

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук
(повна назва)
Кафедра _____ Штучного інтелекту
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти _____ другий (магістерський)

_____ Дослідження та порівняння методів екстракції даних з неструктурованих
_____ текстових документів на основі NLP-моделей
(тема)

Виконав:

здобувач _____ другого _____ року
навчання, групи _____ СШМ-24-2

_____ Денис ТЕРНИНКО
(власне ім'я, прізвище)

Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)

Тип програми _____ освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма Системи штучного інтелекту
(повна назва освітньої програми)

Керівник проф. Костянтин ПЕТРОВ
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри ШІ

_____ Лариса ЧАЛА
(підпис) (власне ім'я, прізвище)

2026 р.

Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук _____

Кафедра _____ Штучного інтелекту _____

Рівень вищої освіти _____ другий (магістерський) _____

Спеціальність _____ 122 Комп'ютерні науки _____

(код і повна назва)

Тип програми _____ освітньо-наукова _____

(освітньо-професійна або освітньо-наукова)

Освітня програма _____ Системи штучного інтелекту _____

(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

«_____» _____ 20 ____ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві _____ Тернинку Денису Івановичу _____

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження та порівняння методів екстракції даних з неструктурованих текстових документів на основі NLP-моделей

затверджена наказом університету від 6 квітня 20 26 р. № 269Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 20 травня 20 26 р.

3. Вихідні дані до роботи наукові публікації з тематики обробки природної мови та екстракції інформації, документація бібліотек Python (spaCy, Transformers, PyTorch, scikit-learn, pandas, NumPy, Streamlit), набір даних Resume NER Training Dataset з платформи Kaggle для навчання та тестування моделей розпізнавання іменованих сутностей

4. Перелік питань, що потрібно опрацювати в роботі _____

1) Аналіз предметної галузі та постановка задачі дослідження

2) Методи екстракції інформації з неструктурованих текстових документів

3) Реалізація методів екстракції інформації

4) Результати та порівняльний аналіз методів екстракції інформації

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / термін виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	06.04.2026	виконано
2	Аналіз предметної галузі та постановка задачі дослідження	06.04.2026–09.04.2026	виконано
3	Теоретичне дослідження методів екстракції інформації з текстових документів	10.04.2026–14.04.2026	виконано
4	Вибір методів та моделей, підготовка набору даних	15.04.2026–17.04.2026	виконано
5	Реалізація rule-based підходу	18.04.2026–19.04.2026	виконано
6	Реалізація та навчання моделі spaCy NER	20.04.2026–22.04.2026	виконано
7	Реалізація та навчання трансформерних моделей	23.04.2026–26.04.2026	виконано
8	Формування та реалізація гібридного підходу	27.04.2026–29.04.2026	виконано
9	Проведення експериментів та оцінювання результатів	30.04.2026–04.05.2026	виконано
10	Розробка програмного інтерфейсу	05.05.2026–07.05.2026	виконано
11	Оформлення пояснювальної записки	08.05.2026–12.05.2026	виконано
12	Нормоконтроль	13.05.2026–15.05.2026	виконано
13	Рецензування	16.05.2026	виконано
14	Попередній захист	18.05.2026	виконано
15	Захист перед ЕК	20.05.2026	

Дата видачі завдання 6 квітня 2026 р.

Здобувач _____
(підпис)

Керівник роботи _____
(підпис)

проф. Костянтин ПЕТРОВ
(посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 84 с., 33 рис., 1 дод., 27 джерел.

ЕКСТРАКЦІЯ ДАНИХ, МАШИННЕ НАВЧАННЯ,
НЕСТРУКТУРОВАНІ ТЕКСТОВІ ДОКУМЕНТИ, ОБРОБКА ПРИРОДНОЇ
МОВИ, РОЗПІЗНАВАННЯ ІМЕНОВАНИХ СУТНОСТЕЙ,
ТРАНСФОРМЕРНІ МОДЕЛІ.

Об'єкт дослідження – процес екстракції даних з неструктурованих текстових документів.

Предмет дослідження – методи екстракції даних з текстових документів на основі NLP-моделей.

Мета роботи – дослідження та порівняння методів екстракції даних з неструктурованих текстових документів на основі NLP-моделей.

Методи дослідження – методи обробки природної мови, розпізнавання іменованих сутностей, rule-based підхід, моделі машинного навчання та трансформерні моделі (spaCy, BERT, DistilBERT), методи оцінювання якості (precision, recall, F1-score).

У роботі досліджено та порівняно методи екстракції даних з неструктурованих текстових документів на основі NLP-моделей на прикладі текстових резюме. Реалізовано rule-based підхід, модель на основі spaCy, трансформерні моделі DistilBERT та BERT-base-cased, а також гібридний підхід. Проведено експериментальне оцінювання ефективності методів із використанням метрик якості та виконано їх порівняльний аналіз. Результати дослідження показали доцільність використання комбінованих підходів для підвищення якості екстракції інформації.

ABSTRACT

Master's thesis contains: 84 pp., 33 fig., 1 ann., 27 references.

DATA EXTRACTION, MACHINE LEARNING, NAMED ENTITY RECOGNITION, NATURAL LANGUAGE PROCESSING, TRANSFORMER MODELS, UNSTRUCTURED TEXT DOCUMENTS.

Object of research – the process of data extraction from unstructured text documents.

Subject of research – methods of data extraction from text documents based on NLP models.

Purpose of the work – to study and compare methods of data extraction from unstructured text documents based on NLP models.

Research methods – natural language processing methods, named entity recognition, rule-based approach, machine learning models and transformer models (spaCy, BERT, DistilBERT), evaluation metrics (precision, recall, F1-score).

The paper investigates and compares methods of data extraction from unstructured text documents based on NLP models using text resumes as an example. A rule-based approach, a spaCy-based model, transformer models (DistilBERT and BERT-base-cased), and a hybrid approach are implemented. Experimental evaluation of the methods is carried out using quality metrics, followed by a comparative analysis. The results demonstrate the effectiveness of combining different approaches to improve the overall quality of information extraction.

ЗМІСТ

Вступ.....	8
1 Аналіз предметної галузі та постановка задачі дослідження.....	11
1.1 Теоретичні.....	
основи екстракції інформації з неструктурованих текстових документів	11
1.2 Аналіз існуючих методів та рішень екстракції інформації.....	17
1.3 Постановка задачі дослідження.....	30
2 Методи екстракції інформації з неструктурованих текстових документів	32
2.1 Особливості екстракції інформації з текстових резюме.....	32
2.2 Класичні підходи до екстракції інформації.....	34
2.3 Методи машинного навчання для екстракції інформації.....	36
2.4 Трансформерні моделі для задач обробки тексту.....	40
2.5 Методи розпізнавання іменованих сутностей (NER).....	43
2.6 Метрики оцінювання якості екстракції.....	46
3 Реалізація методів екстракції інформації.....	49
3.1 Опис набору даних.....	50
3.2 Підготовка та попередня обробка даних.....	52
3.3 Реалізація rule-based підходу до екстракції інформації.....	55
3.4 Реалізація моделі NER на основі spaCy.....	57
3.5 Реалізація трансформерних моделей для задачі NER.....	59
3.6 Формування гібридного підходу до екстракції інформації.....	63
4 Результати та порівняльний аналіз методів екстракції інформації.....	65
4.1 Методика проведення експерименту.....	65
4.2 Результати застосування методів екстракції інформації.....	66
4.3 Порівняльний аналіз методів екстракції.....	72
4.4 Аналіз помилок методів екстракції інформації.....	74
4.5 Реалізація програмного інтерфейсу системи екстракції інформації ..	75

Висновки.....	ЗМІСТ	78
Перелік джерел посилання.....		80
Додаток А Відомість кваліфікаційної роботи.....		84

ВСТУП

Сучасний розвиток інформаційних технологій супроводжується стрімким зростанням обсягів даних, значна частина яких представлена у текстовому вигляді. Такі дані формуються в різних сферах діяльності людини, зокрема у бізнесі, освіті, медицині, державному управлінні та інформаційних сервісах. При цьому текст виступає одним із основних способів фіксації та передачі знань, що зумовлює його широке використання. Значна частина цих даних має неструктурований характер, що ускладнює їх автоматизовану обробку, аналіз та подальше використання в інформаційних системах.

Неструктуровані текстові дані характеризуються відсутністю чітко визначеної структури, стандартизованих форматів і однозначних правил представлення інформації. Це означає, що дані не мають фіксованих полів або атрибутів, а інформація подається у вигляді довільного тексту. У результаті традиційні методи обробки даних, орієнтовані на структуровані формати, не забезпечують належної ефективності при роботі з такими даними. Додаткову складність створює варіативність мовних конструкцій, використання синонімів, скорочень та різних стилів викладу, що ускладнює формалізацію інформації та її подальше використання.

У зв'язку з цим виникає необхідність розробки підходів, що дозволяють перетворювати текстову інформацію у структурований вигляд, придатний для подальшого аналізу, зберігання та інтеграції в інформаційні системи. Одним із таких підходів є екстракція даних, яка передбачає автоматизоване виділення значущих фрагментів тексту та їх подальше представлення у формалізованому вигляді. Це дозволяє перетворити неструктурований текст у набір структурованих записів, які можуть бути використані для виконання різних аналітичних задач.

Однією з актуальних практичних задач, що ілюструє зазначену проблему, є обробка текстових резюме. У сфері підбору персоналу щоденно

обробляються значні обсяги таких документів, які містять важливу інформацію про кандидатів, зокрема їхні персональні дані, освіту, професійний досвід, навички та інші характеристики. Однак ця інформація, як правило, подається у довільному текстовому форматі, що ускладнює її автоматизоване використання, пошук та інтеграцію у відповідні інформаційні системи.

Текстові резюме характеризуються значною варіативністю структури та способів подання інформації. Вони можуть містити як чітко виділені логічні блоки, так і суцільний текст або їх комбінацію. Крім того, одна й та сама інформація може бути представлена різними способами, що ускладнює її однозначну інтерпретацію. Це робить задачу автоматизованої обробки резюме складною та потребує використання методів, здатних враховувати контекст і семантичні особливості тексту.

Для розв'язання зазначених задач застосовуються методи обробки природної мови (Natural Language Processing, NLP), які забезпечують можливість автоматизованого аналізу текстових даних. Використання таких методів дозволяє здійснювати виділення значущих елементів тексту, виявлення закономірностей та формування структурованого представлення інформації. Зокрема, важливим напрямом є розпізнавання іменованих сутностей, яке дозволяє виділяти ключові елементи тексту та визначати їх тип.

З розвитком методів машинного навчання та глибокого навчання з'явилися нові можливості для підвищення ефективності обробки текстів. Сучасні NLP-моделі, зокрема трансформерні архітектури, дозволяють враховувати контекст, семантичні зв'язки між словами та складні мовні конструкції, що значно покращує якість екстракції даних. При цьому існує велика кількість різних підходів і моделей, які відрізняються за принципами роботи, точністю, швидкістю та вимогами до обчислювальних ресурсів.

Важливим аспектом при виборі підходів до екстракції інформації є необхідність забезпечення балансу між точністю, повнотою та

обчислювальною ефективністю моделей. У практичних застосуваннях, зокрема при обробці великих масивів текстових документів, важливу роль відіграють швидкість обробки та можливість масштабування рішень. Крім того, різні методи можуть демонструвати різну ефективність залежно від типу сутностей: формалізовані дані краще піддаються шаблонним підходам, тоді як контекстно залежні – потребують використання більш складних моделей. Це зумовлює доцільність дослідження та порівняння різних методів екстракції з метою визначення їх переваг і обмежень у конкретних умовах застосування.

У зв'язку з цим виникає необхідність проведення дослідження, спрямованого на аналіз та порівняння методів екстракції даних з неструктурованих текстових документів на основі NLP-моделей. Особливий інтерес становить оцінювання ефективності таких підходів на прикладі текстових резюме як одного з поширених типів неструктурованих документів, а також визначення доцільності їх використання у практичних інформаційних системах.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

Зростання обсягів текстових даних у цифровому середовищі зумовлює необхідність їх ефективної обробки та аналізу. Значна частина інформації зберігається у неструктурованому вигляді, що ускладнює її використання в автоматизованих системах та аналітичних процесах.

У зв'язку з цим актуальним є дослідження підходів до виділення значущих елементів тексту та перетворення їх у структурований формат. Особливий інтерес становлять текстові документи, що містять важливу інформацію у довільній формі, зокрема резюме, які широко використовуються у сфері підбору персоналу.

1.1 Теоретичні основи екстракції інформації з неструктурованих текстових документів

Дані, що використовуються в інформаційних системах, залежно від ступеня впорядкованості поділяються на структуровані, напівструктуровані та неструктуровані (рисунк 1.1).

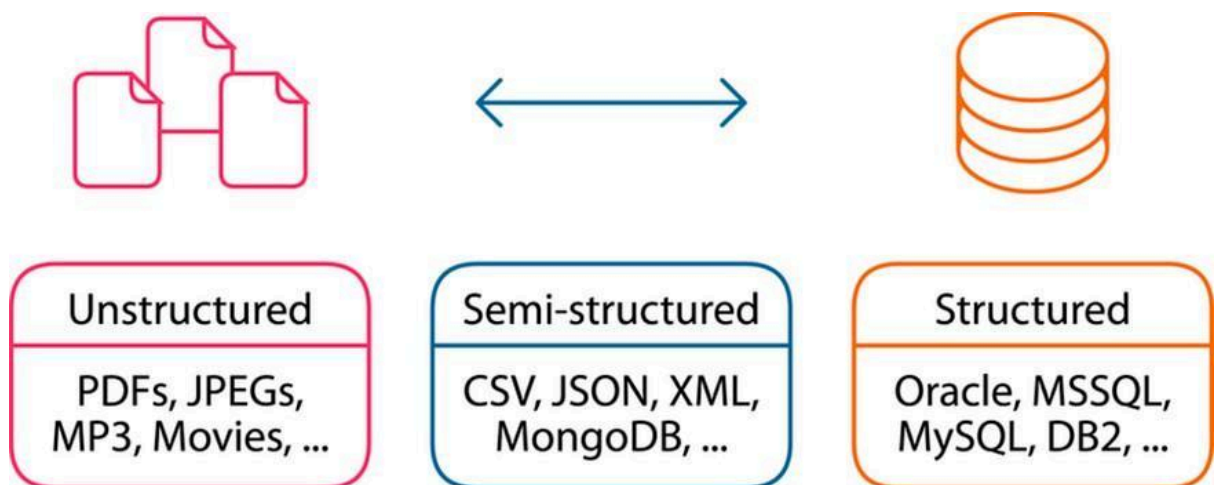


Рисунок 1.1 – Класифікація даних за рівнем структурованості [1]

Структуровані дані мають чітко визначену схему та організовуються у вигляді таблиць або баз даних, що забезпечує їх зручне зберігання, пошук та обробку за допомогою традиційних алгоритмів. Напівструктуровані дані частково зберігають структуру за рахунок використання тегів або ключів (наприклад, XML чи JSON), однак не мають жорстко фіксованої схеми, що дозволяє гнучкіше представляти інформацію та адаптувати її до різних форматів.

Неструктуровані дані, на відміну від наведених типів, не мають наперед визначеної структури та включають текстові документи, електронні повідомлення, веб-сторінки та інші форми подання інформації. Вони не містять чітко визначених полів або атрибутів, що ускладнює їх автоматичну обробку та інтеграцію у інформаційні системи. У результаті значна частина інформації, що міститься у таких даних, залишається неформалізованою та недоступною для подальшого аналізу без застосування спеціалізованих методів.

Значна частина неструктурованих даних представлена у вигляді природної мови, що зумовлює її домінування як засобу фіксації знань у різних предметних галузях. Текстові дані характеризуються високим рівнем варіативності, оскільки одна й та сама інформація може бути виражена різними мовними конструкціями, із використанням синонімів, скорочень або різних стилів викладу. Крім того, структура тексту може суттєво відрізнятися залежно від автора, формату документа та цілей його створення.

Обробка неструктурованих текстових даних ускладнюється відсутністю чіткої структури, залежністю змісту від контексту, наявністю неоднозначностей та надлишкових або нерелевантних фрагментів. Окремі частини тексту можуть містити шум, службову інформацію або неформалізовані вирази, що не несуть корисного змісту для задач аналізу. Додатковою складністю є те, що значення слів і фрагментів тексту

визначається не лише їх формою, а й контекстом використання, що потребує більш складних підходів до інтерпретації.

Такі особливості неструктурованих текстових даних ускладнюють їх формалізацію та подальше використання у інформаційних системах, що обумовлює необхідність застосування методів, здатних виконувати автоматичний аналіз тексту з урахуванням його лінгвістичних та семантичних характеристик.

У зв'язку з труднощами обробки неструктурованих текстових даних виникає потреба у виділенні з них значущої інформації, придатної для подальшого використання в інформаційних системах та аналітичних застосуваннях. Такий підхід реалізується в межах задачі екстракції інформації (information extraction), яка передбачає автоматизоване отримання ключових відомостей із тексту [2]. Основною метою екстракції є перетворення неструктурованого тексту у форму, що дозволяє виконувати подальший аналіз, зберігання та інтеграцію отриманих даних у різні програмні системи.

Результатом екстракції є структуроване представлення даних у вигляді впорядкованих записів, полів або таблиць, що забезпечує можливість їх ефективної обробки програмними засобами. На відміну від тексту природною мовою, такі дані мають чітко визначені елементи та зв'язки між ними, що спрощує виконання операцій пошуку, фільтрації, агрегації та подальшого аналізу. Це дозволяє використовувати отриману інформацію у прикладних задачах, зокрема у системах підтримки прийняття рішень, інформаційно-пошукових системах та аналітичних платформах.

На концептуальному рівні процес екстракції інформації включає послідовні етапи: виявлення релевантних фрагментів тексту, визначення їх семантичного значення та перетворення отриманих даних у формалізований вигляд. На практиці цей процес може включати додаткові кроки, такі як попередня обробка тексту, токенізація, нормалізація та класифікація елементів. Кожен із цих етапів впливає на кінцевий результат, оскільки

помилки на ранніх стадіях можуть призводити до втрати або спотворення інформації.

Залежно від поставленої задачі екстракція інформації може включати виділення різних типів даних, таких як сутності, зв'язки між ними або події. У задачах аналізу текстових документів найбільш поширеним є підхід, орієнтований на виділення іменованих сутностей, який дозволяє локалізувати ключові елементи інформації у тексті.

Одним із базових підходів до виділення інформації з тексту є розпізнавання іменованих сутностей (Named Entity Recognition, NER), яке передбачає ідентифікацію в тексті об'єктів, що мають визначене семантичне значення, зокрема імен осіб, назв організацій, географічних об'єктів, дат та інших категорій [3]. Використання такого підходу дозволяє локалізувати ключові елементи інформації, визначити їх тип та підготувати дані до подальшого структурованого представлення.

Ефективність екстракції інформації суттєво залежить від типу текстових документів, що підлягають аналізу. Різні види текстів відрізняються за структурою, стилем викладу та ступенем формалізації, що безпосередньо впливає на складність їх обробки та вибір методів аналізу. Зокрема, тексти з чітко вираженою структурою або напівструктуровані документи дозволяють використовувати більш прості алгоритми обробки, тоді як документи, у яких інформація представлена у вільній формі, потребують застосування більш складних підходів, здатних враховувати контекст та варіативність тексту.

