

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ
СІКОРСЬКОГО”**

Факультет прикладної математики

Кафедра програмного забезпечення комп’ютерних систем

КУРСОВИЙ ПРОЕКТ

з дисципліни “Бази даних”

спеціальність 121 – Програмна інженерія

на тему:

Моніторингова система терористичних актів у світі

Студент

групи КП-71

Крючек Євгеній Іванович

(підпис)

Викладач

к.т.н, доцент кафедри

СПіСКС

Петрашенко А.В.

(підпис)

Захищено з оцінкою _____

Київ – 2020

Анотація

Метою розробки даного курсового проекту є набуття практичних навичок розробки сучасного програмного забезпечення, що взаємодіє з постріляційними базами даних, а також здобуття навичок оформлення відповідного текстового, програмного та ілюстративного матеріалу у формі проектної документації. У результаті виконання курсового проекту було опановано навик розробляти програмне забезпечення для постріляційних баз даних, володіння основами використання СУБД, а також інструментальними засобами аналізу великих обсягів даних.

Темою даного курсового проекту є створення ПЗ для моніторингової системи для виявлення закономірності у характеристиках терористичних актів у світі. У документі викладена актуальність та проблематика аналізу великого обсягу даних, аналіз використаного інструментарію (опис мови програмування, використаних бібліотек та СКБД), описана структура бази даних, опис розробленого програмного забезпечення (загальний, опис модулів та основних алгоритмів роботи), аналіз функціонування засобів масштабування, та опис результатів проведеного аналізу.

Результатами даного проекту стали діаграми та графіки, що зображають статистику кількості терористичних актів по рокам, кількість актів по країнам та регіонам, оцінка співвідношення вдалих/невдалих актів та актів, які були/не були спробою самогубства та закономірності розподілу типів жертв та використаної зброї.. З даними діаграмами та графіками можна ознайомитися в додатку А.

Зміст

Анотація	1
Зміст	2
Вступ	3
Аналіз інструментарію для виконання курсового проект	5
Аналіз СКБД	5
Обґрунтування вибору мови програмування	9
Обґрунтування вибору бібліотек і фреймворків	10
Структура бази даних	12
Опис результатів аналізу предметної галузі	21
Висновки	23
Література	25
Додаток А	27

Вступ

Станом та поточний рік все більше і більше компаній приходять до того, що дані, якими вони оперують та збирають в процесі своєї роботи потребують агрегації та аналізу. Використання класичних методів аналізу, що базуються на звичайних алгоритмах математичної статистики та ймовірності не проносять очікуваної користі. З плином часу все більшу і більшу актуальність набирають програмні засоби аналізу даних, які інколи базуються на алгоритмах машинного навчання, що дозволяє зробити аналіз даних інтелектуальним (*en. data minning*) Цю науку, яка поєднує в собі програмування, мат. статистику, теорію ймовірності та машинне навчання називаються наукою про дані (*en. Data Science*).

Сучасний Data Science спеціаліст (дослідник даних) має оперувати великими обсягами даних (від одного гігабайта до декількох терабайт) та вміти виокремити з них приховані залежності, на основі яких зробити прогноз про те, як будуть надалі відбуватися ті чи інші явища. Дослідники даних використовують свої дані та аналітичні здібності для пошуку та інтерпретації великих джерел даних; керують великими обсягами даних безвідносно до апаратного та програмного забезпечення і обмежень пропускної здатності; об'єднують джерела даних; забезпечують цілісність наборів даних; створюють візуалізації для кращого розуміння даних; з використанням даних будують математичні моделі; надають тлумачення даних та висновки.

Класичним набором інструментів, яким користується Data Science спеціаліст є мова програмування Python 3 і набір математичних бібліотек до неї (pandas, scikit-learn, numpy, matplotlib), нереляційні бази даних (на кшталт MongoDB, DynamoDB, Cassandra) та закони математичного аналізу, теорії ймовірності та мат. статистики).

Метою створення даного проекту був аналіз даних про закономірності в терористичних актах у світі, з метою встановлення залежностей від різних факторів, підвищення рівня безпеки. Критерієм актуальності було обрано той факт, що в Україні фіксується понад 150 повідомлень про терористичні акти щодня, тому цілком доцільно дослідити та скласти статистику по тероризму у світі.

Дані про терористичні акти у світі було запозичено з відкритого реєстру даних для machine learning ентузіастів <https://www.kaggle.com/START-UMD/gtd>.

Аналіз інструментарію для виконання курсового проєкт

Аналіз СКБД

В процесі виконання цього курсового проєкту перед нами встала потреба кешувати та зберігати дані про терористичні акти між запусками аналізатора. Кожного разу читати дані із CSV-файлів є дуже дорогою операцією, тож було прийняте рішення використати СКБД. В якості СКБД були розглянуті варіанти: PostgreSQL, MongoDB, CassandraDB. З порівняльною характеристикою цих СКБД можна ознайомитися в таблиці 1.

