

**ПЕРЕЯСЛАВСЬКА СВИТЛАНА**

Державний заклад «Луганський національний університет імені Тараса Шевченка»

<https://orcid.org/0000-0001-9873-0447>email: [malinovskaya.yuliya2017@gmail.com](mailto:malinovskaya.yuliya2017@gmail.com)**СМАГІНА ОЛЬГА**

Державний заклад «Луганський національний університет імені Тараса Шевченка»

<https://orcid.org/0000-0002-6024-5152>email: [malinovskaya.yuliya2017@gmail.com](mailto:malinovskaya.yuliya2017@gmail.com)

## ВИВЧЕННЯ МЕТОДІВ ОБРОБКИ ПРИРОДНОЇ МОВИ ДЛЯ СТВОРЕННЯ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ, ТАКИХ ЯК ЧАТ-БОТИ ТА СИСТЕМИ АВТОМАТИЧНОГО ПЕРЕКЛАДУ

У статті представлені результати досліджень методів обробки природної мови (NLP) для розробки інтелектуальних систем, таких як чат-боти та системи автоматичного перекладу. Розглянуто основні підходи до обробки природної мови, включаючи токенізацію, синтаксичний аналіз, глибоке навчання, контекстні моделі та нейронні мережі. Встановлено взаємозв'язок між використовуваним алгоритмом і точністю відповідей системи. Результати цього дослідження можуть бути використані для покращення ефективності сучасних систем штучного інтелекту.

Ключові слова: штучний інтелект, машинне навчання, розпізнавання тексту, машинний переклад, нейронні мережі.

**PEREIASLAVSKA SVITLANA,****SMAHINA OLHA**

Luhansk Taras Shevchenko National University

## STUDY OF NATURAL LANGUAGE PROCESSING METHODS FOR CREATING INTELLIGENT SYSTEMS, SUCH AS CHATBOTS AND AUTOMATIC TRANSLATION SYSTEMS

The article's purpose is to study NLP methods for creating intelligent systems, such as chatbots and automatic translators, and to improve their efficiency, accuracy, and adaptability to user needs.

Scientific novelty. This article discusses modern approaches to natural language processing, including the use of transducers, neural networks, and deep learning for real-time text processing. Optimizations for parsing, text generation, and integration of multilingual models are introduced to improve the quality of translation and dialog systems.

Results. The main methods of natural language processing, such as tokenization, lemmatization, and stemming, as well as their impact on text pre-processing are analyzed. The advantages of contextual models (BERT, GPT) compared to static vector representations of words are revealed, which allows for more efficient consideration of semantic and syntactic dependencies. The effectiveness of attention mechanisms in converters for processing large datasets and multilingual texts is demonstrated, which improves translation accuracy and dialog quality. It was found that optimizing the architecture of deep neural networks (adding global and local attention mechanisms) significantly improved the chatbot's performance during contextual tasks. This article discusses the use of tone analysis and named entity recognition (NER) methods to improve the chatbot's adaptability to different use cases. Recommendations for integrating natural language processing models into automatic translation systems while minimizing computational costs and maintaining high performance are formulated.

Conclusion The development and implementation of modern natural language processing models can significantly improve the efficiency of systems such as chatbots and automatic translators, providing a more natural interaction with users. The use of contextual models, transducers, and attention mechanisms helps to improve the quality of text processing, especially in multilingual environments. Adaptive model integration can expand chatbots' capabilities, for example, in healthcare, education, and customer-oriented industries. Further developments in natural language processing will focus on developing multimodal systems capable of analyzing text, images, and audio, as well as ensuring transparency of model decisions to increase user trust. Key areas for further research include minimizing the resource consumption of models, improving their ability to work in real-time, and expanding the availability of technology to accommodate different languages and cultures.

Keywords: artificial intelligence, machine learning, text recognition, machine translation, neural networks.

### Постановка проблеми у загальному вигляді

#### та її зв'язок із важливими науковими чи практичними завданнями

Обробка природної мови (NLP) є важливою галуззю штучного інтелекту (ШІ), яка дозволяє комп'ютерам обробляти людську мову та спілкуватися з нею. Ця технологія дозволяє створювати інтелектуальні системи, які включають чат-ботів та системи автоматичного перекладу. Удосконалена діалогова система, відома як чат-бот, функціонує з метою створення імітації людських розмов. Віртуальна підтримка клієнтів у різних галузях працює завдяки машинному навчанню та методам NLP. Чат-боти існують для оптимізації задоволеності користувачів шляхом негайного реагування на популярні запити, що призводить до підвищення операційної ефективності обслуговування клієнтів. Ефективність чат-ботів значно покращується, коли дослідники оптимізують певні змінні моделі в архітектурі глибоких нейронних мереж. Чат-бот досягає кращої якості взаємодії з користувачем, коли він використовує як локальні, так і глобальні механізми уваги для ефективної обробки запитів користувачів. Процес навчання стає ефективнішим, коли набори даних розширюються і проходження