Окремим класом текстових документів, що активно використовуються у практичних застосуваннях екстракції інформації, є резюме. Вони широко застосовуються у сфері підбору персоналу та містять узагальнену інформацію про кандидата, зокрема відомості про освіту, професійний досвід, навички та інші характеристики. Автоматизація обробки таких документів є актуальною задачею, оскільки дозволяє суттєво

пришвидшити аналіз великої кількості заявок, зменшити вплив людського фактору та підвищити ефективність відбору кандидатів.

Текстові резюме не мають єдиного стандарту оформлення та характеризуються значною варіативністю структури. Інформація може бути представлена у вигляді суцільного тексту, списків, окремих логічних блоків або їх комбінацій, що ускладнює автоматичне виділення необхідних відомостей. Крім того, різні автори використовують різні підходи до організації інформації, що призводить до відсутності єдиного шаблону, на який можна було б орієнтуватися під час обробки.

Додаткову складність створює те, що однакові за змістом характеристики можуть бути сформульовані різними способами або розміщені в різних частинах документа [4]. Наприклад, інформація про досвід роботи може бути представлена як у вигляді окремого розділу, так і інтегрована у загальний опис професійної діяльності. Аналогічно, навички можуть бути подані як у вигляді списку, так і у складі опису виконуваних задач.

Такі особливості резюме ускладнюють застосування жорстко заданих правил та потребують використання методів, здатних адаптуватися до різних форматів представлення інформації. Зокрема, необхідно враховувати контекст тексту, взаємозв'язки між його елементами та семантичні особливості формулювань. Це обумовлює доцільність застосування сучасних методів обробки природної мови, орієнтованих на аналіз неструктурованих текстових даних.

У межах задачі аналізу резюме виконується виділення ключових характеристик кандидата, таких як освіта, досвід роботи, професійні навички та інші відомості, що використовуються для оцінювання кандидата. При цьому відповідні дані не завжди мають чітко визначені межі у тексті та можуть бути розподілені по різних частинах документа. У багатьох випадках для їх коректного визначення необхідно враховувати контекст,

оскільки окремі фрагменти тексту набувають значення лише у поєднанні з іншими елементами.

Складність задачі екстракції інформації з резюме зумовлена неоднозначністю природної мови, залежністю змісту від контексту та варіативністю способів подання однакових відомостей. Наприклад, одна й та сама інформація про досвід роботи може бути представлена як у вигляді короткого запису з датами, так і у вигляді розгорнутого текстового опису виконуваних обов'язків. Аналогічно, назви посад або навичок можуть мати різні формулювання, що ускладнює їх однозначну ідентифікацію.

Додаткові труднощі пов'язані з використанням скорочень, неформальних формулювань, різних мов або їх поєднання в межах одного документа. Крім того, у резюме часто використовуються списки, аббревіатури та спеціалізована термінологія, що потребує врахування предметної специфіки. Поєднання різних способів подання інформації в межах одного документа призводить до того, що одна й та сама сутність може бути представлена у різних структурних формах, що ускладнює її автоматичне визначення.

Ще одним важливим аспектом є необхідність точного визначення меж сутностей. У випадку багатослівних конструкцій, таких як назви організацій або посад, помилки у визначенні початку або кінця сутності можуть призводити до спотворення результатів екстракції. Це вимагає використання моделей, здатних працювати з послідовностями тексту та враховувати залежності між їх елементами.

Таким чином, екстракція даних з неструктурованих текстових документів, зокрема текстових резюме, є складною задачею, що поєднує проблеми неоднозначності природної мови, варіативності представлення інформації та необхідності врахування контексту. Це обумовлює доцільність застосування сучасних методів обробки природної мови, зокрема моделей машинного навчання та трансформерних архітектур, які

здатні ефективно виділяти та структурувати інформацію у межах різних форматів текстових даних.

1.2 Аналіз існуючих методів та рішень екстракції інформації

Сучасні підходи до екстракції інформації з неструктурованих текстових документів характеризуються значним різноманіттям методів, що відрізняються як за принципами обробки даних, так і за рівнем використання контекстної інформації. Розвиток цієї галузі відбувався від правил, що жорстко задаються, до моделей, здатних автоматично навчатися на даних і враховувати складні мовні залежності.

У наукових дослідженнях запропоновано низку підходів до аналізу текстових документів, зокрема резюме, що охоплюють як класичні методи обробки тексту, так і сучасні моделі машинного навчання та глибинного навчання. Кожен із підходів має власні переваги та обмеження, що визначають доцільність його застосування залежно від структури даних, обсягу навчальної вибірки та вимог до точності результатів.

Екстракція інформації з текстових документів на початкових етапах розвитку здійснювалася переважно за допомогою правил та евристичних підходів. Такі методи базуються на використанні заздалегідь визначених шаблонів, регулярних виразів та словників, що дозволяє ідентифікувати ключові елементи тексту без застосування навчання моделей. Подібні підходи є ефективними у випадках, коли структура документів є відносно стабільною та передбачуваною.

У роботі [5] запропоновано підхід до автоматизованого аналізу резюме, що базується на поєднанні методів обробки природної мови, пошуку за ключовими словами та використання регулярних виразів. Основна ідея полягає у поетапному вилученні структурованої інформації з текстового документа з подальшим оцінюванням відповідності кандидата заданим вимогам.

Процес обробки передбачає попередню конвертацію резюме у текстовий формат, після чого здійснюється виділення окремих категорій інформації, зокрема навичок, освіти та професійного досвіду. Для цього використовуються шаблони, ключові слова та правила позиційного аналізу тексту, що дозволяє локалізувати необхідні фрагменти навіть у випадку різної структури документа.

Отримані дані використовуються для розрахунку часткових оцінок за кожною категорією, які далі агрегуються у загальний показник відповідності кандидата. Такий підхід дозволяє автоматизувати процес первинного відбору резюме та зменшити навантаження на рекрутерів. Загальна схема процесу, що включає етапи попередньої обробки, екстракції інформації та обчислення підсумкової оцінки, наведена на рисунку 1.2.

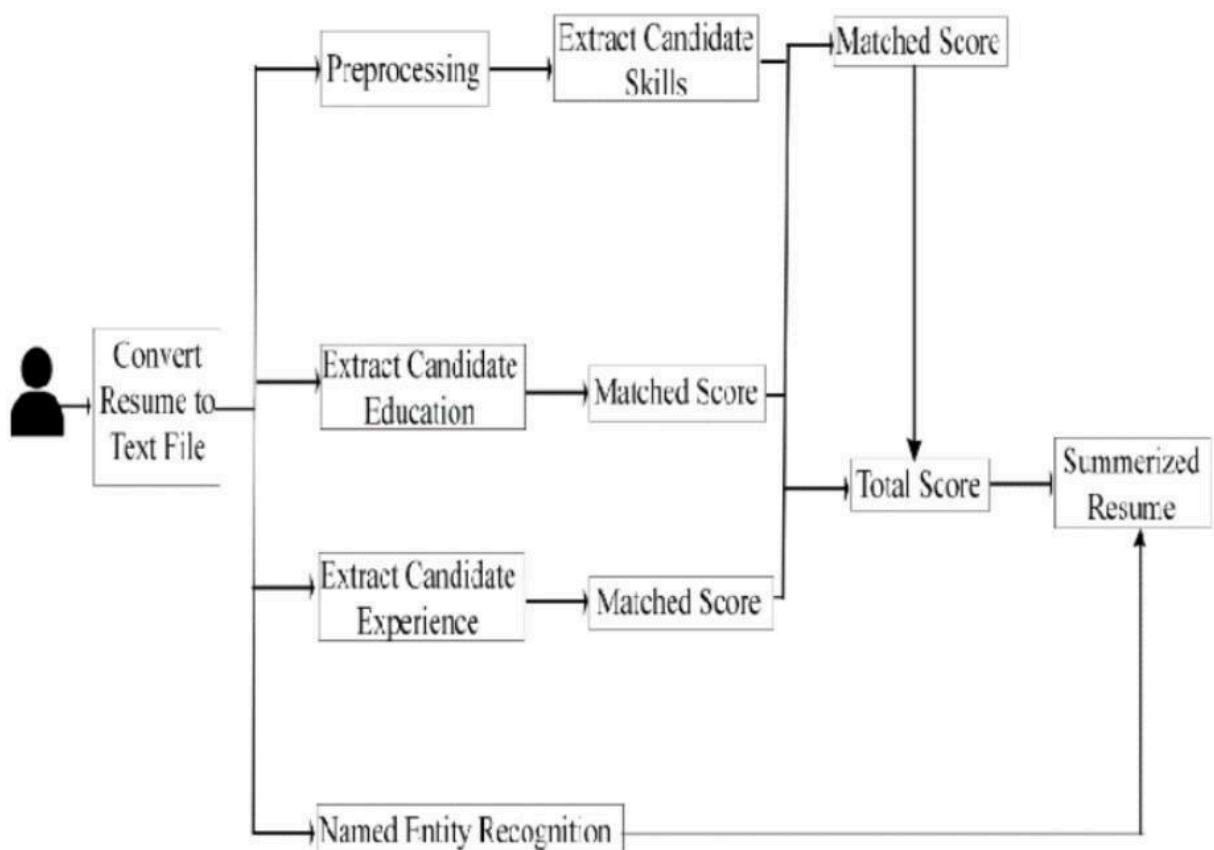


Рисунок 1.2 – Архітектура системи аналізу технічних резюме на основі правил [5]

Перевагою підходу є простота реалізації та інтерпретованість результатів. Водночас залежність від заздалегідь визначених правил і ключових слів обмежує гнучкість системи та ускладнює обробку нестандартних або неформалізованих резюме.

У дослідженні [6] розглядається підхід до автоматизації обробки резюме у межах системи управління кадровим резервом, що базується на використанні методів обробки природної мови. Основна увага приділяється виділенню ключової інформації про кандидатів та її подальшій організації для спрощення пошуку і відбору.

Запропонований підхід передбачає попередню обробку тексту, що включає очищення, токенізацію та нормалізацію даних, після чого здійснюється виділення сутностей, пов'язаних із професійними навичками, освітою та досвідом роботи. Для цього використовуються лексичні ресурси та правила, що дозволяють ідентифікувати релевантні фрагменти тексту.

Отримана інформація структурується та зберігається у вигляді, придатному для подальшого аналізу, зокрема для пошуку кандидатів за заданими критеріями. Такий підхід дозволяє підвищити ефективність роботи з великими обсягами резюме та забезпечує швидкий доступ до необхідних даних.

Разом із тим, використання переважно правил та словників обмежує здатність системи адаптуватися до різних форматів і стилів написання резюме, що знижує точність екстракції у випадках складних або нетипових документів.

Подальший розвиток методів екстракції інформації пов'язаний із переходом до моделей машинного навчання, які дозволяють автоматично виявляти закономірності у текстових даних без використання жорстко заданих правил. У таких підходах задача екстракції формалізується як задача маркування послідовностей, що дає змогу враховувати контекст та залежності між елементами тексту, забезпечуючи більш узгоджене виділення інформації.

У роботі [7] запропоновано нейронну архітектуру для задачі розпізнавання іменованих сутностей, що поєднує двонаправлені рекурентні нейронні мережі з умовними випадковими полями (BiLSTM-CRF). Основною ідеєю підходу є формування контекстного представлення кожного слова з урахуванням як попередніх, так і наступних токенів у послідовності, що дозволяє ефективно враховувати залежності в тексті.

Модель отримує на вході векторні представлення слів, які доповнюються ознаками, отриманими на символічному рівні. Для цього використовується додатковий механізм, що дозволяє враховувати морфологічну структуру слів, зокрема префікси та суфікси, що є особливо важливим для коректного визначення сутностей у випадку рідкісних або невідомих слів.

На наступному етапі двонаправлена LSTM формує узагальнене представлення кожного токена, яке передається до CRF-шару. Використання CRF дозволяє моделювати залежності між мітками в послідовності, що забезпечує узгодженість результатів, наприклад, у випадках, коли сутність складається з декількох слів. Загальна архітектура моделі, що поєднує символічні та словникові представлення з послідовнісним маркуванням, наведена на рисунку 1.3.

Перевагою підходу є здатність автоматично навчатися релевантних ознак без використання ручних правил, а також ефективне врахування контексту. Водночас модель потребує значних обсягів навчальних даних та обчислювальних ресурсів, що може ускладнювати її застосування в умовах обмежених ресурсів.

Подальше вдосконалення моделей послідовнісного маркування пов'язане з урахуванням внутрішньої структури слів та підвищенням якості їх представлення. У роботі [8] запропоновано підхід до розпізнавання іменованих сутностей, що базується на поєднанні двонаправлених LSTM-мереж із згортковими нейронними мережами для обробки символічного рівня. Основною ідеєю є використання CNN для автоматичного вилучення

ознак із послідовностей символів, що дозволяє враховувати внутрішню структуру слова без необхідності ручного задання морфологічних характеристик.

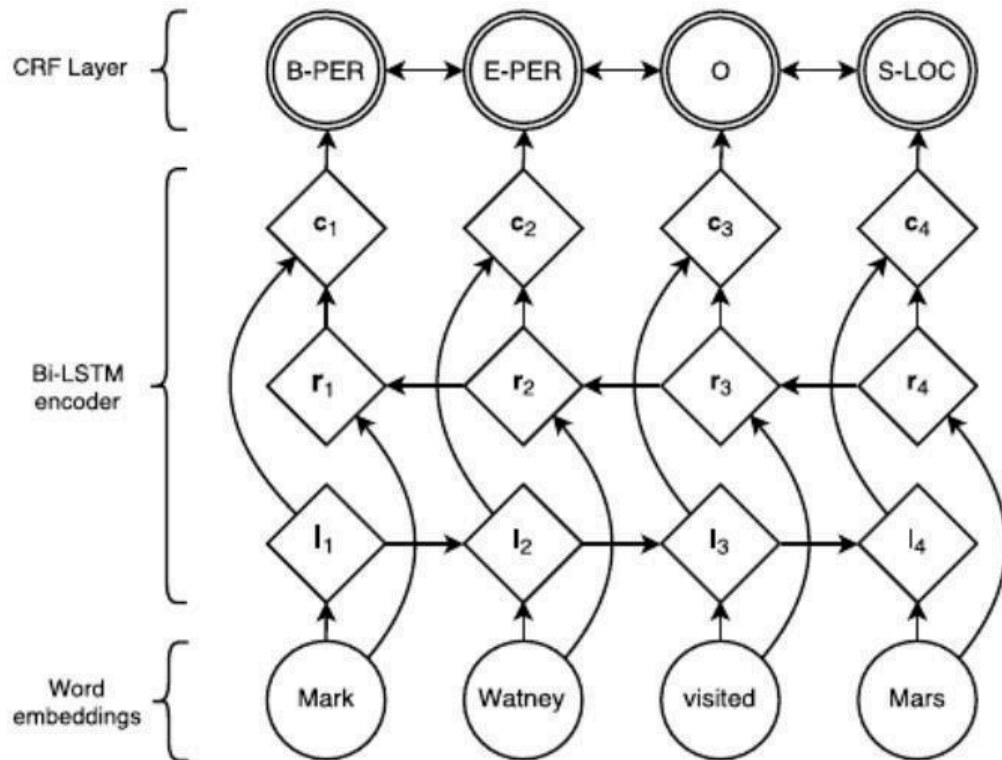


Рисунок 1.3 – Архітектура моделі BiLSTM-CRF для задачі розпізнавання іменованих сутностей [7]

Символьні представлення, отримані за допомогою згорткової мережі, об'єднуються з векторними представленнями слів і подаються на вхід двонаправленої LSTM. Це забезпечує одночасне врахування як локальних ознак слова, так і контексту в межах усього речення. Двонаправлений характер мережі дозволяє моделі аналізувати залежності як зліва направо, так і справа наліво, що є важливим для коректного визначення меж сутностей у тексті.

Додатковою особливістю підходу є використання механізму декодування послідовності міток, який враховує узгодженість між сусідніми тегами, що зменшує кількість некоректних комбінацій

міток. Крім того, модель може інтегрувати зовнішні лексичні ресурси, що дозволяє підвищити точність розпізнавання специфічних типів сутностей.

Важливим аспектом підходу є його здатність ефективно працювати з рідкісними та невідомими словами, що досягається саме за рахунок використання символічного рівня представлення. Загальна архітектура моделі представлена на рисунку 1.4.

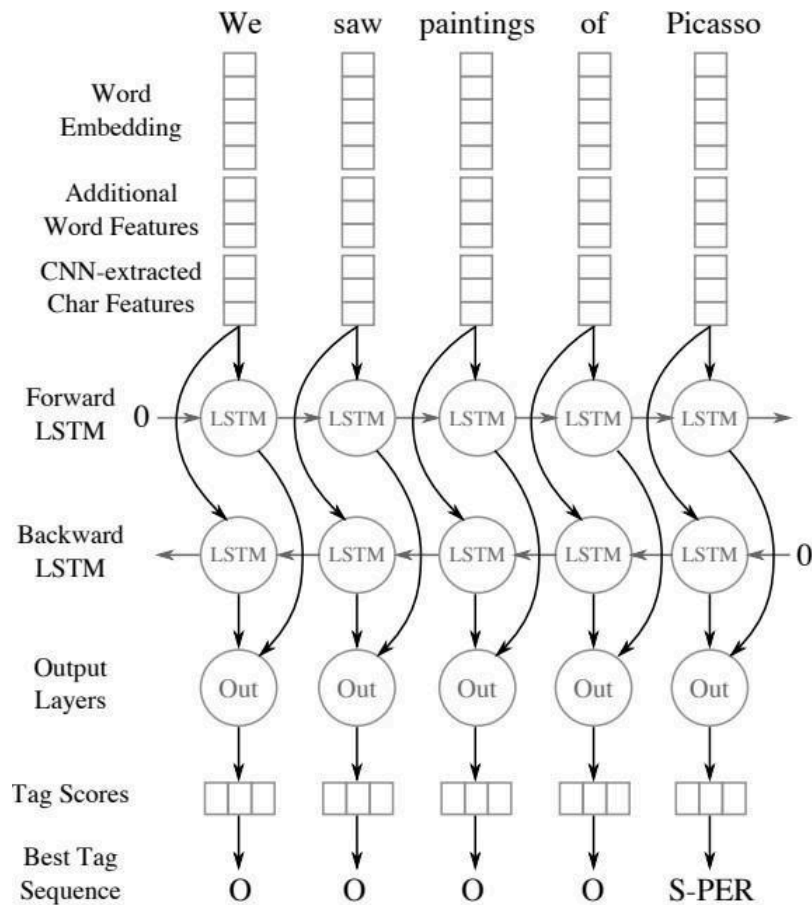


Рисунок 1.4 – Архітектура моделі BiLSTM-CNN для задачі розпізнавання іменованих сутностей [8]

Запропонований підхід демонструє високу ефективність на стандартних наборах даних, перевершуючи попередні методи, зокрема за рахунок поєднання символічного та контекстного рівнів аналізу. Водночас складність архітектури та необхідність узгодження декількох компонентів можуть ускладнювати її практичне застосування.

Розвиток підходів на основі послідовнісного маркування привів до переходу від задачі розпізнавання окремих сутностей до їх спільної екстракції разом із відношеннями між ними.

У роботі [9] запропоновано модель, що формулює задачу екстракції інформації як задачу послідовнісного маркування із використанням спеціальної схеми тегів, яка дозволяє одночасно визначати межі сутностей та типи зв'язків між ними.