таблиця 1. Порівняльна характеристика СКБД

Критерій порівняння	Назва СКБД		
	MongoDB	PostgreSQL	CassandraDB
Має відкритий вихідний код	так	так	так
Схема даних	динамічна	статична і динамічна	статична і динамічна
Підтримка ієрархічних даних	так	так (з 2012)	ні
Реляційні дані	ні	так	так
Транзакції	ні	так	так
Атомарності операцій	всередині документа	по всій БД	всередині партії

Мова запитів	JSON	SQL	CQL
Найлегший спосіб масштабування	горизонтальний	вертикальний	горизонтальний
Підтримка шардингів	так	так (важка конфігурація)	так (може зберігати партії на різних машинах)
Приклад використання	Великі дані (мільярди записів) з великою кількістю паралельних оновлень, де цілісність і узгодженість даних не потрібно.	Транзакційні і операційні програми, вигода яких в нормалізованому формі, об'єднаннях, обмеження даних і підтримки транзакцій.	Багато запитів на запис/читання у одиницю часу, до даних можна задіяти партіціювання за ключем, дані мають лише первинні індекси
Наявність бібліотек для мови програмування Python 3	так	так	так
Підтримка реплікації	так, автоматичне	За принципом master-slave	так, через партіціювання

	переобрання головного процесу		
Засіб збереження та відновлення даних	mongodump	pg_dump	не має окремого доданка, виконується засобами CQL
Форма збереження даних	документи BSON	таблиця	таблиця

За результатами порівняння цих СКБД було прийнято рішення зупинитися на NoSQL рішеннях. Оскільки вони чудово поєднують в собі переваги неструктурованих баз даних та простоту використання горизонтального масштабування. Крім цього класичним прикладом використання NoSQL СКБД є системи збору та аналізу даних, до яких можна застосувати індексування за первинними та вторинними ключами.

Ця база даних є об'єктно орієнтованою та дозволяє зберігати великі масиви неструктурованих даних. На відміну від SQL баз даних ми можемо зберігати дані у “сирому” об'єктному вигляді, який використовується програмою та є більш близьким за структурою до моделі даних, яку буде використовувати ПЗ написане з використанням мови програмування Python. Це пришвидшить збір, збереження та отримання даних програмним забезпеченням. Оскільки MongoDB є представником NoSQL баз даних, вона не потребує жорсткої схеми даних, що дозволяє пришвидшити процес розробки та зробити його більш гнучким. Окрім цього дана СУБД підтримує горизонтальне масштабування за допомогою шардингу з метою зменшення

навантаження на кожен окремий вузол шляхом розподілення навантаження між ними всіма.

Нижче наведено перелік основних переваг:

- Підтримка ієрархічних даних
- Динамічна схема
- Швидкість запису у колекцію
- Швидкість читання із колекції
- Простота масштабування та відновлення даних

Обґрунтування вибору мови програмування

Мовою програмування для ПЗ було обрано Python 3.8. Оскільки це строго динамічно типізована мова програмування. Це дозволяє пришвидшити швидкість розробки ПЗ за рахунок уникнення постійного оголошення типів даних, якими оперує програма, та водночас не дає зробити критичних помилок, прихованих за автоматичним приведенням типів, на відміну від мови JavaScript.

Ключовою особливістю Python є широкий інструментарій засобів розробки систем збору та аналізу даних. Де-факто ця мова є стандартом у світі математичних розрахунків та обробки даних у реальному часі. Тож під час виконання курсового проекту було відносно легко знайти необхідну документацію та приклади роботи із цією мовою.

Обґрунтування вибору бібліотек і фреймворків

Використані бібліотеки:

- **pandas** — бібліотека для обробки та аналізу даних. Pandas є open-source бібліотекою, ліцензована BSD (Berkeley Source Distribution), забезпечує швидкі, гнучкі та експресивні структури даних, призначені для того, щоб працювати з «реляційними» або «помітними» даними як простими, так і інтуїтивно зрозумілими. Він прагне бути основним будівельним блоком високого рівня для здійснення практичного, реального аналізу даних у Python. Можливості бібліотеки: інструменти для обміну даними між структурами в пам'яті і файлами різних форматів, засоби поєднання даних і способи обробки відсутньої інформації, Переформатування наборів даних, в тому числі створення зведених таблиць, зріз даних за значеннями індексу, розширені можливості індексування, вибірка з великих наборів даних, вставка і видалення стовпців даних, можливості угруповання дозволяють виконувати трьохетапні операції типу «поділ, зміна, об'єднання», злиття і об'єднання наборів даних;
- **matplotlib** — це бібліотека Python 2D, яка представляє числові дані у різноманітних форматах та інтерактивних середовищах на різних платформах. Також ця бібліотека - це математичне розширення NumPy. Він надає об'єктно-орієнтований API для вбудови ділянок у додатки, що використовують набір інструментів для загального графічного інтерфейсу, таких як Tkinter, wxPython, Qt або GTK+. Matplotlib може використовуватися в скриптах Python, оболонках Python та IPython, серверах веб-додатків та чотирьох графічних наборах інструментів для користувацького інтерфейсу. Бібліотека **sciPy** використовує **matplotlib**.
- **numpy** — математична бібліотека мови Python, що додає підтримку великих багатовимірних масивів і матриць, разом з великою

бібліотекою високорівневих математичних функцій для операцій з цими масивами. А також інструменти для інтеграції C/C++ і Fortran коду, корисну лінійну алгебру, перетворення Фур'є і можливості випадкових чисел. Крім очевидних наукових застосувань, NumPy також може бути використаний як ефективний багатовимірний контейнер загальних даних. Можна визначити довільні типи даних. Це дає змогу без проблем і швидкої інтеграції NumPy з різноманітними базами даних. NumPy має ліцензію за ліцензією BSD, що дозволяє повторно використовувати кілька обмежень.

