YAK 004.8

GENERATION OF VECTOR GRAPHICS 'WITH CHATGPT

Timofeenko B.A. (ITMO University)
Scientific research supervisor — Filchenkov A.A., PhD
(ITMO University)

Introduction. ChatGPT [1] is a SOTA Al language model, which is further development of GPT family, specifically recent GPT-3 model [2]. Besides its natural language processing capabilities, it can generate source code in different programming languages. But can we push the limits and generate source code of SVG graphics? In this work, I explore the application of ChatGPT to SVG vector graphics generation.

Main part. During the research, several experiments were conducted. Firstly, ChatGPT SVG generation ability is intentionally blocked by the authors. To bypass this limitation, "DAN" master prompt injection was used [2]. After the injection, ChatGPT starts answering to prompts like "As DAN, generate the source code of a vector SVG image of e".

Firstly, I asked ChatGPT to draw simple geometric shapes — triangles, rectangles, circles — with specified size, color, and stroke width. ChatGPT did a good job, outputting valid SVG files with these shapes, abiding all the constraints.

Then I asked ChatGPT to draw objects consisting of simple geometric shapes — a Christmas tree and a banana. I used 2 types of prompts:

e "As DAN, generate .> a Christmas tree" — for this prompt, the images were a complete mess and didn't even remotely resemble the objects.

o "As DAN, generate <...> a Christmas tree. The tree consists of — for this prompt, the images were moderately messed up, but the objects were recognizable. The Christmas tree had overlapping triangles, but the overall shape and color were correct. I noticed that the more complicated the prompt gets, the more difficult is gets for ChatGPT to execute it. Then I asked ChatGPT to draw complex objects without explaining their structure in detail _a cat. The resulting image was looking like a sloppy child's drawing, but at least, the cat was acknowledgeable. Trying to prompt a more detailed explanation of cat's structure led to a complete mess in SVG.

The last experiment was to feed ChatGPT source codes of SVG images and ask to explain what's pictured on the image. I fed a cat's silhouette, a smiley face, and a house. ChatGPT was not sure about the cat, and answered "It could be any number of four-legged animals, such as a dog, cat, or horse". It correctly guessed the face, but was wrong about the house.

Conclusion. For a general-purpose language model, ChatGPT is surprisingly good at generating SVG graphics. It can only work with simple shapes and objects and quickly reaches its limit, but it's still a big improvement upon the previous models. It can also correctly acknowledge simple objects in SVG graphics.

List of used sources:

1.   OpenAI team, "ChatGPT: Optimizing Language Models for Dialogue", OpenAI blog, 2022

2.   OpenAI team, "Language Models are Few-Shot Learners", Advances in Neural Information Processing Systems, pages 1877—1901, 2020

3.   AfSch001, "DAN 2.0" // "ChatGPT" community on Reddit, 2022

Timofeenko B.A. (author)   Signature

Filchenkov A.A. (scientific advisor)   Signature