СЕКЦІЯ КОМП'ЮТЕРНІ НАУКИ

УДК 004.62

Золотуха Роман Андрійович
доктор філософії, старший викладач кафедри інформаційних систем і технологій, Національний університет біоресурсів і природокористування України
ORCID: https://orcid.org/0000-0003-3099-722X
E-mail: r.zolotukha@nubip.edu.ua

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ АВТОМАТИЗАЦІЇ ОБРОБКИ РЕЗЮМЕ КАНДИДАТІВ ДЛЯ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ ПРОЦЕСУ ФОРМУВАННЯ ІТ-КОМАНД

Анотація. *Стаття присвячена розробці інформаційної технології автоматизованої обробки резюме кандидатів у форматі PDF з використанням мови програмування Python. Представлено підхід до вилучення, структуризації та подальшого аналізу даних із застосуванням бібліотек pdfplumber, spaCy та pandas. Запропонований модуль дозволяє визначати ключові елементи резюме, зокрема освіту, навички, контактну інформацію та досвід роботи, з подальшим формуванням структурованих даних у форматі JSON. Особливу увагу приділено забезпеченню універсальності алгоритму для резюме з довільною структурою та україномовним контентом. У роботі розглянуто основні етапи реалізації програмного рішення, наведено діаграми потоків даних, схеми обробки PDF-файлів та приклади юніт-тестування функцій системи. Розроблена технологія може бути використана для автоматизації первинного етапу рекрутингу та інтеграції з HR-аналітичними системами, що підвищує точність і швидкість обробки кандидатських даних у процесі формування IT-команд.*

Ключові слова: *інформаційна технологія, Python, PDF-резюме, автоматизація рекрутингу, spaCy, NLP, HR-система.*

Актуальність. У сучасних умовах динамічного розвитку IT-галузі питання ефективного підбору персоналу набуває стратегічного значення. Різке збільшення кількості кандидатів на одну вакансію після 2022 року [1], зростання конкуренції та дефіцит висококваліфікованих спеціалістів зумовлюють необхідність підвищення швидкості та точності процесу відбору. Зокрема, автоматизація первинної обробки резюме кандидатів дозволяє зменшити навантаження на HR-відділи та знизити ризик суб'єктивності при прийнятті рішень.

Однією з ключових проблем сучасних рекрутингових систем є відсутність інструментів для якісного вилучення та аналізу даних з резюме у форматі PDF, особливо тих, що складені українською або двомовною структурою. Більшість існуючих платформ, таких як LinkedIn Recruiter чи Greenhouse, орієнтовані на англомовний ринок і не враховують специфіку локальних форматів документів, структури резюме та різноманіття форматування. Це ускладнює інтеграцію таких документів у бази даних HR-систем та подальший аналітичний обробіток.

Мета дослідження полягає у розробці інформаційної технології автоматизованої обробки резюме кандидатів у форматі PDF, що забезпечує вилучення, структуризацію та подальший аналіз даних із використанням бібліотек Python, з метою підвищення ефективності процесу підбору персоналу та формування IT-команд.

Аналіз останніх досліджень та публікацій. У численних роботах останніх років показано важливість автоматизації обробки документів та витягування структурованої інформації із PDF-форматів. У роботі Chafiq N., Ghazouani M. та El Gounidi R. [2] запропоновано систему автоматизованої обробки резюме для вступу до магістерських програм, яка ґрунтується на використанні методів обробки природної мови (NLP). Автори застосували попередньо натреновані моделі spaCy та Hugging Face Transformers для розпізнавання сутностей (Named Entity Recognition — NER) і вилучення таких ключових елементів, як освіта, досвід та навички кандидатів. Додатково реалізовано двоетапне узагальнення тексту резюме — екстрактивне (на основі моделей BERT) та абстрактивне (з використанням мовних моделей LLAMA). Система продемонструвала високу ефективність,

досягнувши точності NER 82 % та середнього часу обробки одного резюме 3,84 секунди. Ця робота показує перспективність поєднання класичних NLP-підходів із сучасними трансформерними архітектурами для обробки великих масивів документів у форматі PDF.