Структура бази даних

База даних складається з однієї колекції `gun_cases`, яка зберігає інформацію про збройні напади в США за 2013-2018 рр. Структура колекції наведена в таблиці 2.

таблиця 2. Опис властивостей документа у колекції `gun_cases`

Назва властивості	Тип	Опис
<code>_id</code>	<code>ObjectId</code>	Ідентифікатор запису
<code>year</code>	<code>Int</code>	Рік дати інциденту
<code>country</code>	<code>Str</code>	Назва країни
<code>region</code>	<code>Str</code>	Назва регіону
<code>success</code>	<code>Bool</code>	Чи був акт успішним
<code>suicide</code>	<code>Bool</code>	Чи був акт виконаний терористом-смертником
<code>targtype</code>	<code>Str</code>	Тип цілі акту
<code>weaptype</code>	<code>Str</code>	Тип використаного озброєння

Опис результатів аналізу предметної галузі

В результаті виконання курсового проекту було проаналізовано закономірності в збройних нападах в США.

1. Згідно з рисунком А1

З 2000 року загальна кількість терористичних актів зростала до 2014 року, після нього загальна кількість зменшується до сьогодні.

2. Згідно із рисунком А2

Найбільше терористичних актів відбувається в країнах: Іран, Афганістан та Пакистан.

3. Згідно із рисунком А3

Найбільше терористичних актів відбувається в регіонах Середнього Сходу, Північної Африки, Південної Азії.

4. Згідно із рисунком А4

Більшість (87%) терористичних актів були успішними.

5. Згідно із рисунком А5

Лише 6% терористичних актів були виконані терористами-смертниками.

6. Згідно з рисунком А6

Найчастіше жертвами терористичних актів стають мирні громадяни, силовики та поліція.

7. Згідно з рисунком А7

Найчастіше при терористичних актах використовується вибухівка та вогнепальна зброя.

Висновки

В процесі виконання даного курсового проекту було отримано практичні навички обробки великих масивів даних за допомогою мови програмування Python 3 та СКБД MongoDB.

Було проаналізовано сучасні методи та інструменти для роботи із великими даними, знайдено гарно працюючу комбінацію із мови програмування Python 3 та бібліотек до нього: Pandas, , Matplotlib, Numpy. Було складено порівняльну таблицю із трьох баз даних, які найчастіше використовують для роботи із часовими рядами. На основі цих даних було обрано в якості СКБД NoSQL рішення MongoDB.

На основі зібраних даних було проаналізовано статистику терористичних актів у світі. Були знайдені закономірності кількості терористичних актів по рокам, кількість актів по країнам та регіонам, була дана оцінка співвідношення вдалих/невдалих актів та актів, які були/не були спробою самогубства та створена статистика типів жертв та використаної зброї. Дані аналізу приведені у Додатку А. Текстовий опис надано в відповідному розділі цього курсового проекту.

В ході виконання даного курсового проекту було досягнуто поставленої мети: було набуто практичних навичок розробки сучасного програмного забезпечення, що взаємодіє з NoSQL базами даних, а також були здобуті навички оформлення відповідного текстового, програмного та ілюстративного матеріалу у формі проектної документації. У результаті виконання курсового проекту було покращено навички писати програмне забезпечення для NoSQL баз даних, володіти основами використання СУБД, а також інструментальними засобами аналізу великих обсягів даних, а саме відкритими бібліотеками мови Python.

Література

1. Hashed sharding [Електронний ресурс]. Режим доступу до ресурсу: <https://docs.mongodb.com/manual/core/hashed-sharding/>
2. Aggregation [Електронний ресурс]. Режим доступу до ресурсу: <https://docs.mongodb.com/manual/aggregation/>
3. 5 причин, почему машинное обучение станет технологией 2018 [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: <https://blogs.oracle.com/russia/machine-learning-2018;>
4. Сложности накопления данных для интеллектуального анализа [Електронний ресурс]. – 2012. – Режим доступу до ресурсу: [https://habr.com/post/154787/;](https://habr.com/post/154787/)
5. Почему Python так хорош в научных вычислениях [Електронний ресурс]. – 2018. – Режим доступу до ресурсу: [https://habr.com/post/349482/;](https://habr.com/post/349482/)
6. Python [Електронний ресурс]. – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Python;>
7. PostgreSQL лучше чем MongoDB [Електронний ресурс]. – 2015. – Режим доступу до ресурсу: [https://habr.com/post/272735/;](https://habr.com/post/272735/)
8. 27 шпаргалок п машинному обучению [Електронний ресурс]. – 2017. – Режим доступу до ресурсу: [https://proglab.io/p/ds-cheatsheets/;](https://proglab.io/p/ds-cheatsheets/)
9. Pandas [Електронний ресурс]. – Режим доступу до ресурсу: <https://ru.wikipedia.org/wiki/Pandas;>
10. Scikit-learn [Електронний ресурс]. – Режим доступу до ресурсу: <https://en.wikipedia.org/wiki/Scikit-learn;>
11. Matplotlib [Електронний ресурс]. – Режим доступу до ресурсу: <https://en.wikipedia.org/wiki/Matplotlib;>
12. NumPy [Електронний ресурс]. – Режим доступу до ресурсу: <https://en.wikipedia.org/wiki/NumPy;>

