



українська ▼

англійська ▼

Презентація до курсової  
роботи на тему:  
“Після-Машинний переклад”



Presentation for course work  
on the topic:  
"Post-Machine translation"





1. Основна інформація
2. Історія
3. Типи МП та їхні принципи роботи
4. Схема роботи сучасного машинного перекладу
5. Висновки
6. Література
7. Результати курсового проекту

# ☰ Що таке машинний переклад?

**Машинний переклад** (МП) — технології автоматизованого перекладу текстів (письмових та усних) з однієї природної мови на іншу за допомогою комп'ютера; напрямок наукових досліджень, пов'язаний з побудовою систем автоматизованого перекладу.

Машинний переклад — одна з підгруп комп'ютерної лінгвістики, яка досліджує використання програмного забезпечення для перекладу тексту з однієї мови на іншу



## Загальні тези:

- Початок історії машинного перекладу датується сімнадцятим століттям, коли філософи Лейбніц і Декарт прогнозували створення кодів, що могли б зв'язати слова між мовами.
- Вперше можливість машинного перекладу на практиці передбачив Ч. Беббідж, що у першій половині 19 століття працював над проектом цифрової аналітичної машини – механічного прототипу електронних цифрових обчислювальних машин.
- Перші патенти на створення перекладацьких машин було видано у середині 30-х років минулого століття.
- Першу пропозицію машинного перекладу за допомогою комп'ютера було у 1947 р висунуто Уорреном Вівером, дослідником з Фонду Рокфеллера у його меморандумі.
- 7 січня 1954 у Нью-Йорку в головному офісі IBM було вперше проведено публічну демонстрацію системи машинного перекладу (МП).



# Типи систем машинного перекладу

Існують три принципово різних підходи до побудови алгоритмів машинного перекладу:

- заснований на правилах (rule-based)
- статистичний, або заснований на статистиці (statistical-based)
- з використанням нейромереж

The screenshot displays a web-based machine translation interface with two examples. Each example has a text input area on the left and a translation output area on the right. The interface includes language selection dropdowns at the top of each section and a 'Перевести' (Translate) button.

**Example 1:**

Input (Russian): Уронили мишку на пол, Оторвали мишке лапу. Всё равно его не брошу - Потому что он хороший.

Output (Latin): Ursus fundes super pavimento; Abscissum ursi pede. Et tamen non deficere - Et quia est bonum.

**Example 2:**

Input (Russian): Уронили мишку на пол, Оторвали мишке лапу. Всё равно его не брошу - Потому что он хороший.

Output (Latin): Nести кровь на полу; Лапа отрубленную медведя. Тем не менее, они никогда не сдаваться - И это хорошо.



## Типи систем машинного перекладу

Останніми роками все більшої популярності набирає Гібридний МП (Hybrid machine translation [HMT]), який поєднує в собі переваги 3 видів машинного перекладу: RB, статистичного та нейронного.

Такий підхід дозволяє користуватися перевагами NMT та SMT які в процесі перекладу контролюються правилами RBMT. Єдиним недоліком цієї системи перекладу є невід'ємна складність такої роботи, і робить його одним з найбільш амбіційних проектів сьогодення.



## Схема роботи сучасного машинного перекладу

*“London is the capital and most populous city of England and the United Kingdom. Standing on the River Thames in the south east of the island of Great Britain, London has been a major settlement for two millennia. It was founded by the Romans, who named it Londinium.”*

1. *London is the capital and most populous city of England and the United Kingdom.*
2. *Standing on the River Thames in the south east of the island of Great Britain, London has been a major settlement for two millennia.*
3. *It was founded by the Romans, who named it Londinium.*



*London is the capital and most populous city of England and the United Kingdom.*

