# ЗАСТОСУВАННЯ РЕКУРЕНТНИХ НЕЙРОННИХ МЕРЕЖ ДЛЯ ВИКОНАННЯ МАШИННОГО РЕРАЙТУ

**Шеремет О. І., Запорожець В. С.**

Машинное обучение представляет собой актуальную сферу научного знания, которая интенсивно развивается и имеет очень большие перспективы. В более узком смысле под машинным обучением понимают класс методов искусственного интеллекта, характерной чертой которых является не прямое решение поставленной задачи, а применение для этого специально обученной математической модели. Такая модель учится за счет решения большого количества подобных задач в нужной области. Одной из самых перспективных современных технологий машинного обучения является применение глубинных нейронных сетей, в основе которых лежит применение глубокого обучения. Глубокое обучение – это набор алгоритмов машинного обучения, которые позволяют создавать модели с высоким уровнем абстракции в выходных данных, используя архитектуры нейронных сетей, содержащих нелинейные преобразования сигнала. В статье выполняется демонстрация возможностей, которые предоставляет применение рекуррентных нейронных сетей для решения одной из самых сложных задач, которая стоит перед разработчиками веб-контента – рерайта текстовой информации. Смысл применения машинного обучения для обработки естественных языков заключается в том, что глубинные нейронные сети выполняют работу, на осуществление которой в течение приемлемого промежутка времени нужно было бы применять десятки или даже сотни команд профессиональных лингвистов. Традиционные нейронные сети не имеют возможности принимать текущие решения на основе своих предыдущих суждений. Большое количество задач, решаемых при машинной обработке естественных языков, требует поэтапного анализа данных с учетом предыдущих результатов. Нейронная сеть должна «читать» предложение слово за словом, «осмысливая» его значение исходя из контекста. Рекуррентные нейронные сети содержат обратные связи и позволяют кратковременно хранить информацию, благодаря чему они как нельзя лучше подходят для обработки последовательностей слов и символов, которыми являются предложения естественного языка. Техническую реализацию рерайта предложений предложено осуществить с помощью библиотеки seq2seq, которая входит в состав TensorFlow – программного обеспечения, разработанного компанией Google для решения задач построения и обучения нейронных сетей.

**Ключевые слова:** рекуррентная нейронная сеть, рерайт, обратная связь, кратковременная память.

Машинне навчання являє собою актуальну сферу наукового знання, яка інтенсивно розвивається та має дуже значні перспективи. В більш вузькому розумінні під машинним навчанням розуміють клас методів штучного інтелекту, характерною рисою яких є не пряме рішення поставленої задачі, а застосування для цього спеціально навченої математичної моделі. Така модель навчається за рахунок розв'язання великої кількості подібних задач у потрібній області. Одним з найперспективніших сучасних технологій машинного навчання є застосування глибинних нейронних мереж, в основі яких лежить застосування глибокого навчання. Глибоке навчання – це набір алгоритмів машинного навчання, які дозволяють створювати моделі з високим рівнем абстракції у вихідних даних, використовуючи архітектури нейронних мереж, що містять нелінійні перетворення сигналу. В статті виконується демонстрація можливостей, які надає застосування рекурентних нейронних мереж для розв'язання

однієї з найскладніших задач, що постає перед розробниками веб-контенту – рерайту текстової інформації. Сенс застосування машинного навчання для обробки природних мов полягає в тому, що глибинні нейронні мережі виконують роботу, на здійснення котрої впродовж прийнятного проміжку часу потрібно було б застосовувати десятки чи навіть сотні команд професійних лінгвістів. Традиційні нейронні мережі не мають можливості приймати поточні рішення на основі своїх попередніх суджень. Велика кількість задач, що вирішуються при машинній обробці природніх мов, потребує поетапного аналізу даних з врахуванням попередніх результатів. Нейронна мережа повинна «читати» речення слово за словом, «осмислюючи» його значення виходячи з контексту. Рекурентні нейронні мережі містять зворотні зв'язки і дозволяють короткочасно зберігати інформацію, завдяки чому вони як найкраще підходять для обробки послідовностей слів та символів, якими є речення природної мови. Технічну реалізацію рерайту речень запропоновано здійснити за допомогою бібліотеки seq2seq, котра входить до складу TensorFlow – програмного забезпечення, розробленого компанією Google для вирішення задач побудови і тренування нейронних мереж.

