- STABLE DIFFUSION -

CMPE 491

Project Specifications Report

29.10.2022

Arda Atakol 13624005832

Serdar Hoşver 10061964142

Furkan Özelge 14758028780

Yağız Hikmet Karakuş 44164883604

Senior's Website: https://furkanozelge.github.io/seniorstedu/

1. INTRODUCTION

1.1. DESCRIPTION

This is a Stable Diffusion project which is the program that convert the text to the image. A text-to-image model using deep learning is called Stable Diffusion. Although it can be used for various tasks including inpainting, outpainting and creating image-to-image translations directed by text prompts, its primary usage is to generate detailed visuals conditioned on text descriptions. So, with this information, we will create an application & tool that uses deep learning and artificial intelligence, which creates a design idea for the use of designers by using stable diffusion together with a model of pre-trained open-source data. While doing this, we will work on adding various features by going beyond what has been done. Stable Diffusion will reveal very creative ideas and images for designers. In this way, we will give designers a new perspective.

Application will be two window application which has text box window and the login page window. After the users logon the application, they will see the text-box window and when the user enters his/her text to the text box, the program will convert it to the image. For example, when you enter the input ‘’a photograph of an astronaut riding a horse’’, application will read that command and pull the related images from data which was trained before and it will show the equivalent image. User will see a horse with the astronaut after that input.

1.2. CONTRAINTS

Our overall goal is to introduce stable diffusion and AI to all users and make a good impression. The ability of people to draw their dreams to artificial intelligence with just a few words or sentences. At this point, our general scope is to reach the entire user base. Some reasons arising from this may lead the project to some restrictions. In order to appeal to the entire user base, there should be no +18 content in this AI project in general. This will have a positive impact on our market value. Because one of the rules

of the market is to expand the user base. This will contribute to our budget because we will make a profit that will reduce the cost in the project. In addition, all users will have the right to produce a limited number of photos. Because it costs us a certain cost in each photo generate. For this reason, it is very important to have a border in the middle. At the same time, we will do this by registering and logging in. All users must have an account. There are many reasons for this. We also need this so that we can limit the number of uses. We will also offer suggestions based on previous searches in user accounts, while providing suggested photos to users. We will also get this with the user account. In addition, we will have a very strict limit on terrorism and violence. It will be forbidden to use words of this variety. This limitation is again a necessary limitation to be suitable for all users. But of course, there will be many more sub-headings about cost. First of all we will need a domain for the project. Our project is a big one and we need a nice domain for it and it will cost around $20-30$ per year. In addition to this, there will be an annual fee of $25-$35 for hosting. Apart from these, there may be fees for various frameworks and technologies that we will use on stable diffusion. The most important thing for us is data. We will also need a large server for data. This will cost around $50. In fact, we will try to reduce costs by putting various mini ads in our project. These ads will not be in a way that disturbs the user. We will use advertising to generate an income for us. There will be a premium package so that we can make some profit on the cost. Users with the Premium package will purchase this package for a price. In this way, they will be able to generate unlimited photos and have an ad-free experience. This will contribute to us on the cost.

Time is also one of the most important issues in a project. This project is done in a team of 4 people and it will cost 4 people to work in various areas of this project for 4-5 months. We need to reflect this time positively on the project as quality. All team members will devote at least 10-15 hours per week to this project. It is very important to be able to use this as quality. If we can't spend the time we spend on making this project a successful one, it will cause a great waste of time. For this reason, the tasks and each branch of the project will be carefully divided and finished with a regular study.

Of course, like every project, this project has some risks. It is our greatest desire to make a project worth the effort we spend, the time we spend and the money we spend. But for this, various risk management should be done. We should consider what measures we can take in various scenarios. We will see this with various tests and reviews before it reaches the end user. On top of that, we will make the project perfect with many updates. All projects would be perfect with updates. We will fix the bugs in the first versions in the next versions. In this way, we will have a sustainable project. Sustainability is the most important thing for a project. Our project is not a project that can be consumed quickly. All kinds of users will get magnificent designs and drawings by pushing the limits of imagination.