Додаток А

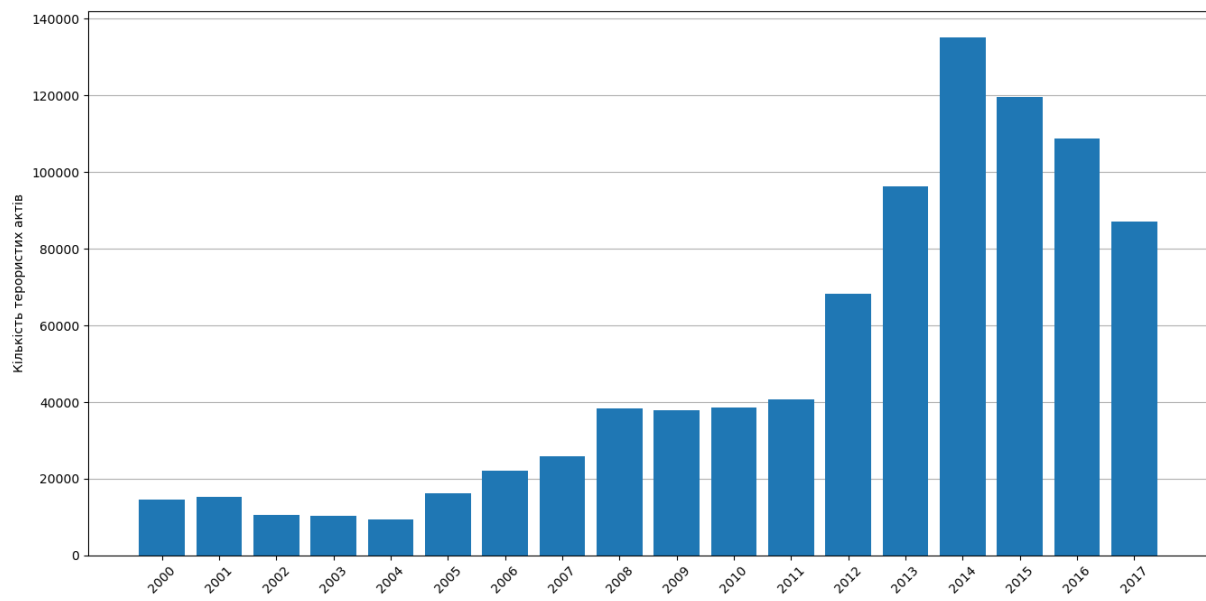


Рис 1. Розподіл кількості терористичних актів по рокам

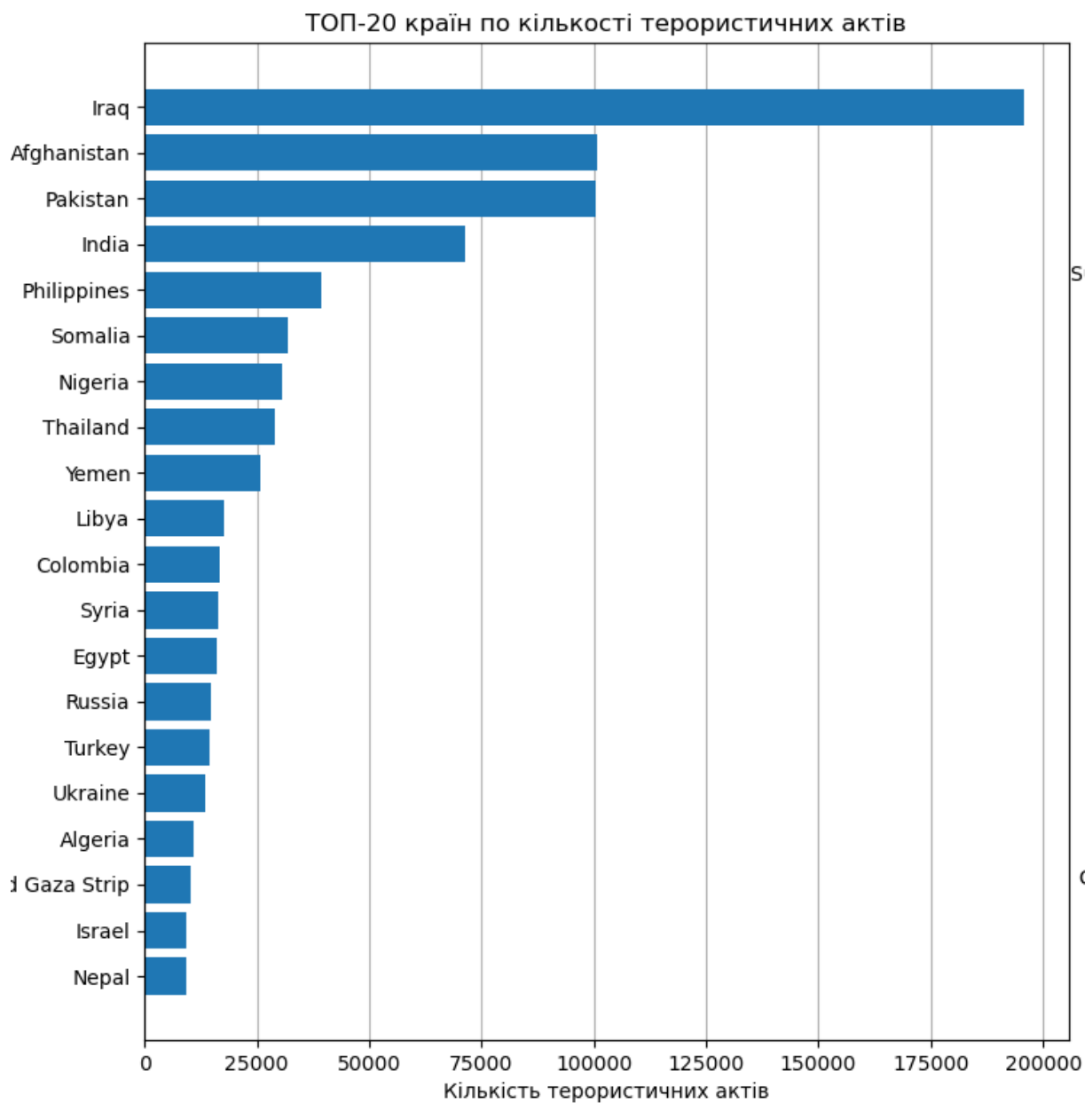


Рис 2. Топ- країн по кількості терористичних актів