**Ключові слова:** рекурентна нейронна мережа, рерайт, зворотний зв'язок, короткочасна пам'ять.

Machine learning is an actual area of scientific knowledge, which is intensively developing and has very significant prospects. In a narrower sense, machine learning is understood as a class of methods of artificial intelligence, the characteristic feature of which is not the direct solution of the problem, but the application of a specially trained mathematical model for this. This model learns by solving a large number of similar tasks in the desired area. One of the most promising modern technologies of machine learning is using of deep neural networks, which is based on the application of deep training. Deep learning is a set of machine learning algorithms that allow creating models with a high level of abstraction in the source data, using the architecture of neural networks that contain nonlinear signal transformations. The article demonstrates the possibilities that the application of recurrent neural networks provides for solving one of the most difficult tasks facing web content developers – the rewriting of textual information. The meaning of the use of machine learning for the processing of natural languages is that the deep neural networks perform work for which, within an acceptable period of time, dozens or even hundreds of teams of professional linguists would have to be used. Traditional neural networks do not have the ability to make current decisions based on their previous judgments. A large number of tasks solved by machine processing of natural languages requires a step-by-step analysis of data taking into account previous results. The neural network should "read" the sentence word by word, "comprehending" its meaning from the context. Recurrent neural networks contain feedbacks and allow you to store information for a short time, so that they are perfectly suited for processing sequences of words and symbols, which are natural language sentences. The technical implementation of the rewriting is proposed to be implemented using the seq2seq library, which is part of TensorFlow, software developed by Google to solve the problems of building and training neural networks.

**Keywords:** recurrent neural network, rewriting, feedback, short-term memory.

Шеремет А. И. д-р техн. наук, зав. каф. ЭСА ДГМА
sheremet-a@mail.ru

Запорожец В. С. магистр кафедры ЭСА ДГМА

ДГМА – Донбасская государственная машиностроительная академия, г. Краматорск.

УДК 004.8

**Шеремет О. І., Запорожець В. С.**

# ЗАСТОСУВАННЯ РЕКУРЕНТНИХ НЕЙРОННИХ МЕРЕЖ ДЛЯ ВИКОНАННЯ МАШИННОГО РЕРАЙТУ

Машинне навчання являє собою актуальну сферу наукового знання, яка інтенсивно розвивається та має дуже значні перспективи [1]. В більш вузькому розумінні під машинним навчанням (англ. Machine Learning) розуміють клас методів штучного інтелекту, характерною рисою яких є не пряме рішення поставленої задачі, а застосування для цього спеціально навченої математичної моделі [2]. Така модель навчається за рахунок розв'язання великої кількості подібних задач у потрібній області.

Одним з найперспективніших сучасних технологій машинного навчання є застосування глибинних нейронних мереж, в основі яких лежить застосування глибокого навчання (англ. Deep Learning). Глибоке навчання – це набір алгоритмів машинного навчання, які дозволяють створювати моделі з високим рівнем абстракції у вихідних даних, використовуючи архітектури нейронних мереж, що містять нелінійні перетворення сигналу [3].

Характерною особливістю алгоритмів глибоко навчання є проходження вхідної інформації через набагато більшу кількість шарів, ніж при традиційному (поверхневому) навчанні. Для нейронних мереж з рекурентною архітектурою шлях, яких проходить інформаційний сигнал від входу до виходу, є теоретично необмеженим (практично він може обмежуватись можливостями застосованого програмного забезпечення) [4].

Завдяки використанню глибинних нейронних мереж розв'язуються задачі комп'ютерного бачення, обробки природних мов, розпізнавання музики, прогнозування перебігу подій, інтелектуальної фільтрації даних, побудування чат-ботів та багато інших. Велика кількість звичних та зручних сервісів компанії Google була б абсолютно неможливою без застосування глибинних нейронних мереж.

