

B. V. Shefkin, I. V. Krasiuk, V. O. Khomenchuk, K. P. Storchak, A. M. Tushych

### RESEARCH AND IMPLEMENTATION OF NN BASED ON TENSORFLOW

TensorFlow is Google's open-source machine learning and deep learning framework, which is convenient and flexible to build the current mainstream deep learning model. Convolutional neural network is a classical model of deep learning, the advantage lies in its powerful feature extraction capabilities of convolutional blocks. A neural network in the simplest case is a mathematical model consisting of several layers of elements that perform parallel calculations. Initially, such an architecture was created by analogy with the small computing elements of the human brain — neurons. The minimal computing elements of an artificial neural network are also called neurons. Neural networks typically consist of three or more layers: an input layer, a hidden layer (or layers), and an output layer. An important feature of the neural network is its ability to learn by example, this is called learning with a teacher. The neural network is trained on a large number of examples consisting of input-output pairs (corresponding to each other input and output). In object recognition problems, such a pair will be the input image and the corresponding label — the name of the object. Neural network learning is an iterative process that reduces the deviation of the network output from a given «teacher response» — a label that corresponds to a given image. This process consists of steps called epochs of learning (they are usually calculated in thousands), each of which is the adjustment of the «weights» of the neural network — the parameters of the hidden layers of the network. Upon completion of the learning process, the quality of the neural network is usually good enough to perform the task for which it was trained, although the optimal set of parameters that perfectly recognizes all the images, it is often impossible to choose. Based on the TensorFlow platform, a convolutional neural network model with two-convolution-layers was built. The model was trained and tested with the MNIST data set. The test accuracy rate could reach 99,15%, and compared with the rate of 98,69% with only one-convolution-layer model, which shows that the two-convolution-layers convolutional neural network model has a better ability of feature extraction and classification decision-making.

**Keywords:** neural network; deep learning; convolution layers; TensorFlow.

УДК 004.432.2

DOI: 10.31673/2412-9070.2020.062628

Д. В. КРАВЕЦЬ, асистент;

А. М. ТУШИЧ, ст. викладач;

В. В. ШКАПА, доцент;

В. Р. МИКОЛАЙЧУК, ст. викладач,

Державний університет телекомунікацій, Київ

## ОГЛЯД СТАНДАРТНОЇ БІБЛІОТЕКИ ДЛЯ СТВОРЕННЯ GUI МОВОЮ Python

*Розглянуто стандартну бібліотеку для створення графічних інтерфейсів мовою Python. Було проаналізовано актуальність засобів створення графічних інтерфейсів, значення графічного інтерфейсу у сприйнятті користувачем програми, останні дослідження і публікації, що пов'язані з бібліотекою Tkinter. Досліджено історію розвитку бібліотеки, основні віджети та менеджери геометрії. Визначено всі функції, що відповідають за розміщення елементів графічного інтерфейсу у вікні та зроблено висновки щодо недоліків і переваг даної бібліотеки.*

**Ключові слова:** програмування; мова програмування; графічний інтерфейс; інтерфейс; вікно; віджет; бібліотека; менеджер геометрії; клас; функція; Python; Tkinter; GUI; Tk.

### ВСТУП

**Постановка проблеми.** На сучасному етапі розвитку інформаційних технологій пересічний користувач звик до графічних інтерфейсів і може мати складнощі у процесі роботи із суто текстовими інтерфейсами. Через це важливо знати не лише методи реалізації логіки програм, а й те, як зробити спілкування користувача з програмою максимально комфортним.

**Аналіз останніх досліджень і публікацій.** Стандартна бібліотека для створення графічних інтерфейсів мовою Python тісно пов'язана з бібліотекою Tk, що відома давно та багато де описана. Напри-

клад праця [1] досі має попит в інтернет-магазині amazon.com, незважаючи на те, що її було написано 2000 року. Проте час швидкоплинний, як і розвиток мови програмування Python. Говорячи про використання новіших версій бібліотеки Tkinter, добре згадати про підручник [2]. Однак найактуальнішим джерелом інформації завжди залишатиметься документація бібліотеки [3], що викладена на сайті документації мови програмування Python.

**Мета статті.** Метою статті є огляд стандартної бібліотеки для створення графічних інтерфейсів мовою Python.

© Д. В. Кравець, А. М. Тушич, В. В. Шкапа, В. Р. Миколайчук, 2020

**ОСНОВНА ЧАСТИНА****Бібліотека Tkinter**

Tkinter — багатоплатформна графічна бібліотека інтерфейсів на основі засобів Tk (широко популяризована у світі GNU/Linux та інших UNIX подібних систем, портована зокрема і на Microsoft Windows, Apple Mac OS), поширювана з відкритими вихідними кодами, яку написали Стін Лумхольт і Гвідо ван Россум і яка належить до стандартної бібліотеки Python.

