
 - run code and receive PP presentation
6. pull PP presentations to bot

pptx_creator_bot
bot

/start

12:53

✓✓

Hi, this is a pptx_creator_bot for creating presentations based on the pdf version of the course book and lecture plan. To start creating presentations, send a pdf file with the course materials.

12:53

CV_coursebook.pdf

4.9 MB

12:53

✓✓

Please wait while the file is being written to the database. It may take some time. Done!

12:53

Mariia Shmakova

CV_coursebook.pdf

207 documents have been successfully saved in the database.

13:03

Mariia Shmakova

CV_coursebook.pdf

Please provide me the course content by highlighting each lecture in a separate line so that I can include it in pptx.

13:03

Representation of images and videos (Computer representation, Rescaling/manipulating images)
Image Classification (Loss Functions, Backpropagation)
Neural Networks (Training)
Convolutional Neural Networks (Training, Architectures)
Recurrent Neural Networks (Training, Architectures)
Image Segmentation and object detection (Techniques)

13:04

✓✓

Mariia Shmakova

Representation of images and videos (Computer re...

Please wait until the bot sends you the pptx files.

13:04

Coursebook have initially 235 pages, after the preprocessing it converts to 207 document that contain titles and texts.

An example of using a telegram bot to create presentations for a CV course based on a textbook and sections from the course description.

Representation of images and videos (Computer representation, Rescaling/manipulating images)
Image Classification (Loss Functions, Backpropagation)
Neural Networks (Training)
Convolutional Neural Networks (Training, Architectures)
Recurrent Neural Networks (Training, Architectures)
Image Segmentation and object detection (Techniques)

13:04

✓✓

Mariia Shmakova

Representation of images and videos (Computer re...

Please wait until the bot sends you the pptx files.

13:04

Lecture1.pptx

48.8 KB

13:10

Lecture2.pptx

44.7 KB

13:10

Lecture3.pptx

56.1 KB

13:10

Lecture4.pptx

62.3 KB

13:10

Lecture5.pptx

37.5 KB

13:10

Lecture6.pptx

44.7 KB

13:10

Example of presentation

Presentations vary in the form of a text presentation

Also, each presentation contains only the text part, that is, formulas, pictures, etc. must be inserted at will, since this framework is not fully suitable for this.

Convolutional Neural Networks

Lecture 4

Example of presentation

Presentations vary in the form of a text presentation

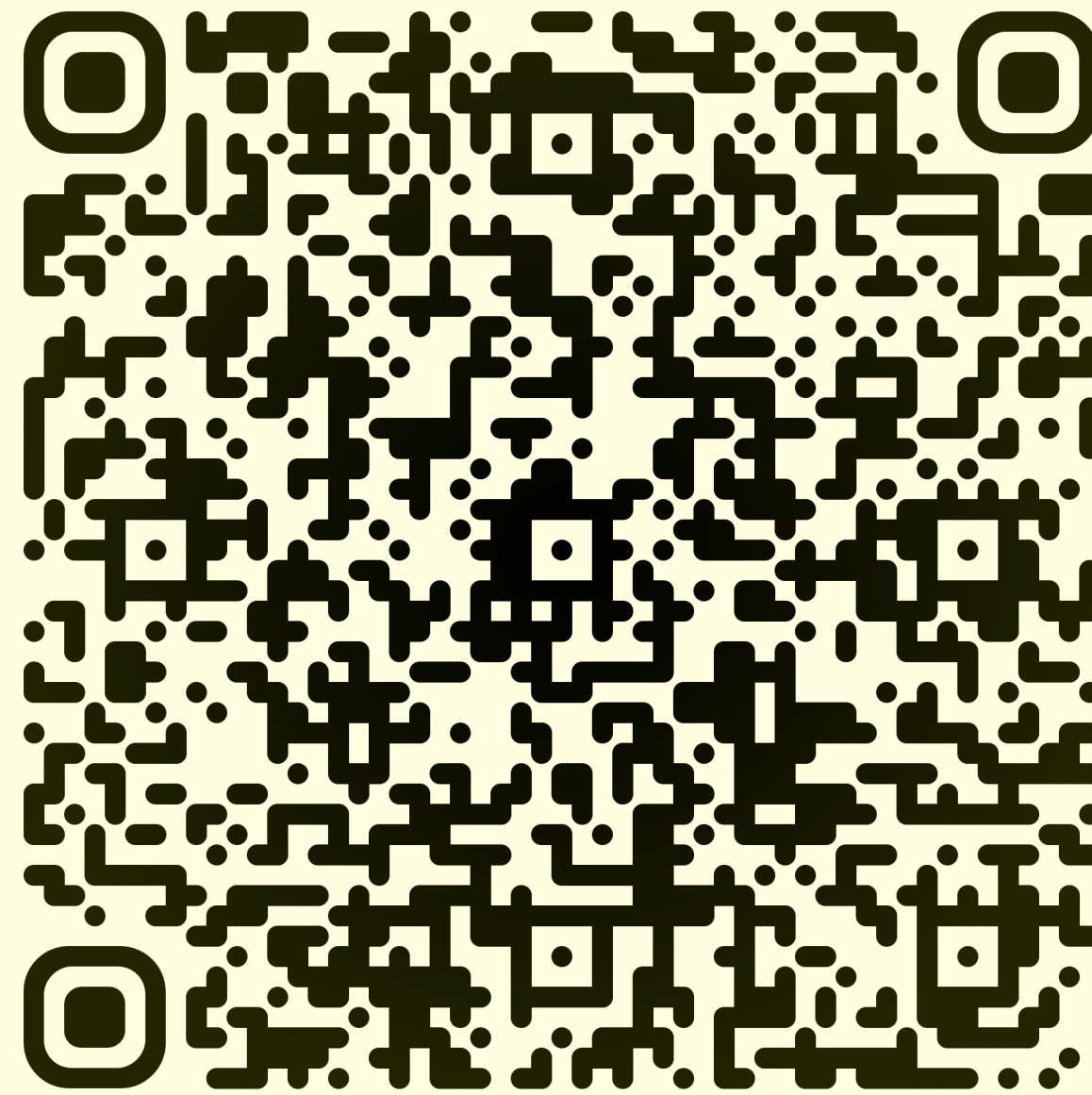
Also, each presentation contains only the text part, that is, formulas, pictures, etc. must be inserted at will, since this framework is not fully suitable for this.

Image Segmentation and Object Detection

Thanks for your time



Link to telegram bot



Link to GitLab



Link to GitHub