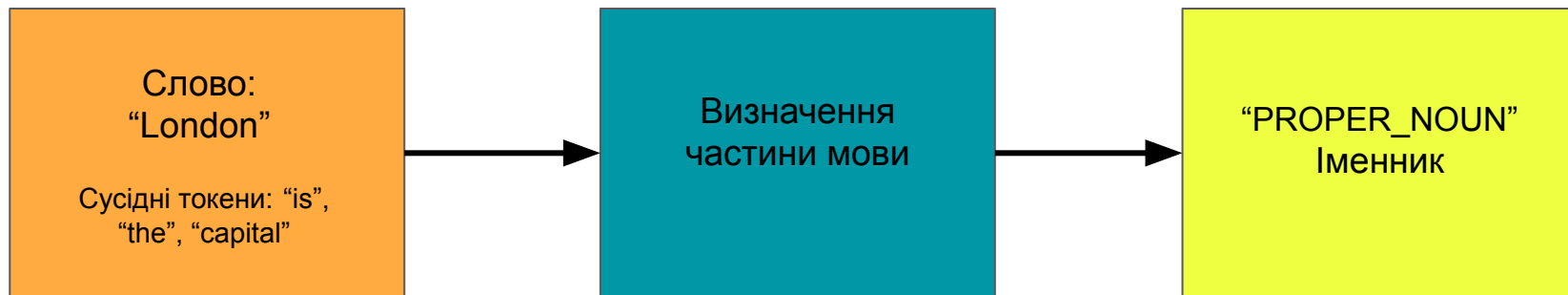


## Крок 2. Токенізація, або виділення слів

*«London», «is», «the», «capital», «and», «most», «populous», «city», «of»,  
«England», «and», «the», «United», «Kingdom», «. »*

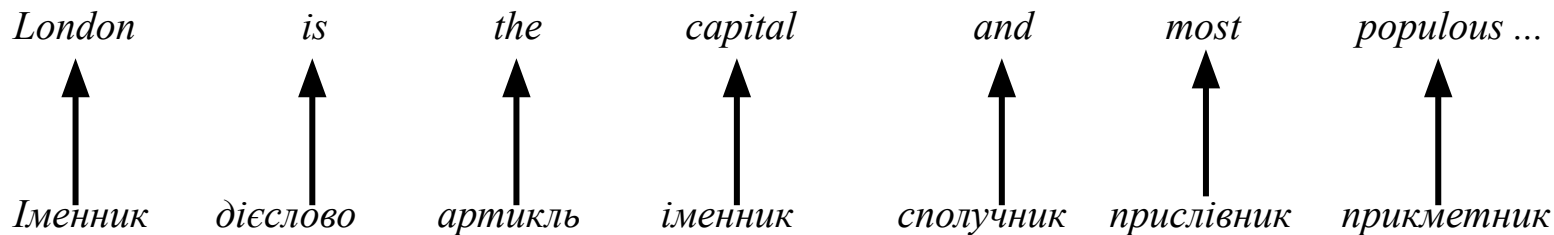


## Крок 3. Визначення частин мови





## Крок 3. Визначення частин мови





## Крок 4. Лематизація

*I had a **pony**.*

*I had two **ponies**.*



## Крок 4. Лематизація

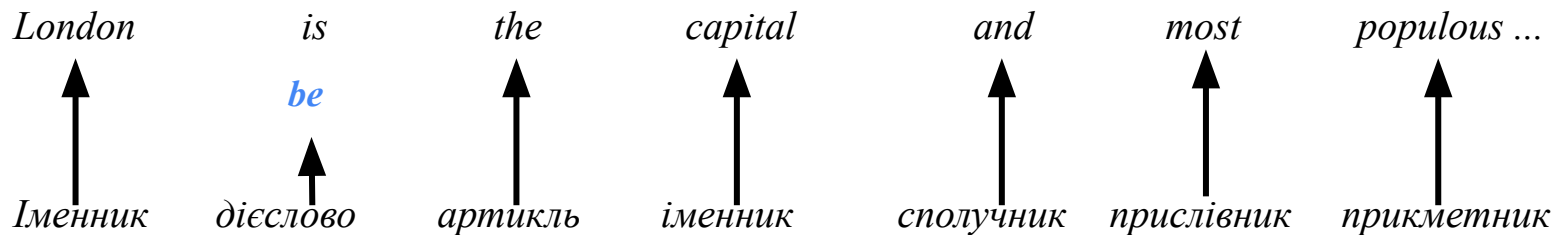
«*I had two ponies*»