Дослідження Sandanayake T. C., Limesha G. A. I., Madhumali T. S. S., Mihirani W. P. I. та Peiris M. S. A. [3] зосереджене на автоматичному аналізі та ранжуванні резюме кандидатів для підбору персоналу в IT-сфері. Запропонований авторами інструмент витягує релевантну інформацію з неструктурованих текстів резюме та формує рейтинг кандидатів відповідно до заданих критеріїв. Особливістю цього підходу є інтеграція зовнішніх джерел даних, таких як Stack Overflow, GitHub та професійні блоги, для створення повного профілю кандидата. Розроблена система орієнтована на вакансії в галузі інформаційних технологій, що дозволяє значно зменшити час ручного перегляду резюме та підвищити точність відбору.

У роботі Ahmed F., Anannya M., Rahman T. та Khan R. T. [4] розглянуто можливість поєднання автоматизованої обробки резюме з психометричним аналізом у процесі підбору кадрів. Автори запропонували концепцію соціальної мережі для шукачів роботи та роботодавців, яка автоматично зіставляє кандидатів із вакансіями за заданими критеріями. Система враховує результати психометричних тестів для визначення відповідності особистісних якостей кандидата вимогам компанії, що, на думку дослідників, підвищує точність відбору й рівень задоволеності працівників після працевлаштування.

Ben Azzou K. та Talei H. [5] запропонували машинно-навчальний підхід до автоматизованого аналізу даних резюме та визначення профілю кандидата. Розроблена ними система використовує методи NLP для вилучення з текстів резюме структурованих даних про освіту, досвід та навички, після чого застосовує алгоритми класифікації для автоматичного зіставлення кандидатів із відповідними посадами. Автори підкреслюють, що такий підхід дає змогу істотно скоротити навантаження на HR-відділи та знизити суб'єктивність відбору завдяки стандартизованій оцінці текстових даних.

Проведений аналіз свідчить, що сучасні дослідження у сфері автоматизації обробки резюме орієнтовані на поєднання методів обробки природної мови, машинного навчання та інтеграції зовнішніх джерел даних. Спільною тенденцією є прагнення зменшити трудомісткість і суб'єктивність процесу відбору кандидатів за рахунок впровадження інтелектуальних алгоритмів, здатних опрацьовувати великі обсяги неструктурованої інформації. Разом із тим більшість розробок сфокусовані на англомовному ринку, що зумовлює актуальність дослідження систем, адаптованих до україномовних резюме у форматі PDF.

Матеріали і методи дослідження. Для тестування алгоритмів сортування та демонстрації проблеми локалізації важливо мати реалістичний набір тестових даних. Вибірку досліджуваних склали студенти 1 курсу факультету інформаційних технологій Національного університету біоресурсів і природокористування України. До експерименту було залучено 4 групи студентів ІПЗ-2301, ІПЗ-2302, ІСТ-2301, ІСТ-2302 (Рисунок 1). За планом експерименту для застосування алгоритму - кожен учасник дослідження надіслав своє резюме у PDF форматі для подальшої його обробки та формування команд. Для чистоти експерименту резюме були заповнені у вільному форматі без запропонованого шаблону.