відбувається одночасно, що призводить до покращення розуміння лексики та логічної реакції чат-ботів. Системи машинного перекладу є фундаментальними компонентами для розвитку міжнародної торгівлі та полегшення комунікації. Традиційні процеси перекладу демонструють неефективність і неточність при роботі з багатьма мовними комбінаціями. Поєднання обробки природної мови з машинним перекладом і нейронними мережами робить проблеми, пов'язані з неефективністю перекладу, менш значущими, що сприяє підвищенню ефективності багатомовного перекладу.

### **Аналіз досліджень та публікацій**

Досліджуючи з 2000 року, Koç and Savaş [1] оцінюють, як чат-боти на основі ШІ підтримують вивчення англійської мови та які навчальні переваги отримують студенти від цих систем. Chen Y., Wang H., Yu K., Zhou R. [2] дають повний опис методів NLP AI та детально описують, що було досягнуто за допомогою цих методів. Chen Y. та ін. [3] аналізують численні методи ШІ в NLP, щоб довести їхню важливу роль у розвитку галузі. Pothuri V. [4] визначає, як NLP допомагає розвивати функції розмовного ШІ, пояснюючи, як користувачі взаємодіють із системами ШІ за межами машин. Shiva K. [5, 6] розглядає використання NLP у чат-ботах для обслуговування клієнтів. Ortiz-Garces I. et al. [7] демонструють, що досконалий аналіз структури речень підвищує здатність систем штучного інтелекту природно реагувати при вирішенні завдань з обслуговування клієнтів. Onyshchenko, K., & Daniel, Y. [8] досліджують, як LSTM можна використовувати для фіксації та обробки довгострокових залежностей у мовних даних, що має вирішальне значення для різних завдань NLP. Dalayli, F. [9] розглядає ефективність методів NLP ChatGPT у перекладацькому контексті. Chakraborty, C., Pal, S., Bhattacharya, M., Dash, S., & Lee, S. S. [10] окреслюють потенціал чат-ботів зі штучним інтелектом у медичній науці, зокрема в догляді за пацієнтами, медичній освіті та дослідженнях.

### **Формулювання цілей статті**

**Метою роботи є:** виявлення методів для розпізнання текстів за допомогою ШІ з демонстрацією їх роботи на Python коді.

### **Виклад основного матеріалу**

Сучасні інтелектуальні системи перекладу складаються з функціональних модулів, що оптимізують процес перекладу. Використовуючи передові алгоритми, ці системи досягають високої точності без втручання людини і демонструють значні переваги порівняно з традиційними методами як за швидкістю, так і за точністю. Використовуючи методи глибокого навчання, ці системи здатні аналізувати величезні обсяги голосових даних, що робить їх незамінними інструментами в сучасному взаємопов'язаному світі. Область NLP працює в рамках ШІ як підрозділ, що займається взаємодією між машинами та людьми за допомогою природної розмовної мови. Завдяки його алгоритмам і статистичним моделям комп'ютери отримують можливість обробляти, розуміти і генерувати дані в межах природної мови. NLP підтримує розробку інтелектуальних технологічних систем, таких як чат-боти та платформи автоматичного перекладу. Аналіз вводу користувача за допомогою NLP дозволяє чат-ботам обробляти як письмовий текст, так і вимовлені голосові команди, щоб ідентифікувати реальні потреби користувача.

Здатність чат-ботів обробляти природну мову дозволяє їм давати відповіді, які відповідають інформації, що вводиться користувачем. Система отримує дані зі своєї бази знань, а також виконує запити в Інтернеті, перш ніж надати креативну відповідь або здійснити пошук потрібної інформації.

Завдяки контекстному розумінню NLP дозволяє чат-ботам ефективно відповідати на розмови, що призводить до безперешкодного діалогу та підвищує залученість користувачів.

Системи автоматичного перекладу, відомі як системи машинного перекладу, працюють шляхом перетворення письмового або усного контенту між мовами. Для розробки точних систем перекладу необхідне застосування NLP, оскільки це є їхньою фундаментальною вимогою.

Аналізатор мовного джерела в системах NLP визначає, як граматику, синтаксис і семантична структура тексту існують у мовному тексті.