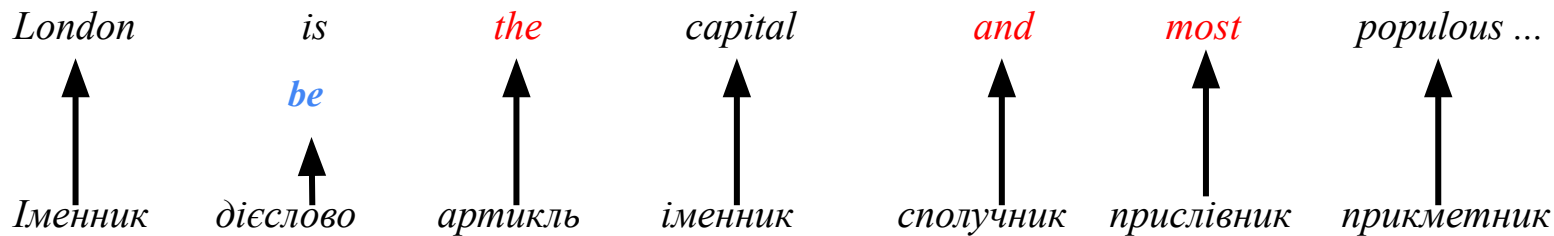
«*I [have] two [pony]*»



