

USPTO Technical Drawing Search Engine

Muntabir Choudhury
Ph.D. Student
Department of Computer Science
Old Dominion University

Introduction

- I am from **Dhaka, Bangladesh**
- **Bachelor of Science** in Computer Engineering
 - Graduated from **Elizabethtown College**, PA, USA – 2018
 - On Campus Jobs: (a) Grader (b) Database Assistant (c) International Leadership Assistant
 - Summer Research – Data Analytics
- Worked with **Resource 9 Group** – a New York based start-up
 - Application Performance Engineer (Specialized on **AppDynamics**)
 - Contractor – **John Deere**
- Joined **Old Dominion University** as a Ph.D. student in Fall 2019.
 - Advisor: Dr. Jian Wu
 - Our research lab is known as **LAMP-SYS** (Lab for Applied Machine Learning and Natural Language Processing and Systems)
 - We are also collaborating with **WSDL lab** (Web Science and Digital Libraries, ODU)
 - Two Summer Internships:
 - **Research Intern – Los Alamos National Laboratory**, New Mexico – 2020
 - **Machine Learning Intern – Bihle Applied Research Inc**, VA – 2021

Building Search Engine

- Dataset: **DeepPatent3** – contains 50,000 technical drawings
- We have two tasks –
 - Front End – is a user interface (UI) that will connect to the database
 - Mostly focusing on the design aspects of the website
 - Connect to database so that it can return search results
 - Features: a **search bar** which will allow users to query through text and image
 - Example: [LAION-5B Search Demo](#)
 - We plan to use [Django](#) – a python framework to build the UI
 - Back End – focuses on the server-side application which you can not see in the UI
 - We will use Deep Learning based model ([OpenAI's Clip Model](#)) to embed text and image into vectors
 - Store these vectors in the database (vector database)
 - [Autofaiss](#) to create a KNN index for retrieval
 - Returns a best score using a similarity search algorithm (e.g., [cosine similarity](#))
 - Sorting the score in **descending order** to return the result of the user's query

Front End – User Interface

Two Features:

- Text Query
- Image Query

SEARCH



Clip Model

Text Encoder

Image Encoder

Embedding Phase

T_1 T_2 T_3 T_4

I_1	I_1T_1	I_1T_2
I_2	I_2T_1	I_2T_2
I_3
I_4

Sorting best result in descending order

Vector DB

Use Autofaiss for indexing

- KNN index
- Similarity search
- Return similarity score

Back End

Dot Product of Image and Text Vectors

Raw Text + Images

DB

It's a pencil



Tools / Technologies and Frameworks

- Version Control and Code Sharing: [GitHub](#)
- AI model: Clip model (pre-trained model for embeddings)
- Indexing tools: Autofaiss
- Framework: Django for front end
- HPC Cluster: [Wahab Cluster](#)
 - Jupyter Notebook
- Database: ES, MySQL, MongoDB (options)
- Tutorials:
 - Django W3 Schools: <https://www.w3schools.com/django/>
 - Django Official Website: <https://docs.djangoproject.com/en/4.2/>
 - Clip Model: <https://github.com/openai/CLIP>
 - Autofaiss: <https://github.com/criteo/autofaiss>
 - GitHub Installation: <https://www.pluralsight.com/guides/using-git-and-github-on-windows> (however, first open an account into GitHub)