Report

Advanced Programming

Sudoku Solver Bot

Project was done by: Arukhan Otarova and Aida Erbulat

Link to Github: https://github.com/aruhash/Sudoku-bot

Link to Youtube: https://www.youtube.com/watch?v=GrcCTXgTo6Q

Introduction:

1. Problem (Idea)

The idea behind the creation of this bot was to build an autonomous system that can analyze Sudoku grids, figure out the missing pieces of the puzzle, and fill the grid. This bot will logically guess the missing pieces of sudoku. We will also add Picture recognition as an addition to the main idea. So, the bot will work not only on online pictures, but also user will be able to scan the sudoku images. Bot will extract the sudoku grids from input image. We found this idea interesting, that's why we chose this topic. The main problem of our project is that it is not easy to solve sudoku and it takes a lot of time.

2. Brief description of our solution:

As a solution to our project, we decided to use machine learning for making people's life easier. Now sudoku will be solved by machine (Bot) , not by people themselves.

3. Available solutions with links:

- https://medium.com/swlh/how-to-solve-sudoku-using-artificial-intelligence-8d5d3841b872?source=friends_link&sk=c9055ce222a4ed4e6ea060a0f8bcfab2
- Python Sudoku Solver - Computerphile - YouTube

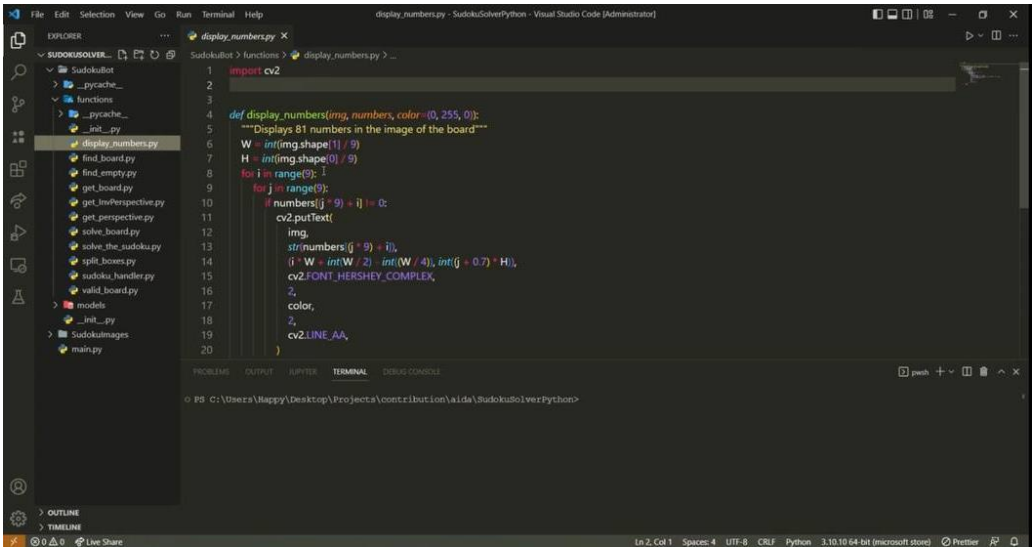
4. Current work:

In this project we used libraries as tensorflow, opencv-python, numpy. Here we have briefly described each file and what it does.

