

РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ІНТЕЛЕКТУАЛЬНОЇ КЛАСТЕРИЗАЦІЇ НЕСТРУКТУРОВАНИХ ФАЙЛОВИХ МАСИВІВ

Сучасний стан інформаційних технологій характеризується парадоксом: попри досконалість систем управління базами даних (СУБД), вони залишаються ефективними лише за умови високої якості, повноти та уніфікованості вхідної інформації. У реальній практиці, особливо в сегменті малого бізнесу та індивідуального використання, дані часто існують у вигляді неструктурованих масивів файлів із нечіткою логікою найменування та гетерогенним вмістом. Класичні підходи до організації таких даних вимагають значних інтелектуальних та часових витрат на етапі препроцесингу, що робить стандартні цифрові інструменти малоефективними для пересічного користувача.

Головною науково-практичною проблемою є дефіцит доступних рішень, здатних систематизувати неповні дані та виконувати операцію «зворотного структурування» – перетворення розрізнених фрагментів інформації у цілісну логічну схему. Наявні комерційні продукти зазвичай є вузькоспеціалізованими або вимагають високої кваліфікації персоналу, що залишає нішу простої автоматизації «брудних» даних відкритою.

На основі цього було розроблено універсальну систему первинного сортування та групування інформації. На сьогодні на ринку практично відсутні комерційні рішення для ефективної систематизації неповних та гетерогенних даних. Спостерігається дефіцит загальнодоступних алгоритмів автоматичного зниження складності: якщо візуалізація структурованого контенту у схемі є тривіальним завданням, то зворотна операція – структурування хаотичних масивів – залишається трудомістким процесом. Більшість наявних програмних продуктів є вузькоспеціалізованими та унітарними. З огляду на це, актуальною є розробка універсальної системи первинного сортування масивів для оперативного пошуку необхідних знань. Запропонований додаток орієнтований на індивідуальне використання, не потребує спеціальної підготовки користувача та має інтуїтивно зрозумілий інтерфейс. За своєю суттю інструмент надає можливість доступної автоматизації при роботі зі слабоструктурованим контентом. Функціонал системи базується на використанні методів інтелектуального аналізу даних, а саме, ієрархічних (Agglomerative, Birch) та щільнісних (DBSCAN, HDBSCAN) [1,2]. В автоматичному режимі програма формує каталог наявних файлів та здійснює витяг їхнього вмісту (первинна машинна

обробка). Сформована адміністрована база дозволяє групувати інформаційні ресурси для швидкого опрацювання без втрат. Гнучкість ідентифікації забезпечується набором фільтрів, а можливість створення користувацьких макросів дозволяє автоматизувати рутинні операції. Це дає змогу швидко класифікувати об'єкти за контекстом, ігноруючи точність написання назв, що критично для масивів, де стандартна логіка є недоцільною.

```
DB_EXT= ['.accdb','.db', '.sqlite','.mdb', '.sql']
TXT_EXT= ['.txt','.log', '.ini','.json', '.xml','.bat', '.conf','.cfg']
DOC_EXT= ['.docx','.odt', '.rtf','.doc']
TBL_EXT= ['.xlsx','.xls', '.csv','.ods', '.xslm']
IMG_EXT= ['.jpg','.jpeg', '.png','.gif', '.bmp','.mp4', '.avi','.mkv', '.mov']
ARCHIVE_EXTENSIONS= ['.zip','.rar', '.7z','.tar', '.gz','.bz2']
```

Рисунок 1 – Приклади файлових форматів

Для програмної реалізації системи було обрано мову Python, що обумовлено її високою синтаксичною лаконічністю та наявністю розгалуженої екосистеми спеціалізованих бібліотек для інтелектуального аналізу даних і розробки користувацьких інтерфейсів [3,4]. Ефективність рішення було протестовано на фрагменті реальної бази обсягом 400 МБ, що містить понад 650 файлів. Через специфіку контенту традиційні методи пошуку виявилися неефективними: повний перебір займає 15–20 хвилин, а мовні бібліотеки за умови повного словника покривають лише 45–55% запитів. Запропонований додаток дозволяє проводити багаторівневу обробку запитів у реальному часі. Програмний продукт реалізовано у форматі десктопного додатка, здатного працювати з поширеними форматами файлів, що мають нестандартні назви та нечітку структуру вмісту.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Real Python: Python Tutorials and Guides. URL: <https://realpython.com>. (дата звернення : 10.04.2026).
2. TkDocs: Comprehensive Tkinter documentation. URL: <https://tkdocs.com>. (дата звернення : 10.04.2026).
3. Васильєв О.Ю. Програмування мовою Python. Київ : Навчальна книга – Богдан, 2023. 504 с.
4. Офіційна документація Python. URL: <https://docs.python.org>. (дата звернення : 10.04.2026).