## Крок 4. Лематизація



## Крок 5. Визначення стоп-слів





## Крок 5. Визначення стоп-слів

ВІЗНАЧИТИ МОВУ

АНГЛІЙСЬКА

РОСІЙСЬКА

УКРАЇНСЬКА

↕

АНГЛІЙСЬКА

УКРАЇНСЬКА

РОСІЙСЬКА

↕

"The The" is popular american rock-group

×

"The" - популярна американська рок-група

☆

"The" - populyarna amerykans'ka rok-hrupa

40/5000

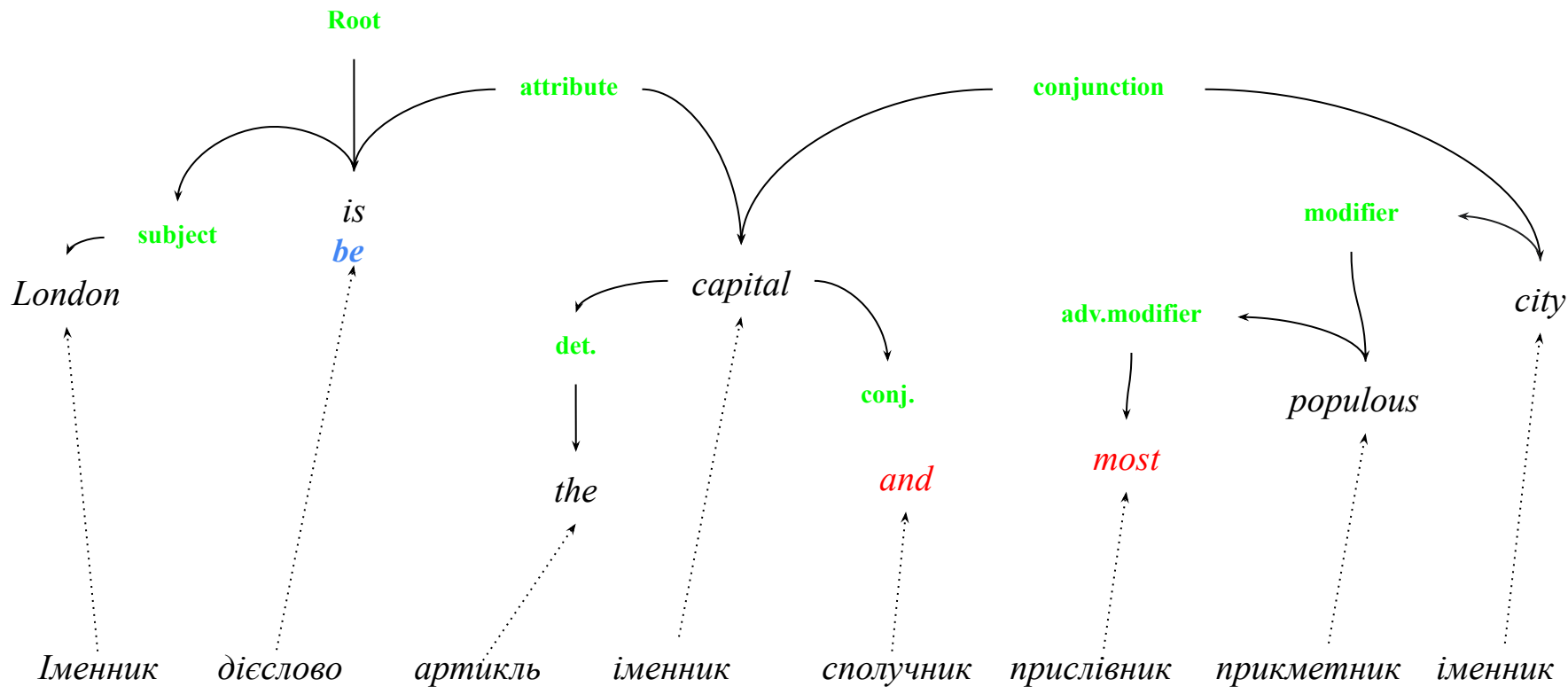
🔊

📄

✎

🔗

# Крок 6. Парсинг залежностей





## Крок 6б. Пошук груп іменників

<i>London</i>	<i>is</i>	<i>the</i>	<i>capital</i>	<i>and</i>	<i>most</i>	<i>populous ...</i>