Бібліотека Tkinter не реалізує власний інтерфейс до бібліотеки Tk, а забезпечує конвертування звернень Python у звернення Tcl — мови, яка тісно інтегрована з Tk. Отже, Tkinter є так званою обгорткою для Tcl/Tk [3].

Пакет tkinter — це тонкий об'єктно-орієнтований шар поверх Tcl/Tk. Щоб використовувати tkinter, вам не потрібно писати код Tcl, але вам доведеться звернутися до документації Tk, а іноді і до документації Tcl. Tkinter — це набір обгортки, які реалізують віджети Tk як класи Python. Крім того, внутрішній модуль `_tkinter` забезпечує безпечний механізм, який дає можливість взаємодіяти Python і Tcl.

Tk — крос-платформенна бібліотека базових елементів графічного інтерфейсу, поширювана з відкритими вихідними кодами. Tk було розроблено Джоном Остерхутом як розширення для інтерпретованої мови програмування Tcl. Tk може використовуватись із C/C++, Perl, Python, Ruby (в окремих випадках може знадобитись Tcl).

Компонент Tk вже давно є невід'ємною частиною Python. Він забезпечує надійний і незалежний від платформи набір інструментів для вікон, який доступний програмістам Python за допомогою пакета tkinter та його розширення, модулів `tkinter.tix` та `tkinter.ttk`. Tk надає розробнику набір Tcl-команд, призначених для створення візуальних компонентів і виконання різних дій із ними. Компонент являє собою вікно у складі графічного інтерфейсу, що має певний зовнішній вигляд і виконує визначені функції [4].

**Елементи вікна в Tkinter**

Головним елементом GUI Tkinter є вікно. Вікнами називають контейнери, в яких розміщено всі GUI елементи. Дані GUI елементи, до яких належать текстові бокси, ярлики і кнопки, називаються віджетами. Віджети містяться всередині вікон.

Віджети є основою GUI фреймворка Tkinter в Python. Це елементи, через які користувачі взаємодіють із програмою. У Tkinter кожний віджет визначено класом.

Віджети Label використовуються для відображення тексту або картинок. Текст на віджеті Label не може редагуватися користувачем, він тільки показується.

Віджети Button потрібні для створення клікабельних кнопок. Їх можна налаштувати в такий спосіб, щоб під час натискання викликала певна функція. Існує багато подібностей між віджетами Button і Label. За своєю суттю кнопка — це просто ярлик, на який можна клікнути. Для створення стилів віджетів ярлика і кнопки також використовуються однакові аргументи [5].

У разі, коли потрібно отримати текстову інформацію від користувача, на кшталт адреси електронної пошти, застосовують віджет Entry. Він відображає невеликий текстовий бокс, куди користувач може ввести текст.

Віджети Text використовуються для введення тексту, як і віджети Entry. Різниця полягає лише в тому, що Text може містити кілька рядків тексту. З віджетом Text користувач може вводити цілі параграфи або сторінки тексту.

Checkbutton — це віджет, який дає змогу позначити «галочкою» певний пункт у вікні. Якщо використовуються кілька пунктів, потрібно кожному з них присвоїти свою змінну.

Віджет Radiobutton виконує функцію, схожу з функцією віджета Checkbutton, але у віджеті Radiobutton користувач може вибрати лише один із пунктів.

Віджет Progressbar відображає рівень завантаження, процес збільшення змінної тощо.

**Менеджери геометрії**

Існує три функції для розміщення віджетів у вікні: `pack()`, `grid()` та `place()`.

Функція `pack()` автоматично розміщує віджети в батьківському вікні. Це найпростіша для роботи функція, однак вона не дає свободи розміщення елементів — усі віджети, що додаються, будуть додаватися по чергову у перший рядок вікна, а коли у вікні стане замало місця для додавання елемента — він з'явиться на початку другого рядка і т.д.

Для невеликої кількості віджетів у вікні це обмеження можна обійти за допомогою певних значень параметра `side` (`left`, `right`, `top`, `bottom`, `both`) для розміщення відповідно ліворуч, праворуч, у верхній, нижній або центральній частинах вікна). Однак для великої кількості елементів, які мають бути розміщені не порядково, а довільно чи у стовпчик — ця функція не підходить.