Архітектура моделі базується на використанні BiLSTM для формування контекстних представлень тексту, після чого застосовується механізм декодування, що генерує послідовність тегів, які кодують як інформацію про сутності, так і про їх взаємозв'язки. Такий підхід дозволяє уникнути поділу задачі на окремі етапи та враховувати залежності між сутностями у процесі навчання.

Завдяки цьому модель здатна безпосередньо формувати структуровані представлення знань у вигляді трійок «сутність–зв'язок–сутність», що є базовим форматом у задачах екстракції інформації. Архітектура моделі, що поєднує етапи кодування та декодування для спільної екстракції, представлена на рисунку 1.5.

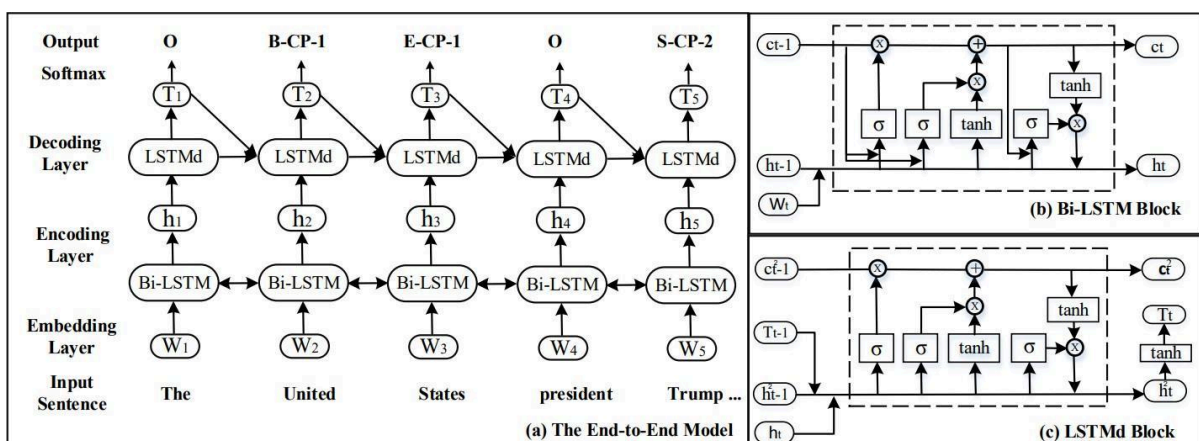


Рисунок 1.5 – Архітектура моделі спільної екстракції сутностей та зв'язків на основі BiLSTM та послідовнісного маркування [9]

Запропонований підхід демонструє ефективність за рахунок об'єднання двох взаємопов'язаних задач у межах єдиної моделі, що дозволяє зменшити накопичення помилок і підвищити узгодженість результатів. Водночас використання фіксованої схеми тегування накладає обмеження на гнучкість моделі, зокрема у випадках складних або вкладених структур сутностей, що ускладнює її застосування для більш комплексних текстових даних. Незважаючи на це, підхід є важливим етапом розвитку методів екстракції інформації, оскільки демонструє можливість ефективного переходу від послідовних pipeline-рішень до моделей спільної екстракції.

Подальший розвиток методів екстракції інформації пов'язаний із появою трансформерних моделей, які дозволяють ефективно враховувати контекст у межах усього тексту. На відміну від попередніх підходів, такі моделі використовують механізм самоуваги, що забезпечує більш глибоке розуміння семантичних зв'язків між словами та підвищує точність виділення інформації.

У роботі [10] представлено модель BERT, яка стала основою для сучасних підходів до обробки природної мови, зокрема задач екстракції інформації. Основна ідея підходу полягає у використанні двонаправленого трансформера, що дозволяє враховувати контекст кожного слова з обох боків, формуючи більш інформативні представлення тексту порівняно з попередніми моделями.

Модель проходить етап попереднього навчання на великих корпусах текстів із використанням задач маскування слів (Masked Language Modeling) та визначення взаємозв'язку між реченнями (Next Sentence Prediction). Це дозволяє сформувати універсальні контекстні embedding-и, які можуть бути адаптовані до різних прикладних задач. Для задач екстракції інформації, зокрема розпізнавання іменованих сутностей, BERT використовується у режимі донавчання, де на вихід трансформера додається класифікаційний шар для присвоєння тегів кожному токenu.

Загальна схема попереднього навчання та подальшого донавчання моделі для різних задач, включаючи екстракцію інформації, представлена на рисунку 1.6.

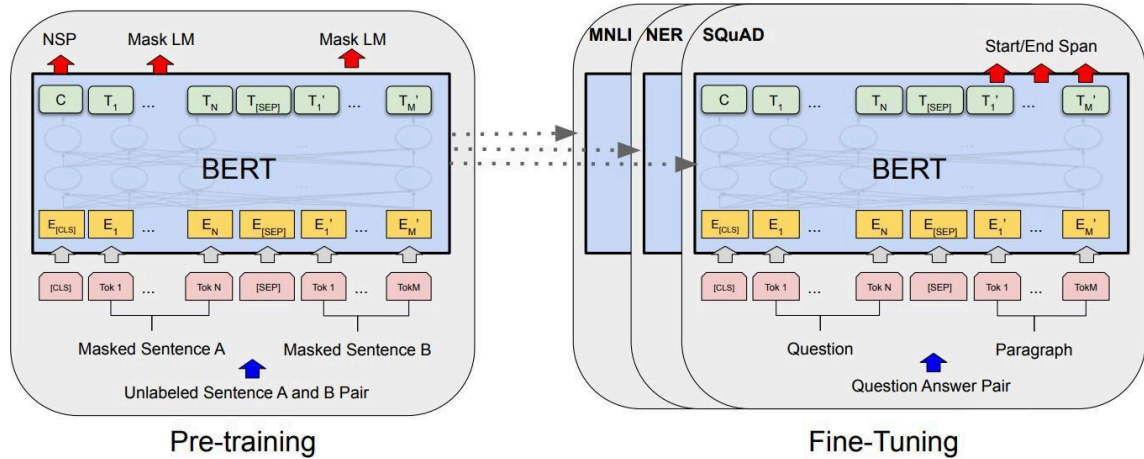


Рисунок 1.6 – Загальна схема попереднього навчання та донавчання моделі BERT [10]

На відміну від попередніх підходів, де контекст враховувався обмежено або лише в одному напрямку, BERT забезпечує глибоке моделювання залежностей між словами у всьому тексті, що є критично важливим для коректного виділення сутностей у складних або неоднозначних реченнях. Завдяки цьому модель демонструє значне підвищення точності у задачах NER, витягування фактів та інших задачах екстракції інформації.

Подальший розвиток трансформерних моделей у задачах екстракції інформації спрямований на більш точне врахування семантики сутностей у тексті.

У роботі [11] запропоновано модель LUKE, яка розширює архітектуру BERT шляхом введення окремих векторних представлень для сутностей та модифікації механізму самоуваги. Основна ідея підходу полягає в тому, що сутності розглядаються як окремі об'єкти, які мають власні embedding-и та беруть участь у процесі обчислення уваги нарівні зі словами. Це дозволяє

моделі явно враховувати взаємодію між словами та сутностями, а також між самими сутностями, що є критично важливим для задач екстракції інформації.

На відміну від класичної моделі BERT, де вся інформація представлена у вигляді послідовності токенів, LUKE доповнює вхідні дані спеціальними векторами сутностей та вводить додаткові параметри для моделювання зв'язків між ними. У процесі попереднього навчання використовуються задачі, спрямовані на відновлення замаскованих сутностей та слів, що дозволяє сформувати більш інформативні представлення. Завдяки цьому модель демонструє покращені результати у задачах розпізнавання іменованих сутностей та витягування зв'язків між ними.

Особливості попереднього навчання та механізм entity-aware self-attention, який забезпечує взаємодію між словами та сутностями, представлено на рисунку 1.7.

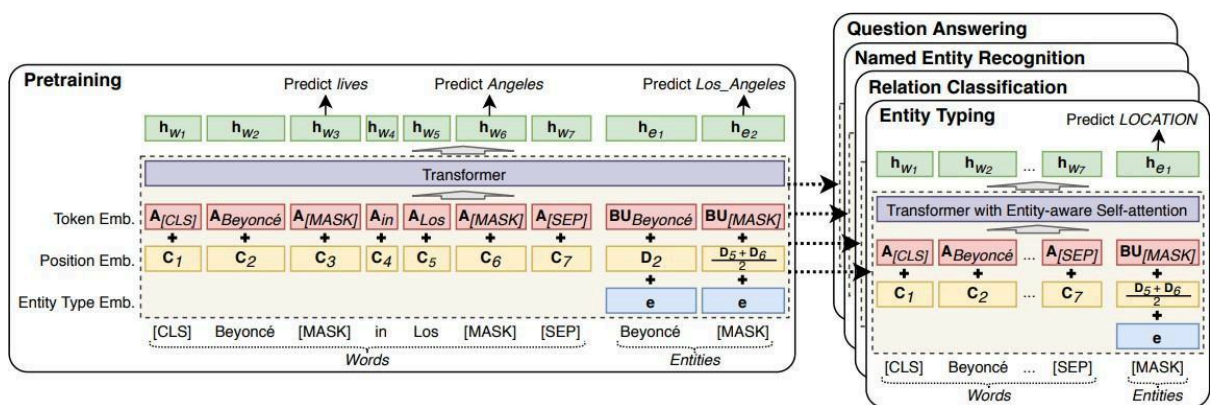


Рисунок 1.7 – Архітектура моделі LUKE з урахуванням сутностей у механізмі самоуваги [11]

У роботі [12] запропоновано модель LayoutLM, орієнтовану на задачі екстракції інформації з документів, яка поєднує текстові представлення з просторовими та візуальними ознаками. Основна ідея підходу полягає у

врахуванні не лише змісту тексту, а й його розташування на сторінці, що дозволяє моделювати структуру документа як цілісну систему взаємопов'язаних елементів.

На відміну від стандартних трансформерних моделей, у LayoutLM кожному токеноу додатково ставляться у відповідність координати його розміщення (bounding box), які кодуються у вигляді позиційних embedding-ів. Це дозволяє враховувати відносне розташування текстових блоків, що є важливим для інтерпретації документів, де значення інформації залежить від її позиції (наприклад, у таблицях або структурованих формах). Додатково використовується інформація, отримана за допомогою OCR, що забезпечує інтеграцію текстових та візуальних характеристик документа.

Модель поєднує текстові embedding-и, просторові координати та візуальні ознаки в єдиному представленні, яке використовується для подальших задач, таких як розпізнавання сутностей, класифікація блоків та екстракція структурованої інформації. Архітектура моделі, що інтегрує різні типи ознак у межах трансформерної структури, представлена на рисунку 1.8.

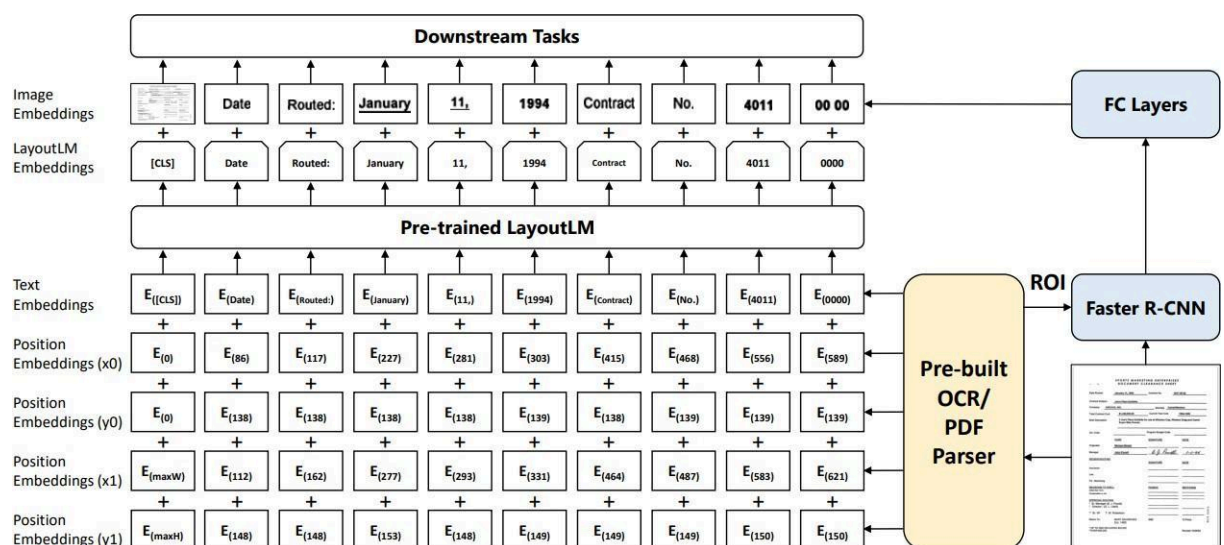


Рисунок 1.8 – Архітектура моделі LayoutLM із урахуванням текстових, просторових та візуальних ознак документа [12]

У роботі [13], де запропоновано модель LayoutLMv2, що розширює можливості LayoutLM шляхом інтеграції мультимодальних ознак. На відміну від попередньої моделі, у цьому підході поєднуються текстові представлення, просторове розташування елементів документа та візуальні характеристики, що отримуються за допомогою окремого візуального енкодера.

Архітектура моделі та основні етапи її попереднього навчання, що поєднують текстові, просторові та візуальні ознаки документа, представлено на рисунку 1.9.

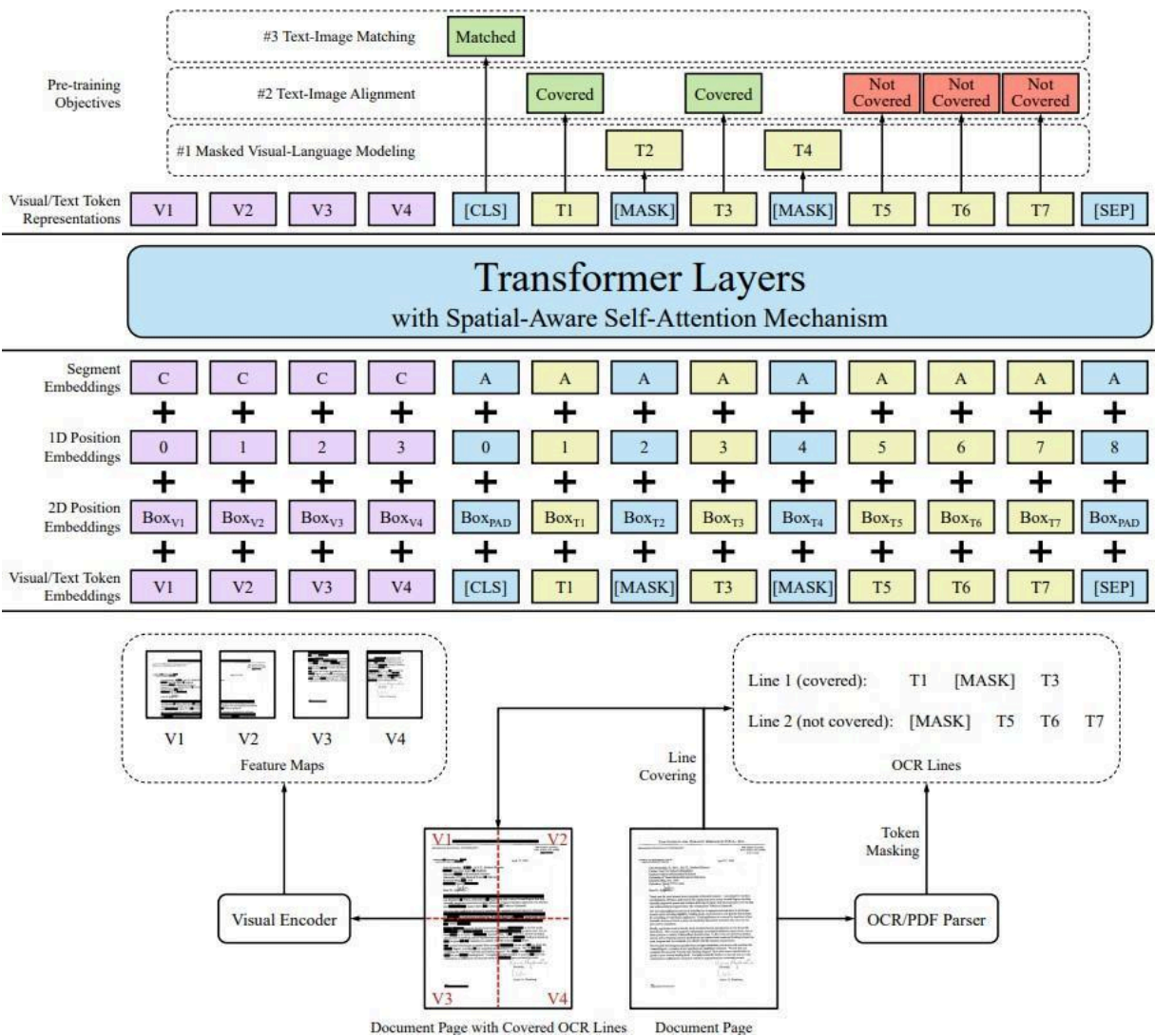


Рисунок 1.9 – Архітектура моделі LayoutLMv2 та стратегії мультимодального попереднього навчання [13]

Ключовою особливістю LayoutLMv2 є використання механізму просторово-орієнтованої самоуваги, який дозволяє враховувати взаємне розташування текстових елементів на сторінці, а також застосування спеціальних стратегій попереднього навчання, спрямованих на узгодження текстових та візуальних ознак.

У роботі [14] запропоновано підхід TPLinker, який базується на механізмі так званого «зв'язування пар токенів» (token pair linking) для одночасного визначення меж сутностей і відношень між ними.

Загальна схема роботи моделі TPLinker, що ілюструє процес формування зв'язків між токенами та подальшу екстракцію структурованих відношень, наведена на рисунку 1.10.