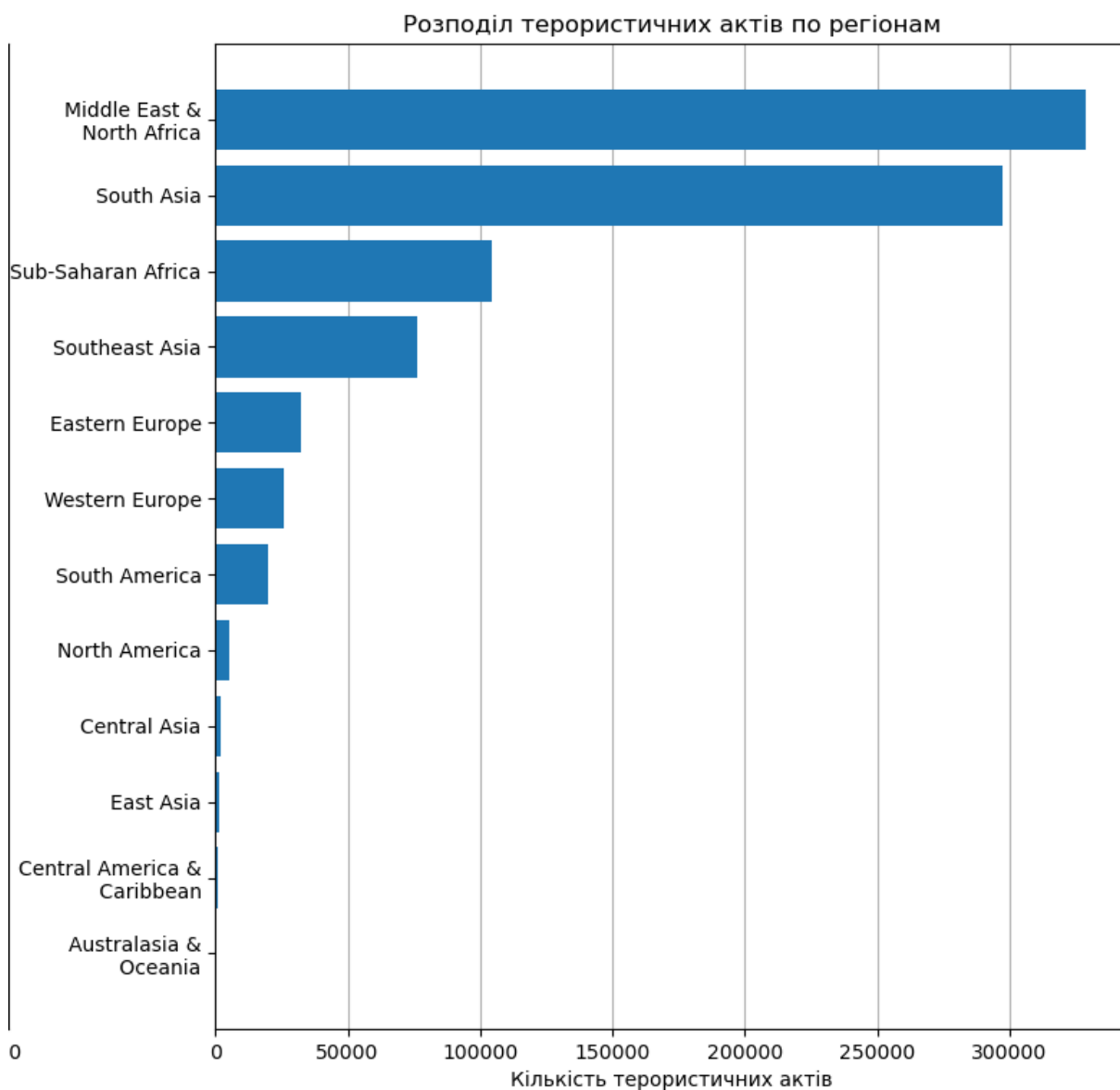


Рис 3. Розподіл терористичних актів по регіонам

Співвідношення успішних/неуспішних
терористичних актів

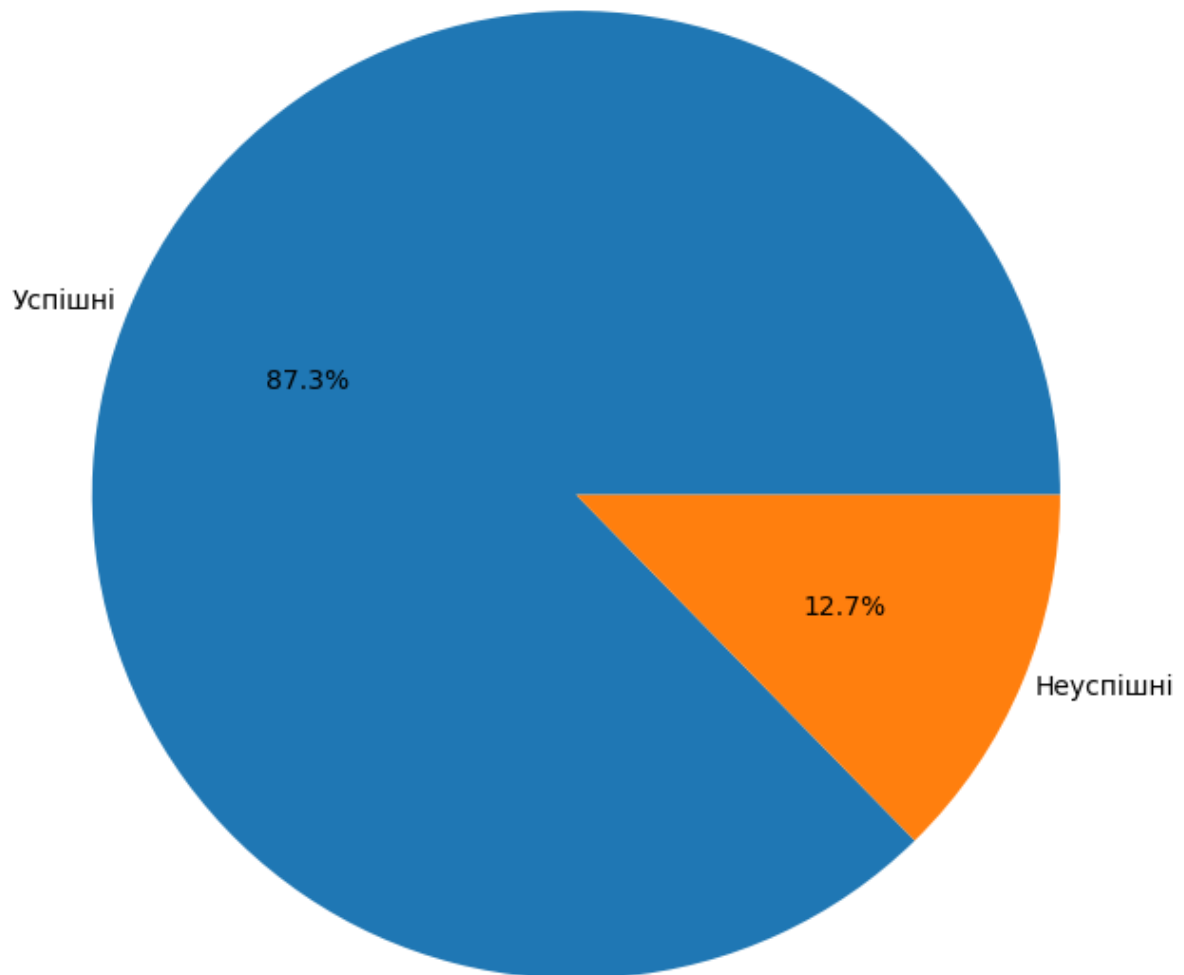


Рис 4. Співвідношення вдалих/невдалих актів