Функція `grid()` розміщує віджети на сітці. Основні параметри: `row` (рядок), `column` (стовпець), `rowspan` (скільки рядків займає віджет), `columnspan` (скільки стовпців займає віджет). Ця функція дає більшу свободу в розміщенні елементів всередині вікна.

Функція `place()` уможливорює розміщення віджетів у зазначених координатах із зазначеними розмірами. Основні параметри: `x` (координата верхньої межі віджета ліворуч за віссю абсцис),

у (координата верхньої межі віджета ліворуч за віссю ординат), width (ширина елемента), height (висота елемента).

#### Висновки

Головні переваги Tkinter полягають у тому, що він швидкий і зазвичай постачається в комплекті з Python. Tkinter також відомий тим, що має застарілий зовнішній вигляд, що значно покращено в Tk 8.5.

Проте щодо використання, Tkinter є відносно простим на противагу іншим бібліотекам. Це відмінний вибір для створення GUI додатків у Python, особливо якщо сучасний вигляд не в пріоритеті для програми, а велику роль відіграє функціональність і крос-платформенна швидкість.

#### Список використаної літератури

1. Grayson J. *Python and Tkinter Programming*. 2000. 688 p.
2. Програмування числових методів мовою Python: підруч. / А. В. Анісімов, А. Ю. Дорошенко, С. Д. Погорілий, Я. Ю. Дорогий; за ред. А. В. Анісімова. Київ: Видавничо-поліграфічний центр «Київський університет», 2014. 640 с.
3. Tkinter — Python interface to Tcl/Tk [Електронний ресурс] // Python Documentation. 2020. URL: <https://docs.python.org/3/library/tkinter.html>.
4. Tk [Електронний ресурс] // Wikipedia. 2016. URL: <https://uk.wikipedia.org/wiki/Tk>.
5. Tkinter.ttk — Tk themed widgets [Електронний ресурс] // Python Documentation. 2020. URL: <https://docs.python.org/3/library/tkinter.ttk.html#ttk-widgets>.

Д. В. Кравец, А. Н. Тушич, В. В. Шкапа, В. Р. Миколайчук

#### ОБЗОР СТАНДАРТНОЙ БИБЛИОТЕКИ ДЛЯ СОЗДАНИЯ GUI НА ЯЗЫКЕ PYTHON

Рассмотрена стандартная библиотека для создания графических интерфейсов на языке Python. Были проанализированы актуальность средств создания графических интерфейсов, значение графического интерфейса в восприятии пользователем программы, последние исследования и публикации, связанные с библиотекой Tkinter. Исследована история развития библиотеки, основные виджеты и менеджеры геометрии. Рассмотрены все функции, отвечающие за размещение элементов графического интерфейса в окне и сделаны выводы относительно недостатков и преимуществ данной библиотеки.

**Ключевые слова:** программирование; язык программирования; графический интерфейс; интерфейс; окно; виджет; библиотека; менеджер геометрии; класс; функция; Python; Tkinter; GUI; Tk.

D. Kravets, A. Tushych, V. Shkapa, V. Mykolaichuk

#### OVERVIEW OF THE STANDARD LIBRARY FOR CREATING A GUI IN PYTHON

This article discusses the standard library for creating graphical user interfaces in Python. The relevance of graphical user interface creation tools, the value of the graphical user interface, the latest research and publications related to the Tkinter library, the history of the library, the main widgets and geometry managers were analyzed. All the functions responsible for placing the elements of the graphical interface in the window are considered. There are three functions for placing widgets in the window — pack (), grid () and place (). Conclusions are drawn about the disadvantages and advantages of this library.

The standard Python GUI library is closely related to the long-known and widely described Tk library, but the most relevant source of information will always be the library documentation on the Python programming language documentation site.

The purpose of this article is to review the standard library for creating graphical interfaces in Python.

Tkinter is a multi-platform graphical interface library based on Tk tools (widespread in the world of GNU/Linux and other UNIX-like systems, ported to Microsoft Windows, Apple Mac OS), distributed with open source, written by Wall Lumholt and Guido van Rossum. Included in the standard Python library. Tk has long been an integral part of Python. It provides a robust and platform-independent set of tools for windows that is available to Python programmers. However, in terms of usage, Tkinter is relatively easy compared to other libraries. This is a great choice for creating GUI applications in Python, especially if the modern look is not a priority for the program, and a big role is played by functionality and cross-platform speed.

**Keywords:** programming; programming language; graphical user interfaces; interface; window; widget; library; geometry manager; class; function; Python; Tkinter; GUI; Tk.