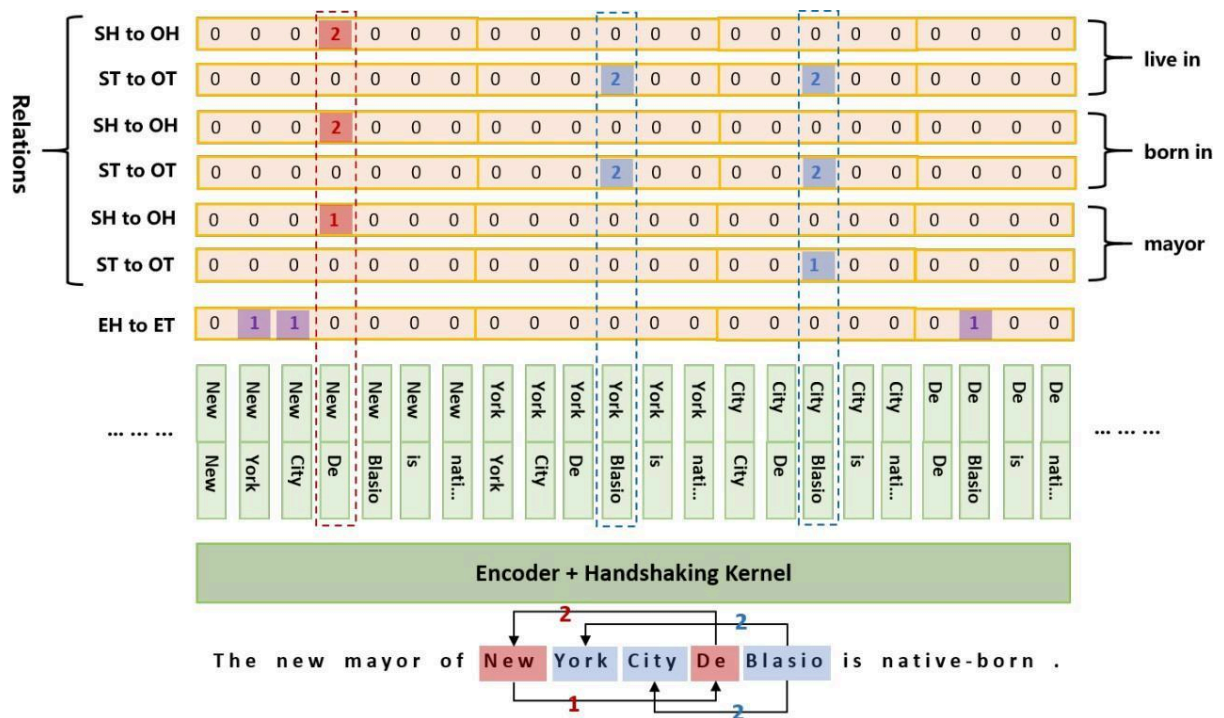


Рисунок 1.10 – Схема моделі TPLinker для спільної екстракції сутностей та відношень на основі механізму зв'язування пар токенів [14]

На відміну від традиційних pipeline-підходів, де спочатку визначаються сутності, а потім окремо встановлюються зв'язки, TPLinker представляє текст у вигляді матриці можливих пар токенів і виконує їх

спільну класифікацію. Такий підхід дозволяє уникнути накопичення помилок між етапами та забезпечує більш узгоджене виділення структурованої інформації.

Архітектура моделі включає енкодер, що формує контекстні представлення токенів, та спеціальний механізм «handshaking tagging», який встановлює зв'язки між початками та кінцями сутностей, а також між пов'язаними елементами тексту. Це дозволяє моделі одночасно визначати як самі сутності, так і типи відношень між ними, формуючи структуровані трійки вигляду «сутність-зв'язок-сутність».

1.3 Постановка задачі дослідження

Проведений аналіз сучасних підходів до екстракції інформації з текстових документів свідчить про еволюцію методів від простих правил та статистичних моделей до складних нейромережових архітектур, здатних враховувати контекст, структуру та взаємозв'язки між елементами тексту. При цьому класичні підходи характеризуються обмеженою гнучкістю, методи машинного навчання забезпечують кращу узагальнювальну здатність, тоді як сучасні трансформерні моделі демонструють найвищу ефективність завдяки глибокому врахуванню контексту та семантики.

Водночас жоден із розглянутих підходів не є універсальним, оскільки ефективність екстракції значною мірою залежить від типу документів, структури тексту та специфіки предметної галузі.

Метою роботи є дослідження та порівняння методів екстракції даних з неструктурованих текстових документів на прикладі текстових резюме.

Для досягнення поставленої мети необхідно вирішити такі задачі:

- проаналізувати особливості неструктурованих текстових документів та їх вплив на процес екстракції інформації;

- розглянути існуючі методи екстракції інформації на основі rule-based підходів, моделей машинного навчання та трансформерних архітектур;
- виконати підготовку та попередню обробку даних для задачі розпізнавання іменованих сутностей;
- реалізувати методи екстракції інформації з використанням різних підходів;
- провести експериментальне дослідження ефективності реалізованих методів;
- виконати порівняльний аналіз результатів та визначити найбільш ефективний підхід.

МЕТОДИ ЕКСТРАКЦІЇ ІНФОРМАЦІЇ З НЕСТРУКТУРОВАНИХ ТЕКСТОВИХ ДОКУМЕНТІВ

Сучасні задачі обробки текстової інформації характеризуються значною складністю, зумовленою різноманітністю форматів представлення даних та відсутністю уніфікованої структури. Особливо це стосується документів, що створюються у довільній формі, де логічні зв'язки між елементами тексту не завжди явно виражені. У таких умовах ефективна екстракція структурованої інформації потребує застосування спеціалізованих методів, здатних враховувати як синтаксичні, так і семантичні особливості тексту.

Розвиток підходів до розв'язання цієї задачі відбувався від використання жорстко заданих правил до впровадження методів машинного навчання та сучасних нейромережових моделей. Кожен із цих підходів має власні переваги та обмеження, що визначають доцільність його застосування залежно від специфіки даних та поставленої задачі. Це зумовлює необхідність їх детального розгляду та порівняння з метою обґрунтування вибору методів для подальшої реалізації.

2.1 Особливості екстракції інформації з текстових резюме

Текстові документи у більшості практичних задач обробки інформації характеризуються відсутністю чітко визначеної структури, що ускладнює їх автоматизований аналіз. На відміну від структурованих даних, де кожен елемент має фіксоване місце, формат і тип, текст подається у вільній формі без жорстко заданої схеми. Це призводить до неоднорідності представлення інформації, варіативності синтаксичних конструкцій та відсутності єдиних правил оформлення, що ускладнює застосування традиційних алгоритмів обробки даних та потребує використання спеціалізованих методів аналізу природної мови.

Резюме як окремий тип текстових документів займає проміжне положення між структурованими та неструктурованими даними. З одного боку, воно зазвичай містить типові логічні блоки, такі як особисті дані, освіта, досвід роботи та навички, що формують певну семантичну структуру документа. З іншого боку, спосіб подання цих блоків не є стандартизованим, що призводить до значної варіативності у форматуванні, порядку розташування та способах опису інформації [15]. У межах одного документа ці блоки можуть бути представлені як у вигляді чітко відокремлених секцій, так і у вигляді змішаного тексту, що ускладнює їх автоматичне виділення.

Додатковою складністю є різноманітність стилів оформлення резюме. Один і той самий тип інформації може бути представлений у різних формах: у вигляді суцільного тексту, маркованих або нумерованих списків, таблиць або коротких фраз. Крім того, назви секцій можуть суттєво відрізнятися (наприклад, «Work Experience», «Professional Background», «Career History») або взагалі бути відсутніми, що ускладнює їх однозначну ідентифікацію під час автоматичної обробки. Така варіативність призводить до того, що одна й та сама інформація може мати різні лінгвістичні та структурні представлення, які необхідно коректно інтерпретувати.

Суттєвою проблемою також є неоднозначність інтерпретації окремих фрагментів тексту. Одні й ті самі слова або словосполучення можуть відповідати різним сутностям залежно від контексту. Наприклад, назви організацій можуть збігатися із загальноживаними словами, а назви навичок – використовуватися у різних значеннях [16]. Аналогічно, назви посад можуть перетинатися з назвами напрямів діяльності або технологій. Це вимагає врахування контекстуального оточення, синтаксичних зв'язків та семантичних залежностей для коректного визначення типу інформації.

Окремо слід відзначити проблему варіативності довжини та структури сутностей. Деякі з них можуть складатися з одного слова (наприклад, назва технології), тоді як інші є багатокomпонентними конструкціями (наприклад, повна назва посади або організації). Це



ускладнює задачу визначення

точних меж сутностей та потребує використання моделей, здатних працювати з послідовностями різної довжини.

Таким чином, задача екстракції структурованої інформації з текстових резюме ускладнюється поєднанням напівструктурованості, варіативності представлення, неоднозначності інтерпретації та контекстної залежності даних. Це обґрунтовує необхідність застосування методів, здатних адаптуватися до різних форматів тексту, враховувати його семантичні особливості та забезпечувати узагальнення на нові приклади.

2.2 Класичні підходи до екстракції інформації

Класичні підходи до екстракції інформації з текстових документів ґрунтуються на використанні наперед визначених правил та шаблонів, що дозволяють виділяти необхідні фрагменти тексту за заданими ознаками. Такі методи передбачають формалізацію структури даних у вигляді набору умов, які перевіряються під час обробки тексту, зокрема із застосуванням регулярних виразів, словників та шаблонів пошуку. У межах задач екстракції інформації це означає, що система не «розуміє» текст, а виконує пошук відповідностей між вхідними даними та заздалегідь заданими правилами.

Одним із найбільш поширених інструментів у межах цього підходу є регулярні вирази, які ефективно застосовуються для виявлення інформації з чітко визначеним форматом, такої як адреси електронної пошти, номери телефонів або дати. Використання регулярних виразів дозволяє формалізувати шаблони пошуку у вигляді спеціальних конструкцій, що описують послідовності символів та їх допустимі варіації [17]. Це робить їх особливо корисними у задачах, де структура даних є строго визначеною та не допускає значної варіативності.

Завдяки цьому регулярні вирази забезпечують високу точність у задачах, де структура шуканих даних є стабільною та добре

визначеною. Крім того, вони характеризуються високою швидкістю обробки та не потребують значних обчислювальних ресурсів, що робить їх зручними для використання у простих системах обробки тексту або як складова більш складних рішень.

Водночас ефективність регулярних виразів суттєво знижується у випадках, коли інформація представлена у довільній формі або містить варіації у написанні. Навіть незначні зміни у форматі тексту, такі як пропуски, додаткові символи або інший порядок слів, можуть призвести до некоректного спрацювання шаблону або пропуску необхідної інформації. У задачах обробки резюме це проявляється особливо яскраво, оскільки один і той самий тип даних може бути представлений у великій кількості варіантів. До класичних підходів також належать словникові методи, що базуються на використанні заздалегідь сформованих списків ключових слів або термінів. У таких підходах процес екстракції полягає у пошуку збігів між елементами тексту та записами у словнику, що дозволяє ідентифікувати певні типи сутностей. При цьому текст зазвичай проходить попередню обробку, яка може включати нормалізацію, приведення до нижнього регістру або видалення службових символів.

Наприклад, для визначення навичок може використовуватися перелік технологій, мов програмування або інструментів, з яким порівнюється текст документа [18]. Подібні методи часто застосовуються у поєднанні з простими правилами обробки, що дозволяє підвищити їх точність у межах обмеженої предметної галузі.

Перевагою словникових підходів є можливість швидкого впровадження та зрозумілості результатів, оскільки кожне знайдене значення безпосередньо відповідає елементу словника. Крім того, такі методи легко контролювати та модифікувати, що є важливим у прикладних задачах, де необхідно забезпечити передбачуваність результатів.

Однак їх застосування обмежується повнотою та актуальністю самого словника. У випадках, коли текст містить нові терміни, синоніми,

скорочення або нетипові формулювання, такі методи можуть не виявляти релевантну інформацію. Це особливо актуально для резюме, де часто використовуються індивідуальні формулювання та нестандартні позначення.

Перевагами класичних підходів у цілому є їхня простота реалізації, прозорість логіки роботи та відсутність потреби у навчанні на розмічених даних. Результат роботи таких систем легко інтерпретується, оскільки кожне рішення базується на чітко визначених правилах або збігах із шаблонами. Це робить їх доцільними для використання у задачах, де необхідно забезпечити високу точність та контрольованість результатів.

Крім того, подібні методи забезпечують стабільні результати у межах вузько визначених задач, де структура тексту є відносно сталою. У таких умовах вони можуть виступати ефективним інструментом або базовим компонентом більш складних систем екстракції інформації.

Водночас основним недоліком класичних підходів є їхня низька гнучкість та слабка здатність до узагальнення. Будь-які зміни у структурі або формулюванні тексту потребують ручного коригування правил, що ускладнює масштабування та підтримку таких рішень. Зі збільшенням обсягу та різноманітності даних кількість необхідних правил зростає, що призводить до підвищення складності системи, зниження її ефективності та зростання витрат на підтримку.

2.3 Методи машинного навчання для екстракції інформації

Методи машинного навчання стали логічним етапом розвитку підходів до екстракції інформації, оскільки дозволили відійти від жорстко заданих правил і перейти до моделей, здатних самостійно виявляти закономірності у даних. На відміну від класичних підходів, такі методи базуються на навчанні з використанням розмічених текстових корпусів, що дає змогу враховувати різноманітність способів подання інформації у

документах та адаптуватися до варіативності тексту без необхідності ручного задання правил.

Задача екстракції інформації в межах методів машинного навчання зазвичай формулюється як задача послідовнісного маркування (sequence labeling), у якій кожному токenu тексту ставиться у відповідність певна мітка, що визначає його належність до конкретного типу сутності або відсутність такої. Такий підхід дозволяє моделі враховувати не лише окремі слова, а й їхнє оточення в межах тексту, що є критично важливим для коректного визначення меж сутностей [19]. Зокрема, у задачах обробки резюме це дозволяє розрізняти схожі за формою фрагменти тексту залежно від контексту їх використання.

До класичних моделей машинного навчання, що застосовуються для розв'язання задачі екстракції інформації, належать, зокрема, приховані марковські моделі (HMM) та умовні випадкові поля (CRF). Ці моделі орієнтовані на обробку послідовностей і дозволяють враховувати залежності між сусідніми елементами тексту, що є критично важливим у задачах типу sequence labeling [20].

Приховані марковські моделі ґрунтуються на припущенні, що спостережувана послідовність слів породжується прихованою послідовністю станів, яка відповідає міткам сутностей. Вони моделюють імовірності переходів між станами та появи слів, що дозволяє визначати найбільш ймовірну послідовність міток для заданого тексту. Однак такі моделі мають обмеження, пов'язані з необхідністю робити спрощувальні припущення щодо незалежності ознак, що знижує їх ефективність у складних текстах із великою кількістю варіацій.

Більш гнучким підходом є умовні випадкові поля, які дозволяють безпосередньо моделювати умовну ймовірність послідовності міток за заданим текстом. На відміну від HMM, CRF можуть враховувати широкий набір ознак для кожного елемента послідовності, включаючи як локальні характеристики слова (форма, регістр, суфікси), так і інформацію про його

контекст. Це дає змогу точніше визначати межі сутностей та зменшувати кількість помилок, пов'язаних із некоректними комбінаціями міток.

Завдяки здатності враховувати залежності між сусідніми мітками такі моделі забезпечують узгодженість результатів у межах всієї послідовності, що є суттєвою перевагою порівняно з rule-based підходами, які зазвичай обробляють окремі фрагменти тексту незалежно один від одного. Це дозволяє уникати ситуацій, коли частина сутності визначається коректно, а інша – пропускається або класифікується неправильно.

Водночас ефективність методів машинного навчання значною мірою залежить від якості та обсягу навчальних даних. У випадку недостатньої кількості прикладів або нерівномірного розподілу сутностей моделі можуть демонструвати нестабільні результати та гіршу здатність до узагальнення. Крім того, класичні підходи машинного навчання потребують ручного формування ознак, що ускладнює їх застосування у задачах із високою варіативністю тексту.

Таким чином, незважаючи на суттєве покращення якості порівняно з класичними rule-based методами, підходи машинного навчання мають обмеження, пов'язані з необхідністю ручного інжинірингу ознак та обмеженою здатністю враховувати складні контекстні залежності, що обґрунтовує подальший розвиток більш потужних моделей обробки тексту.

Подальшим розвитком підходів до екстракції інформації стали нейромережеві методи, зокрема двонаправлені рекурентні нейронні мережі, такі як BiLSTM. На відміну від класичних моделей машинного навчання, вони не потребують явного задання ознак, оскільки здатні автоматично формувати внутрішні представлення тексту безпосередньо на основі даних.

Це дозволяє уникнути трудомісткого процесу інжинірингу ознак та підвищує здатність моделей адаптуватися до різних типів текстів.

Рекурентні нейронні мережі призначені для обробки послідовностей і передають інформацію від одного елемента до наступного, що дозволяє враховувати контекст у межах тексту. Проте стандартні RNN мають

обмеження, пов'язані зі складністю збереження інформації на довгих відрізках послідовності, що ускладнює обробку текстів зі складною структурою. Для подолання цієї проблеми використовуються мережі типу LSTM, які містять спеціальні механізми керування потоком інформації, що дозволяє зберігати важливі залежності протягом довших фрагментів тексту.

Двонаправлена архітектура BiLSTM поєднує два процеси обробки: один у прямому напрямку (зліва направо), інший – у зворотному (справа наліво). Це дозволяє моделі враховувати як попередній, так і наступний контекст слова, що є особливо важливим у задачах розпізнавання іменованих сутностей, де значення слова часто визначається його оточенням. Такий підхід забезпечує більш повне представлення контексту порівняно з однонаправленими моделями.

У поєднанні з шаром умовних випадкових полів (рисунок 2.1) такі моделі забезпечують узгодженість передбачень у межах всієї послідовності.

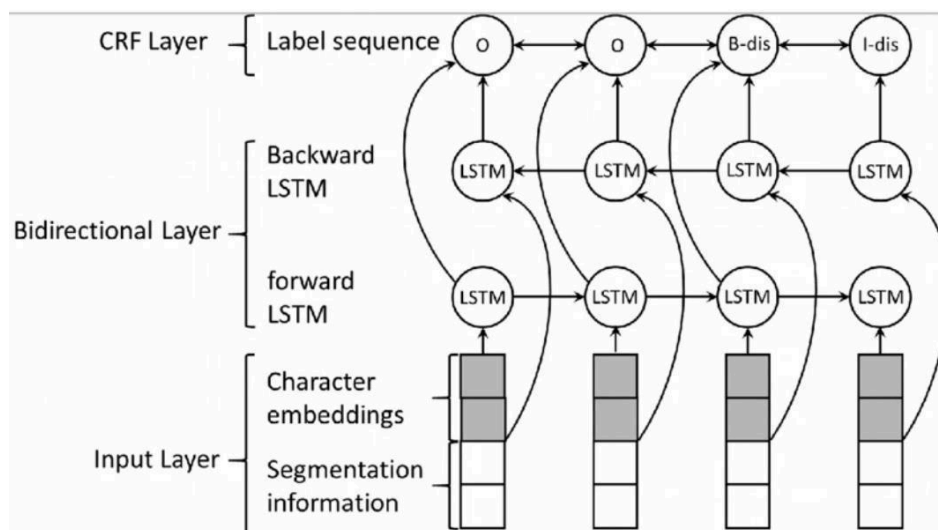


Рисунок 2.1 – Архітектура моделі BiLSTM-CRF [21]

Якщо BiLSTM відповідає за формування контекстно залежних представлень слів, то CRF дозволяє враховувати залежності між мітками та уникати некоректних їх комбінацій. Це особливо важливо у задачах sequence labeling, де правильність окремої мітки залежить від сусідніх. Таке

поєднання стало одним із базових підходів у задачах розпізнавання сутностей до появи більш сучасних моделей.

Попри суттєве покращення якості екстракції порівняно з класичними методами, нейромережеві підходи також мають певні обмеження. Вони потребують значної кількості розмічених даних для навчання, а також характеризуються складнішою архітектурою та більшими обчислювальними витратами. Крім того, навіть моделі типу BiLSTM мають обмеження у врахуванні довгострокових залежностей, особливо у випадку довгих або складно структурованих текстів.

У задачах обробки резюме це проявляється у складності коректного визначення сутностей, що залежать від віддаленого контексту або представлені у нетиповій формі. Зокрема, взаємозв'язки між різними частинами документа можуть не враховуватися повною мірою, що впливає на якість розпізнавання.

Таким чином, нейромережеві методи, зокрема архітектура BiLSTM-CRF, забезпечують більш гнучкий та ефективний підхід до екстракції інформації порівняно з класичними моделями машинного навчання. Водночас їх обмеження, пов'язані з обробкою довгострокових залежностей та складною структурою тексту, обґрунтовують подальший розвиток більш потужних моделей, здатних краще враховувати глобальний контекст.

