DOI 10.31891/2307-5732-2022-311-4-22-25
УДК 637.5.02

БАСИСТЮК Олег
Національний університет «Львівська політехніка»
https://orcid.org/0000-0003-0064-6584
e-mail: oleh.a.basystiuk@lpnu.com
МЕЛЬНИКОВА Наталія
Національний університет «Львівська політехніка»
https://orcid.org/0000-0002-2114-3436
e-mail: nataliia.i.melnykova@lpnu.ua

МУЛЬТИМОДАЛЬНЕ РОЗПІЗНАВАННЯ МОВЛЕННЯ НА ОСНОВІ ЗВУКОВИХ І ТЕКСТОВИХ ДАНИХ

Глибоке навчання повністю змінило підхід до машинного перекладу. Дослідники в галузі глибокого навчання створили прості рішення на основі машинного навчання, які перевершують найкращі експертні системи. У цій роботі розглянуто основні особливості машинного перекладу на основі рекурентних нейронних мереж. У статті також висвітлено переваги систем на основі RNN, що використовують модель послідовності до послідовності, порівняно зі статистичними системами трансляції. Дві системи машинного перекладу, засновані на моделі послідовності до послідовності, були створені з використанням бібліотек машинного навчання Keras і PyTorch. На основі отриманих результатів проведено аналіз бібліотек та порівняння їх продуктивності.

Ключові слова: машинний переклад, глибоке навчання, рекурентні нейронні мережі, продуктивність, keras, pytorch, sequence-to-sequence.

BASYSTIUK Oleh, MELNYKOVA Nataliia
Lviv Polytechnic National University

MULTIMODAL SPEECH RECOGNITION BASED ON AUDIO AND TEXT DATA

Systems of machine translation of texts from one language to another simulate the work of a human translator. Their performance depends on the ability to understand the grammar rules of the language. In translation, the basic units are not individual words, but word combinations or phraseological units that express different concepts. Only by using them, more complex ideas can be expressed through the translated text.

The main feature of machine translation is different length for input and output. The ability to work with different lengths of input and output provides us with the approach of recurrent neural networks.

A recurrent neural network (RNN) is a class of artificial neural network that has connections between nodes. In this case, a connection refers to a connection from a more distant node to a less distant node. The presence of connections allows the RNN to remember and reproduce the entire sequence of reactions to one stimulus. From the point of view of programming, such networks are analogous to cyclic execution, and from the point of view of the system, such networks are equivalent to a state machine. RNNs are commonly used to process word sequences in natural language processing. Usually, a hidden Markov model (HMM) and an N-program language model are used to process a sequence of words.

Deep learning has completely changed the approach to machine translation. Researchers in the deep learning field has created simple solutions based on machine learning that outperform the best expert systems. In this paper was reviewed the main features of machine translation based on recurrent neural networks. The advantages of systems based on RNN using the sequence-to-sequence model against statistical translation systems are also highlighted in the article. Two machine translation systems based on the sequence-to-sequence model were constructed using Keras and PyTorch machine learning libraries. Based on the obtained results, libraries analysis was done, and their performance comparison.

Keywords: machine translation, deep learning, recurrent neural networks, performance, keras, pytorch, sequence-to-sequence.

Постановка проблеми у загальному вигляді та її зв’язок із важливими науковими чи практичними завданнями

Системи машинного перекладу текстів з однієї мови на іншу моделюють роботу людини-перекладача. Їхня продуктивність залежить від здатності розуміти правила граматики мови. У перекладі основними одиницями є не окремі слова, а словосполучення або фразеологізми, що виражають різні поняття. Тільки використовуючи їх, через текст перекладу можна висловити більш складні ідеї [1].

Головною особливістю машинного перекладу є різна довжина для введення та виведення. Можливість працювати з різною довжиною входу та виходу надає нам підхід рекурентних нейронних мереж [1–5].