Одиниці перекладу з'являються завдяки технології NLP, яка дозволяє системам виявляти відповідні компоненти перекладу на рівні слів, словосполучень або речень.

Процес генерації цільової мови NLP створює текст, який відповідає особливостям цільової мови, враховуючи при цьому контекст перекладу.

NLP використовує кілька методів для побудови інтелектуальних систем:

Токенізація тексту поділяє матеріал на окремі слова або токени.

Розмітка частин мови (POS) ідентифікує кожне слово в реченні через його лінгвістичну класифікацію на іменники, дієслова, прикметники та інші частини мови.

NER відстежує конкретні об'єкти в текстових документах, які включають імена разом з місцями та організаціями.

Синтаксичний аналіз: Аналізує граматичну структуру речення, наприклад, зв'язок між підметом і дієсловом.

Машинне навчання: використовує алгоритми машинного навчання для навчання моделей на великих наборах даних, щоб підвищити точність завдань обробки природної мови.

Глибоке навчання: архітектури глибокого навчання, як-от рекурентні нейронні мережі (RNN) і згорткові нейронні мережі (CNN), використовують для аналізу та генерації даних природною мовою.

Системи, засновані на правилах: часто використовуються в поєднанні з методами машинного

навчання і глибокого навчання.

Як стверджують Koç F. Ş., Savaş P.: «Чат-боти на основі ШІ демонструють великий потенціал для поліпшення результатів вивчення мови, особливо в контексті англійської як другої мови. Його здатність забезпечувати миттєвий зворотний зв'язок, персоналізований навчальний процес і занурювальну практику спілкування робить його цінним інструментом у сучасній освіті» [1].

Як зазначають Chen Y., Wang H., Yu K., Zhou R.: «Останні досягнення в галузі ШІ, зокрема, глибоке навчання і моделі на основі перетворювачів, зробили революцію в NLP. Ці технології забезпечують більш точне розуміння мови, генерацію та переклад, торуючи шлях до більш досконалих діалогових агентів і систем перекладу» [2].

Згідно з Pothuri V.: «Системи розмовного ШІ на основі NLP трансформують сфери обслуговування клієнтів, освіти та охорони здоров'я, надаючи можливість вести діалоги, схожі на людські. Інтеграція аналізу настроїв і контекстної обробки значно підвищила ефективність цих систем» [4].

Shiva K. et al. відзначають: «Чат-боти на основі NLP все частіше використовуються для обробки складних запитів клієнтів, забезпечення цілодобової підтримки та зниження експлуатаційних витрат. Використання розпізнавання намірів і вилучення сутностей зробило ці системи більш ефективними та зручними для користувачів» [5].

Ortiz-Garces I. et al. зауважують: «Розширені методи аналізу, такі як аналіз залежностей і маркування семантичних ролей, допомагають підвищити точність і швидкість реагування чат-ботів. Ці методи дозволяють краще зрозуміти наміри та контекст користувача» [7].

Виділяють такі методи обробки природної мови (NLP) з їх порівняльною характеристикою:

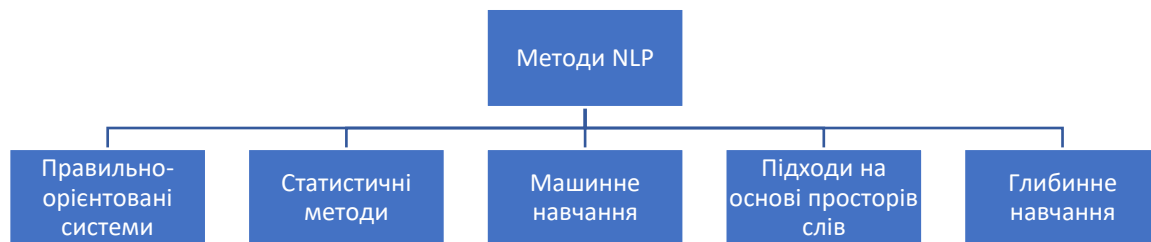
Системи на основі правил: інтерпретують мову за допомогою набору правил, розроблених експертами.

Статистичні методи: використовують ймовірності, що базуються на частоті вхідних даних.

Машинне навчання: робить прогнози за допомогою алгоритмів, які навчаються на прикладах.

Глибоке навчання: використовує нейронні мережі для виконання складних завдань обробки природної мови.

Підхід на основі простору слів: уявіть слова у векторному просторі та допоможіть комп'ютеру зрозуміти їхні взаємозв'язки (Рис. 1 і Таблиця 1).