2.4 Трансформерні моделі для задач обробки тексту

Обмеження методів машинного навчання, пов'язані з недостатнім врахуванням довгострокових залежностей у тексті та обмеженою здатністю до моделювання складного контексту, зумовили розвиток нових підходів до обробки природної мови. Зокрема, навіть моделі типу BiLSTM, незважаючи на використання механізмів пам'яті, демонструють зниження ефективності при роботі з довгими або складно структурованими текстами, де залежності між словами можуть бути значно віддаленими. Це стало передумовою для

появи трансформерних моделей, які на сьогодні є домінуючими у більшості задач аналізу текстових даних.

Ключовою ідеєю трансформерів є використання механізму самоуваги (self-attention), який дозволяє оцінювати важливість кожного слова у тексті відносно інших слів у межах тієї ж послідовності. На відміну від рекурентних моделей, де обробка здійснюється послідовно і залежить від порядку проходження тексту, трансформери встановлюють зв'язки між усіма словами одночасно. Це забезпечує більш ефективне врахування глобального контексту та дозволяє моделі виявляти залежності навіть між віддаленими елементами тексту, що є критично важливим у задачах екстракції інформації.

Механізм self-attention працює шляхом обчислення ваг для кожного слова відносно інших слів у реченні. Таким чином модель визначає, які частини тексту є найбільш релевантними для інтерпретації конкретного токена. Це дозволяє виділяти значущі зв'язки між словами незалежно від їх позиції у тексті, що суттєво відрізняє трансформери від попередніх підходів. Крім того, використання багатоголової уваги (multi-head attention) дозволяє одночасно враховувати різні типи залежностей, наприклад синтаксичні та семантичні.

Важливою особливістю трансформерних моделей є формування контекстуально залежних представлень слів. Це означає, що векторне представлення одного й того самого слова змінюється залежно від його оточення у тексті. Такий підхід дозволяє більш точно інтерпретувати значення слів у різних контекстах, що є критичним для задач екстракції інформації, де правильне визначення сутностей часто залежить від контексту їх використання. Наприклад, одна й та сама назва може позначати як організацію, так і загальне поняття, і лише контекст дозволяє зробити правильний вибір.

Загальна архітектура трансформерної моделі наведена на рисунку 2.2. Вона включає послідовність блоків самоуваги та повнозв'язних шарів, які

забезпечують формування контекстних представлень тексту. Крім того, у таких моделях використовуються позиційні кодування (positional encoding), які дозволяють враховувати порядок слів у послідовності, оскільки механізм самоуваги сам по собі не містить інформації про позицію елементів.

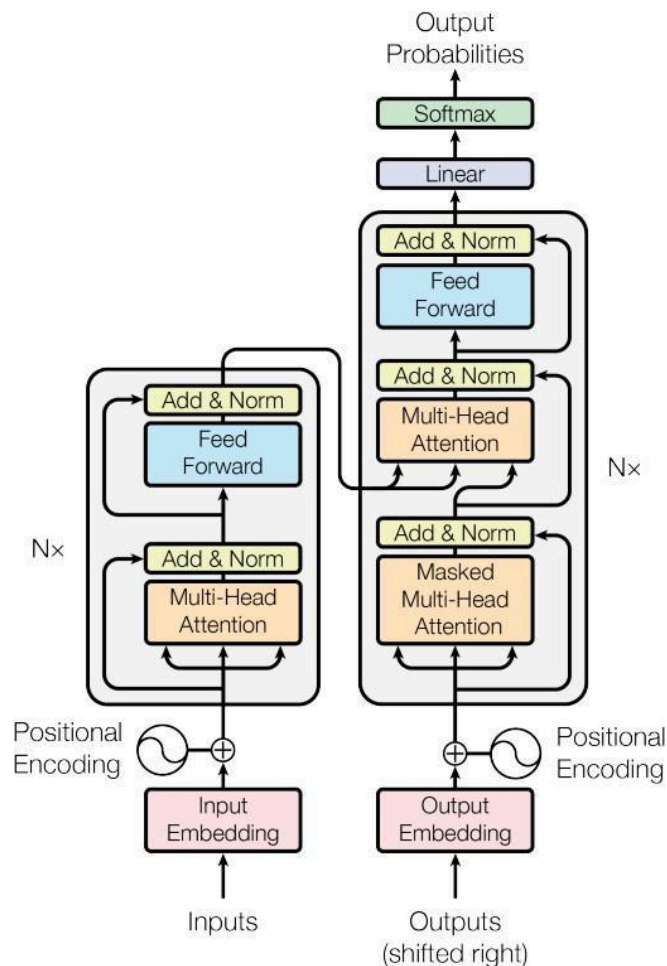


Рисунок 2.2 – Узагальнена архітектура трансформерної моделі [22]

Суттєву роль у поширенні трансформерів відіграє концепція попереднього навчання (pretraining) на великих обсягах текстових даних з подальшим донавчанням (fine-tuning) для конкретної задачі. Попереднє навчання дозволяє моделі засвоїти загальні мовні закономірності, такі як синтаксис та семантика, тоді як донавчання адаптує ці знання до специфіки конкретної предметної галузі. Такий підхід значно зменшує потребу у великих обсягах розмічених даних та підвищує якість результатів.

У задачах екстракції інформації, зокрема для текстових резюме, це має особливе значення, оскільки дозволяє ефективно працювати навіть за обмеженої кількості анотованих прикладів. Модель, попередньо навчена на великих текстових корпусах, уже має базове розуміння мовних конструкцій, що дозволяє їй швидше адаптуватися до нових даних.

Серед найбільш відомих трансформерних моделей, що застосовуються у задачах обробки тексту, можна виділити BERT, RoBERTa та DistilBERT. Модель BERT заклала основу підходу з використанням глибоких двонапрямлених представлень тексту, що дозволяє враховувати контекст з обох боків від слова. RoBERTa є її вдосконаленою версією з оптимізованим процесом навчання та покращеною якістю результатів. DistilBERT, у свою чергу, є спрощеною моделлю, що зберігає більшу частину якості при значно меншій обчислювальній складності, що робить її придатною для практичних застосувань.

Таким чином, трансформерні моделі забезпечують якісно новий рівень обробки текстових даних порівняно з попередніми підходами, поєднуючи ефективне врахування глобального контексту, гнучкість представлення слів та можливість перенесення знань між задачами. Це робить їх найбільш доцільним вибором для задач екстракції структурованої інформації з неструктурованих текстових документів, зокрема текстових резюме, де важливо враховувати складні контекстні залежності та варіативність представлення інформації.

2.5 Методи розпізнавання іменованих сутностей (NER)

Задача розпізнавання іменованих сутностей (Named Entity Recognition, NER) є однією з ключових у сфері обробки природної мови та лежить в основі більшості систем екстракції інформації. Вона полягає у виявленні у тексті фрагментів, що відповідають певним категоріям сутностей, та віднесенні їх до відповідних класів, таких як імена осіб,

організації, локації, дати або інші типи даних. Саме NER забезпечує перехід від суцільного тексту до структурованого представлення інформації, яке може бути використане для подальшого аналізу, збереження або інтеграції у інформаційні системи.

Фактично, NER виконує роль базового рівня інтерпретації тексту, оскільки дозволяє виділити найбільш значущі елементи документа та надати їм формалізованого вигляду. Без етапу розпізнавання сутностей подальша автоматизована обробка тексту є суттєво ускладненою або неможливою, оскільки інформація залишається у неструктурованому вигляді.

У загальному випадку перелік сутностей залежить від конкретної задачі та предметної галузі. У контексті аналізу текстових резюме доцільно виділяти кілька основних груп сутностей: персональні дані (ім'я, контактна інформація), освітні дані (назви навчальних закладів, спеціальності), професійний досвід (назви компаній, посади), а також навички та інструменти. Така класифікація дозволяє сформувати структуровану модель даних, яка відображає ключові характеристики кандидата та може бути використана у системах підбору персоналу, аналітичних сервісах або автоматизованих системах обробки резюме.

Формально задача NER розглядається як задача послідовнісного маркування, у якій кожному токеноу тексту присвоюється мітка, що визначає його належність до певної сутності або її відсутність. Такий підхід дозволяє враховувати не лише окремі слова, а й їх положення та взаємозв'язки у межах тексту. У цьому контексті модель розглядає текст як послідовність взаємопов'язаних елементів, де правильне визначення однієї мітки залежить від сусідніх.

Для реалізації цього підходу використовуються спеціальні схеми розмітки, серед яких найбільш поширеною є BIO-схема. Вона передбачає використання трьох типів міток: B (beginning) – початок сутності, I (inside)

продовження сутності, та O (outside) – токен, що не належить до жодної сутності (рисунок 2.3).

Winnie	B-PERSON
the	I-PERSON
Pooh	I-PERSON
is	O
naïve	O
and	O
slow-witted	O

Рисунок 2.3 – Приклад схеми BIO [23]

Застосування BIO-схеми дозволяє коректно обробляти багатослівні сутності, які часто зустрічаються у текстових документах. Наприклад, назви організацій або навчальних закладів можуть складатися з кількох слів, і правильне визначення їх меж є важливим для подальшої інтерпретації. Крім того, така схема дозволяє уникати неоднозначностей, коли окремі слова можуть належати до різних сутностей залежно від контексту.

Процес розпізнавання іменованих сутностей є складовою більш загального процесу екстракції інформації, що включає кілька етапів обробки тексту. Як показано на рисунку 2.4, цей процес може включати сегментацію тексту, токенізацію, визначення частин мови, формування ознак та безпосереднє застосування моделі розпізнавання. Кожен із цих етапів впливає на кінцевий результат, оскільки помилки на ранніх стадіях можуть призводити до некоректного визначення сутностей.

У задачі екстракції інформації з текстових резюме NER виступає базовим інструментом, що забезпечує виділення ключових елементів документа. Особливістю цієї задачі є висока варіативність представлення інформації, що вимагає від моделі здатності узагальнювати та коректно працювати з різними форматами тексту. Отримані сутності можуть бути використані для формування структурованих записів, які надалі

застосовуються у різноманітних інформаційних системах, зокрема системах підбору персоналу, рекомендаційних сервісах або аналітичних платформах.

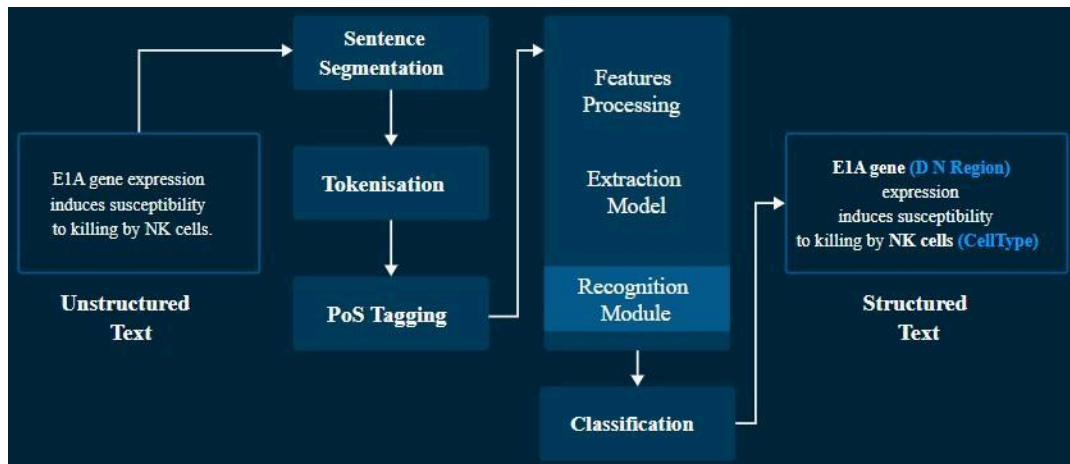


Рисунок 2.4 – Загальна схема процесу екстракції інформації з тексту з використанням NER [24]

Таким чином, ефективність роботи системи екстракції інформації значною мірою визначається якістю розпізнавання іменованих сутностей. Саме на цьому етапі формується структуроване представлення даних, від якого залежить точність подальшого аналізу та практична придатність всієї системи.

2.6 Метрики оцінювання якості екстракції

Оцінювання якості екстракції інформації є важливим етапом дослідження, оскільки дозволяє кількісно визначити ефективність застосованих методів. У задачах розпізнавання іменованих сутностей результати роботи моделі зазвичай порівнюються з еталонною розміткою, що дає змогу визначити правильність передбачень для кожного класу сутностей.

Основними метриками оцінювання є точність (precision), повнота (recall) та F1-міра, які базуються на поняттях істинно

позитивних (TP), хибно позитивних (FP) та хибно негативних (FN) передбачень.

Точність характеризує частку правильно визначених сутностей серед усіх знайдених моделлю і визначається як:

$$Precision = \frac{TP}{TP+FP} \quad (2.1)$$

Ця метрика показує, наскільки отримані результати є коректними, тобто чи не включає модель зайві або помилкові сутності.

Повнота відображає здатність моделі знаходити всі релевантні сутності у тексті та визначається як:

$$Recall = \frac{TP}{TP+FN} \quad (2.2)$$

Високе значення повноти свідчить про те, що модель не пропускає важливі елементи, проте саме по собі не гарантує відсутності помилкових спрацьовувань.

Для узагальненої оцінки якості використовується F1-міра, яка є гармонійним середнім точності та повноти:

$$F1 = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)} \quad (2.3)$$

Ця метрика дозволяє врахувати баланс між точністю та повнотою, що є особливо важливим у задачах екстракції інформації, де необхідно одночасно мінімізувати як пропуски сутностей, так і кількість хибних



визначень. У задачах розпізнавання іменованих сутностей оцінювання зазвичай здійснюється на рівні цілих сутностей, а не окремих токенів. Правильним вважається лише той випадок, коли модель коректно визначає як межі сутності, так і її тип, що відповідає підходу exact match [25].

Використання метрики асигасу у даному контексті є обмеженим, оскільки вона враховує всі токени, включаючи ті, що не належать до жодної сутності. У задачах NER більшість tokenів зазвичай мають мітку O, що може призводити до завищення значення асигасу навіть при наявності значної кількості помилок у визначенні сутностей.

Таким чином, для оцінювання якості екстракції інформації доцільно використовувати комбінацію метрик precision, recall та F1-міри, що дозволяє отримати об'єктивне уявлення про ефективність застосованих методів та коректно порівняти їх між собою.

3 РЕАЛІЗАЦІЯ МЕТОДІВ ЕКСТРАКЦІЇ ІНФОРМАЦІЇ

У процесі реалізації системи екстракції даних з неструктурованих текстових документів було використано низку програмних засобів і бібліотек, вибір яких обумовлений специфікою задачі, необхідністю роботи з текстовими даними та реалізації моделей обробки природної мови.

Основною мовою програмування обрано Python, оскільки вона є однією з найбільш поширених у сфері обробки даних та штучного інтелекту. Python має велику кількість готових бібліотек для роботи з текстом, машинним навчанням і глибоким навчанням, що значно спрощує реалізацію складних алгоритмів та дозволяє зосередитися на дослідницькій частині роботи.

Для реалізації задачі розпізнавання іменованих сутностей було використано бібліотеку spaCy. Вибір spaCy зумовлений її високою швидкістю роботи, зручністю інтеграції та наявністю готових інструментів для навчання моделей NER. Крім того, бібліотека підтримує роботу з власними наборами даних та дозволяє гнучко налаштовувати процес навчання.

Для реалізації трансформерних моделей використано бібліотеку HuggingFace Transformers. Ця бібліотека надає доступ до великої кількості попередньо навчених моделей, зокрема BERT та DistilBERT, що дозволяє уникнути навчання моделей з нуля та значно скоротити час розробки. Використання предтренуваних моделей забезпечує високу якість обробки тексту завдяки вже засвоєним мовним закономірностям.

Бібліотека PyTorch використовується як основа для роботи з нейронними мережами. Вона забезпечує гнучкість у побудові моделей, підтримує використання графічних процесорів та добре інтегрується з бібліотекою Transformers. Це дозволяє ефективно виконувати навчання моделей та обробку великих обсягів даних.

Для роботи з даними та їх аналізу використано бібліотеки pandas та NumPy. Вони дозволяють зручно обробляти табличні дані, виконувати статистичний аналіз та готувати результати до подальшої інтерпретації.

Для оцінювання якості моделей використано бібліотеку scikit-learn, яка містить реалізації основних метрик та інструменти для аналізу результатів. Це дозволяє стандартизовано порівнювати різні підходи та оцінювати їх ефективність.

Для навчання та експериментів із моделями було використано середовище Google Colab. Вибір цього інструменту обумовлений можливістю використання обчислювальних ресурсів, зокрема графічних процесорів, без необхідності налаштування локального середовища. Це дозволяє значно пришвидшити процес навчання трансформерних моделей та забезпечує зручний доступ до необхідних бібліотек і інструментів. Крім того, середовище підтримує інтеграцію з хмарним сховищем, що спрощує збереження результатів і повторне використання моделей.

Для створення програмного інтерфейсу використано бібліотеку Streamlit. Вибір цього інструменту обумовлений його простотою у використанні та можливістю швидкого створення інтерактивних веб-застосунків без необхідності глибоких знань у веб-розробці. Це дозволило реалізувати інтерфейс, який демонструє роботу системи та забезпечує взаємодію з користувачем.

3.1 Опис набору даних

У якості вхідних даних для дослідження використано набір текстових резюме, призначений для задачі розпізнавання іменованих сутностей – Resume NER Training Dataset [26]. Датасет містить 5960 стандартизованих прикладів резюме, сформованих шляхом об'єднання кількох відкритих джерел, зокрема Kaggle ATS Dataset, HuggingFace NER Dataset, Resume

Corpus Dataset та Doccano Dataset. Таке поєднання дозволяє охопити різні формати представлення резюме та підвищити різноманітність даних.

Кожен документ у датасеті являє собою неструктурований текст резюме, що містить інформацію про кандидата, включаючи персональні дані, освіту, досвід роботи та професійні характеристики. Для кожного тексту задано розмітку сутностей, яка використовується для навчання моделей екстракції інформації.

У початковому варіанті датасет містить 14 типів сутностей, серед яких ключовими є SKILL, DESIGNATION, LOCATION, EXPERIENCE, PERSON, EDUCATION, EMAIL та COMPANY. Розподіл сутностей є нерівномірним: деякі категорії (наприклад, SKILL) представлені значно більшою кількістю прикладів, тоді як інші (зокрема COMPANY або CERTIFICATION) мають обмежену кількість входжень (рисунок 3.1). Це створює додаткові труднощі для навчання моделей та впливає на якість розпізнавання окремих типів сутностей.

Дані представлені у вигляді послідовностей токенів із відповідними мітками, що визначають належність кожного токена до певного типу сутності.

Така структура дозволяє застосовувати сучасні підходи до задачі NER, зокрема методи на основі нейронних мереж і трансформерних моделей.

Типи сутностей та їх кількість:

```
SKILL: 549465
OTHER: 10267
DESIGNATION: 4301
LOCATION: 4073
EXPERIENCE: 3544
PERSON: 3122
EDUCATION: 2124
EXPERTISE: 1045
EMAIL: 815
COMPANY: 218
COLLABORATION: 187
LANGUAGE: 159
ACTION: 133
CERTIFICATION: 122
```

Рисунок 3.1 – Розподіл сутностей

Приклад фрагмента розміченого тексту наведено на рисунку 3.2.