Рекурентна нейронна мережа (RNN) – це клас штучної нейронної мережі, яка має зв'язки між вузлами. У цьому випадку з'єднання відноситься до з'єднання від більш віддаленого вузла до менш віддаленого вузла. Наявність зв'язків дозволяє РНН запам'ятовувати і відтворювати всю послідовність реакцій на один стимул. З точки зору програмування в таких мережах є аналог циклічного виконання, а з точки зору системи - такі мережі еквівалентні кінцевому автомату. RNN зазвичай використовуються для обробки послідовності слів у обробці природної мови [4–7]. Зазвичай для обробки послідовності слів використовують приховану модель Маркова (HMM) і модель мови N-програм.

Прихована модель Маркова (HMM) - статистична модель, яка імітує роботу процесу, схожого на процес Маркова з невідомими параметрами, і завдання полягає в тому, щоб вгадати невідомі параметри на основі спостережуваних. Отримані параметри можна використовувати в подальший аналіз. У нормальній моделі Маркова стан відомий спостерігачеві, тому ймовірність переходів є одним з параметрів. У NMM

можна спостерігати лише змінні, на які впливає цей стан. Кожен стан має ймовірнісний розподіл серед усіх можливих вихідних значень. Тому послідовність слів, згенерована NMM, дає інформацію про послідовність станів. NMM можна вважати найпростішою байєсівською мережею.

Байєсова мережа – графічна модель у вигляді спрямованого ациклічного графа, кожна вершина якого відповідає випадковій величині, а дуги графа кодують відносини умовної незалежності між цими змінними. Вершини можуть представляти змінні будь-якого типу, бути зваженими параметрами, прихованими змінними або гіпотезами. Існують ефективні методи, які використовуються для розрахунку та дослідження байєсівських мереж. Для проведення ймовірнісного виведення в байєсівських мережах використовуються як точні, так і наближені алгоритми [8, 9].

Виклад основного матеріалу

Основна ідея RNN полягає у використанні рекурсії для формування вектора фіксованої розмірності з вхідної послідовності символів. Припустимо, що на кроці t вектор h_{t-1} h_{t-1}, який є історією всіх попередніх слів. RNN обчислить новий вектор h_t h_t (його внутрішній стан), який поєднує всі попередні слова $(x_1, x_2, \ldots, x_{t-1})$ $(x_1, x_2, \ldots, x_{t-1})$ і новий символ x_t x_t за допомогою:

$$h_t = \varphi_\theta(x_t, h_{t-1}), \quad (1)$$

У цьому рівнянні присутні наступні параметри: φ_θ φ_θ – функція, параметризована θ, яка отримує нове вхідне слово x_t x_t та історію слів h_{t-1} h_{t-1} до $(t-1)$ -N слова. По-перше, можна вважати, що h_0 h_0 - нульовий вектор. Рекурентна функція активації φ зазвичай реалізується як афінне перетворення, за яким слідує нелінійна функція:

$$h_t = \tanh(Wx_t + Uh_{t-1} + b), \quad (2)$$

У цьому рівнянні присутні такі параметри: вхідна вагова матриця W, рекурентна вагова матриця U та вектор зміщення b. Зверніть увагу, що це не єдиний варіант. Існує широкий простір для розробки нових повторюваних функцій активації.

Детальніше про роботу методу перекладу тексту на основі нейронних мереж. Ідея цього алгоритму, по суті, проста і складається з наступних кроків:

1. Кодування вхідного тексту мови A в набір даних;
2. Декодування набору даних мовою B.

Розглянемо алгоритм кодування тексту на прикладі пропозиції: «Приклад нейронної мережі»:

Виконавши таку просту операцію, ми отримуємо закодований текст, який виглядає як набір числових даних. На початковому етапі навчання ці числа є випадковими і генеруються алгоритмом також випадково. Наступне проходження тексту, який уже закодовано, RNN буде оцінюватися до того самого числового набору даних. Алгоритм декодування тексту працює як кодування, тільки навпаки – на вхід надходить набір числових даних і виводиться ймовірний текст, який відповідає цим даним [5–7].