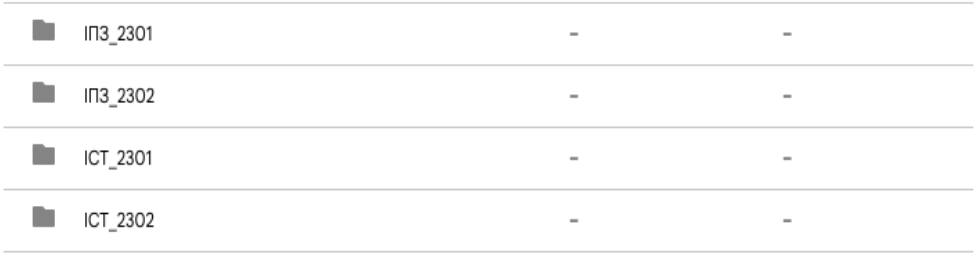
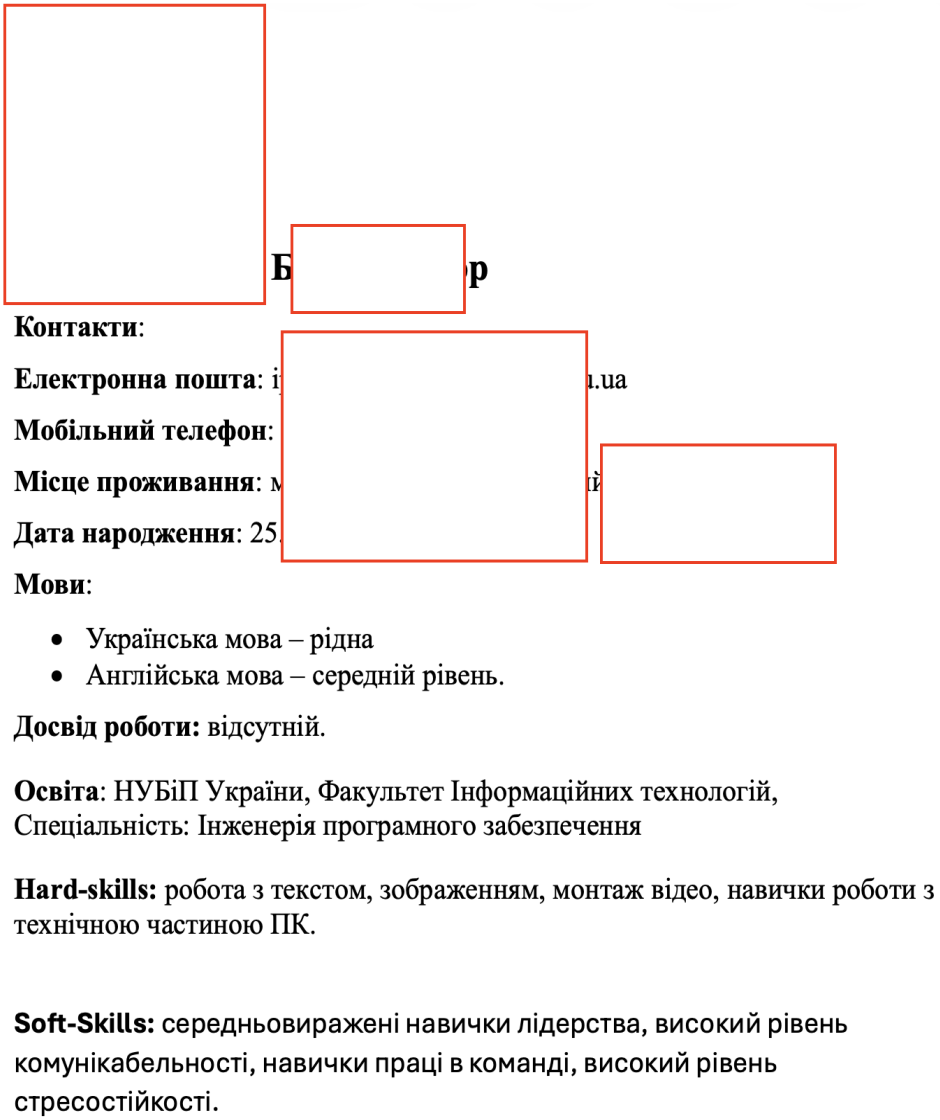


Рисунок 1 – ZIP-архів з резюме учасників експерименту.

На рис. 2 продемонстровано резюме у PDF форматі одного з учасників експерименту. Усі поля з персональними даними навмисне зафарбовані для збереження конфіденційності даних.