Співвідношення терористичних актів,
які були або не були суїцидом

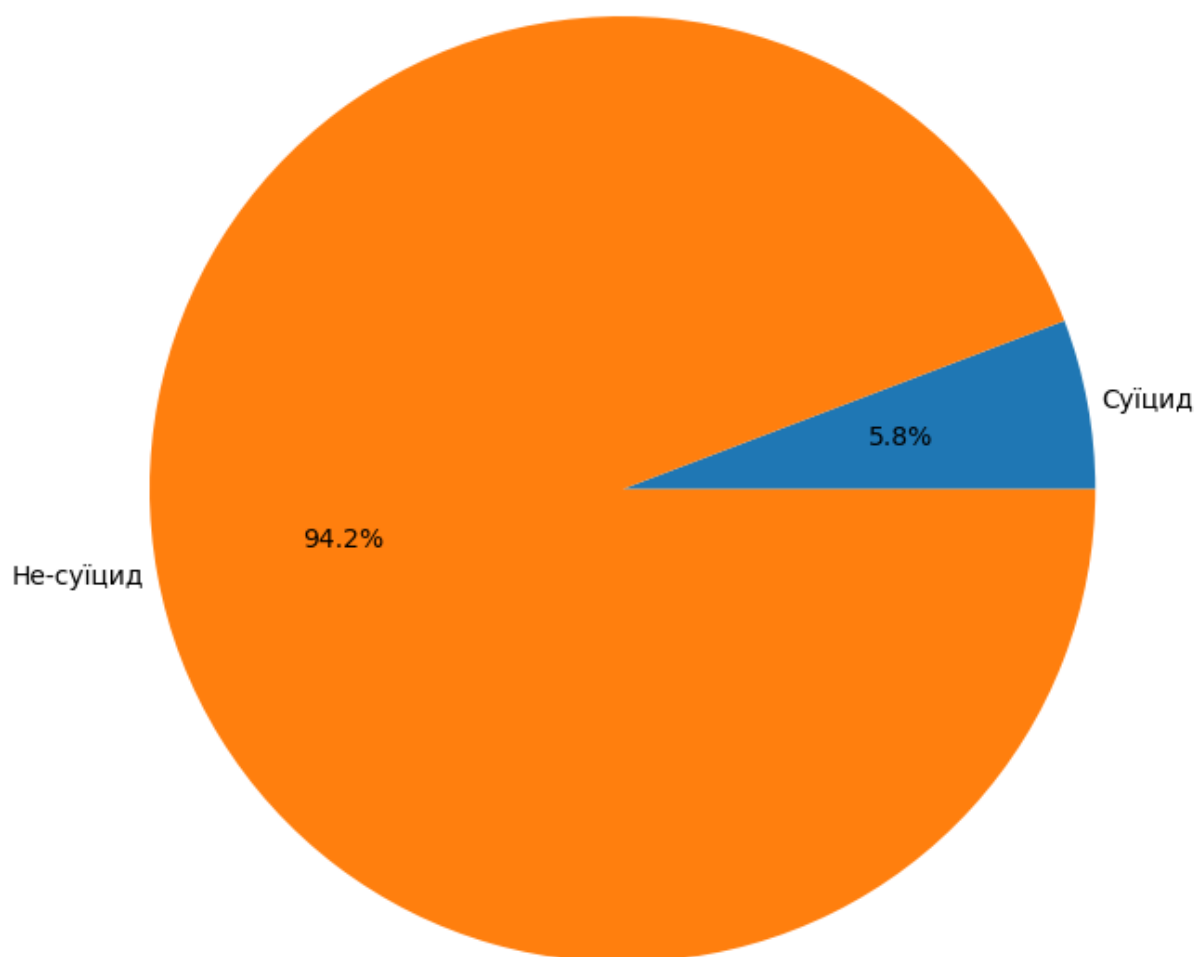


Рис 5. Співвідношення терористичних актів які були/не були виконані терористом-смертником