Метою даної статті є демонстрація можливостей, які надає застосування рекурентних нейронних мереж для розв'язання однієї з найскладніших задач, що постає перед розробниками веб-контенту – рерайту текстової інформації.

Під обробкою природних мов (англ. Natural Language Processing, NLP) розуміють створення систем з ознаками штучного інтелекту, які певним чином обробляють мовну інформацію з метою виконання певних задач [5]. До таких задач належать:

– чат-боти або формування відповідей на запитання користувача;
– визначення характеру емоційного забарвлення висловлювань;
– машинний переклад з однієї мови на іншу;
– розпізнавання мов;
– перевірка правопису;
– визначення частин мови в реченні і їх анотування;
– рерайт текстової інформації для створення веб-контенту [6].

Сенс застосування машинного навчання для обробки природних мов полягає в тому, що глибинні нейронні мережі виконують роботу, на здійснення котрої впродовж прийнятного проміжку часу потрібно було б застосовувати десятки чи навіть сотні команд професійних лінгвістів.

Традиційні нейронні мережі не мають можливості приймати поточні рішення на основі своїх попередніх суджень. Велика кількість задач, що вирішуються при машинній обробці природніх мов потребує поетапного аналізу даних з врахуванням попередніх результатів. Нейронна мережа повинна «читати» речення слово за словом, «осмислюючи» його значення виходячи з контексту.

Рекурентні нейронні мережі (англ. Recurrent Neural Networks, RNN) – це мережі, що містять зворотні зв'язки і дозволяють зберігати інформацію (рис. 1) [7].

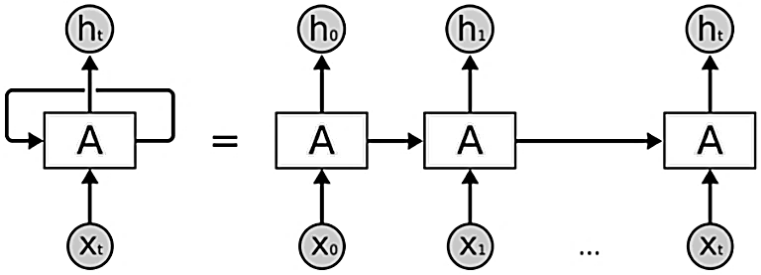


Рис. 1. Рекурентна мережа у розгортці

На схемі вище фрагмент нейронної мережі A приймає вхідний значення $x_t$ і повертає значення $h_t$. Наявність зворотного зв'язку дозволяє передавати інформацію від одного кроку навчання мережі до іншого.

Одним з різновидів RNN є LSTM-мережі [8]. LSTM (англ. Long Short Term Memory) – це RNN здатні до навчання довготривалими залежностями. LSTM-мережі складаються з повторюваних елементів. Кожен такий елемент містить чотири шари і відрізняється тим, що має комірку довгої короткочасної пам'яті (рис. 2) [7].