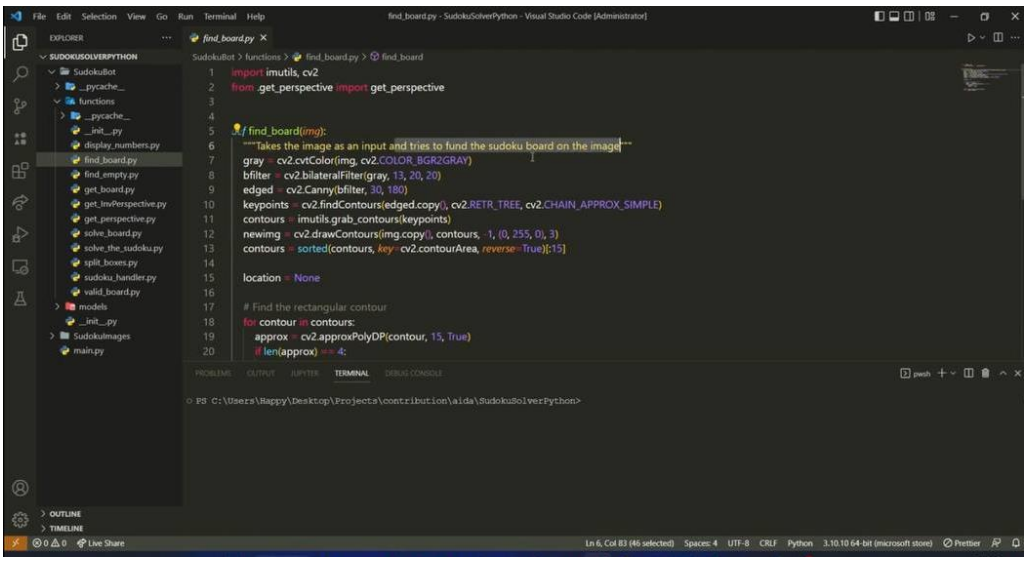
Main.py - The main root file of the project with all the imports of libraries and modules for the launch and performance of the telegram bot itself

```python
import asyncio
from aiogram import Bot, Dispatcher, executor
from aiogram.types import *
from aiogram.contrib.fsm_storage.memory import MemoryStorage as ms
from SudokuBot import sudoku_handler

bot = Bot(token="5856094549:AAEFnk5XCnRiNzfNudeDc0OelJhq6pq5dS0", parse_mode="html")
dp = Dispatcher(bot, storage=ms())


async def throttled(*args, **kwargs):
    m = args[0]
    await m.answer("Do not flood, please ;) \nYou can use bot only once per second!")


@dp.message_handler(commands=['start', 'help'])
@dp.throttled(rate=1, on_throttled=throttled)
async def start_command_handler(message: Message):
    await message.reply("Send me image of sudoku, and I will try to solve it for you! \nTry and you will receive the solution in a few seconds!")
```

- **Display_numbers.py** - Displays all digits, puts digits in empty spaces.
- **Find_board.py** - Takes a picture and finds a Sudoku square.
- **Find_empty.py** - Function to check if a cell is empty or not.



- **Get_board.py** - Takes a 9x9 sudoku matrix and returns the solved version
- **Get_InvPerspective.py** - Takes the original/original Sudoku picture and applies the "Perspective Transform" algorithm to it
- **Get_perspective.py** - Returns the selected area of the picture with the "Perspective Transform" algorithm applied to it
- **Solve_board.py** - Function (algorithm) that solves Sudoku
- **Solve_the_sudoku.py** - Main function that accepts a sudoku pattern and a path/name of the main model as parameters
- **Split_boxes.py** - Splits a sudoku board/table into 81 boxes/frequent
- **Sudoku_handler.py** - Message handler with "photo" type for telegram bot to respond/react to pictures sent by the user
- **Valid_board.py** - Checks and validates every row, column and square/box



5. Data and Methods

   All data that our project contains is taken from internet sources. For example, the formula of sudoku game:

   https://sudoku.com/how-to-play/sudoku-rules-for-complete-beginners/#:~:text=Sudoku%20is%20played%20on%20a,the%20row%2C%20column%20or%20square.

   And the examples of sudoku grids:

5	3	4	6	7	8	9	1	2
6	7	2	1	9	5	3	4	8
1	9	8	3	4	2	5	6	7
8	5	9	7	6	1	4	2	3
4	2	6	8	5	3	7	9	1
7	1	3	9	2	4	8	5	6
9	6	1	5	3	7	2	8	4
2	8	7	4	1	9	6	3	5
3	4	5	2	8	6	1	7	9