Stable diffusion is not a subject that has too much diversity as a resource. For this reason, we will be very limited on research. But in this case, we will step in. In scientific studies or projects, you should be a resource yourself where the resource ends. For this reason, we will not be afraid to develop new technologies ourselves in places where there are no resources.

1.3. PROFESSIONAL AND ETHICAL ISSUES

This Stable Diffusion application will be an open source and because of that there are some problems we may face. All team members should be considerate of one another and open-minded when it comes to ethical problems. Racism, discrimination, and toxic behavior are not permitted under any circumstances. Team members should behave professionally and attempt to resolve conflicts in a constructive and courteous way whenever they arise. All team members should be receptive to constructive criticism and recommendations. The confidentiality of the project's information is everyone's duty, both personally and professionally.

For professional reasons, each team member should make every effort to participate as effectively as they can to the development process to produce a high-quality result. To stay up with the changes and sprints, every member should be knowledgeable with cutting-edge methods and technologies. Additionally, team

members must communicate clearly, effectively, and informatively. Each team member should update the others on how their work and sprints are going. Code review and sprint suggestions must be politely acknowledged. Pair programming can also be used to guarantee code quality for time-consuming activities. Each assignment should be given to a team member who is interested in doing that specific work. Software components should be modular so they can be used.

We are responsible for harmful images produced under the Open RAIL-M license. For this reason, the report image button will be added to make the application ethically correct. In this way, we will prevent things that will be ethically bad.