**Рис. 1 Методи NLP**  
Джерело: авторська розробка

Таблиця 1

**Компаративний аналіз методів NLP**

Назва методу	Опис	Приклади	Переваги	Недоліки
Правильно-орієнтовані системи	Використання набору правил, написаних спеціалістами, для інтерпретації мови.	Граматичні правила для синтаксичного аналізу. Шаблони для розпізнавання іменованих сутностей.	Точність у вузьких областях.	Складність масштабування та обмеженість правилами.
Статистичні методи	Використання ймовірностей на основі частот слів або фраз у тексті.	Модель n-грам для передбачення наступного слова. Статистичний машинний переклад (SMT).	Простота реалізації, ефективність для базових завдань	Обмежена здатність до узагальнення.
Машинне навчання (ML)	Використання алгоритмів, які навчаються на прикладах для передбачення або класифікації	Класифікація текстів за допомогою SVM або Random Forest. Аналіз тональності.	Гнучкість, здатність до узагальнення.	Потреба у великих обсягах даних для навчання.

Продовження таблиці 1

Глибинне навчання (Deep Learning)	Використання нейронних мереж для вирішення складних завдань NLP.	Моделі LSTM, GRU для обробки послідовностей. Трансформери (Transformer) для машинного перекладу (наприклад, BERT, GPT).	Висока точність, здатність до роботи з контекстом.	Затратність обчислень, потреба у великих датасетах.
Підходи на основі просторів слів (Word Embeddings)	Представлення слів у векторному просторі для відображення їх семантичних зв'язків.	Word2Vec, GloVe, FastText. Використання векторів для знаходження синонімів або аналогій.	Здатність захоплювати семантичні відносини.	Обмежена здатність до роботи з контекстом (без використання контекстуальних моделей).

Джерело: авторська розробка на основі [1-10]

Дослідження в галузі обробки природної мови створюють інтелектуальні системи, такі як чат-боти та інструменти автоматичного перекладу. Ось основні кроки та методи, які використовує NLP для створення цих систем.

#### 1. Попередня обробка тексту.

Процес розбиття тексту розбиває документ на окремі слова, які називаються токенами.

Процеси стандартизації нормалізують вербалізацію шляхом перетворення на малі літери та видалення розділових знаків.

Алгоритми нормалізують слова до їхніх кореневих форм під час обробки (позмінна робота перетворюється на прогулянку).

Відкидання малозначущих слів, таких як «and», «in» або «on», для покращення семантичної якості.

#### 2. Векторизація тексту.

Текст стає вектором за допомогою функції Bag of Words, яка показує, як часто слова з'являються в даному тексті. При обчисленні за допомогою алгоритмів TF-IDF слова виділяються в одному документі порівняно з іншими документами.

Word Embeddings працює з векторами слів, включаючи продукти Word2Vec, GloVe і FastText, які вимірюють слова на основі їхніх сусідніх значень.

Системи BERT і GPT вивчають значення слів, дивлячись на їхню позицію в реченнях.

#### 3. Моделювання та аналіз тексту.

Моделі сортують текст на заздалегідь визначені групи, застосовуючи машинне навчання (машини опорних векторів або випадковий ліс) і глибоке навчання (мережі довготривалої короткочасної пам'яті або згорткові нейронні мережі).

Аналіз настроїв: визначення емоційного забарвлення тексту (позитивного, негативного, нейтрального). Замість розпізнавання об'єктів (NER) знаходить і маркує всі власні імена, що зустрічаються в тексті.

Набори LM знаходять найважливіші теми, що існують у колекції текстів, за допомогою LDA (Latent Dirichlet Allocation).

#### 4. Генерація тексту.

Модель GPT допомагає створювати текстовий вивід на основі доступної контекстної інформації.

Використовуйте архітектури Transformer або LSTM у моделях Seq2Seq для задач, які потребують зміни послідовності тексту, зокрема для машинного перекладу.

#### 5. Машинний переклад.

SMT працює зі статистичними моделями для перетворення письмового тексту з однієї мови на іншу. Нейронний машинний переклад (НМТ) покладається на нейронні мережі, зокрема Transformer, для створення якісніших і гладших перекладів.

#### 6. Розробка чат-ботів.

Шаблони та правила допомагають сканувати певні проблеми.

Чат-боти стають розумнішими та краще реагують на індивідуальні потреби завдяки системам машинного навчання Rasa та Dialogflow.

Персонаж GPT генерує миттєві відповіді завдяки архітектурі генеративної моделі.