Кількість записів: 5960

Приклад одного запису:

Abhishek Jha Application Development Associate - Accenture Bengaluru, Karnataka - Email me on Indeed: [indeed.com/r/Abhishek-Jha/10e7a8cb732bc43a](https://www.indeed.com/r/Abhishek-Jha/10e7a8cb732bc43a) • To work for an organization which provides me the opportunity to improve my skills and knowledge for my individual and company's growth in best possible ways. Willing to relocate to: Bangalore, Karnat...

Анотації (перші 12):

1296,	1622,	"SKILL "
993,	1154,	"SKILL "
939,	957,	"PERSON"
883,	905,	"PERSON"
429,	433,	"EDUCATION"
771,	814,	"PERSON"
727,	769,	"DESIGNATION"
49,	58,	"COMPANY"
13,	46,	"DESIGNATION"
95,	145,	"EMAIL "
60,	69,	"LOCATION"
0,	12,	"PERSON"

Рисунок 3.2 – Приклад запису

3.2 Підготовка та попередня обробка даних

Після аналізу структури датасету було виконано етап підготовки даних, спрямований на приведення їх до вигляду, придатного для навчання моделей екстракції інформації. Основною метою даного етапу було зменшення шуму, усунення некоректних анотацій та виділення релевантних типів сутностей відповідно до задачі дослідження.

На першому етапі було обрано підмножину сутностей, що безпосередньо відповідають задачі екстракції інформації з резюме, а саме: PERSON, EMAIL, EDUCATION, EXPERIENCE, DESIGNATION та COMPANY. Інші типи сутностей, представлені у датасеті, не розглядалися в межах даного дослідження.

Додатково було виконано перевірку коректності індексів анотацій відносно тексту, а також відфільтровано записи, що не містили жодної

релевантної сутності. У результаті кількість записів було зменшено з 5960 до 988, що дозволило зосередитися на інформативних прикладах. Водночас кількість анотацій скоротилась з 579575 до 14124, що пояснюється значною часткою нерелевантних або надлишкових розміток у початковому наборі даних.

Розподіл сутностей після виконання первинної фільтрації наведено на рисунку 3.3.

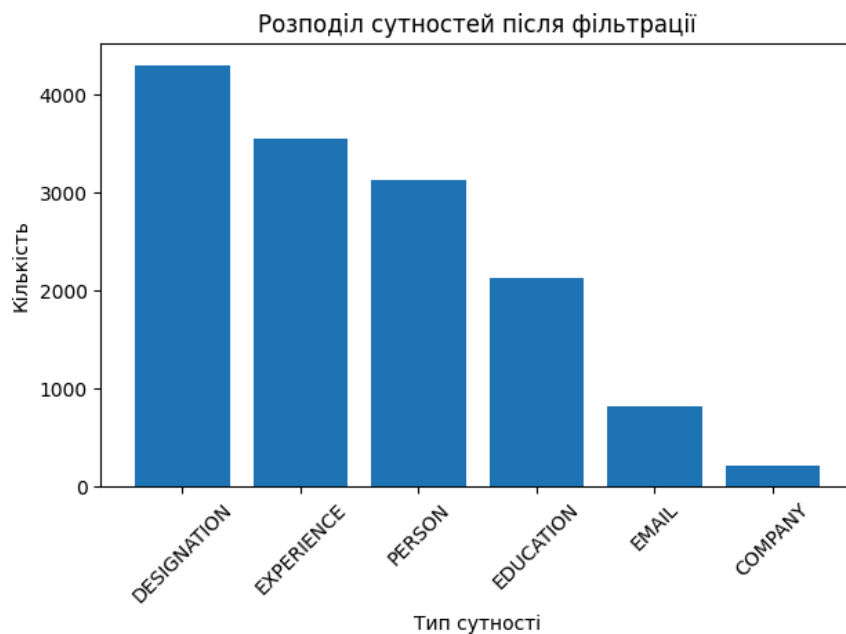


Рисунок 3.3 – Розподіл кількості прикладів для обраних типів сутностей після первинної фільтрації даних

На другому етапі було виконано додаткове очищення даних від шуму. До шумових анотацій віднесено некоректно визначені або частково перекривні фрагменти тексту, що не відповідають сутності або містять службову інформацію. У результаті було видалено 1638 анотацій, після чого загальна кількість валідних анотацій склала 12486.

Наступним кроком було перетворення даних у формат, придатний для навчання моделей NER. На цьому етапі виконувалася конвертація тексту та анотацій у токенизоване представлення з відповідними мітками. У процесі

конвертації було здійснено додаткову перевірку коректності анотацій, зокрема відповідності меж сутностей та відсутності їх перетинів.

У результаті з 14124 анотацій було отримано 14051 валідну сутність. Частина анотацій була відкинута через невідповідність меж (57 випадків) або перетини (16 випадків). Випадків, пов'язаних із некоректним кодуванням символів, виявлено не було.

Фінальний розподіл сутностей після всіх етапів обробки наведено на рисунку 3.4.

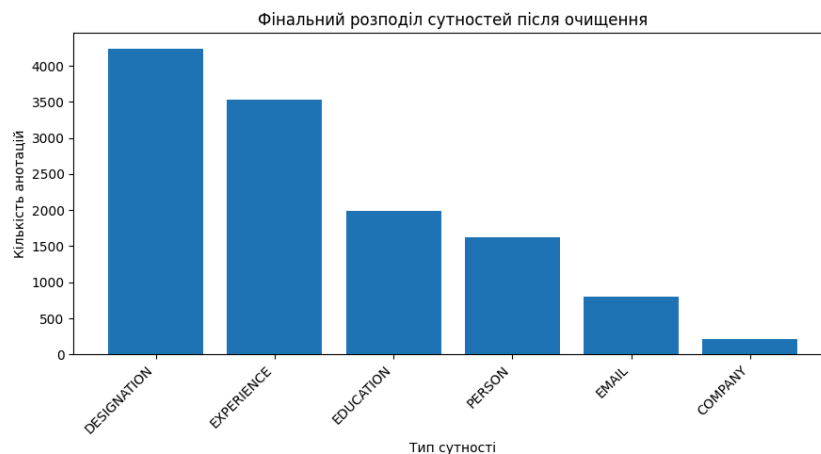


Рисунок 3.4 – Розподіл кількості прикладів для типів сутностей після завершення попередньої обробки даних

Таким чином, у результаті проведеної підготовки було сформовано очищений та узгоджений набір даних, що забезпечує коректність навчання моделей та дозволяє отримати більш достовірні результати під час подальшого порівняльного аналізу методів екстракції інформації.

Приклад підготовленого запису з анотаціями наведено на рисунку 3.5. Після завершення попередньої обробки дані було розподілено на навчальну, валідаційну та тестову вибірки у співвідношенні 80%, 10% та 10% відповідно. Розподіл виконано таким чином, щоб забезпечити коректну оцінку якості моделей та уникнути перенавчання. У результаті сформовано

навчальну вибірку обсягом 790 записів, валідаційну – 98 записів та тестову – 100 записів.

Кількість документів: 988

TEXT:

Abhishek Jha Application Development Associate - Accenture Bengaluru, Karnataka - Email me on Indeed: indeed.com/r/Abhis

ENTITIES:

Abhishek Jha -> PERSON

Application Development Associate -> DESIGNATION

Accenture -> COMPANY

Indeed: indeed.com/r/Abhishek-Jha/10e7a8cb732bc43a -> EMAIL

Рисунок 3.5 – Приклад підготовленого запису з анотаціями

На рисунку 3.6 наведено розподіл сутностей у навчальній вибірці

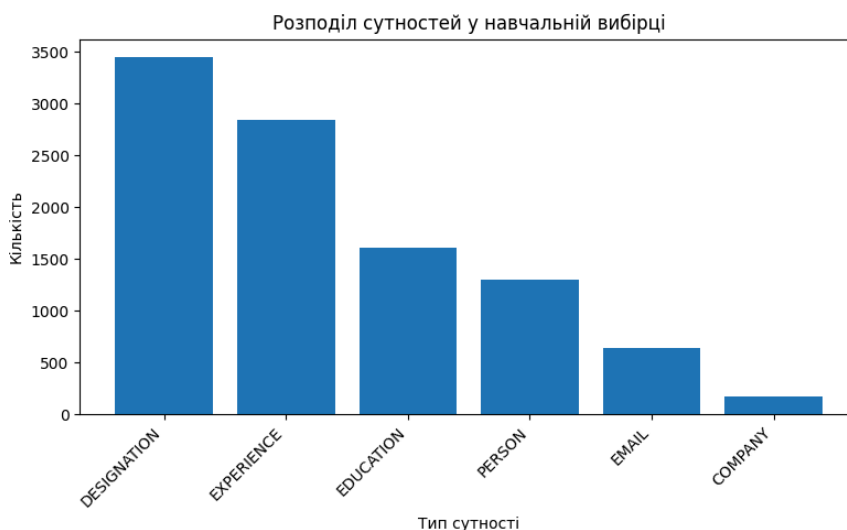


Рисунок 3.6 – Розподіл сутностей у навчальній вибірці

3.3 Реалізація rule-based підходу до екстракції інформації

Після підготовки набору даних першим етапом практичної реалізації було застосовано rule-based підхід. Його використання є доцільним для тих типів сутностей, які мають відносно сталу зовнішню форму та можуть бути описані за допомогою регулярних виразів. У межах роботи такий підхід

було застосовано не до всіх сутностей датасету, а лише до тих, для яких шаблонне виділення є технічно обґрунтованим: EMAIL, EXPERIENCE та EDUCATION.

Для сутності EMAIL було використано регулярний вираз, що визначає послідовність символів до знака @, доменну частину та домен верхнього рівня. Такий тип інформації має найбільш формалізовану структуру, тому його вилучення за допомогою шаблону є достатньо стабільним і не потребує контекстного аналізу речення.

Для сутності EXPERIENCE було задано шаблон, орієнтований на типові текстові форми подання досвіду роботи в резюме. Зокрема, враховувалися записи на кшталт кількості років досвіду, скорочені варіанти з yrs, значення з позначкою +, діапазони років, а також окремий випадок less than 1 year. На відміну від електронної пошти, така сутність є менш формалізованою, тому rule-based підхід у цьому випадку покриває лише найбільш типові варіанти запису.

Для сутності EDUCATION було сформовано перелік характерних ключових слів і скорочень, що можуть вказувати на освітній рівень або кваліфікацію: Bachelor, Master, PhD, Degree, Diploma тощо. Такий підхід дозволяє виділяти частину освітніх згадок, однак його якість залежить від повноти сформованого словника та не враховує складніші формулювання освітнього досвіду.

Реалізована функція `rule_based_extract_extended` приймає текст резюме та повертає словник із трьома ключами, що відповідають обраним типам сутностей. Для кожного типу сутності виконується пошук за відповідним регулярним виразом, а знайдені значення додатково приводяться до множини для усунення повторів. Таким чином, результатом роботи rule-based підходу є набір знайдених фрагментів тексту для кожної з трьох формалізованих категорій.

Оцінювання якості rule-based підходу виконувалося на тестовій вибірці. Для кожного документа перевірялася наявність істинної сутності

певного типу та наявність відповідного передбачення, отриманого за допомогою регулярних виразів. Тобто оцінювання здійснювалося на рівні факту виявлення сутності в документі, а не на рівні точного збігу меж текстового фрагмента. Для кожного з трьох типів сутностей було розраховано значення precision, recall та F1-score.

Такий спосіб оцінювання дозволив перевірити, наскільки добре шаблонний підхід визначає наявність відповідної інформації у резюме. Водночас він не відображає повною мірою якість встановлення точних меж сутностей, тому отримані результати доцільно розглядати як базовий орієнтир для подальшого порівняння з моделями NER, а не як повноцінну заміну контекстному розпізнаванню сутностей.

3.4 Реалізація моделі NER на основі spaCy

Оскільки rule-based підхід може бути застосований лише до частини формалізованих сутностей і не враховує контекст тексту, наступним етапом було реалізовано модель розпізнавання іменованих сутностей на основі бібліотеки spaCy. Використання даного підходу дозволяє враховувати контекст тексту та здійснювати виділення більш складних сутностей, які не можуть бути описані за допомогою шаблонів.

Для навчання моделі дані було представлено у внутрішньому форматі бібліотеки spaCy – DocBin, який забезпечує компактне зберігання текстів разом із анотаціями сутностей. Завантаження та збереження даних виконувалося з використанням відповідних функцій, що дозволяє працювати з датасетом як із набором об'єктів Doc.

У процесі підготовки даних було виявлено проблему некоректних меж сутностей, зокрема наявність пробілів або розділових знаків на початку чи в кінці анотацій. Такі випадки призводять до помилок типу `invalid whitespace entity spans` під час навчання моделі. Для усунення цієї проблеми було реалізовано окрему процедуру очищення, яка для кожної сутності

виконує коригування її меж шляхом послідовного видалення небажаних символів на початку та в кінці.

У випадках, коли після очищення довжина сутності ставала нульовою, така анотація видалялася. Додатково застосовувалася функція `filter_spans`, яка дозволяє усунути перекриття сутностей, залишаючи найбільш релевантні варіанти. У результаті виконання даного етапу було скориговано межі значної кількості сутностей, що свідчить про наявність неточностей у вихідній розмітці.

У результаті обробки було отримано такі зміни:

- для навчальної вибірки (790 документів) скориговано межі 210 сутностей та видалено 1 некоректну анотацію;
- для валідаційної вибірки (98 документів) скориговано межі 23 сутностей;
- для тестової вибірки (100 документів) скориговано межі 30 сутностей.

Це підтверджує, що основна проблема полягала саме у неточності меж, а не у повній некоректності анотацій.

Після очищення даних було виконано перевірку їх коректності за допомогою вбудованого інструменту `spacy debug data`, який дозволяє виявити потенційні проблеми у розмітці перед початком навчання. Застосування даного інструменту забезпечує виявлення неконсистентних анотацій, помилок у межах сутностей та інших недоліків датасету, що дозволяє уникнути збоїв під час тренування моделі та підвищити стабільність процесу навчання.

Навчання моделі виконувалося з використанням стандартного пайплайну `spaCy`, що включає компоненти `tok2vec` для формування векторних представлень тексту та `ner` для розпізнавання сутностей. У процесі навчання модель багаторазово проходила по навчальній вибірці з поступовим оновленням параметрів, а якість її роботи контролювалася на валідаційній вибірці.

Додатково було виконано аналіз характеристик сутностей у підготовленому датасеті. Для кожної сутності визначалися її довжина у символах та кількість токенів, що дозволяє оцінити особливості їх представлення у тексті. Отримані результати наведено на рисунку 3.7.

label	char_len								token_len							
	count	mean	std	min	25%	50%	75%	max	count	mean	std	min	25%	50%	75%	max
COMPANY	174.0	12.431034	7.394996	3.0	7.0	9.0	15.0	34.0	174.0	1.821839	1.116034	1.0	1.0	1.0	2.0	6.0
DESIGNATION	3451.0	19.418429	9.418757	1.0	14.0	18.0	24.0	182.0	3451.0	2.621849	1.497777	1.0	2.0	2.0	3.0	29.0
EDUCATION	1606.0	27.098381	17.684047	2.0	13.0	27.0	38.0	101.0	1606.0	4.118929	2.434729	1.0	2.0	4.0	6.0	15.0
EMAIL	643.0	28.916019	12.504047	10.0	19.0	24.0	42.0	64.0	643.0	1.259720	0.692221	1.0	1.0	1.0	1.0	7.0
EXPERIENCE	2841.0	15.443858	8.224690	1.0	9.0	14.0	22.0	100.0	2841.0	3.350581	1.674248	1.0	3.0	3.0	4.0	38.0
PERSON	1303.0	16.432080	10.050095	2.0	11.0	14.0	19.0	148.0	1303.0	2.590944	1.488557	1.0	2.0	2.0	3.0	20.0

Рисунок 3.7 – Статистичні характеристики сутностей у підготовленому датасеті

Аналіз показує, що сутності мають різну довжину як у символах, так і у токенах, що створює додаткову складність для задачі розпізнавання. Зокрема, наявність як дуже коротких, так і довгих сутностей ускладнює процес узагальнення моделі, оскільки такі випадки по-різному впливають на якість передбачення меж сутностей. Отримані статистичні характеристики враховуються під час подальшого аналізу ефективності застосованих методів екстракції інформації.

3.5 Реалізація трансформерних моделей для задачі NER

Наступним етапом дослідження стало застосування трансформерних моделей для задачі розпізнавання іменованих сутностей. На відміну від підходу на основі spaCy, використання трансформерів дозволяє більш ефективно враховувати контекст у межах усього речення, що є критично важливим для обробки неструктурованих текстів резюме.

Початково для навчання трансформерних моделей було використано підготовлені на попередньому етапі дані у форматі spaCy (DocBin). Обсяг вибірок становив 790 документів для навчання, 98 – для валідації та 100 – для тестування.

Перед подальшою обробкою було виконано додаткове очищення даних з метою усунення перетинів між вибірками. Для цього для кожного документа обчислювався хеш нормалізованого тексту, що дозволило виявити дублікати між навчальною, валідаційною та тестовою вибірками. У результаті з валідаційної та тестової вибірок було видалено документи, що повторювалися у навчальній вибірці, що дозволило уникнути витоку інформації між наборами даних та забезпечити коректність подальшого оцінювання моделей.

У результаті кількість записів у валідаційній вибірці зменшилась з 98 до 77, а у тестовій – зі 100 до 75, тоді як навчальна вибірка залишилась без змін (790 записів).

На наступному етапі дані було перетворено у формат, придатний для навчання трансформерних моделей. Для цього кожен документ було представлено у вигляді послідовності токенів та відповідних міток у форматі BIO (Begin-Inside-Outside), де кожному токenu відповідає певна категорія сутності або відсутність сутності.

Формування BIO-міток здійснювалося на основі початкових анотацій: перший токен сутності позначався як B-<label>, наступні – як I-<label>, тоді як усі інші токени отримували мітку O. Такий підхід є стандартним для задачі розпізнавання іменованих сутностей і дозволяє моделі враховувати межі сутностей у тексті.

Приклад такого перетворення наведено на рисунку 3.8.

Після формування BIO-міток було створено відображення між текстовими мітками та числовими ідентифікаторами, що необхідно для навчання моделі. На основі отриманих даних було сформовано датасет у

форматі бібліотеки HuggingFace Transformers, що містить окремі підвибірки для навчання, валідації та тестування.

Приклад токенів:

```
['Navas', 'Koya', '\n', 'Test', 'Engineer', '\n\n', 'Mangalore', ',', 'Karnataka',
```

Приклад міток:

```
['B-PERSON', 'I-PERSON', 'O', 'B-DESIGNATION', 'I-DESIGNATION', 'O', 'O', 'O', 'O',
```

Рисунок 3.8 – Перетворення у формат BIO

Наступним етапом було виконано токенизацію тексту з використанням токенизатора моделі DistilBERT (distilbert-base-uncased). Оскільки трансформерні моделі можуть розбивати слова на підслова, виникає необхідність у коректному вирівнюванні початкових BIO-міток із новою токенизованою послідовністю. Для цього для кожного токена визначався відповідний індекс слова, після чого мітки переносилися з урахуванням розбиття на підслова. У випадку, якщо один токен відповідає частині слова, використовувалося перетворення мітки з B- у I-, що забезпечує коректне представлення сутності. Приклад результату токенизації та вирівнювання міток наведено на рисунку 3.9.