Після того як ми зрозуміли суть кодування і декодування тексту, переходимо до самої суті нашого завдання - машинного перекладу і його загального алгоритму. Для цього нам просто потрібно поєднати ці дві RNN - для кодування та декодування - і отримати наступний результат:

Таким чином, ми отримуємо загальний спосіб перетворення послідовності українських слів на еквівалентну послідовність англійських слів, це так званий, послідовний метод мовного перекладу Sequence-to-Sequence. Основними перевагами методу є [3–6]:

- цей підхід обмежений обсягом набору навчальних даних і обчислювальною потужністю, яку ви можете виділити для перекладу. Дослідники машинного навчання винайшли цей метод лише кілька років тому, але такі системи вже працюють краще, ніж статистичні системи машинного перекладу, які розвивалися протягом останніх 20 років;
- система не залежить від знання будь-яких правил мови. Алгоритм сам визначає ці правила і постійно адаптується.

Експерименти

Давайте проведемо експерименти на основі двох бібліотек машинного навчання, написаних на Python — PyTorch і Keras. Основою алгоритму є метод послідовного навчання. Отже, ми маємо навчити нашу майбутню модель, яка перекладатиме наш текст. Для цього давайте створимо простий набір даних для навчання:

Таблиця 1

Набір даних для навчання RNN

English	Українська
An example of a neural network .	Приклад нейронної мережі .
Hello !	Привіт !
How are you ?	Як справи ?
I'm fine !	Все добре !

Thank you !	Дякую !
Fine .	Добре .
Good .	Добре .
What are you waiting for ?	Чого ти чекаєш ?
Recurrent neural network .	Рекурентна нейронна мережа .
Success !	Успіх !

Як бачите, навчальний набір даних складається з 10 фраз, ми проведемо ці дані через наші моделі, після чого оцінимо швидкість і точність моделей [10].

В результаті цих двох алгоритмів ми отримуємо такі дані:

Таблиця 2

Порівняння результатів Keras та PyTorch

Назва бібліотеки	**Час навчання**	**Цикли навчання**	**Коефіцієнти втрат**	**Точність перекладу**
Keras	4150millis	400	0.0027	100%
PyTorch	5800millis	650	0.0021	100%

Розглянемо ці дані детальніше:

- Час навчання. Значення, яке показує час навчання моделі. Головним чином залежить від середовища, де запускався сценарій. Середовище означає поточні характеристики ПК; обчислювальна потужність процесора та її завантаження іншими процесами.
- Тренувальні петлі. Значення, яке показує цикли навчання моделі. Ми самі даємо.
- Коефіцієнт втрат. Значення, яке показує точність навченої моделі. Це показник того, наскільки хороша ваша модель.
- Точність перекладу. Значення, яке показує у відсотках термін значення речень правильного перекладу.

Таким чином, збірка моделі на основі бібліотеки Keras виявилася ефективнішою, ніж модель PyTorch, порівняння базується на часу навчання, циклах навчання та частоті помилок. Через невеликий набір навчальних даних обидва алгоритми демонструють максимальну точність перекладу. У разі збільшення обсягу навчального набору даних, моделі забезпечать зовсім інший коефіцієнт втрат і точність перекладу, час навчання і коефіцієнт втрат збільшиться, а точність зменшиться.

Результати

У статті описано проектування та створення двох систем перекладу тексту з однієї мови на іншу на основі бібліотек Keras та PyTorch ML. Результат роботи можна побачити на рисунку. 3, де алгоритм правильно розпізнавав (переклав) речення в кожному випадку.

```
Input text: Приклад нейронної мережі .
Decoded text: Example of the neural network .

-
Input text: Привіт !
Decoded text: Hello !

-
Input text: Як справи ?
Decoded text: How are you ?

-
Input text: Все добре !
Decoded text: Everything is good !

-
Input text: Дякую !
Decoded text: Thank you !

-
Input text: Добре .
Decoded text: Fine .

-
Input text: Класно .
Decoded text: Cool .

-
Input text: Чого ти чекаєш ?
Decoded text: What are you waiting for ?

-
Input text: Рекурентна нейронна мережа .
Decoded text: Recurrent neural network .

-
Input text: Успіх !
Decoded text: Success !
```