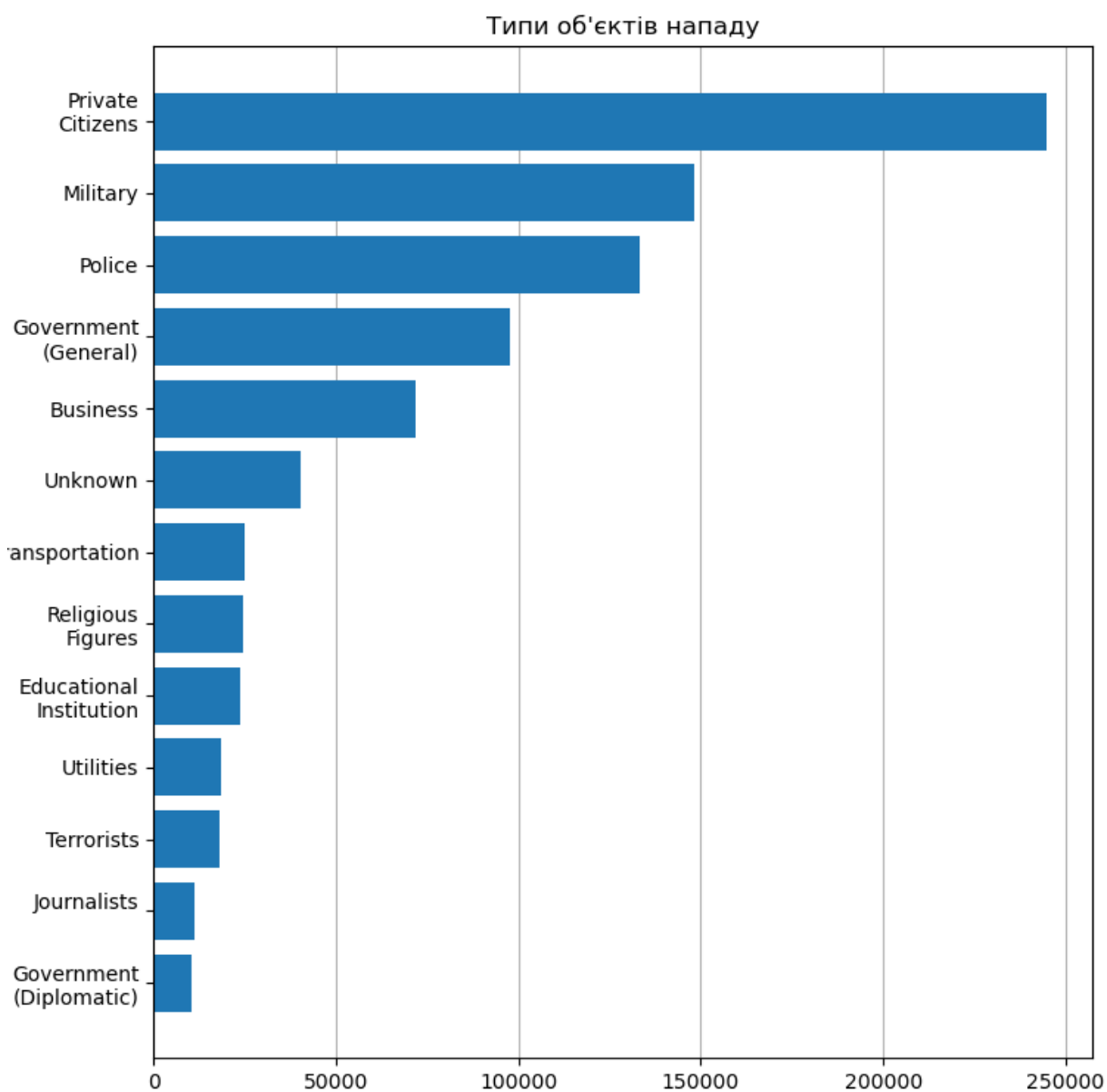


Рис 6. Типи об'єктів нападу терористичних актів