#### 7. Оцінювання та вдосконалення моделей.

Для перевірки якості моделі ми застосовуємо вимірювання точності та повноти, а також середньоквадратичне значення F1, а також оцінки машинного перекладу на основі BLEU.

Система покращує продуктивність після навчання моделей на вибраних наборах даних.

Вивчайте помилки моделі, щоб знайти закономірності та внести оновлення для їх виправлення.  
8. *Інтеграція та розгортання.*

Розробка API, щоб зробити моделі NLP доступними у веб-додатках, мобільних додатках та інших платформах

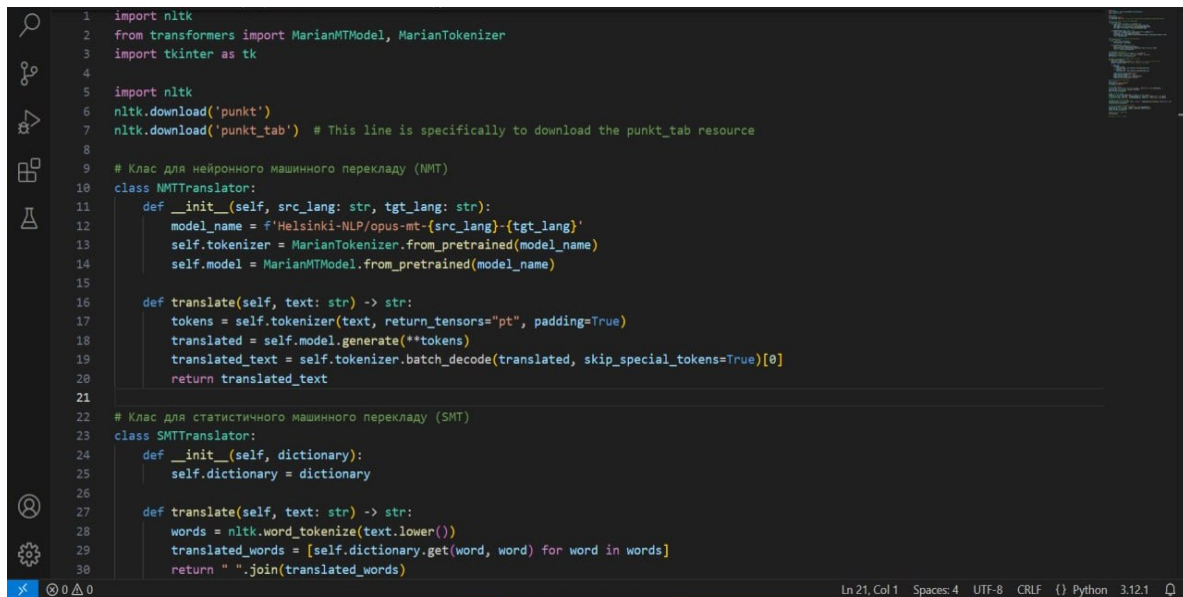
Команда розміщує та вдосконалює системи NLP на таких платформах, як AWS, Google Cloud та Azure за допомогою хмарних сервісів.

9. *Етика та конфіденційність.*

Захист даних: забезпечення конфіденційності даних користувачів.

Етична розробка систем НЛП вимагає моделей, що дозволяють уникнути упередженості при їх розробці. Дослідження цих шляхів призводить до розробки автоматизованих текстових систем, які працюють точно і легко дозволяють спілкуватися.

Розроблений код на Python у Visual Studio демонструє створення графічного інтерфейсу для перекладу тексту з використанням двох методів: нейронного машинного перекладу (NMT) і статистичного машинного перекладу (SMT) (Рис. 2):



```

1 import nltk
2 from transformers import MarianMTModel, MarianTokenizer
3 import tkinter as tk
4
5 import nltk
6 nltk.download('punkt')
7 nltk.download('punkt_tab') # This line is specifically to download the punkt_tab resource
8
9 # Клас для нейронного машинного перекладу (NMT)
10 class NMTTranslator:
11     def __init__(self, src_lang: str, tgt_lang: str):
12         model_name = f'Helsinki-NLP/opus-mt-{src_lang}-{tgt_lang}'
13         self.tokenizer = MarianTokenizer.from_pretrained(model_name)
14         self.model = MarianMTModel.from_pretrained(model_name)
15
16     def translate(self, text: str) -> str:
17         tokens = self.tokenizer(text, return_tensors="pt", padding=True)
18         translated = self.model.generate(**tokens)
19         translated_text = self.tokenizer.batch_decode(translated, skip_special_tokens=True)[0]
20         return translated_text
21
22 # Клас для статистичного машинного перекладу (SMT)
23 class SMTTranslator:
24     def __init__(self, dictionary):
25         self.dictionary = dictionary
26
27     def translate(self, text: str) -> str:
28         words = nltk.word_tokenize(text.lower())
29         translated_words = [self.dictionary.get(word, word) for word in words]
30         return " ".join(translated_words)