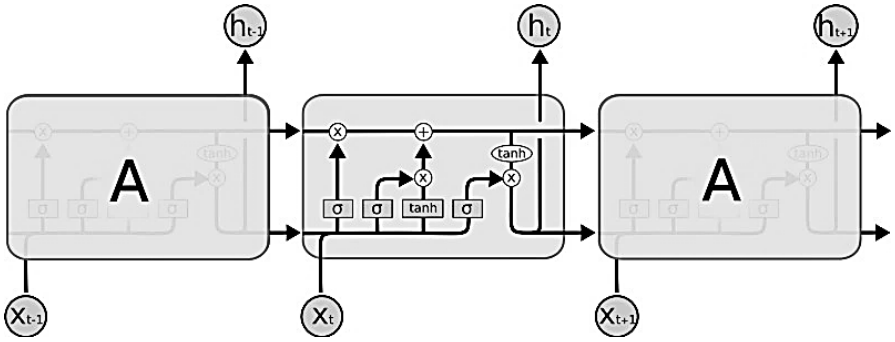


Рис. 2. Повторюваний модуль в LSTM, що містить чотири взаємодіючих шари

Ідея, покладена в основу концепції Sequence-to-sequence полягає в тому, щоб використовувати одну LSTM-мережу для читання вхідної послідовності крок за кроком, щоб отримати векторний простір даних фіксованої розмірності, а потім використовувати іншу LSTM-мережу для формування вихідної послідовності з цього вектора (рис. 3). Можливість LSTM-мереж успішно вивчати дані з довготривалими залежностями робить їх природним вибором для розв'язання задач, у котрих як вхідна, так і вихідна інформація представляються у вигляді послідовностей деяких елементів (наприклад, літер, слів, речень) [9].

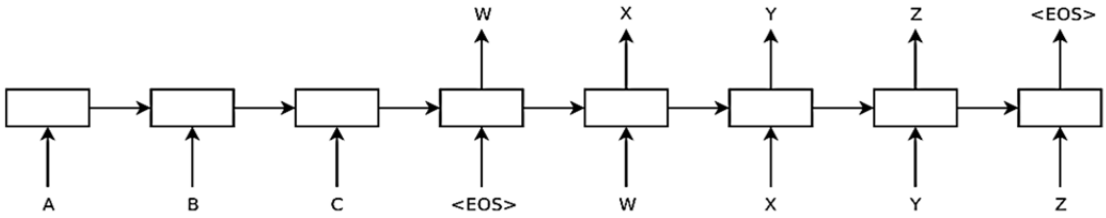


Рис. 3. Модель, що читає вхідну послідовність «ABC» та генерує вихідну "WXYZ"

Концепція Sequence-to-sequence, реалізована за допомогою LSTM-мереж є основою для розв'язання задач машинного переводу, перевірки помилок у тексті та рерайту.

Технічну реалізацію рерайту речень можна здійснити за допомогою бібліотеки seq2seq, котра входить до складу TensorFlow – програмного забезпечення, розробленого компанією Google для вирішення здач побудови і тренування нейронних мереж.

Процес розв'язання будь-якої задачі з області обробки природних мов можна розбити на три етапи:

1. Підготовка даних для навчання.
2. Навчання нейронної мережі (створення моделі).
3. Тестування одержаної моделі.

Розглянемо ці етапи більш докладно. Для навчання будь-якої нейронної мережі потрібно мати суттєвий об'єм даних (екземплярів) для навчання. Як правило для навчання мереж, що розв'язують задачі з області обробки природних мов використовують корпуси – філологічно-компетентні масиви мовних даних.

Для розв'язання поставленої задачі скористаємось корпусом Гутенберга [10], який містить близько 54000 безкоштовних електронних книг різних авторів. Послідовність підготовки потрібних даних:

1. Виймання речень з «сирого» тексту, що міститься у корпусі Гутенберга. Для цього було створено спеціальний парсер на основі Beautiful Soup [11].
2. Унікалізація одержаних речень за допомогою теорії множин.
3. Випадкове перемішування речень та зберігання їх в окремому текстовому файлі.
4. Отримання тегів з одержаних речень. При цьому на ту позицію, яку займає слово в реченні, ставиться POS-тег [12] (табл. 1) з символом «_» перед ним, який відповідає частині мови, до якої відноситься слово (табл. 2). Замість розділових знаків ставиться тег _PUNCT.

Таблиця 1

Перелік стандартних POS-тегів

| POS-тег | Значення |
|---|---|
| ADJ | adjective |
| ADP | adposition |
| ADV | adverb |
| AUX | auxiliary |
| CCONJ | coordinating conjunction |
| DET | determiner |
| INTJ | interjection |
| NOUN | noun |
| NUM | numeral |
| PART | particle |
| PRON | pronoun |
| PROPN | proper noun |
| PUNCT | punctuation |
| SCONJ | subordinating conjunction |
| SYM | symbol |
| VERB | verb |
| X | other |

5. Фільтрація прикметників у шаблонах за наступним принципом: декілька тегів _ADJ, що йдуть один за одним у шаблоні, замінюються одним тегом _ADJ. Одержані шаблони, що мають абсолютну частоту повторень від 10 і більше зберігаються в окремому текстовому файлі. Bash-скрипт для виконання представлених п'яти пунктів.