↑	↑	↑	↑	↑	↑	↑
<i>be</i>						
Іменник	дієслово	артикль	іменник	сполучник	прислівник	прикметник

<i>London</i>	<i>is</i>	<i>the</i>	<i>capital</i>	<i>and</i>	<i>most populous city...</i>
↑	↑	↑	↑	↑	↑
<i>be</i>					
Іменник	дієслово	артикль	іменник	сполучник	іменник

*«London», «capital», «city», «England», «United Kingdom»*

*London is the capital and most populous city of England and the United Kingdom.*



*Географічна  
сутність*



*Географічна  
сутність*

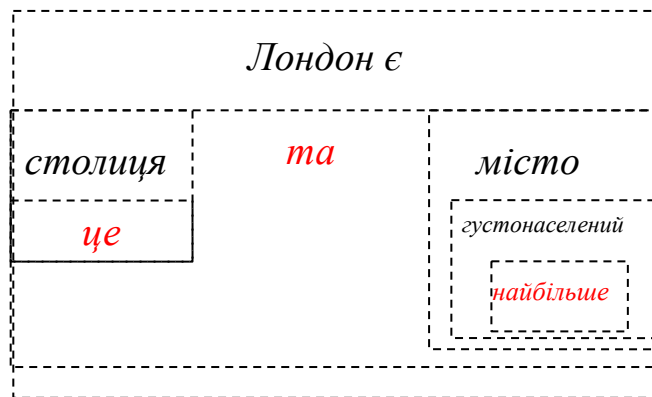
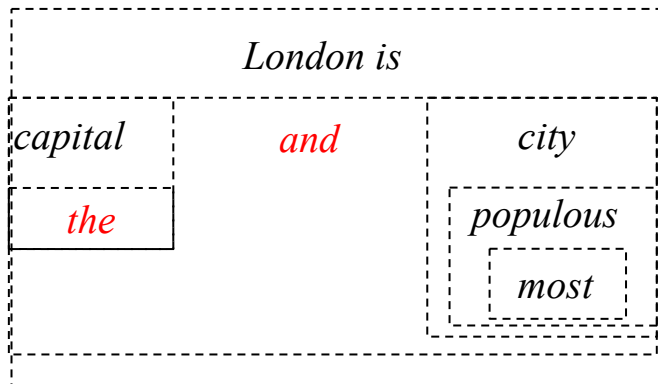