```

Рис. 2. Робочий код на Python

Джерело: авторська розробка

Нижче наведено детальне пояснення кроків коду:

1. *Імпорт бібліотек*

python

Copy

import nltk

from transformers import MarianMTModel, MarianTokenizer

import tkinter as tk

nltk: Бібліотека для обробки природної мови (Natural Language Toolkit), що використовується для токенизації тексту.

transformers: Бібліотека від Hugging Face, яка надає доступ до моделей NLP - MarianMT для машинного перекладу.

tkinter: Стандартна бібліотека Python для створення графічних інтерфейсів.

2. *Завантаження ресурсів NLTK*

python

Copy

nltk.download('punkt')

nltk.download('punkt\_tab')

punkt представляє собою ресурс для перетворення текстів в окремі слова..

punkt\_tab слугує додатковим ресурсом для токенизації, який може знадобитися для роботи зі специфічними мовами. (Рис. 3):



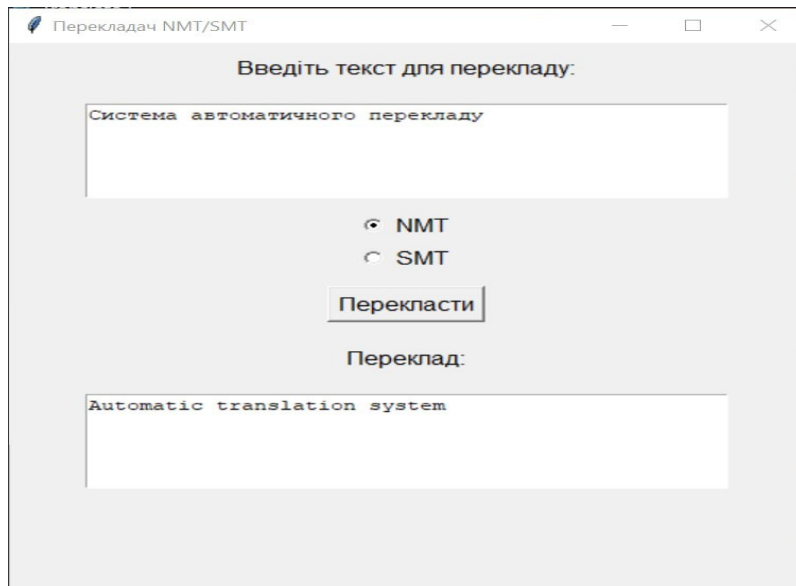


```

method = method_var.get() # Отримати вибраний метод перекладу
if input_text:
    «nmt» if method == «nmt»:
        translated_text = nmt_translator.translate(input_text)
    elif method == «smt»:
        translated_text = smt_translator.translate(input_text)
    output_text_box.config(state=«normal»)
    output_text_box.delete(«1.0», «end»)
    output_text_box.insert(«1.0», translated_text)
    output_text_box.config(state=«disabled»)

```

Ця функція перекладає текст на основі обраного нами методу перекладу (NMT або SMT). Вона витягує текстову інформацію з поля введення. Визначає, який метод перекладу має обробляти вхідні дані. Вона показує результати перекладу в області виведення. (Рис. 4):



**Рис. 4. Інтерфейс програми**  
Джерело: авторська розробка

#### 7. Створення графічного інтерфейсу

```
python
```

Скопіювати

```
root = tk.Tk()
```

```
root.title(«Перекладач NMT/SMT»)
```

```
root.geometry(«500x350»)
```

Програма виведе на екран головне вікно із зазначеними заголовком і розмірами.

Поле введення тексту

```
python
```

Копіювати

```
tk.Label(root, text=«Введіть текст для перекладу:», font=(«Arial», 12)).pack(pady=10)
```

```
input_text_box = tk.Text(root, height=5, width=50)
```

```
input_text_box.pack(pady=10)
```

Додано текстове поле для введення.

Вибір методу перекладу

```
python
```

Копіювання

```
method_var = tk.StringVar(value=«NMT») # Значення за замовчуванням
```

```
tk.Radiobutton(root, text=«NMT», variable=method_var, value=«NMT», font=(«Arial», 12)).pack()
```

```
tk.Radiobutton(root, text=«SMT», variable=method_var, value=«SMT», font=(«Arial», 12)).pack()
```

Користувачі можуть обирати між NMT та SMT за допомогою опцій кнопки.