Після підготовки даних було ініціалізовано модель класифікації токенів на основі архітектури DistilBERT (distilbert-base-uncased). Модель налаштовувалася з урахуванням кількості класів, що відповідають сформованому набору BIO-міток, а також відображень між текстовими мітками та їх числовими ідентифікаторами.

Навчання моделі здійснювалося з використанням бібліотеки HuggingFace Transformers. Було визначено основні параметри тренування, зокрема швидкість навчання, розмір батчу, кількість епох та коефіцієнт регуляризації. Додатково застосовувався механізм ранньої зупинки (early stopping), що дозволяє припинити навчання у випадку відсутності

покращення якості на валідаційній вибірці та запобігти перенавчанню моделі.

TOKEN	LABEL
[CLS]	IGN
na	B-PERSON
##vas	I-PERSON
ko	I-PERSON
##ya	I-PERSON
test	B-DESIGNATION
engineer	I-DESIGNATION
manga	0
##lore	0
,	0
karnataka	0
system	B-DESIGNATION
engineer	I-DESIGNATION
info	0
##sy	0
##s	0
-	0
august	B-EXPERIENCE
2014	I-EXPERIENCE
to	I-EXPERIENCE
present	I-EXPERIENCE
.	0

Рисунок 3.9 – Приклад токенизації тексту та вирівнювання BIO-міток

Оцінювання якості моделі здійснювалося з використанням метрик precision, recall, F1-score та accuracy на основі порівняння передбачених та істинних BIO-міток із застосуванням бібліотеки seqeval.

З метою порівняння ефективності трансформерних підходів додатково було реалізовано модель на основі архітектури BERT (bert-base-cased). На відміну від DistilBERT, дана модель є повною версією BERT та зберігає регістр символів, що потенційно дозволяє точніше обробляти власні назви, аббревіатури та інші чутливі до регістру фрагменти тексту.

Процес підготовки даних для даної моделі є аналогічним і включає токенизацію тексту, вирівнювання BIO-міток та формування датасету у відповідному форматі, тому додаткові етапи обробки не виконувалися. Відмінність полягає у використанні іншого токенизатора та відповідної попередньо навченої моделі.

Навчання моделі BERT також здійснювалося з використанням бібліотеки HuggingFace Transformers із подібними параметрами тренування, однак із меншою кількістю епох, що зумовлено більшою складністю моделі та вищими обчислювальними витратами.

Таким чином, у межах даного етапу було реалізовано дві трансформерні моделі для задачі розпізнавання іменованих сутностей, що дозволяє надалі виконати їх порівняльний аналіз та оцінити доцільність використання кожної з них у задачі екстракції інформації з текстових резюме.

3.6 Формування гібридного підходу до екстракції інформації

Після реалізації окремих підходів до екстракції інформації було сформовано гібридний метод, який поєднує rule-based підхід та трансформерну модель на основі DistilBERT. Основна ідея полягає у використанні різних методів для різних типів сутностей залежно від їх структури та способу представлення у тексті.

Зокрема, для формалізованих сутностей, таких як EMAIL, EXPERIENCE та EDUCATION, використовується rule-based підхід, заснований на регулярних виразах. Такі сутності мають відносно стабільний формат запису, що дозволяє ефективно виділяти їх без застосування складних моделей.

Для сутностей, що залежать від контексту, а саме PERSON, DESIGNATION та COMPANY, застосовується трансформерна модель DistilBERT, попередньо навчена на підготовленому датасеті. Використання моделі дозволяє враховувати семантичні зв'язки між словами та коректно визначати межі сутностей у складних текстових фрагментах.

У процесі реалізації гібридного підходу було використано збережену модель DistilBERT, отриману на етапі навчання. Для обробки вхідного тексту застосовується відповідний токенизатор, після чого виконується

передбачення міток для кожного токена. На основі отриманих ВІО-міток здійснюється відновлення текстових фрагментів сутностей із використанням позицій токенів у вихідному тексті.

Оскільки модель виконує передбачення для всіх типів сутностей, на етапі постобробки здійснюється фільтрація результатів: залишаються лише ті категорії, що використовуються у гібридному підході. Формування сутностей виконується шляхом об'єднання послідовних токенів із мітками типу В- та І-, що відповідають одній і тій самій категорії.

Підсумковий результат екстракції формується шляхом об'єднання результатів обох підходів. Для кожного тексту окремо обчислюються сутності, отримані за допомогою регулярних виразів, та сутності, передбачені трансформерною моделлю. Далі вони інтегруються в єдину структуру, що містить усі визначені типи сутностей.

Такий підхід дозволяє поєднати переваги обох методів: високу точність для формалізованих сутностей та здатність до контекстного узагальнення для складніших категорій. Це забезпечує більш повне та узгоджене вилучення інформації з текстових резюме порівняно з використанням кожного з підходів окремо.

РЕЗУЛЬТАТИ ТА ПОРІВНЯЛЬНИЙ АНАЛІЗ МЕТОДІВ ЕКСТРАКЦІЇ ІНФОРМАЦІЇ

4.1 Методика проведення експерименту

Для оцінювання ефективності методів екстракції інформації з текстових резюме було проведено серію експериментів із використанням підготовленого набору даних. У межах дослідження розглядалися такі підходи: rule-based метод, модель на основі spaCy, трансформерні моделі DistilBERT та BERT-base-cased, а також гібридний підхід.

Оцінювання виконувалося на тестовій вибірці, яка не використовувалася під час навчання моделей. Для моделей машинного навчання додатково застосовувалася валідаційна вибірка, що використовувалася для контролю процесу навчання та вибору найкращих параметрів.

Як основні метрики якості використовувалися precision, recall та F1-score. Для трансформерних моделей додатково враховувалася метрика assuarcy, що обчислювалася на рівні токенів. Обчислення метрик здійснювалося з використанням бібліотеки seqeval, яка є стандартним інструментом для оцінювання задач розпізнавання іменованих сутностей.

Оцінювання виконувалося як у цілому по всіх сутностях, так і окремо для кожного типу (PERSON, EMAIL, EDUCATION, EXPERIENCE, DESIGNATION, COMPANY). Такий підхід дозволяє детальніше проаналізувати сильні та слабкі сторони кожного методу.

Для забезпечення коректності порівняння всі методи оцінювалися на однаковому тестовому наборі даних. Отримані результати представлено у вигляді таблиць та графіків у наступних підрозділах.

Основною метрикою для вибору найкращої моделі виступав показник F1-score, що враховувався під час навчання та застосовувався як критерій для збереження найкращих ваг моделі.

4.2 Результати застосування методів екстракції інформації

Першим було оцінено rule-based підхід, який застосовувався для екстракції формалізованих сутностей (EMAIL, EXPERIENCE, EDUCATION). Результати оцінювання наведено на рисунку 4.1.

Entity	Precision	Recall	F1-score
EMAIL	1.000000	0.475410	0.644444
EXPERIENCE	0.803922	0.506173	0.621212
EDUCATION	0.987179	0.802083	0.885057

Рисунок 4.1 – Результати роботи rule-based підходу для окремих типів сутностей

Як видно з наведених результатів, для сутності EMAIL досягнуто максимальної точності (Precision = 1.0), що пояснюється чіткою формалізацією даного типу даних та можливістю його коректного опису за допомогою регулярних виразів. Для сутностей EDUCATION та EXPERIENCE також отримано високі значення precision, однак спостерігається зниження показника recall, що свідчить про пропуск частини релевантних випадків. Це зумовлено більш варіативними способами представлення таких сутностей у тексті, які не повністю покриваються заданими шаблонами.

Узагальнені значення метрик становлять: precision – 0.930, recall – 0.595 та F1-score – 0.717, що дозволяє розглядати rule-based підхід як базовий орієнтир для подальшого порівняння з більш складними моделями.

Результати навчання моделі на основі spaCy наведено на рисунку 4.2, де відображено динаміку зміни значень функції втрат та основних метрик якості в процесі тренування.

E	#	LOSS TOK2VEC	LOSS NER	ENTS_F	ENTS_P	ENTS_R	SCORE
0	0	0.00	316.50	0.00	0.00	0.00	0.00
0	200	2013.24	10653.27	3.02	14.50	1.64	0.03
0	400	1574.53	3204.02	41.74	57.39	31.18	0.41
0	600	2118.62	3033.82	56.74	51.98	57.83	0.56
1	800	552.42	2880.10	51.26	75.56	36.95	0.52
1	1000	308.64	2339.94	63.23	59.21	62.93	0.63
1	1200	317.59	2389.99	63.19	69.49	54.47	0.63
1	1400	463.80	2499.31	67.85	73.02	59.46	0.68
2	1600	332.06	2227.10	68.13	64.28	67.26	0.68
2	1800	391.16	2050.86	67.19	66.84	63.02	0.67
2	2000	532.01	2011.98	64.74	74.07	54.17	0.64
2	2200	434.34	2203.22	66.93	70.42	59.75	0.67
3	2400	454.71	1885.49	67.00	65.09	64.28	0.67
3	2600	455.33	1789.34	65.69	68.76	58.89	0.66
3	2800	482.24	1756.45	63.92	66.33	57.74	0.64
3	3000	441.87	1938.67	67.61	66.18	64.37	0.68
4	3200	433.73	1718.50	69.86	62.48	73.12	0.70
4	3400	525.80	1513.02	66.67	61.38	67.55	0.67
4	3600	747.71	1864.36	70.86	72.79	64.57	0.71
4	3800	552.19	1707.91	67.59	79.18	55.62	0.68
5	4000	530.56	1684.15	68.25	73.41	59.85	0.68
5	4200	657.53	1445.74	67.37	75.99	56.96	0.68
5	4400	617.30	1632.72	67.94	60.54	71.40	0.68
5	4600	673.79	1630.43	68.32	62.00	70.34	0.68
6	4800	613.67	1491.44	67.16	66.32	63.41	0.67
6	5000	733.74	1299.10	69.18	67.17	66.39	0.69
6	5200	802.96	1605.99	67.49	65.86	64.47	0.68

Рисунок 4.2 – Динаміка зміни метрик якості моделі spaCy під час навчання

У процесі навчання спостерігається поступове зменшення значень функції втрат, що свідчить про ефективне налаштування параметрів моделі. Водночас метрики якості (precision, recall та F1-score) змінюються нерівномірно, що є характерним для задач розпізнавання іменованих сутностей у текстах із неоднорідною структурою.

Коливання значень метрик пояснюється складністю вхідних даних та нерівномірним розподілом сутностей у датасеті. Частина сутностей представлена значно більшою кількістю прикладів, що впливає на процес навчання та здатність моделі узагальнювати інформацію для менш представлених категорій.

Найкраще значення F1-score, досягнуте під час навчання, становило 0.7086, при цьому відповідні значення precision та recall склали 0.7279 та 0.6457 відповідно. Подальші ітерації не призвели до суттєвого

покращення результатів, що свідчить про стабілізацію процесу навчання та досягнення моделлю певного рівня узагальнення.

Результати навчання трансформерної моделі DistilBERT наведено на рисунках 4.3–4.4.

Epoch	Training Loss	Validation Loss	Precision	Recall	F1-score	Accuracy
1	0.254346	0.166012	0.410989	0.403268	0.421726	0.960143
2	0.124322	0.118996	0.487666	0.676067	0.587893	0.970520
3	0.092026	0.100133	0.572830	0.701483	0.653983	0.975981
4	0.092598	0.095845	0.619841	0.721815	0.691460	0.977278
5	0.072876	0.093982	0.626329	0.743842	0.705097	0.976713
6	0.056516	0.098027	0.647906	0.762481	0.726316	0.978199
7	0.053695	0.098730	0.633740	0.801452	0.734076	0.977362
8	0.047310	0.102828	0.606942	0.794675	0.713898	0.975646
9	0.042414	0.104139	0.650253	0.791286	0.740248	0.978011
10	0.042520	0.105517	0.642729	0.769259	0.726148	0.977718
11	0.040290	0.104634	0.669631	0.754009	0.735273	0.978429

Рисунок 4.3 – Результати навчання моделі DistilBERT за епохами

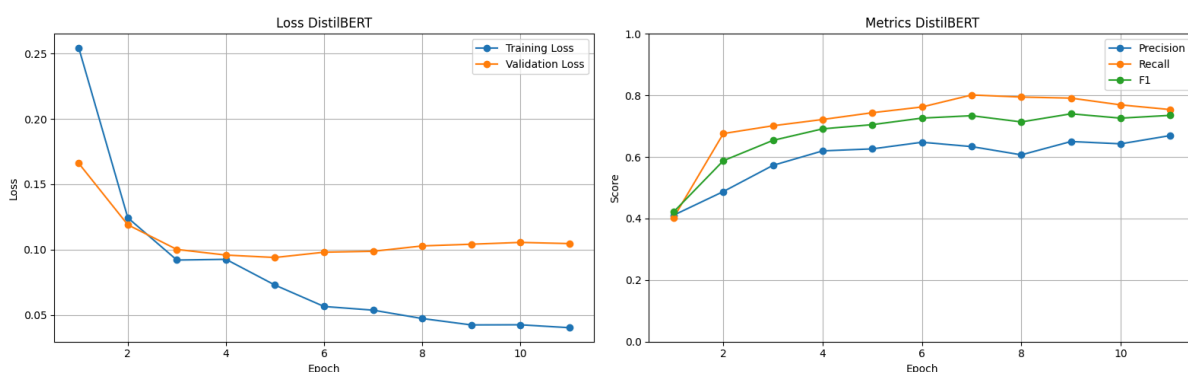


Рисунок 4.4 – Динаміка функції втрат та метрик якості моделі DistilBERT

З рисунка 4.3 видно, що на початкових епохах спостерігається суттєве покращення якості моделі: значення F1-score зростає з 0.42 на першій епосі

до понад 0.70 вже на 5–6 епохах. Подальше навчання не призводить до значного підвищення якості, а значення метрик коливаються в межах досягнутого рівня.

Аналіз рисунка 4.4 показує, що значення функції втрат на тренувальній вибірці стабільно зменшується протягом усього процесу навчання. Водночас валідаційна функція втрат після початкового зниження починає поступово зростати, що свідчить про появу ефекту перенавчання на пізніших етапах.

Метрики якості (precision, recall та F1-score) демонструють швидке зростання на перших епохах та подальшу стабілізацію. Найкраще значення F1-score становить 0.7402, що досягається на 9 епосі навчання. Подальші ітерації не забезпечують суттєвого покращення, що підтверджує доцільність використання механізму ранньої зупинки.

Отримані результати свідчать про ефективність використання трансформерної моделі для задачі екстракції інформації, а також про досягнення моделлю стабільного рівня узагальнення.

Результати навчання трансформерної моделі BERT-base-cased наведено на рисунках 4.5–4.6.

Epoch	Training Loss	Validation Loss	Precision	Recall	F1-score	Accuracy
1	0.161231	0.140642	0.390097	0.513962	0.453677	0.970146
2	0.120467	0.115495	0.472203	0.712742	0.581207	0.974248
3	0.094332	0.101751	0.528097	0.696625	0.614492	0.977294
4	0.094616	0.101949	0.556487	0.709160	0.637815	0.977744
5	0.078229	0.105838	0.504299	0.753930	0.618327	0.976256
6	0.061291	0.098415	0.578689	0.748558	0.667639	0.979111
7	0.060772	0.100731	0.567377	0.757512	0.663642	0.979180
8	0.056543	0.102632	0.554729	0.753930	0.653815	0.979007

Рисунок 4.5 – Результати навчання моделі BERT-base-cased за епохами

Як видно з рисунка 4.5, на початкових епохах спостерігається зростання значень метрик якості, зокрема F1-score підвищується з 0.45 до приблизно 0.66–0.67. Подальше навчання не забезпечує суттєвого покращення результатів, а значення метрик коливаються в межах досягнутого рівня.

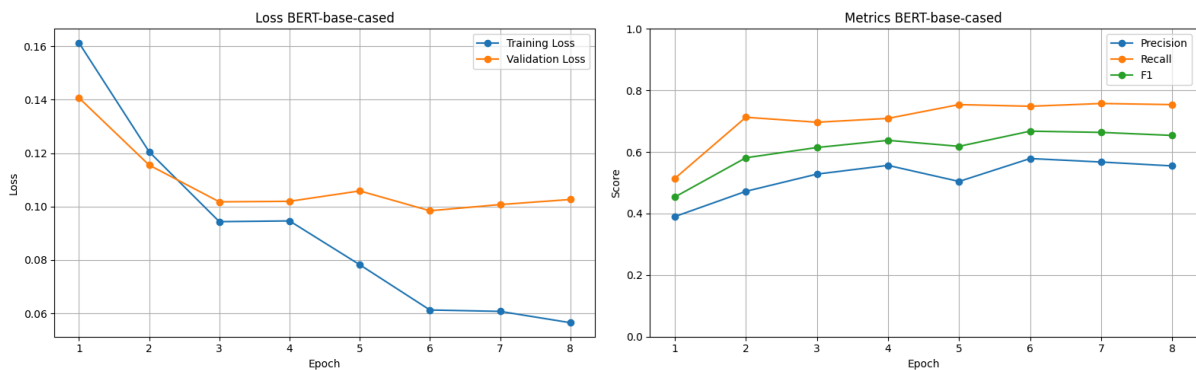


Рисунок 4.6 – Динаміка функції втрат та метрик якості моделі BERT-base-cased

Аналіз рисунка 4.6 показує, що значення функції втрат на тренувальній вибірці поступово зменшується, тоді як валідаційна втрата після початкового зниження демонструє незначні коливання. Це вказує на стабілізацію процесу навчання з можливими ознаками перенавчання на пізніших етапах.

Метрики якості (precision, recall та F1-score) демонструють подібну до DistilBERT динаміку: швидке зростання на перших епохах та подальшу стабілізацію. Максимальне значення F1-score становить близько 0.66–0.67, що є нижчим порівняно з моделлю DistilBERT.

Отримані результати свідчать про досягнення моделлю стабільного рівня якості, однак без подальшого суттєвого покращення на пізніших етапах навчання.

Результати роботи гібридного підходу наведено на рисунку 4.7.

Entity	Precision	Recall	F1-score
EMAIL	1.000000	0.594262	0.745501
EXPERIENCE	0.844118	0.582099	0.689040
EDUCATION	0.987179	0.802083	0.885057
PERSON	0.945652	0.956044	0.950820
DESIGNATION	0.979167	0.989474	0.984294
COMPANY	0.660000	0.626086	0.642596

Рисунок 4.7 – Результати гібридного підходу для окремих типів сутностей

Як видно з отриманих результатів, гібридний підхід демонструє високі значення метрик якості для більшості типів сутностей. Зокрема, для сутностей PERSON та DESIGNATION досягнуто найвищих значень F1-score – 0.9508 та 0.9843 відповідно, що свідчить про ефективність використання трансформерної моделі для контекстно залежних категорій.

Для формалізованих сутностей (EMAIL, EDUCATION, EXPERIENCE) також спостерігаються стабільно високі значення precision, що зумовлено використанням rule-based підходу. Водночас показник recall для цих категорій є вищим порівняно з окремим застосуванням шаблонного методу, що свідчить про більш повне виявлення сутностей.

Для сутності COMPANY значення метрик є нижчими порівняно з іншими категоріями, що може бути пов'язано з варіативністю способів представлення назв компаній у тексті та складністю їх однозначного визначення.