*Географічна  
сутність*



## Крок 8. Статистичний переклад компонентів схеми

*London is the capital and most populous city of England and the United Kingdom.*



*Лондон є столицею та найбільш густонаселеним містом Англії та Великобританії.*



## Крок 8. Статистичний переклад компонентів схеми

*“London is the capital and most populous city of England and the United Kingdom. Standing on the River Thames in the south east of the island of Great Britain, London has been a major settlement for two millennia. It was founded by the Romans, who named it Londinium.”*

*“Лондон є столицею та найбільш густонаселеним містом Англії та Великобританії. Стоячи на річці Темза на південному сході острова Великобританія, Лондон протягом двох тисячоліть був великим поселенням. Він був заснований римлянами, які назвали його Лондініум.”*



## Результати розробки проекту: Актуальність проблеми

*Сучасні системи машинного перекладу вже є досить досконалими, але враховуючи той факт, що вони застосовуються для перекладу самих різних текстів, то не дивлячись на складність їх алгоритмів, такі системи ще роблять помилки.*

### Результати розробки проекту

*Найпростішим вирішенням даної проблеми є персоніфікація систем перекладу для вирішення задач перекладу текстових документів, схожих за стилістикою та іншими синтаксичними характеристиками.*



*У ході розробки проекту була застосовані концепції з NLP та Deep Learning. Була розроблена нейронна мережа типу seq2seq яка складається з двох підмереж: кодера и декодера, які будуть імітувати для комп'ютера процес перекладу з однієї мови на іншу.*

## Результати розробки проекту

*Для імітації процесу виправлення перекладу, було реалізовано застосунок `muny spell translation checker`*



## Результати розробки проекту: Схема проекту



*Для реалізації проекту була використана наступна схема*

*Наступні слайди описують кожен модуль більш детально*



## Результати розробки проекту: Отримання даних перекладача

*В цій частині за допомогою застосування GoogleAPI генерується стартовий переклад з однієї мови на іншу.*

```
from deep_translator import GoogleTranslator
to_translate = 'I want to translate this text'
translated = GoogleTranslator(source='auto', target='ru').translate(to_translate)
print(to_translate)
print(translated)
```

### Результати розробки проекту

*Маємо наступний output:*

```
C:\Users\PlaZZma\AppData\Local\Programs\Python\Python38\python.exe "C:/Users/PlaZZma/Desktop/derjy v 4 course/MM/input_data.py"
I want to translate this text
Я хочу перевести этот текст
|
```



## Результати розробки проекту: Отримання даних перекладача

*В цій частині за допомогою застосування GoogleAPI генерується стартовий переклад з однієї мови на іншу.*

```
from deep_translator import GoogleTranslator
to_translate = 'I want to translate this text'
translated = GoogleTranslator(source='auto', target='ru').translate(to_translate)
print(to_translate)
print(translated)
```

### Результати розробки проекту

*Маємо наступний output:*

```
C:\Users\PlaZZma\AppData\Local\Programs\Python\Python38\python.exe "C:/Users/PlaZZma/Desktop/derjy v 4 course/MM/input_data.py"
I want to translate this text
Я хочу перевести этот текст
|
```



## Результати розробки проекту Нормалізація даних

*Наступним кроком є нормалізація даних, згідно принципів зазначених вище (Токенізація, лематизація, визначення стоп слів, усунення сторонніх назв, що заважають перекладу*

*Ось декілька прикладів валідованого перекладу*

```
[she advised him to give up drinking] => [sie riet ihm mit dem trinken aufzuhoren]
[she advised him to go to the police] => [sie riet ihm zur polizei zu gehen]
[she advised him to go to the police] => [sie riet ihm sich an die polizei zu wenden]|
[she always reminds me of her mother] => [sie erinnert mich immer an ihre mutter]
[she belongs to the democratic party] => [sie gehort der demokratischen partei an]
[she blamed him for all her problems] => [sie gab ihm die schuld fur all ihre probleme]
[she came down to breakfast at eight] => [sie kam um acht zum fruhstucken herunter]
```



## Результати розробки проекту Визначення та класифікація

*Після пройденої валідації, дані оригіналу і перекладу повинні пройти процедури семантичного аналізу. Це потрібно для того, щоб визначити, до якого стилю належить поданий текст, та які основні морфологічні зв'язки пов'язують слова та речення у ньому. Це робота для NLP*

*Ось приклад, для, згадуваного вище, тексту про Лондон для оригіналу*

- the capital and most populous city of England and the United Kingdom.  
- a major settlement for two millennia.