Б р

Контакти:

Електронна пошта: і .ua

Мобільний телефон:

Місце проживання: м й

Дата народження: 25.

Мови:

- Українська мова – рідна
- Англійська мова – середній рівень.

Досвід роботи: відсутній.

Освіта: НУБіП України, Факультет Інформаційних технологій,
Спеціальність: Інженерія програмного забезпечення

Hard-skills: робота з текстом, зображенням, монтаж відео, навички роботи з технічною частиною ПК.

Soft-Skills: середньовиражені навички лідерства, високий рівень комунікабельності, навички праці в команді, високий рівень стресостійкості.

Рисунок 2 – Приклад резюме у форматі PDF з тестової вибірки.

Результати дослідження та їх обговорення. Для розробки модулю була розроблена модель потоків даних процесу подачі резюме кандидатом (рисунок 3). Кандидат подає заявку на вакансію, прикріплюючи резюме у форматі PDF. Резюме обробляється системою і виділяє ключову інформацію про кандидата: контактну інформацію, місце навчання та навички. Ця інформація структурується і вноситься в базу даних HR, де HR-спеціаліст може бачити потрібних кандидатів, використовуючи задані фільтри.

Визначені вимоги до інформаційної технології обробки даних з PDF резюме кандидатів відображено на діаграмі варіантів використання (рис. 4).

Для реалізації даного алгоритму ми використали універсальні можливості мови програмування Python. Надійна екосистема бібліотек Python полегшила наші зусилля в розборі та вилученні релевантної інформації зі складної структури резюме кандидатів.