Отримані результати свідчать про ефективність поєднання різних підходів у межах єдиної системи екстракції, що дозволяє досягти високої якості розпізнавання як формалізованих, так і контекстно залежних сутностей. Це пояснюється тим, що rule-based компонент забезпечує точне визначення сутностей із чіткою структурою, для яких можливе використання шаблонів, тоді як трансформерна модель компенсує його

обмеження за рахунок здатності враховувати контекст тексту та обробляти варіативні формулювання.

4.3 Порівняльний аналіз методів екстракції

Узагальнені результати оцінювання методів екстракції інформації наведено на рисунку 4.8.

Method	Precision	Recall	F1-score
Rule-based	0.930367	0.594555	0.716905
spaCy NER	0.727900	0.645700	0.708600
DistilBERT	0.650253	0.791286	0.740248
BERT-base-cased	0.578689	0.748558	0.667639
Hybrid	0.902686	0.758341	0.816218

Рисунок 4.8 – Порівняння якості методів екстракції інформації

Аналіз результатів показує, що різні підходи демонструють принципово відмінну поведінку залежно від типу сутностей та способу їх представлення у тексті.

Rule-based підхід забезпечує найвище значення precision (0.930), що пояснюється жорстко заданими шаблонами, які майже не допускають хибнопозитивних спрацювань. Водночас низьке значення recall (0.595) є прямим наслідком обмеженості цих шаблонів: будь-які нетипові або варіативні формулювання сутностей не розпізнаються. Таким чином, даний підхід є ефективним лише для строго формалізованих категорій і не здатний узагальнювати нові варіанти запису.

Модель на основі spaCy демонструє більш збалансовані значення метрик (F1-score = 0.709), що свідчить про здатність враховувати контекст. Проте її ефективність обмежується менш потужним механізмом

представлення тексту, що ускладнює розпізнавання складних або довгих залежностей у резюме. У результаті модель частково компенсує недоліки rule-based підходу, але не забезпечує достатнього рівня узагальнення.

Трансформерні моделі демонструють іншу поведінку: зниження precision супроводжується зростанням recall. Для DistilBERT це проявляється у досягненні найвищого значення recall (0.791) серед усіх окремих моделей, що свідчить про здатність моделі виявляти значно більшу кількість сутностей, включаючи складні та неявні випадки. Це пояснюється використанням контекстного представлення тексту, яке дозволяє враховувати залежності між словами в межах усього документа.

Водночас порівняння DistilBERT та BERT-base-cased показує, що більша складність моделі не гарантує кращих результатів. Незважаючи на повну архітектуру BERT, її F1-score (0.668) є нижчим, ніж у DistilBERT (0.740). Це може бути пов'язано з обмеженим обсягом навчальних даних, за якого більш складна модель гірше узагальнює та більш схильна до перенавчання.

Гібридний підхід демонструє найкращі результати (F1-score = 0.816), що пояснюється поєднанням сильних сторін різних методів. Використання rule-based підходу дозволяє зберегти високу точність для формалізованих сутностей, тоді як трансформерна модель компенсує його основний недолік – низьку повноту – за рахунок контекстного аналізу. У результаті досягається одночасне підвищення як precision, так і recall, що безпосередньо впливає на зростання F1-score.

Окремо варто відзначити вплив розподілу даних на якість моделей. Для сутності COMPANY, яка представлена лише 217 прикладами, спостерігається зниження якості розпізнавання. У той же час сутності з більшою кількістю прикладів (DESIGNATION, EXPERIENCE) демонструють стабільно кращі результати. Це підтверджує, що дисбаланс даних є одним із ключових факторів, що обмежують ефективність моделей у задачі NER.

Таким чином, проведений аналіз показує, що жоден із окремих підходів не забезпечує оптимального результату для всіх типів сутностей. Найкращі показники досягаються при комбінуванні методів, що підтверджує доцільність використання гібридного підходу для задачі екстракції інформації з текстових резюме.

4.4 Аналіз помилок методів екстракції інформації

Незважаючи на отримані результати, усі розглянуті підходи демонструють певні обмеження, що проявляються у вигляді помилок під час екстракції сутностей.

Одним із характерних типів помилок є неповне виділення сутностей. Це спостерігається переважно для багатослівних назв посад або організацій, де модель визначає правильний тип сутності, але не включає всі відповідні токени. Така поведінка пов'язана зі складністю визначення точних меж сутностей у тексті.

Іншим типом помилок є хибне віднесення токенів до сутностей. Зокрема, окремі слова можуть помилково класифікуватися як PERSON або DESIGNATION у випадках, коли контекст є недостатньо однозначним. Це характерно для трансформерних моделей, які активно використовують контекст і можуть узагальнювати на схожі мовні конструкції.

Також спостерігаються пропуски сутностей, особливо для категорій із малою кількістю прикладів у датасеті. Найбільш показовим є випадок сутності COMPANY, для якої наявно лише 217 прикладів, що суттєво обмежує здатність моделей до її коректного розпізнавання.

Для rule-based підходу помилки пов'язані переважно з обмеженістю шаблонів: нетипові або варіативні формулювання не відповідають заданим регулярним виразам, що призводить до пропуску сутностей. У свою чергу, трансформерні моделі демонструють кращу здатність до узагальнення, однак залишаються чутливими до неоднозначного контексту.

Гібридний підхід дозволяє частково зменшити кількість помилок за рахунок поєднання різних методів, проте не усуває повністю проблеми, пов'язані з контекстною неоднозначністю та дисбалансом даних.

Таким чином, основними причинами помилок є складність природної мови, варіативність представлення сутностей та нерівномірний розподіл даних у датасеті.

4.5 Реалізація програмного інтерфейсу системи екстракції інформації

З метою демонстрації практичного застосування розробленого підходу було реалізовано програмний інтерфейс на основі бібліотеки Streamlit. Розроблена система забезпечує взаємодію користувача з моделлю та дозволяє виконувати екстракцію структурованої інформації з текстових резюме у зручному режимі.

Інтерфейс передбачає можливість введення тексту резюме вручну або завантаження файлу у форматах TXT, PDF або DOCX. У випадку завантаження файлу виконується попереднє вилучення текстового вмісту, після чого отримані дані передаються на обробку. Таким чином, система підтримує різні сценарії використання та не обмежується лише текстовим введенням.

Обробка вхідних даних здійснюється із застосуванням гібридного підходу, що поєднує rule-based метод та трансформерну модель DistilBERT. Зокрема, для формалізованих сутностей (EMAIL, EDUCATION, EXPERIENCE) використовуються регулярні вирази, що забезпечують високу точність їх визначення.

Для контекстно залежних категорій (PERSON, DESIGNATION, COMPANY) застосовується навчена модель NER, яка дозволяє враховувати контекст тексту та коректно ідентифікувати сутності різної структури. Результати обох підходів об'єднуються у єдине структуроване представлення.

Приклад інтерфейсу системи та результатів екстракції наведено на рисунку 4.9.

Resume Information Extraction System

Hybrid approach: rule-based extraction + DistilBERT NER

Upload resume file

[Upload](#) 200MB per file • TXT, PDF, WORD

Or paste resume text

Software Engineer

Work Experience:

Accenture Bengaluru, Karnataka (June 2019 – Present)

- Developed backend services using Python and Django

- Worked with REST APIs and microservices

[Extract information](#)

Extracted structured information

PERSON

• John Smith

EMAIL

• john.smith.dev@gmail.com

DESIGNATION

• Software Engineer

EDUCATION

• Bachelor

• Master

EXPERIENCE

• 7 years

COMPANY

• Accenture

Highlighted resume text

John Smith (PERSON)

Email: john.smith.dev@gmail.com (EMAIL)

Software Engineer (DESIGNATION)

Рисунок 4.9 – Інтерфейс системи екстракції інформації та приклад результату

Результати роботи системи відображаються у декількох формах. По-перше, формується структурований список знайдених сутностей, згрупованих за типами (PERSON, EMAIL, EDUCATION, EXPERIENCE, DESIGNATION, COMPANY). Такий спосіб подання дозволяє швидко отримати узагальнену інформацію з резюме та оцінити результати екстракції.

По-друге, реалізовано візуалізацію результатів у вигляді підсвічування сутностей безпосередньо у тексті. Для цього використовуються позиції знайдених фрагментів (span), що дозволяє точно

відобразити межі сутностей та уникнути накладання виділених областей. Такий підхід забезпечує узгодженість між текстовим представленням і результатами екстракції та дозволяє візуально перевірити коректність роботи системи.

Додатково результати екстракції представлені у форматі JSON, що забезпечує можливість їх подальшого використання в інформаційних системах або інтеграції з іншими програмними сервісами. Такий формат є зручним для автоматизованої обробки, оскільки дозволяє чітко структурувати дані та передавати їх між різними компонентами системи.

Приклад сформованого JSON-виходу наведено на рисунку 4.10.



Рисунок 4.10 – Приклад структурованого JSON-виходу системи

Таким чином, реалізований програмний інтерфейс демонструє повний цикл обробки даних – від введення тексту резюме до отримання структурованої інформації. Використання гібридного підходу у поєднанні з візуалізацією результатів підтверджує практичну придатність розробленого рішення для задачі автоматизованої екстракції інформації.

ВИСНОВКИ

У роботі проведено дослідження та порівняння методів екстракції даних з неструктурованих текстових документів на основі NLP-моделей. Як приклад неструктурованих документів було використано текстові резюме, що містять різні типи сутностей та характеризуються варіативною структурою подання інформації.

У теоретичній частині проаналізовано існуючі підходи до екстракції інформації, зокрема rule-based методи, класичні моделі розпізнавання іменованих сутностей та трансформерні архітектури. Визначено їх основні переваги та обмеження у задачі обробки неструктурованих текстів.

У практичній частині виконано підготовку набору даних, що включала очищення текстів, перевірку коректності анотацій, усунення помилок розмітки та перетворення даних у формат, придатний для навчання моделей. Проведений аналіз показав нерівномірний розподіл сутностей, що впливає на якість їх розпізнавання.

У межах експериментального дослідження реалізовано та оцінено декілька підходів до екстракції інформації: rule-based метод, модель на основі spaCy, трансформерні моделі DistilBERT та BERT-base-cased, а також гібридний підхід. Оцінювання виконувалося на тестовій вибірці з використанням метрик Precision, Recall та F1-score.

Отримані результати показали, що rule-based підхід забезпечує високу точність (Precision = 0.93), однак має низьку повноту (Recall = 0.59) через обмеженість шаблонів. Модель spaCy демонструє більш збалансовані результати (F1-score $\approx$ 0.71), проте поступається трансформерним моделям за здатністю враховувати контекст. Серед трансформерних моделей кращі результати отримано для DistilBERT (F1-score $\approx$ 0.74), тоді як модель BERT-base-cased показала нижчу ефективність (F1-score $\approx$ 0.67).

Найвищі результати продемонстрував гібридний підхід (F1-score $\approx$ 0.82), який поєднує rule-based метод для формалізованих сутностей та

трансформерну модель для контекстно залежних. Це дозволило досягти більш збалансованих значень precision та recall та підвищити загальну якість екстракції інформації.

Додатково встановлено, що якість розпізнавання сутностей суттєво залежить від їх представленості у датасеті. Зокрема, для сутності COMPANY, яка має найменшу кількість прикладів, спостерігається зниження значень метрик якості, що підтверджує вплив дисбалансу даних на ефективність моделей.

У межах роботи також реалізовано програмний інтерфейс на основі бібліотеки Streamlit, що демонструє практичне застосування розробленого підходу та дозволяє виконувати екстракцію інформації з текстових документів у зручному вигляді.

Таким чином, результати дослідження підтверджують доцільність використання комбінованих підходів для задачі екстракції даних з неструктурованих текстових документів та ефективність застосування сучасних NLP-моделей у даній предметній галузі.

Перспективи подальших досліджень полягають у розширенні набору даних та балансуванні розподілу сутностей, що дозволить підвищити якість розпізнавання рідкісних категорій. Доцільним є також використання більш сучасних трансформерних моделей або донавчання моделей на доменно- специфічних корпусах, що може покращити здатність до узагальнення. Окрім цього, перспективним напрямом є розширення підходу на інші типи неструктурованих текстових документів, а також інтеграція розробленої системи з інформаційними сервісами для автоматизованої обробки та аналізу текстових даних.

Апробація результатів дослідження. Окремі положення дослідження було висвітлено у доповіді на 2 Міжнародній науково-практичній конференції «Сучасні інформаційні технології та системи штучного інтелекту (MIT&AIS-2026)» (27–29 квітня 2026 р., м. Харків-Яремче) [27].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Machado D. Understanding structured, semi-structured, and unstructured data. *Medium*. URL: <https://dorian599.medium.com/data-101-understanding-structured-semi-structured-and-unstructured-data-4c0cfc313037> (date of access: 06.04.2026).
2. A survey of information extraction based on deep learning / Y. Yang et al. *Applied sciences*. 2022. Vol. 12, no. 19. P. 9691. URL: <https://doi.org/10.3390/app12199691> (date of access: 06.04.2026).
3. Named entity recognition: a comprehensive guide to NLP's key technology. *Medium*. URL: <https://medium.com/@kanerika/named-entity-recognition-a-comprehensive-guide-to-nlps-key-technology-636a124eaa46> (date of access: 07.04.2026).
4. Saatci M., Kaya R., Ünlü R. Resume screening with natural language processing (NLP). *Alphanumeric journal*. 2024. URL: <https://doi.org/10.17093/alphanumeric.1536577> (date of access: 07.04.2026).
5. Pant D., Pokhrel D., Poudyal P. Automatic software engineering position resume screening using natural language processing, word matching, character positioning, and regex. *2022 5th international conference on advanced systems and emergent technologies (IC_ASET)*, Hammamet, Tunisia, 22–25 March 2022. 2022. URL: https://doi.org/10.1109/ic_aset53395.2022.9765916 (date of access: 07.04.2026).
6. Trinh Q., Dang T.-T. Automatic process resume in talent pool by applying natural language processing. *International conference on logistics and industrial engineering*. 2021. P. 234–240.
7. Neural architectures for named entity recognition / G. Lample et al. *Proceedings of the 2016 conference of the north american chapter of the association for computational linguistics: human language technologies*, San

Diego, California. Stroudsburg, PA, USA, 2016.

URL: <https://doi.org/10.18653/v1/n16-1030> (date of access:07.04.2026).

8. Chiu J. P. C., Nichols E. Named entity recognition with bidirectional LSTM-CNNs. *Transactions of the association for computational linguistics*. 2016. Vol. 4. P. 357–370. URL: https://doi.org/10.1162/tacl_a_00104 (date of access: 07.04.2026).

9. Joint extraction of entities and relations based on a novel tagging scheme / S. Zheng et al. *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: long papers)*, Vancouver, Canada. Stroudsburg, PA, USA, 2017. URL: <https://doi.org/10.18653/v1/p17-1113> (date of access: 08.04.2026).

10. BERT: pre-training of deep bidirectional transformers for language understanding / J. Devlin et al. *arXiv*. 2019. URL: <https://doi.org/10.48550/arXiv.1810.04805> (date of access: 08.04.2026).

11. LUKE: deep contextualized entity representations with entity-aware self-attention / I. Yamada et al. *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, Online. Stroudsburg, PA, USA, 2020. URL: <https://doi.org/10.18653/v1/2020.emnlp-main.523> (date of access: 09.04.2026).

12. LayoutLM: pre-training of text and layout for document image understanding / Y. Xu et al. *KDD '20: the 26th ACM SIGKDD conference on knowledge discovery and data mining*, Virtual Event CA USA. New York, NY, USA, 2020. URL: <https://doi.org/10.1145/3394486.3403172> (date of access: 09.04.2026).

13. LayoutLMv2: multi-modal pre-training for visually-rich document understanding / Y. Xu et al. *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing (volume 1: long papers)*, Online. Stroudsburg, PA, USA, 2021. URL: <https://doi.org/10.18653/v1/2021.acl-long.201> (date of access: 09.04.2026).

14. TPLinker: single-stage joint extraction of entities and relations through token pair linking / Y. Wang et al. *Proceedings of the 28th international conference on computational linguistics*, Barcelona, Spain (Online). Stroudsburg, PA, USA, 2020. URL: <https://doi.org/10.18653/v1/2020.coling-main.138> (date of access: 09.04.2026).
15. Joshi S. Top 10 resume formatting challenges for recruitment. *Peerlist*. URL: <https://peerlist.io/snehaljoshi/articles/resume-formatting-challenges-for-recruitment> (date of access: 10.04.2026).
16. Why 75% of resumes fail ATS screening: a technical analysis of resume parsing systems. *DEV Community*. URL: https://dev.to/vicente_sansilvestrecha/why-75-of-resumes-fail-ats-screening-a-technical-analysis-of-resume-parsing-systems-1hmg (date of access: 10.04.2026).
17. Kim K. Regular expressions: crafting patterns for text processing. *Medium*. URL: <https://medium.com/@kevinkoech265/regular-expressions-crafting-patterns-for-text-processing-165c178bc829> (date of access: 10.04.2026).
18. Khaouja I., Mezzour G., Kassou I. Unsupervised skill identification from job ads. *2021 IEEE 22nd international conference on information reuse and integration for data science (IRI)*, Las Vegas, NV, USA, 10–12 August 2021. 2021. URL: <https://doi.org/10.1109/iri51335.2021.00026> (date of access: 10.04.2026).
19. Pogeant N. Sequence Labeling – the basis of NLP. *Medium*. URL: <https://npogeant.medium.com/sequence-labeling-the-basis-of-nlp- a52c8fffa567> (date of access: 11.04.2026).
20. Karimi A. Hidden Markov models vs. conditional random fields. *Baeldung*. URL: <https://www.baeldung.com/cs/hidden-markov-vs-crf> (date of access: 11.04.2026).
21. Clinical named entity recognition from chinese electronic health

records via machine learning methods / Y. Zhang et al. *JMIR medical informatics*.



2018. Vol. 6, no. 4. P. e50. URL: <https://doi.org/10.2196/medinform.9965> (date of access: 11.04.2026).

22. Attention is all you need / A. Vaswani et al. *arXiv*. 2023. URL: <https://doi.org/10.48550/arXiv.1706.03762> (date of access: 26.04.2026).

23. Rong S. Named entity recognition – annotation schemes. *Medium*. URL: <https://medium.com/@rongqianhui/named-entity-recognition-annotation-schemes-e684f9cd5a56> (date of access: 12.04.2026).

24. A comprehensive overview of named entity recognition (NER). *LeewayHertz – AI Development Company*. URL: <https://www.leewayhertz.com/named-entity-recognition/#How-to-perform-named-entity-recognition-using-Python> (date of access: 12.04.2026).

25. Named entity recognition: entity types, BIO tagging & evaluation – interactive | michael brenndoerfer. *Michael Brenndoerfer | Data & AI, Private Equity, Technology*. URL: <https://mbrenndoerfer.com/writing/named-entity-recognition-ner-tutorial> (date of access: 13.04.2026).

26. Resume NER training dataset. *Kaggle: The World's AI Proving Ground*. URL: <https://www.kaggle.com/datasets/yashpwrr/resume-ner-training-dataset> (date of access: 15.04.2026).

27. Єремєєв Є., Тернинко Д., Золотухін О. Біоінспіровані ройові методи в задачах побудови адаптивних інтелектуальних систем. *Сучасні інформаційні технології та системи штучного інтелекту MIT&AIS-2026 : матеріали 2-ї Міжнар. науково-практ. конф., м. Харків - Яремче, 27–29 квіт. 2026 р. Харків: ХНУРЕ, 2026. С. 67–68.*