*І для перекладу*

- столиця та найбільш густонаселене місто Англії та Великобританії.



## Результати розробки проекту Персоніфікований переклад

*Результати, отримані на попередньому етапі експортуються в наступний модуль, створення персоніфікованого (заснованого на класифікації) перекладу.*

*У ході розробки системи персоніфікованого була застосована нейронна мережа типу **Seq2Seq** яка складається з двох підмереж **LSTM**: кодера и декодера, які будуть імітувати для комп'ютера процес перекладу з однієї мови на іншу. Таким чином ми отримуємо *recurrent neural networks*, яка персоніфіковано навчена на відповідних дата-сетах. (Зокрема, були використані паралельно перекладані датасети з визначень на сайті *Wikipedia.org* та пари розмовних речень на двох мовах*

*Приклад з датасету Вікіпедії для навчання нейронної мережі*

### Результати розробки проекту

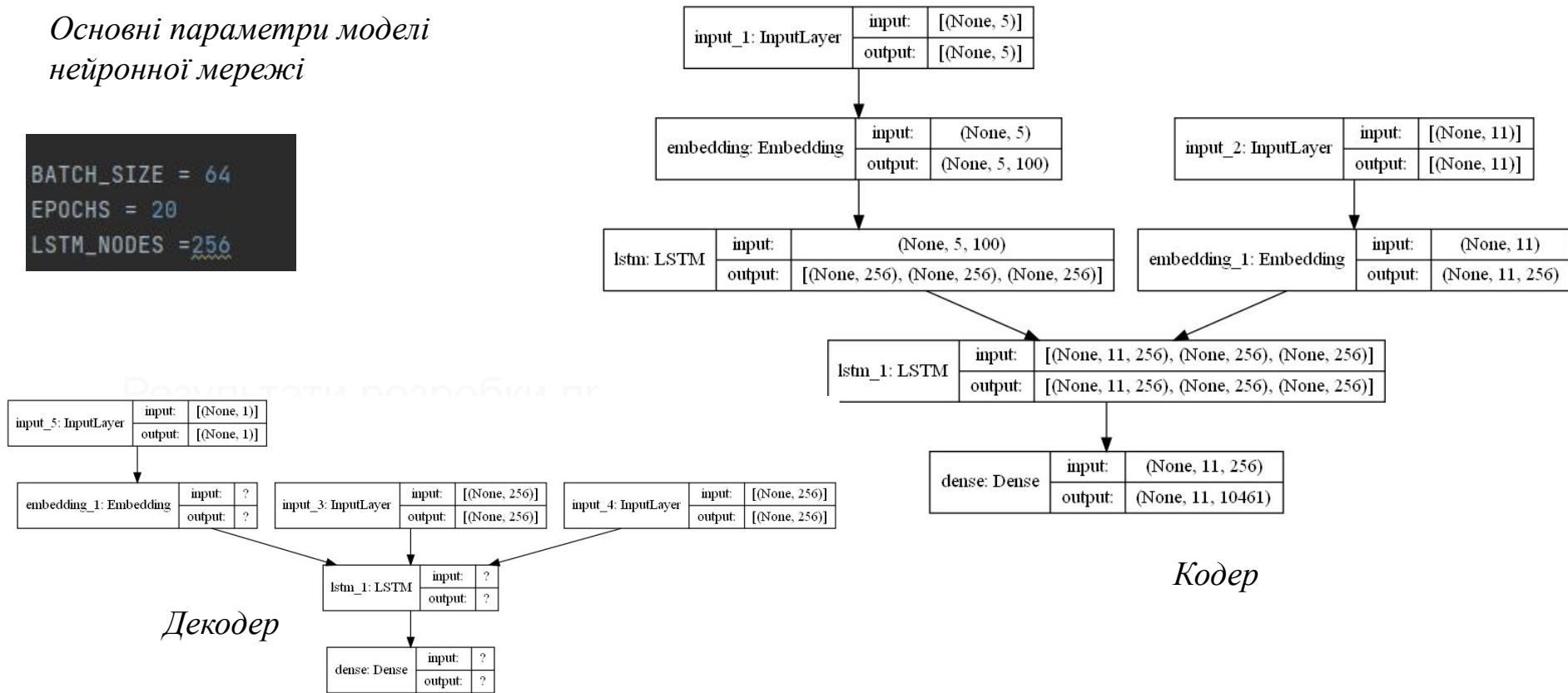
```
<tuv xml:lang="en"><seg>Stressing that, as stated in the Programme of Action of the International Conference on Population and Development, Report of the International Conference on Population and Development, Cairo, 5-13 September 1994 (United Nations publication, Sales No. E.95.XIII.18), chap. I, resolution 1, annex. family reunification of documented migrants is an important factor in international migration and that remittances by documented migrants to their countries of origin often constitute a very important source of foreign exchange and are instrumental in improving the well-being of relatives left behind,</seg></tuv>
<tuv xml:lang="ru"><seg>подчеркивая, что, согласно Программе действий Международной конференции по народонаселению и развитию Доклад Международной конференции по народонаселению и развитию, Каир, 5-13 сентября 1994 года (издание Организации Объединенных Наций, в продаже под № R.95.XIII.18), глава I, резолюция 1, приложение., воссоединение семей зарегистрированных мигрантов является важным фактором международной миграции, а денежные переводы зарегистрированных мигрантов в их страны происхождения зачастую составляют очень важный источник валютных поступлений и способствуют повышению благосостояния оставшихся в странах происхождения родственников,</seg></tuv>
>
```



