

ДОЦІЛЬНІСТЬ ВИКОРИСТАННЯ ФРЕЙМВОРКУ DJANGO ДЛЯ СТВОРЕННЯ СИСТЕМИ РОЗПІЗНАВАННЯ ОБЛИЧЬ

Хома Д.Ю., магістр, dmytro.khoma.kita@donntu.edu.ua;

Ануфрієв П.О., аспірант, pavlo.anufriiev@donntu.edu.ua

*Донецький національний технічний університет,
м. Покровськ, Україна*

Розпізнавання обличь є дуже складною та трудомісткою задачею, що відноситься до високопродуктивних обчислень. Тому, при створенні такого додатку важливо обрати такі технології з розробки програмного забезпечення, які будуть давати ряд переваг над іншими: швидкість, еластичність, безпека.

На даний момент існує багато мов програмування, що використовуються для створення систем, пов'язаних з машинним навчанням (далі – МН). За версією GitHub [1], найпопулярнішими з них є: Python, C++, JavaScript, Java, C#.

Популярність Python для застосування в МН зумовлена наявністю великого набору бібліотек та фреймворків для аналізу даних, створення і впровадження моделей та алгоритмів МН. Такими бібліотеками є:

NumPy – це універсальна бібліотека, де можна з легкістю працювати з багатовимірними масивами, матрицями, робити операції з лінійної алгебри та числові перетворення. За даними GitHub, використовується майже в трьох чвертях проектів МН та наук про дані.

TensorFlow – фреймворк для МН, що має всеосяжну, гнучку екосистему інструментів, бібліотек і ресурсів, що дозволяє дослідникам пропонувати новітні технології штучного інтелекту, а розробникам створювати й розгортати програми, в основі яких лежать алгоритми машинного та глибокого навчання.

Pandas – це бібліотека, що забезпечує високопродуктивні, прості у використанні структури даних та інструменти аналізу даних для мови програмування Python.

Matplotlib – гнучка бібліотека для створення і візуалізації графіків.

Scikit-Learn – пакет для МН, великою перевагою якого є висока швидкість роботи, що реалізує широкий спектр алгоритмів МН та дозволяє використовувати цілу низку функцій, таких як регресія, кластеризація, вибір моделі, попередня обробка, класифікація та інше.

При створенні додатку для розпізнавання обличь постає питання про вибір платформи, на якій додаток буде працювати. Існує безліч платформ, наприклад, веб-додатки, які працюють через Інтернет, або класичні десктопні програми для відповідної операційної системи (ОС), що встановлюються, запускаються та працюють локально. Крім того, для створення додатку, що використовує МН, потрібно брати до уваги мову програмування, за

допомогою якого розроблена сама модель МН. В нашому випадку це Python, тому цю мову слід використовувати і для створення системи.

Веб-додатки [2], на відміну від десктопних програм:

- мають якісну кросплатформеність, адже не залежать від операційної системи;
- доступні повсюдно, з будь-якого пристрою при наявності інтернету;
- не вимагають попередньої підготовки середовища;
- дозволяють заощадити на розробці, часі та апаратних ресурсах розробників.

Серед усіх засобів будування веб-додатків на мові програмування Python [3], можна виділити лише декілька найбільш використовуваних фреймворків у розробці програмного забезпечення: Django та Flask.

Flask – фреймворк, який надає мінімальну основу для веб-додатку. В ньому немає вбудованої підтримки використання баз даних і вбудованої панелі адміністрування.

Django – це безкоштовний MVT (Model View Template) фреймворк, завдяки якому можна створювати як невеликі проєкти, так і складні веб-сайти. Завдяки використанню об'єктно-реляційного відображення [4] (ORM) для роботи з базами даних, Django може працювати з різними базами даних (Фласк тоже может): PostgreSQL, MySQL, SQLite та Oracle. При створенні додатка на Django, можна користуватись всіма бібліотеками, що використовуються в Python, без труднощів створювати нові моделі в базах даних, а за допомогою вбудованого шаблонізатора Jinja швидко створювати сторінки сайту. На відміну від Flask, в Django є вбудована сторінка адміністрування.

В якості фреймворка доцільно використовувати Django через швидкість розробки веб-додатків та підтримки бібліотек для МН. В якості бази даних – PostgreSQL, яка може зберігати будь-які типи даних, включаючи зображення.

Література

1. The State of the Octoverse: machine learning [Електроний ресурс] – Режим доступу: <https://github.blog/2019-01-24-the-state-of-the-octoverse-machine-learning/>
2. Десктопное или веб-приложение: плюсы и минусы [Електроний ресурс] – Режим доступу: <https://vc.ru/services/297762-desktopnoe-ili-veb-prilozhenie-plyusy-i-minusy>
3. Best Python Frameworks [Електроний ресурс] – Режим доступу: <https://hackr.io/blog/python-frameworks>
4. Object-relational mapping [Електроний ресурс] – Режим доступу: https://en.wikipedia.org/wiki/Object-relational_mapping

Анотація

Розглянуто питання вибору мови програмування, фреймворку та баз даних для розробки системи з розпізнавання обличь. Представлені вимоги до створюваної системи. Розглянуто особливості застосування деяких фреймворків та баз даних в створенні додатка з використанням машинного навчання.

Ключові слова: розпізнавання обличь, фреймворк, веб-додаток, машинне навчання.

Аннотация

Рассмотрены вопросы выбора языка программирования, фреймворка и баз данных для разработки системы распознавания лиц. Представлены требования к создаваемой системе. Рассмотрены особенности применения некоторых фреймворков и баз данных в создании приложения с использованием машинного обучения.

Ключевые слова: распознавание лиц, фреймворк, веб-приложение, машинное обучение.

Abstract

The issue of choosing a programming language, framework and databases for the development of a face recognition system is considered. The requirements to the created system are presented. Features of application of some frameworks and databases in creation of the application with use of machine learning are considered.

Keywords: face recognition, framework, web application, machine learning.