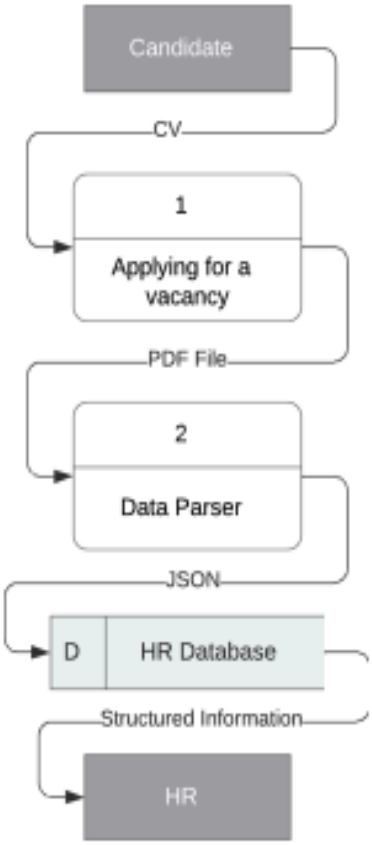


Рисунок 3 – Схема руху даних від кандидата до HR

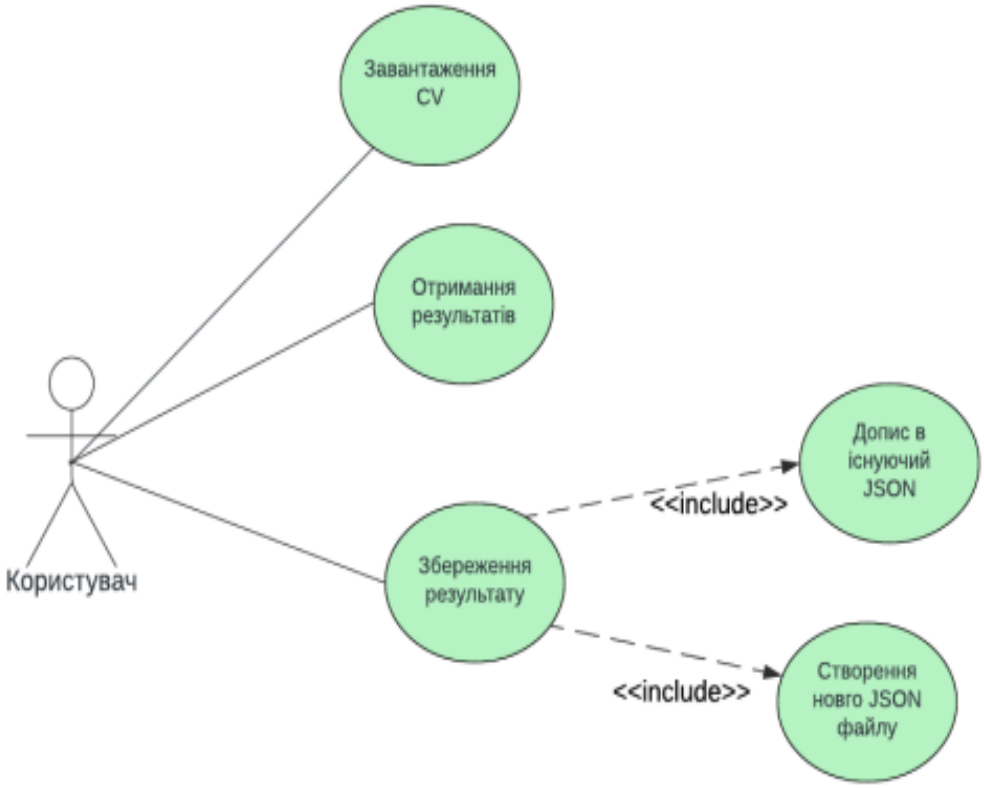


Рисунок 4 – Діаграма варіантів використання для користувача

Щоб розібратися в тонкощах PDF-документів, ми використовували бібліотеку "PDFplumber", яка дозволила нам точно витягувати текстовий контент. Ця бібліотека надає можливість переглядати макет PDF-резюме та виокремлювати необхідні текстові сегменти для подальшого аналізу. Для обробки природної мови та розпізнавання текстових шаблонів ми використовували бібліотеку "spacy". Цей потужний інструмент НЛП дозволив токенізувати, тегувати та аналізувати текст, що дало нам змогу виявити шаблони та сутності, важливі для виокремлення навичок та освіти. Модуль "Matcher" у складі "spacy" допоміг виявити конкретні лінгвістичні патерни, впорядкувавши наш процес визначення ключової інформації. Щоб полегшити організацію та зберігання наших результатів, ми використали модуль "csv" для створення та управління структурованими наборами даних. Бібліотека "pandas" запропонувала нам ефективний засіб для маніпулювання та аналізу цих наборів даних, що дозволило нам отримати уявлення та тенденції з отриманої інформації. Як невід'ємну частину нашої реалізації ми використали можливості вбудованої бібліотеки Python під назвою "json".

Запропонований нами процес перетворення PDF документу, що містить інформацію про кандидата в JSON файл готовий до подальшого аналізу можна візуалізувати наступним чином. (рис. 5).