Кнопка для виконання перекладу

```
python
```

Копіювати

Кнопка translate\_button запускає функцію perform\_translation і відображає текст шрифтом Arial розміром 12.

```
translate_button.pack(pady=10)
```

Кнопка запускає операцію перекладу.

Додаток показує кінцевий переклад у цьому полі

```
python
```

Копіювати

```
tk.Label(root, text=«Переклад:», font=(«Arial», 12)).pack(pady=10)
```

```
output_text_box = tk.Text(root, height=5, width=50, state=«disabled»)
```

```
output_text_box.pack(pady=10)
```

Зробіть пробіл для відображення результату перекладу.

8. Програма переходить в основний цикл відображення вікна

```
python
```

Копіюємо

```
root.mainloop()
```

Програма починає показувати вікно через центральний цикл обробки подій.

9. Перевірка роботи програми

Введіть текст: "Привіт, як справи".

Виберіть метод перекладу (NMT або SMT).

Натисніть кнопку "Перекласти".

Результат:

NMT: "Hello, how are you?"

SMT: "hello how are you"

У цьому коді показано, як створити простий перекладач, використовуючи два методи: нейронний переклад і статистичний машинний переклад. Графічний інтерфейс робить програму зручною для користувача.

Незважаючи на прогрес, досягнутий у NLP, все ще існують деякі проблеми та напрямки для майбутнього:

Боротьба з неоднозначністю: системи NLP часто стикаються з неоднозначною мовою, коли одне слово або фраза має кілька значень.

Розуміння контексту: сучасні системи NLP мають вдосконалювати свої можливості у розпізнаванні ситуативних особливостей мовлення, враховуючи наміри користувача, емоційний колорит та стилістику мовлення.

Мультимодальна інтеграція: для більш повноцінного аналізу природної мови необхідно забезпечити взаємодію різних типів даних – текстових, голосових та візуальних.

Прозорість алгоритмів: результати роботи систем NLP повинні бути зрозумілими для користувачів, а також обґрунтовуватися логічними поясненнями щодо прийнятих рішень.

Підтримка багатьох мов: важливо розширити функціональність систем до опрацювання різномовних даних, що сприятиме міжнародному спілкуванню та кооперації.

У цьому контексті NLP виступає як фундамент для розробки розумних технологій, таких як чат-боти та перекладачі. Глибше знання принципів й завдань NLP допоможе створити більш точні, продуктивні та ефективні інтелектуальні системи, які значно поліпшать якість життя.

### **Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі**

Методи обробки природної мови (NLP) є ключовими для розробки інтелектуальних систем, таких як чат-боти та служби автоматичного перекладу. Завдяки машинному навчанню, глибокому навчанню та складним алгоритмам, ці системи покращують користувацький досвід і полегшують комунікацію, долаючи мовні бар'єри. Постійний прогрес у сфері NLP робить ці методи все більш ефективними та зручними для користувачів. Створення інтелектуальних систем, таких як чат-боти та системи автоматичного перекладу, є одним із найперспективніших напрямків розвитку NLP. Вони допомагають людям отримувати інформацію, виконувати завдання, підтримувати взаємодію та краще розуміти один одного, збільшуючи свою корисність і привабливість на ринку. Майбутнє NLP обіцяє відкрити нові можливості для більш природної взаємодії між людиною і комп'ютером, наближаючи її до людської мови.

### **Література**

1. Koç, F. Ş., & Savaş, P. (2025). The use of artificially intelligent chatbots in English language learning: A systematic meta-synthesis study of articles published between 2010 and 2024. *ReCALL*, 37(1), 4–21. <https://doi.org/10.1017/S0958344024000168>
2. Chen, Y., Wang, H., Yu, K., & Zhou, R. (2024). Artificial intelligence methods in natural language processing: A comprehensive review. *Highlights in Science, Engineering and Technology*, 85, 545–550. <https://doi.org/10.54097/vfwgas09>