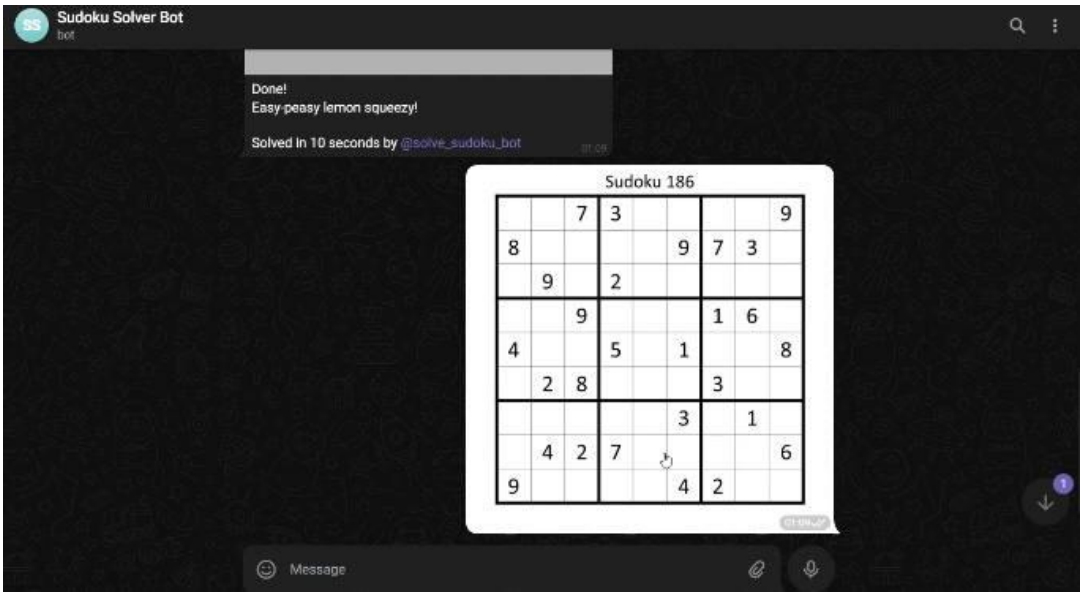
	7	2			4	9		
3		4		8	9	1		
8	1	9			6	2	5	4
7		1					9	5
9					2		7	
			8		7		1	2
4		5			1	6	2	
2	3	7				5		1
				2	5	7		

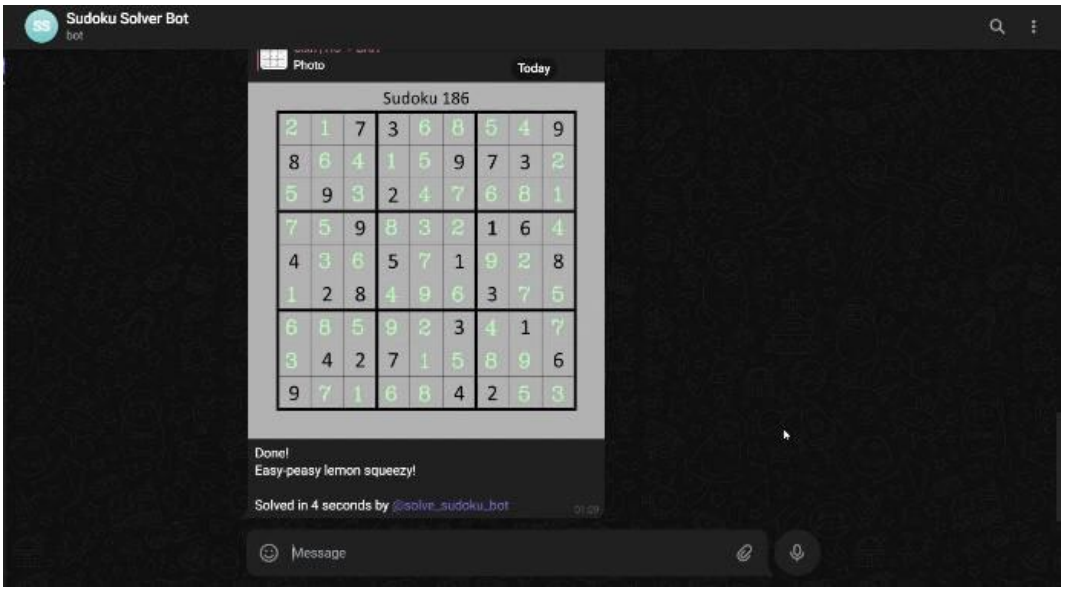
ML Models:

We used supervised ML. In supervised machine learning, the algorithm is provided an input dataset, and is rewarded or optimized to meet a set of specific outputs. For example, supervised machine learning is widely deployed in image recognition, utilizing a technique called classification. Supervised machine learning is also used in predicting demographics such as population growth or health metrics, utilizing a technique called regression.