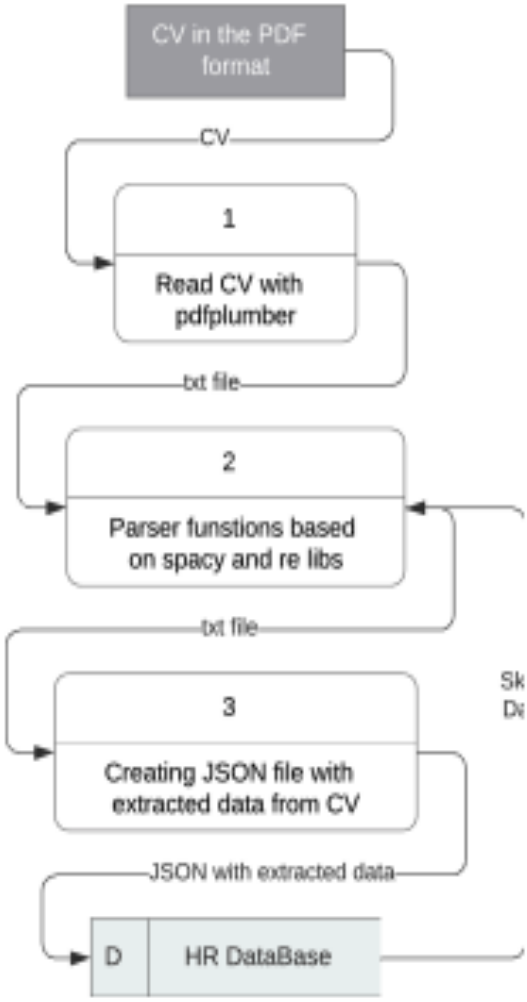


Рисунок 5 – Схема реалізації модулю обробки резюме в PDF форматі

Результатом роботи нашого алгоритму є ретельно структурований JSON-файл, зображений на рисунку нижче (рис. 6).

```python
if __name__ == '__main__':
    resume_text = extracted_text

    name = extract_name(resume_text)
    if name:
        print("Name:", name)
    else:
        print("Name not found")

    contact_number = extract_contact_number_from_resume(resume_text)
    if contact_number:
        print("Contact Number:", contact_number)
    else:
        print("Contact Number not found")

    email = extract_email_from_resume(resume_text)
    if email:
        print("Email:", email)
    else:
        print("Email not found")

    skills_list = csv_data_list
    extracted_skills = extract_skills_from_resume(resume_text, skills_list)
    if extracted_skills:
        print("Skills:", extracted_skills)
    else:
        print("No skills found")

    extracted_education = extract_education_from_resume(resume_text)
    if extracted_education:
        print("Education:", extracted_education)
    else:
        print("No education information found")
```

```
Name:
Contac
Email:
Skills: [redacted] p', 'api', 'mailchimp', 'economics', 'gmail', 'automation', 'pytho
n', 'subscribe', 'analytics', 'google analytics', 'firebase', '2014', 'communication', 'improvement', '.com', 'engl
ish', 'travel', 'british', 'data collection', 'web', 'digital', 'navigation', 'com', 'facebook', 'russian', 'analyt
```

Рисунок 6 – Реалізований модуль в середовищі Jupyter Notebook

Для технічної валідація реалізованого рішення було підготовлено серія unit-тестів. На рис. 7 наведений код для тестування однієї з функції інформаційної технології – обробки тексту з PDF файлів у модулі обробки PDF резюме кандидатів.

```python
import unittest
import pdfplumber
from io import BytesIO

def extract_text_from_pdf(pdf_path):
    with pdfplumber.open(pdf_path) as pdf:
        full_text = ""
        for page in pdf.pages:
            text = page.extract_text()
            full_text += text
    return full_text

class TestPDFExtraction(unittest.TestCase):

    def setUp(self):
        self.pdf_content = b'%PDF-1.4\n1 0 obj\n<< /Type /Catalog /Pages 2 0
        self.pdf_path = "test.pdf"
        with open(self.pdf_path, "wb") as f:
            f.write(self.pdf_content)

    def tearDown(self):
        import os
        os.remove(self.pdf_path)

    def test_extract_text(self):
        expected_text = "Hello, world!\n"
        extracted_text = extract_text_from_pdf(self.pdf_path)
        self.assertEqual(extracted_text.strip(), expected_text.strip())

if __name__ == "__main__":
    unittest.main()
```

Рисунок 7 – Юніт-тест для функції обробки тексту з PDF-файлу

Даний тест складається з функції extract_text_from_pdf, яка витягує текст з PDF-файлу, та тестового класу TestPDFExtraction, який перевіряє правильність роботи цієї функції. Цей тест складається з 4 етапів: створення тимчасового PDF-файлу; обробка тексту з PDF-файлу за допомогою функції extract_text_from_pdf; перевірка тексту на відповідність очікуваному тексту; видалення тимчасового PDF-файлу після завершення тестування. За допомогою юніт-тестування для модуля обробки резюме у форматі PDF вдалось верифікувати функції: обробки тексту з PDF-файлів; функцій, що відповідають за обробку тексту за ключовими словами в резюме кандидатів; функцію збереження структурованих даних у форматі JSON; стандартизацію структури збережених даних.