3. Chen, Y., Wang, H., Yu, K., & Zhou, R. (2024). Artificial intelligence methods in natural language processing: A comprehensive review. *Highlights in Science, Engineering and Technology*, 85, 545–550. <https://doi.org/10.54097/vfwgas09>
4. Vijaya, P. (2024). Natural language processing and conversational AI. *International Research Journal of Modernization in Engineering Technology and Science*, 6, 436–440. <https://doi.org/10.56726/IRJMETS61417>
5. Krishnateja, Etikani, P., Venkata, V., Kamireddy, V. V. S. R. B., Kaur, J., Thakkar, D., Kanchetti, D., & Munirathnam, R. (2024). Natural language processing for customer service chatbots: Enhancing customer experience. *International Journal of Intelligent Systems and Applications in Engineering*, 155–164.
6. Shiva, K. (2024). Natural language processing for customer service chatbots: Enhancing customer experience. *International Journal of Intelligent Systems and Applications in Engineering*, 12(22s), 155–164.
7. Ortiz-Garces, I., Govea, J., Andrade, R. O., & Villegas-Ch, W. (2024). Optimizing chatbot effectiveness through advanced syntactic analysis: A comprehensive study in natural language processing. *Applied Sciences*, 14(5), Article 1737. <https://doi.org/10.3390/app14051737>
8. Onyshchenko, K., & Daniil, Y. (2023). Using long short-term memory networks for natural language processing. *Bulletin of National Technical University KhPI. Series: System Analysis, Control and Information Technologies*, 89–96. <https://doi.org/10.20998/2079-0023.2023.01.14>
9. Dalayli, F. (2023). Use of NLP techniques in translation by ChatGPT: Case study. In *Proceedings of the Workshop on Computational Terminology in NLP and Translation Studies (ConTeNTS) Incorporating the 16th Workshop on Building and Using Comparable Corpora (BUCC)* (pp. 19–25). INCOMA Ltd.
10. Chakraborty, C., Pal, S., Bhattacharya, M., Dash, S., & Lee, S. S. (2023). Overview of chatbots with special emphasis on artificial intelligence-enabled ChatGPT in medical science. *Frontiers in Artificial Intelligence*, 6, Article 1237704. <https://doi.org/10.3389/frai.2023.1237704>

## References

1. Koç, F. Ş., & Savaş, P. (2025). The use of artificially intelligent chatbots in English language learning: A systematic meta-synthesis study of articles published between 2010 and 2024. *ReCALL*, 37(1), 4–21. <https://doi.org/10.1017/S0958344024000168>
2. Chen, Y., Wang, H., Yu, K., & Zhou, R. (2024). Artificial intelligence methods in natural language processing: A comprehensive review. *Highlights in Science, Engineering and Technology*, 85, 545–550. <https://doi.org/10.54097/vfwgas09>
3. Chen, Y., Wang, H., Yu, K., & Zhou, R. (2024). Artificial intelligence methods in natural language processing: A comprehensive review. *Highlights in Science, Engineering and Technology*, 85, 545–550. <https://doi.org/10.54097/vfwgas09>
4. Vijaya, P. (2024). Natural language processing and conversational AI. *International Research Journal of Modernization in Engineering Technology and Science*, 6, 436–440. <https://doi.org/10.56726/IRJMETS61417>
5. Krishnateja, Etikani, P., Venkata, V., Kamireddy, V. V. S. R. B., Kaur, J., Thakkar, D., Kanchetti, D., & Munirathnam, R. (2024). Natural language processing for customer service chatbots: Enhancing customer experience. *International Journal of Intelligent Systems and Applications in Engineering*, 155–164.
6. Shiva, K. (2024). Natural language processing for customer service chatbots: Enhancing customer experience. *International Journal of Intelligent Systems and Applications in Engineering*, 12(22s), 155–164.
7. Ortiz-Garces, I., Govea, J., Andrade, R. O., & Villegas-Ch, W. (2024). Optimizing chatbot effectiveness through advanced syntactic analysis: A comprehensive study in natural language processing. *Applied Sciences*, 14(5), Article 1737. <https://doi.org/10.3390/app14051737>
8. Onyshchenko, K., & Daniil, Y. (2023). Using long short-term memory networks for natural language processing. *Bulletin of National Technical University KhPI. Series: System Analysis, Control and Information Technologies*, 89–96. <https://doi.org/10.20998/2079-0023.2023.01.14>
9. Dalayli, F. (2023). Use of NLP techniques in translation by ChatGPT: Case study. In *Proceedings of the Workshop on Computational Terminology in NLP and Translation Studies (ConTeNTS) Incorporating the 16th Workshop on Building and Using Comparable Corpora (BUCC)* (pp. 19–25). INCOMA Ltd.
10. Chakraborty, C., Pal, S., Bhattacharya, M., Dash, S., & Lee, S. S. (2023). Overview of chatbots with special emphasis on artificial intelligence-enabled ChatGPT in medical science. *Frontiers in Artificial Intelligence*, 6, Article 1237704. <https://doi.org/10.3389/frai.2023.1237704>