2. REQUIREMENTS

2.1. Home Screen

2.1.1. When the user first opens our website if They are not logged in they see our home page.

2.1.2. The home page contains information about the application.

2.1.3. On the home page, they can see the announcement and latest updates for our app

2.1.4. There will be a how to use content on the home page

2.1.5. In the navigation bar, they can see the contact info button.

2.1.6. In the navigation bar, they can see the login button.

2.1.7. In the navigation bar, the user can see the register button.

2.2. Login and Sign Up

2.2.1. Create a back-end for login

2.2.2. User face with username and passwords field

2.2.3. If the information matches our database user logged in to the app

2.2.4. There will be an keep me logged in the check box

2.2.5. If the user checks the keep me signed in user doesn’t need to log in the for the next time

2.2.6. If the information that the user gives us is missing or faulty. If the user faces an error message

2.2.7. There will be a forgot my password button

2.2.8. If user clicks forgot my password button. They will be routed to the password recovery page

2.2.9. If the user clicks the register button. They will be routed to the register page

2.2.10. There will be a register button

2.3. Register System

2.3.1. Creates a back end for the registration page

2.3.2. User face with register input boxes

2.3.3. User provides e-mail, username, Name, Surname, and password information.

2.3.4. Bottom of the page there will be a register button

2.3.5. If the user fills correctly the required information user will be registered and added to the user database

2.3.6. If the user doesn't fill in successfully. The error message is sent to the user by the system.

2.3.7. After successful registration, they are routed to the login page.

2.4. Password Recovery

2.4.1. User face with an email input box to recover the password

2.4.2. The recovered mail will be sent to the users mail

2.4.3. Users see a verification area and put the code they receive from the e-mail

2.4.4. Then they can face new passwords and repeat new password fields

2.4.5. If the user gives the same password in both fields user's password will be updated in the database

2.4.6. If the user gives the wrong password one of the fields users sees an error message

2.4.7. The user will be routed to the login page after a successful recovery

2.5. MVC Models

2.5.1. Creates a MVC model to user-specific page

2.5.2. Models in the timelines containing the created content from other users.

2.5.3. Creators' user info and content like counts.

2.5.4. Our controller is mainly a recommender system that uses the user search and liked posts

2.6. Recommendation System

2.6.1. We use a matrix factorization for the recommendation system

2.6.2. Each image that is created by stable diffusion has a label(keyword)

2.6.3. With help of users liked and disliked posts create a recommendation to the user

2.7. User Timeline

2.7.1. User timeline created dynamically

2.7.2. The data came from the recommendation matrix shown in the user timeline

2.7.3. User can like or dislike the posts in the timeline

2.7.4. If users click the post's creators name, they will be routed to their profile

2.7.5. There will be a report image button on the posts if user clicks it report feature is triggered

2.8. Navigation Bar Update

2.8.1. After a successful login navigation bar will be updated

2.8.2. Users can see text to image, image to image, transfer art, timeline, and profile buttons

2.8.3. Style Transfer API

2.8.4. For style transfer API we create our models and use a pre-trained vgg19 model in python

2.8.5. Take content and style images and output the content image in each style

2.8.6. Our API adds the output content as

2.9. Style Transfer API

2.9.1. For style transfer API we create our models and use a pre-trained vgg19 model in python

2.9.2. Take content and style images and output the content image in each style

2.9.3. Our API adds the output content as

2.10. Stable Diffusion API

2.10.1. Our stable diffusion API we are going to use pre-trained weights and models.

2.10.2. We will create custom pipeline for stable diffusion.

2.10.3. We take text and tokenize the text with the help of our custom stable diffusion pipeline and create text embeddings this is help us to evaluate sentences in context.

2.10.4. With the latent diffusion model of stable diffusion we generate images and send them to the user as output.

2.11. User Profile Page

2.11.1. User profile picture

2.11.2. The user can upload their own profile picture or select one of the preset photos

2.11.3. User information

2.11.4. At the top of the page, the user can see their own information.

2.11.5. User-generated images

2.11.6. In the field below the user information, the user can see what he has selected among the visuals he has produced on his profile.

2.11.7. Favorite visuals.

2.11.8. The user can add likes to their favorites within the images produced by other users

2.11.9. Following and Followers.

2.12. Text to Image Page

2.12.1. Visuals will be produced with the entered key Word.

2.12.2. At the top of the page, there will be a text box for entering key Word or sentences

2.12.3. There will be generate button under the text box.

2.12.4. After clicking the generate button image will be generated and user routed to result page

2.13. Image to Image Page

2.13.1. With an image to be uploaded to the system, new visuals will be produced based on that image.

2.13.2. At the top of the page, there will be an upload area to upload sample images.

2.13.3. It can be found on the Internet and a sample image can be uploaded with a link.

2.13.4. An image in the computer can be uploaded to the system as an example visual.

2.13.5. After clicking the generate button stable diffusion API will be executed, image will be generated, and user routed to result page

2.14. Transfer Art Page

2.14.1. New visuals will be produced based on the input entered based on the visuals of other users or based on the style of a famous art.

2.14.2. There will be an area on the page where the style art is uploaded to the system.

2.14.3. There will be an area for the content image which is going to change.

2.14.4. If user click the transfer button our style transfer API I is going to executed and user routed to result page.

2.15. Revision Feature

2.15.1. There will be a %(percentage) bar for the Revision property. With this revision bar, feedback will be returned to the application.

2.15.2. Based on this feedback, a new revised output image will be returned to meet the user's requests.

2.16. Result page

2.16.1. In this page user see the outcome of their search

2.16.2. There will be a report button and revision feature will seen in this page

2.16.3. There will be a share button. If user clicks it the image become publicly available under his profile.

2.17. Report Feature

2.17.1. If a user reports an image the pop-up screen shown to the user

2.17.2. In the pop-up screen user select the reason of the report

2.17.3. If the reason user wants doesn't listed in the screen user can manually enter the input.

2.17.4. If user click the report button the report will send to our data base

3. REFERENCES

https://www.acm.org/code-of-ethics

https://www.ieee.org/about/corporate/governance/index.html

https://plato.stanford.edu/entries/ethics-computer/

https://dataconomy.com/2022/09/stable-diffusion-ai-art-generator/

https://www.howtogeek.com/830179/how-to-run-stable-diffusion-on-your-pc-to-generate-ai-images/#autotoc_anchor_1

Gotterbarn, Donald. "Software engineering ethics." Encyclopedia of Software Engineering 2 (2001).

Rashid, Awais, John Weckert, and Richard Lucas. "Software engineering ethics in a digital world." Computer 42.6 (2009): 34-41.