УДК 004.89:[004.931:81`36]

ЗАСТОСУВАННЯ МЕТОДІВ ГЛИБОГО НАВЧАННЯ В ЗАДАЧАХ NLP: ГРАМАТИЧНА КОРЕКЦІЯ ТЕКСТІВ

Харченко М.В., Рябова Н.В.
e-mail: maksym.kharchenko1@nure.ua, nataliya.ryabova@nure.ua
Харківський національний університет радіоелектроніки, каф. ШІ,
м. Харків, Україна.

Modern approaches to solving the problem of text Grammatical Error Correction (GEC), based on deep learning methods, are considered. The main algorithmic methods, including rule-based, dictionary-based and more recent neural network-based methods, are considered. The problem of regeneration of whole input sequence in sequence-to-sequence models is described. The sequence-to-edits method with an example of such architecture is described as a solution to the regeneration problem. A perspective approach to combining algorithmic and neural network methods is proposed.

Розвиток методів глибокого навчання та генеративного штучного інтелекту надали нові можливості при вирішенні задач обробки природної мови (Natural Language Processing, NLP). Однією з таких задач є граматична корекція текстів (Grammatical Error Correction), ціллю якої є виправлення орфографічних, пунктуаційних, стилістичних та інших помилок у текстах написаних людьми. Складність завдання полягає у великій варіативності можливих помилок, необхідність оброблювати тексти різних стилів зберігаючи зміст.

Традиційними методами граматичної корекції текстів є алгоритмічні підходи. Ці підходи орієнтовані на правила, що задаються людьми. Такі системи, як «LanguageTool» спираються на правила, що використовують такі інструменти, як регулярні вирази та виявлення частин мови для певних слів. Для виправлення орфографічних помилок використовуються лінгвістичні словники. Вони використовуються для виявлення помилок та для знаходження можливих варіантів виправлення, за допомогою пошуку схожих слів у словнику. Перевагою таких методів є їх швидкодія та детермінованість. Недоліками є низька якість виправлень, так як різноманітність можливих помилок є великою та складно описується правилами, а також потребує детального розуміння контексту.

Більш сучасними методами вирішення цієї проблеми є використання нейронних мереж. Використання моделей послідовності до послідовності (sequence-to-sequence) дозволило обробляти увесь вхідний контекст, та генерувати текст з виправленими помилками. До таких архітектур відносяться рекурентні нейронні мережі (RNN), моделі довгої короткочасної пам’яті (LSTM). Архітектура моделей трансформерів (Transformers) надала новий поштовх до вирішення проблем NLP [1], зокрема й проблеми граматичної корекції текстів. Такі моделі, як «BART»,

«T5», «GPT» та інші показують хороші результати завдяки можливості оброблювати багато даних під час навчання та знаходити складні залежності. Недоліком моделей послідовність до послідовності полягає у тому, що у випадку, коли вхідний текст не потребує ніяких перетворень моделі все одно необхідно перегенерувати весь текст, не вносячи ніяких змін. Так як такі моделі є авторегресивними, тобто генерують послідовність покроково, така перегенерація усієї послідовності використовує значну кількість обчислювальних потужностей.

Вирішенням проблеми використання великої кількості ресурсів на обробку текстів, що не потребують змін є моделі послідовність до виправлень (sequence-to-edits). Такі моделі оброблюють вхідну послідовність, та надають виправлення до неї, якщо ж текст не потребує виправлень, то не буде згенеровано й виправлень, а отже, не буде витрачено обчислювальні потужності. Прикладом такої системи є модель «GECToR» [2] від «Grammarly». В її основі лежить трансформер кодувальник (Transformer Encoder), такий як «BERT», «RoBERTa», «XLNet» або інші, що обробляє вхідну послідовність, та класифікатор, що виконує роль декодувальника. Класифікатор застосовується до кожного токену вхідної послідовності, що надає дію, яку потрібно зробити з цим токеном: залишити, видалити, замінити або додати інший токен. Так як, за одну ітерацію таким способом не завжди виходить виправити усі помилки, після застосування виправлень, послідовність знову проходить через всю систему, доки усі помилки не будуть виправлені. Такий метод обробки збільшує ефективність використання обчислювальних ресурсів в десятки разів.