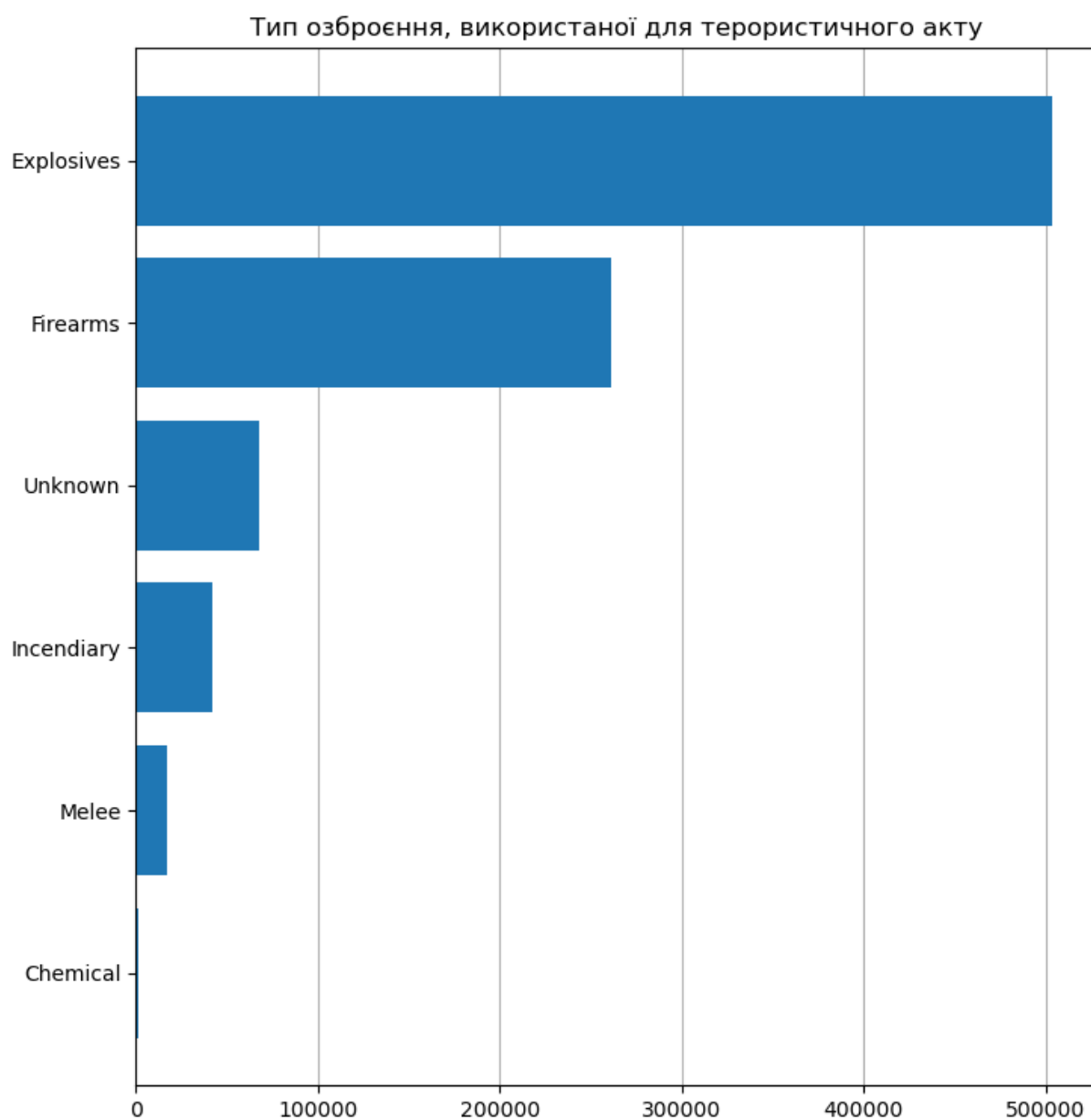


Рис 7. Типи використовуваного при терористичних актах озброєння