Висновок. За допомогою реалізованої інформаційної технології автоматизації обробки резюме кандидатів ми отримали поля які нас цікавлять у резюме: ПІБ, контактний номер телефону, контактний Email кандидата, навички кандидата, освіта кандидата. Алгоритм обробив 98 резюме у довільному форматі і показав ефективність у 96%. Як показало дослідження, для реалізованого алгоритму не потрібно готувати конкретний шаблон резюме. Проте, було виявлено і ряд обмежень, зокрема, коли кандидати використовувати англіцизми українською. В цьому контексті додавання слів виключень би стало одним із варіантів покращення даного алгоритму. Отриману інформацію з CV в подальшому можна імпортувати в JSON форматі у зручне сховище HR-бази, яке іт-компанії використовують під час процесу рекрутингу.

Список використаних джерел

1. Золотуха Р.А., Глазунова О.Г. Прогнозування розвитку ринку праці в ІТ галузі України методом часових рядів. Interdisciplinary research: scientific horizons and perspectives: VI International Scientific and Theoretical Conference, м. Вільнюс, Литва, 13 жовтня 2023 року: тези доповіді. Вільнюс, 2023. С. 31–36.
2. Chafiq N., Ghazouani M., El Gounidi R. From Manual Review to AI Automation: An NLP-Powered System for Efficient CV Processing in Academic Admissions. LatIA, 2025; № 3: 315. DOI: 10.62486/latia2025315.
3. Sandanayake T.C., Limesha G. A.I., Madhumali T.S.S., Mihirani W.P.I., Peiris M.S.A. Automated CV Analyzing and Ranking Tool to Select Candidates for Job Positions. Proceedings of the 2020 ACM/IEEE International Conference on Automated Software Engineering. ACM, 2020. DOI: 10.1145/3301551.3301579.
4. Ahmed F., Anannya M., Rahman T., Khan R.T. Automated CV processing along with psychometric analysis in job recruiting process. Proceedings of the 2015 International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), Savar, Bangladesh, 21–23 May 2015. IEEE, 2015. DOI: 10.1109/ICEEICT.2015.7307521.
5. Ben Azzou K., Talei H. A Machine Learning Approach for Automated CV Data Analysis and Job Profile Identification. Proceedings of the 2024 Sixth International Conference on Intelligent Computing in Data Sciences (ICDS), Marrakech, Morocco, 23–24 October 2024. IEEE, 2024. DOI: 10.1109/ICDS62089.2024.10756435.

Zolotukha Roman
PhD, Senior Lecturer, Department of Information Systems and Technologies,
National University of Life and Environmental Sciences of Ukraine
ORCID: https://orcid.org/0000-0003-3099-722X
E-mail: r.zolotukha@nubip.edu.ua

INFORMATION TECHNOLOGIES FOR AUTOMATING THE PROCESSING OF CANDIDATE CV TO INCREASE THE EFFICIENCY OF IT TEAM FORMATION

Abstract. *The article is devoted to the development of an information technology for automated processing of candidate CV in PDF format using the Python programming language. The approach to extracting, structuring, and*

further analyzing data with the use of the pdfplumber, spaCy, and pandas libraries is presented. The proposed module enables the identification of key resume elements, including education, skills, contact information, and work experience, followed by the formation of structured data in JSON format. Special attention is given to ensuring the universality of the algorithm for CV with arbitrary structure and Ukrainian-language content. The paper describes the main stages of implementing the software solution, including data flow diagrams, PDF processing schemes, and examples of unit testing of system functions. The developed technology can be used to automate the initial stage of recruitment and integrate with HR analytics systems, thereby improving the accuracy and speed of candidate data processing in the IT team formation process.

Keywords: *information technology, Python, CV, recruitment automation, spaCy, NLP, HR system.*