6. Results:

Here is how our project turned out:

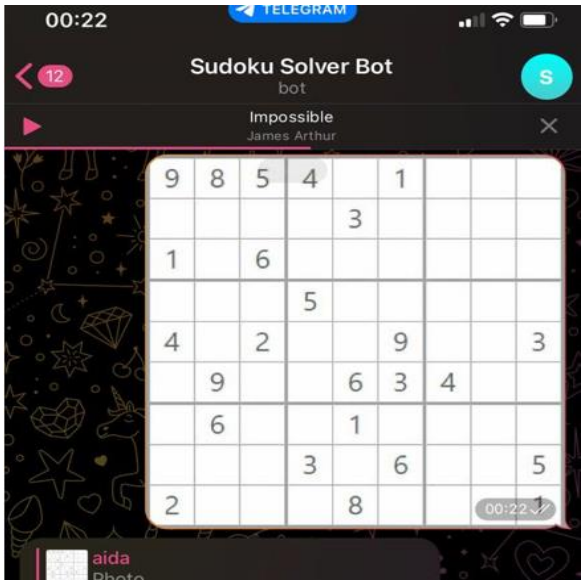




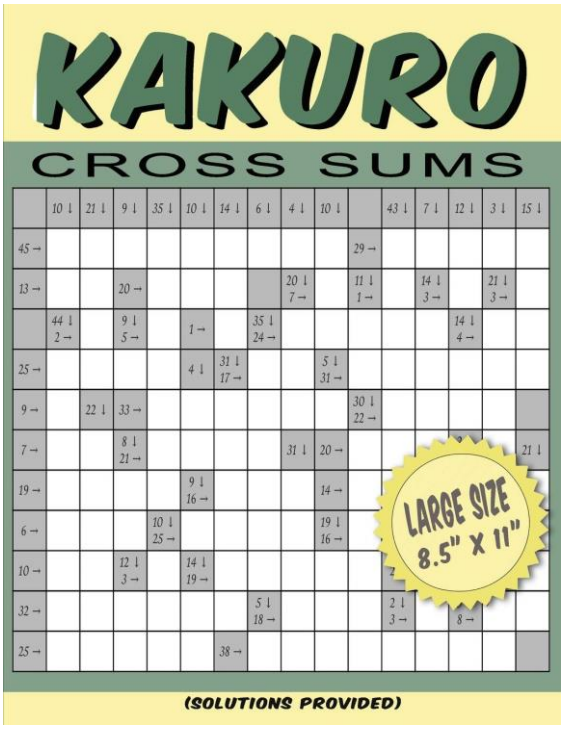
It displays the same image as an output, but with filled numbers. It is comfortable for users. They would not have to fill it up. It also gives information about the period of time the sudoku was solved.

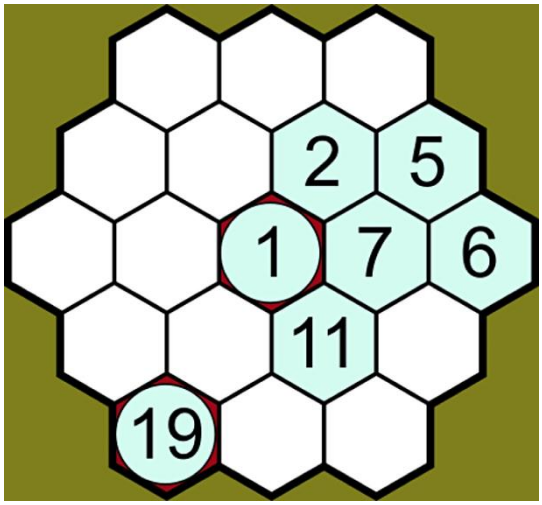
7. Critical review of results and next steps:

If we consider disadvantages of the project, it can be the limited opportunities, which means this bot is single purposed. (Only solves Sudoku) Another limitation that bot sometimes not able to scan the import picture. (If it's too dark/unclear)



Therefore, in the future we plan to develop this project to a new level, so that it could solve not only Sudoku but other types of mind games. Also, add some games (quizzes) according to them.





References:

- https://www.youtube.com/watch?v=qOXDoYUgNlU&t=567s

We gained basic functions and discovered how algorithms should work.

- https://www.geeksforgeeks.org/program-sudoku-generator/

Main algorithm for deducing missing numbers

- https://www.youtube.com/watch?v=cOC-ad0BsY0&t=150s

More about image processing