Деякі позиції, що потребують виправлень можуть бути обчислені без використання моделей глибокого навчання. Прикладом такого є орфографічні помилки, неправильно написані слова можуть бути знайдені за допомогою словникових систем. Традиційні системи для виправлення орфографії можуть дуже точно знаходити неправильно написані слова, але мають проблеми із пропонуванням виправлення, так як для того, щоб обрати найбільш доцільне виправлення потрібно досліджувати контекст, для чого підходять методи глибокого навчання. Запропонований метод вирішення задачі використовує класифікатор у комбінації з трансформером кодувальником, як у архітектурі «GECToR» [2], але лише для визначення токенів, що потребують виправлення, що зводить задачу до бінарної класифікації. Далі, використовуючи техніку підсвічування частини вхідної послідовності для трансформера декодувальника, описану у «Multi-headed Architecture Based on BERT for Grammatical Errors Correction» [3]. Ця техніка дозволяє декодувальнику сфокусуватись на потрібній частині та згенерувати виправлення саме для підсвіченої частини послідовності. Таким чином, позиції що потребують виправлення, згенеровані моделлю та отримані за допомогою алгоритмічного підходу

можливо об’єднати за допомогою оператора «АБО» та вже після цього відправити у декодувальник для генерації виправлень саме для підсвічених частин тексту.

Загальна запропонована архітектура зображена на рисунку 1.

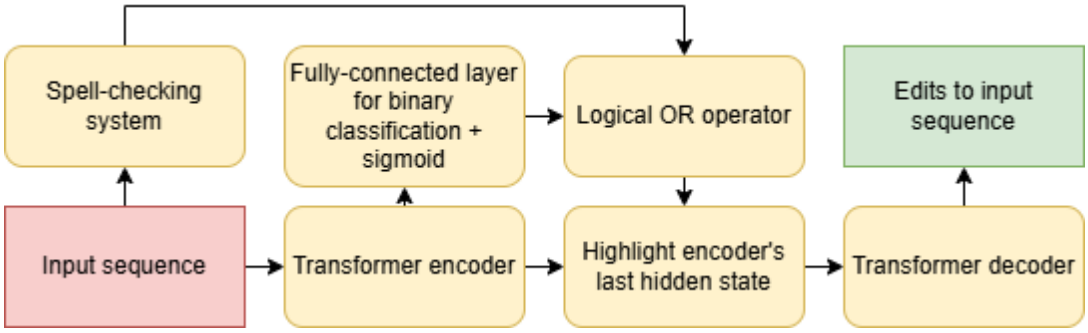

Рисунок 1 – Повна діаграма запропонованої системи

Отже, використання нейронних мереж у задачі граматичної корекції текстів є необхідним для досягання високих результатів якості. Але, недоліком систем глибокого навчання є непередбачуваність результатів, що може призводити до того, що одна й та сама помилка може бути виправлена та не виправлена у різних контекстах. Запропонована архітектура дозволяє поєднувати результати класифікатора та алгоритмічних підходів, таких як системи знаходження орфографічних помилок. Таке поєднання дозволяє зробити результат роботи системи більш передбачуваним та стійким.

Список використаних джерел:

1. Shatalov O., Ryabova N. Towards Russian Text Generation Problem Using OpenAI’s GPT-2. Proc. 5th Int. Conf. On Computational Linguistics and Intelligent Systems (COLINS), Volume I: Main Conference. CEUR Workshop Proceedings, 2021, 2870, pp.141-153.

2. GECToR – Grammatical Error Correction: Tag, Not Rewrite / K. Omelianchuk та ін. arXiv.org. URL: https://arxiv.org/abs/2005.12592 (дата звернення: 05.03.2025).

3. Didenko B., Shaptala J. Multi-headed Architecture Based on BERT for Grammatical Errors Correction. Association for Computational Linguistics. 2019. С. 246–251. URL: https://doi.org/10.18653/v1/W19-4426 (дата звернення: 05.03.2025).