Таблиця 2

Частини мови та відповідні їм теги шаблонів

| Частина мови | Тег |
|---|---|
| Іменник (Noun) | _NOUN |
| Прикметник (Adjective) | _ADJ |
| Дієслово (Verb) | _VERB |
| Прислівник (Adverb) | _ADV |
| Займенник (Pronoun) | _PRON |

6. Фільтрація шаблонів за довжиною (кількістю тегів). Найбільш вживані шаблони мають довжину від 5 до 16 тегів.

7. Відбір речень, що відповідають відфільтрованим шаблонам та збереження їх у окремому текстовому файлі. Оскільки при цьому потрібно обробляти великі об'єми даних, то ця задача вирішується у декілька потоків.

8. Одержаний у пункті 7 файл уособлює бажані вихідні послідовності нейронної мережі (коректно побудовані англомовні речення). Вхідна послідовність готується у відповідності до вибраної стратегії на основі речень та відповідних їм шаблонів.

При цьому власні імена замінюються тегами _NAME1, _NAME2, _NAME3, … як у вхідній, так і у вихідній послідовності.

Фразеологічні дієслова як у вхідній, так і у вихідній послідовностях розглядаються як одне ціле, частини котрого поєднуються символом «_». Наприклад, дієслову give_up відповідатиме один тег – _VERB.

Найбільш ефективною виявилася змішана стратегія з наявністю ймовірної складової:

– синонімізація прикметників (тег _ADJ) у вхідній послідовності (за допомогою WordNet), інші слова (не прикметники) замінюються їх лема-формами з вірогідністю 0,5 (в іншому випадку залишається відповідний тег);

– синонімізація прикметників (тег _ADJ) у вхідній послідовності (за допомогою WordNet), інші слова (не прикметники) замінюються їх лема-формами з вірогідністю 0,4 (в іншому випадку залишається відповідний тег);

– прикметники додаються до вхідної послідовності у вигляді лема-форм (без синонімізації), а всі інші слова (не прикметники) замінюються їх лема-формами з вірогідністю 0,5 (в іншому випадку залишається відповідний тег);

– слова, що відносяться до найбільш "впливових" частин мови додаються до вхідної послідовності у вигляді їх лема-форм (_NOUN, _ADJ, _VERB, _PRON), інші ж слова або розділові знаки у вхідній послідовності представляються у вигляді відповідних їм тегів;

– лема-форми та теги у вхідній послідовності представлені з однаковою вірогідністю – 0,5;

– теги власних імен (_NAME1, _NAME2, _NAME3, …) додаються із стовідсотковою вірогідністю, а інші слова замінюються своїми лема-формами з вірогідністю 0,5 (в іншому випадку залишається відповідний тег).

В результаті розглянутої стратегії і для кожного речення формується шість різних варіації вхідної послідовності при одній вихідній. Такий підхід надає користувачеві широких можливостей щодо завдання вхідної послідовності.

9. Виконується розподілення даних: 80 % одержаних даних, як для вхідної, так і для вихідної послідовностей відводяться на тренування (навчання) моделі, а 20 % – на її тестування, котре виконується безпосередньо у процесі навчання.

Після завершення етапу підготовки даних можна безпосередньо займатись навчанням моделі. Модель будується на основі архітектури нейронної мережі, що використовується при машинному перекладі. Як компроміс для побудування окремих моделей для кожної з можливих пар довжин вхідної та вихідної послідовностей при цьому застосовуються «ковші» (англ. buckets) [13].