Рис. 3. Результати перекладу

RNN, як й інші класи нейронних мереж, розвиваються настільки швидко, що стає все важче відслідковувати нові, більш цікаві та складніші моделі для вирішення більш складних і заплутаних завдань. Ці послідовні методи навчання нейронних мереж можуть бути використані в інших сферах, не тільки в машинному перекладі. Простими прикладами є моделі, які можуть створювати словесні описи зображення, розпізнавати голос і підтримувати розмову. На нашу думку, розвиток RNN призведе до появи розумних помічників, здатних розпізнавати голос власника і правильно сприймати завдання. На даний момент RNN є найбільш часто використовуваними в машинному перекладі, і ми думаємо, що ця сфера також буде модернізована найближчим часом.

Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі

Згідно з результатами експерименту, модель на основі бібліотеки Keras більш ефективна для поточного набору навчальних даних. Зауважте, що результати дослідження можна вважати релевантними лише для невеликих наборів даних, а якість перекладу та час навчання будуть змінені після збільшення обсягу набору навчальних даних. Наступний етап цього дослідження може складатися з навчання моделі у великих обсягах даних з аналізом і порівнянням якості та швидкості її роботи.

References

1. Yu D., Deng L. Automatic Speech Recognition: A Deep Learning Approach. Springer-Verlag Longon, 2015. DOI: 10.1007/978-1-4471-5779-3.

2. Dey N. Intelligent Speech Signal Processing Academic Press, 2019. DOI: 10.1016/C2018-0-03271-5.

3. Shakhovska N., Basystiuk O., Shakhovska K. Development of the speech-to-text chatbot interface based on Google API. In: CEUR Workshop Proceedings, 2019, vol. 2386, pp. 212–221.

4. Melnykova N. Semantic search personalized data as special method of processing medical information. Advances in Intelligent Systems and Computing, 2017: 315-325.

5. Basystiuk O., Shakhovska N., Bilynska V., Syvokon O., Shamuratov O., Kuchkovskiy V. The Developing of the System for Autimatic Audio to Text Conversion. IT&AS'2021: Symposium on Information Technologies & Applied Sciences, March 5–6, 2021, Bratislava, Slovak Republic.

6. Buss E., Leibold L. J., Porter H. L., Grose J. H. Speech recognition in one- and two-talker maskers in school-age children and adults: Development of perceptual masking and glimpsing. The Journal of the Acoustical Society of America, 2017. DOI: 10.1121/1.4979936.

7. Nataliya Boyko, Lesya Mochurad, Uliana Parpan, Oleh Basystiuk. Usage of Machine-based Translation Methods for Analyzing Open Data in Legal Cases. In: Proc. of the Intl Workshop on Cyber Hygiene (CybHyg-2019) co-located with 1st International Conference on Cyber Hygiene and Conflict Management in Global Information Networks (CyberConf, 2019), Kyiv, Ukraine, November 30, 2019, pp. 328–338. CEUR-WS.org, online CEUR-WS.org/Vol-2654/paper26.pdf.

8. Melnykova N., Shakhovska N., Gregušml M., & Melnykov V. (2019). Using big data for formalization the patient's personalized data. Paper presented at the Procedia Computer Science, 155 624-629.

9. Zoryana Rybchak, Oleh Basystiuk. (2017). Analysis of methods and means of text mining. ECONTECHMOD. AN INTERNATIONAL QUARTERLY JOURNAL, 6(2), 73-78.

10. GitHub Repository "Speech recognition algorithms". https://github.com/obasys/speech-recognition-algorithms. (accessed Aug. 15, 2022)

Надійшла/Paper received : 10.09.2022 р. Надрукована/Printed :01.11.2022 р.