# Результати розробки проекту Персоніфікований переклад

Основні параметри моделі  
нейронної мережі

```
BATCH_SIZE = 64  
EPOCHS = 20  
LSTM_NODES = 256
```







# Результати розробки проекту Персоніфікований переклад

## Процес навчання нейронної мережі

```
Epoch 1/20
282/282 [=====] - 380s 1s/step - loss: 2.5152 - accuracy: 0.7223 - val_loss: 1.7507 - val_accuracy: 0.7698
Epoch 2/20
282/282 [=====] - 149s 531ms/step - loss: 1.4493 - accuracy: 0.8049 - val_loss: 1.5908 - val_accuracy: 0.7970
Epoch 3/20
282/282 [=====] - 139s 493ms/step - loss: 1.2609 - accuracy: 0.8269 - val_loss: 1.4800 - val_accuracy: 0.8090
Epoch 4/20
282/282 [=====] - 138s 490ms/step - loss: 1.1452 - accuracy: 0.8382 - val_loss: 1.4265 - val_accuracy: 0.8143
Epoch 5/20
282/282 [=====] - 141s 501ms/step - loss: 1.0545 - accuracy: 0.8478 - val_loss: 1.4067 - val_accuracy: 0.8195
Epoch 6/20
282/282 [=====] - 142s 502ms/step - loss: 0.9857 - accuracy: 0.8552 - val_loss: 1.3788 - val_accuracy: 0.8170
Epoch 7/20
282/282 [=====] - 141s 502ms/step - loss: 0.9427 - accuracy: 0.8603 - val_loss: 1.3715 - val_accuracy: 0.8219
Epoch 8/20
282/282 [=====] - 145s 515ms/step - loss: 0.8957 - accuracy: 0.8659 - val_loss: 1.3699 - val_accuracy: 0.8238
Epoch 9/20
282/282 [=====] - 140s 496ms/step - loss: 0.8555 - accuracy: 0.8706 - val_loss: 1.3477 - val_accuracy: 0.8239
Epoch 10/20
282/282 [=====] - 142s 505ms/step - loss: 0.8218 - accuracy: 0.8755 - val_loss: 1.3367 - val_accuracy: 0.8266
Epoch 11/20
282/282 [=====] - 140s 497ms/step - loss: 0.7950 - accuracy: 0.8797 - val_loss: 1.3587 - val_accuracy: 0.8252
Epoch 12/20
282/282 [=====] - 141s 501ms/step - loss: 0.7668 - accuracy: 0.8830 - val_loss: 1.3594 - val_accuracy: 0.8233
Epoch 13/20
282/282 [=====] - 136s 483ms/step - loss: 0.7491 - accuracy: 0.8860 - val_loss: 1.3673 - val_accuracy: 0.8215
Epoch 14/20
282/282 [=====] - 138s 490ms/step - loss: 0.7288 - accuracy: 0.8891 - val_loss: 1.3851 - val_accuracy: 0.8187
Epoch 15/20
282/282 [=====] - 150s 531ms/step - loss: 0.7160 - accuracy: 0.8911 - val_loss: 1.4045 - val_accuracy: 0.8181
Epoch 16/20
282/282 [=====] - 146s 520ms/step - loss: 0.7149 - accuracy: 0.8917 - val_loss: 1.4181 - val_accuracy: 0.8156
Epoch 17/20
282/282 [=====] - 143s 508ms/step - loss: 0.6938 - accuracy: 0.8948 - val_loss: 1.4361 - val_accuracy: 0.8155
Epoch 18/20
282/282 [=====] - 141s 499ms/step - loss: 0.6893 - accuracy: 0.8955 - val_loss: 1.4412 - val_accuracy: 0.8165
Epoch 19/20
282/282 [=====] - 142s 503ms/step - loss: 0.6795 - accuracy: 0.8977 - val_loss: 1.4566 - val_accuracy: 0.8157
Epoch 20/20
282/282 [=====] - 139s 494ms/step - loss: 0.6781 - accuracy: 0.8993 - val_loss: 1.4821 - val_accuracy: 0.8134
```

Відсоток під  
час навчання:  
96.7%

Відсоток під  
час  
тестування  
76.7%



# Результати розробки проекту Порівняння і з'єднання перекладів

*У проекті порівняння текстів буде здійснюватись з урахуванням значення кожного токена в реченні, і у разі невідповідності попереджати про некоректність стартового перекладу. Це реалізовано за допомогою застосування ПЗ, що засновано на застосуванні відстані Левенштейна*

Input: "I'm(vb) furious."

Response: "я в ярости(сущ)."

Input: "I'm(adj) furious."

Response: "Я яростный(Прил)."

*Приклад правильного і неправильного перекладів*

Результати розробки проекту



В залежності від особливостей морфології, синтаксису та семантики конкретної мовної пари, а також напряму перекладу, загальний алгоритм перекладу може включати й інші етапи, а також модифікації названих етапів або порядку їх проходження, але варіації такого роду в сучасних системах, як правило, незначні. Аналіз та синтез може проводитись як пофразно, так і для всього тексту, введеного в пам'ять комп'ютеру; в останньому випадку алгоритм перекладу передбачає визначення так званих анафоричних зв'язків. Програма – перекладач - це, перш за все, інструмент, котрий дозволяє вирішити проблеми перекладу або підвищити ефективність праці перекладача тільки в тому випадку, якщо він грамотно використовується.



- [https://uk.wikipedia.org/wiki/Маши́нний\\_переклад](https://uk.wikipedia.org/wiki/Маши́нний_переклад)
- <https://medium.com/@ageitgey/natural-language-processing-is-fun-9a0bff37854e>
- <https://explosion.ai/blog/parsing-english-in-python>
- <https://explosion.ai/demos/displacy>
- <https://explosion.ai/demos/displacy-ent>
- <https://huggingface.co/coref/>
- <https://medium.com/huggingface/state-of-the-art-neural-coreference-resolution-for-chatbots-3302365dcf30>



Не Google Перекладач

Знову Google Перекладач

українська ▼

англійська ▼

Дякую за Вашу увагу!



Thank you for your attention!



24.12.2020