Оскільки вхідна та вихідна послідовності мають однакову довжину, то перше та друге значення у ковші співпадають. Наприклад, вхідна послідовність має довжину 5. У відповідності до розробленої стратегії вихідна послідовність матиме також довжину 5, а в цілому така пара буде віднесена до ковша (7, 7) після доповнення двома pad-символами.

Як вхідна, так і вихідна послідовності кодуються кожна на власному словнику з найбільш вживаних 100000 слів. Застосовується просте кодування, при якому словник розглядається як список і код слова – це його позиція у списку. Для невідомих слів як у вхідному, так і у вихідному словнику є спеціальний тег – _UNK.

В процесі навчання відображається поточне значення розгубленості (англ. perplexity) мережі [14]. Навчання можна зупиняти коли воно досягає значення близького до 1 (як правило, на практиці perplexity навченої моделі становить значення від 1,05 до 1,3 – в залежності від якості даних). На етапі тестування модель завантажується у оперативну пам'ять а на вхід подається послідовність, подібна до тих, що використовувались під час навчання моделі.

## ВИСНОВКИ

Таким чином, рекурентні нейронні мережі можуть успішно застосовуватись для здійснення машинного рерайту англомовних речень. Найбільш перспективними нейронними мережами у цьому сенсі постають LSTM-мережі, які є особливим різновидом архітектури рекурентних нейронних мереж, що здатна вивчати тривалим залежності між вхідними та вихідними послідовностями даних. Структура LSTM також нагадує ланцюг, кожна частина якого містять чотири шари, які взаємодіють особливим чином, запам'ятовуючи корисну інформацію. Завдяки такому підходу нейронна мережа здатна послідовно «читати» текст та аналізувати його у контексті раніше отриманих даних.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

*1. Уоссермен Ф. Нейрокомпьютерная техника / Ф. Уоссермен. – М. : Мир, 1992. – 238 с.*

*2. Маннинг К. Введение в информационный поиск / К. Маннинг, П. Рагхаван, Х. Шютце. – М. : Вильямс, 2011. – 528 с.*

*3. Sequence to Sequence Learning with Neural Networks [Електронний ресурс]. – Режим доступу: https://arxiv.org/pdf/1409.3215.pdf.*

*4. Natural Language Processing with Python [Електронний ресурс]. – Режим доступу: http://www.nltk.org/book_1ed.*

*5. Прикладная и компьютерная лингвистика / Под ред. И. С. Николаева, О. В. Митрениной, Т. М. Ландо. – М. : URSS, 2016. – 320 с.*

*6. Автоматическая обработка текстов на естественном языке и компьютерная лингвистика : учеб. пособ. / Большакова Е. И., Клышинский Э. С., Ландэ Д. В., Носков А. А., Пескова О. В., Ягунова Е. В. – М. : МИЭМ, 2011. – 272 с.*

*7. Understanding LSTM Networks [Електронний ресурс]. – Режим доступу: http://colah.github.io/posts/2015-08-Understanding-LSTMs.*

*8. Hochreiter S. Long Short-Term Memory / S. Hochreiter, J. Schmidhuber // Neural Computation. – 1997. – No. 9(8). – P. 1735–1780.*

*9. Sequence to Sequence Learning with Neural Networks [Електронний ресурс]. – Режим доступу: https://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf.*

*10. Free ebooks – Project Gutenberg [Електронний ресурс]. – Режим доступу: http://www.gutenberg.org.*

*11. Beautiful Soup Documentation [Електронний ресурс]. – Режим доступу: https://www.crummy.com/ /software/BeautifulSoup/bs4/doc.*

*12. Universal POS tags [Електронний ресурс]. – Режим доступу: http://universaldependencies.org/u/pos.*

*13. Concept of Bucketing in Seq2Seq model [Електронний ресурс]. – Режим доступу: https://stackoverflow.com/questions/49367871/concept-of-bucketing-in-seq2seq-model.*

*14. What is Machine Learning Perplexity? [Електронний ресурс]. – Режим доступу: https://jamesmccaffrey.wordpress.com/2016/08/16/what-is-machine-learning-perplexity.*

*Стаття надійшла до редакції 11.03.2018 р.*