Исправление грамматических ошибок в домене низкоресурсных языков

Ильдар Хабутдинов

Научный руководитель: А. В. Грабовой
Московский Физико-Технический Институт

depinwhite@gmail.com

17 мая 2025 г.

Задача о нахождении инъективного отображения

Проблема

Построение интерпретируемого автоматического исправления текстовых последовательностей

Задача

Исправление грамматических ошибок в домене низкоресурсных языков.

Метод решения

Метод инъективного отображения из множества произвольных символьных последовательностей в множество наперед заданных целевых последовательностей.

Предложенный метод решения задачи основан на сведении задачи нахождения последовательности корректирующих преобразований к задаче поиска оптимального редакционного предписания между исходной и целевой последовательностями.

Постановка задачи

Задано множество символьных последовательностей,

$$\mathcal{X} = \{s_i | s_i = \{x_1, x_2, ..., x_{n_i}\}\}_{i=0}^{N},$$

которые путем разбиения фиксированным алгоритмом представимы в виде последовательности токенов длины n_i
Задан словарь W корректирующих преобразований размера K

$$W = \{w_i\}_{i=0}^{K}$$

Задано множество целевых последовательностей с разбиением длины m_i :

$$\mathcal{Y} = \{t_i | t_i = \{y_1, y_2, ..., y_{m_i}\}\}_{i=0}^{N}$$

Требуется найти множество всевозможных последовательностей корректирующих преобразований $\mathcal{F}$:

$$\mathcal{F} = \{\{w_1, w_2, ..., w_{n_i}\} : \{w_1, w_2, ..., w_{n_i}\} \circ \{x_1, x_2, ..., x_{n_i}\} \to \{y_1, y_2, ..., y_{m_i}\}, w_j \in W\}$$

Символом $\circ$ обозначено поэлементное применение соответствующего корректирующего преобразования w_j к элементу разбиения x_j.

WordPiece токенизация

Задача WordPiece токенизации состоит в том, чтобы разбить произвольную символьную последовательность S на последовательность символьных подпоследовательностей (WordPiece токенов) обозначаемых как $WT = \{token_1, token_2, ..., token_m\}$.

Таким образом, $S = token_1^* \cup token_2^* \cup ... \cup token_n^*$, где $token_i^* \in WT, n \leq m$. Обозначим применение WordPiece токенизации буквой A.

Утв. Пусть s и t произвольные конечные символьные последовательности. Любую последовательность $A(t)$ можно получить из любой другой последовательности $A(s)$ за конечное число операций вставки, удаления и замены.

Определение редакционного предписания

Множество корректирующих преобразований состоит из элементов:

1. KEEP — оставить токен без изменения
2. REPLACE_t — заменить токен x_j произвольным токеном t
3. DELETE — удалить токен x_j
4. APPEND_t — добавить токен t после токена x_j

Опр. Редакционное предписание - это последовательность корретирующих преобразований, необходимых для получения целевой последовательности из исходной, имеющая минимальное количество операций вставки, замены и удаления.в

Пусть $D_{i,j}$ - это расстояние редактирования между префиксами $s[0..i]$ и $t[0..j]$ длины i и j. Где $D_{0,j} = 0$ и $D_{i,0} = 0$. Остальные значения определяются рекуррентным соотношением:

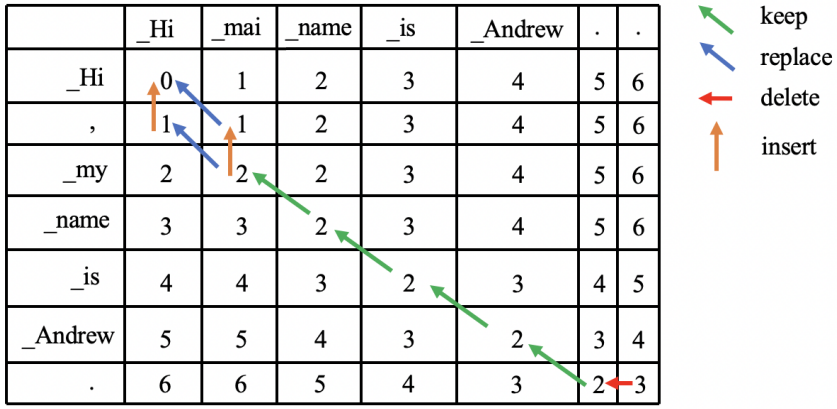
$$D_{i,j} = \begin{cases} D_{i-1,j-1}, s[i] = t[j] \\ 1 + min\{D_{i-1,j}, D_{i,j-1}, D_{i-1,j-1}\}, else \end{cases}$$

Утв. Количество возможных редакционных предписаний равно количеству путей в графе подзадач, имеющих минимальную стоимость.

Граф подзадач для редакционного предписания

Рассмотрим граф подзадач, где каждая вершина соответствует состоянию пары индексов (i, j), и ребра графа отражают возможные корректирующие преобразования между последовательностями s и t длиной n и m соответственно:

- Если символы $s[i]$ и $t[j]$ равны, то можно перейти по диагонали без изменения (операция *keep*).
- Если символы $s[i]$ и $t[j]$ различны, то возможны следующие операции:
  - *Замена* (replace): переход по диагонали $(i-1, j-1)$,
  - *Вставка* (append): переход из $(i, j-1)$,
  - *Удаление* (delete): переход из $(i-1, j)$.



	_Hi	_mai	_name	_is	_Andrew	.	.
_Hi	0	1	2	3	4	5	6
,	1	1	2	3	4	5	6
_my	2	2	2	3	4	5	6
_name	3	3	2	3	4	5	6
_is	4	4	3	2	3	4	5
_Andrew	5	5	4	3	2	3	4
.	6	6	5	4	3	2	3

Редакционные предписания — {delete . ; insert _my ; replace _mai with ,} и {delete . ; insert , ; replace _mai with _my}

Поиск оптимального редакционного предписания

Обозначим $EP_k = \{e_1, e_2, ..., e_{o_k}\}$ множетсво редакционных предписаний для пары последовательностей $(s_k,\ t_k)$, где o_k — количество редакционных предписаний для k-й пары.
Для нахождения оптимального редакционного предписания в k-ой паре, мы учитываем сходство исходных и целевых токенов для коррект. преобр. replace.
Пусть $R_l \subset e_l, l \in \{1, 2, ..., o_k\}$ - множество всех правил replace мощностью p_l для произвольного редакционного предписания e_l:

$$R_l = \{replace\_t_{1i}\_t_{2i}\}_{i=0}^{p_l},$$

где исходный токенов $t_{1i} \in s_k$, целевой токен $t_{2i} \in t_k$.
Введем сходство токенов как:

$$\sigma_l = \sum_{i=0}^{p_l} LevenshteinDist(t_{1i}\_t_{2i})$$

где $LevenshteinDist$ - функция, вычисляющая расстояние Левенштейна между t_{1_i} и t_{2_i} на уровне символов внутри токенов.

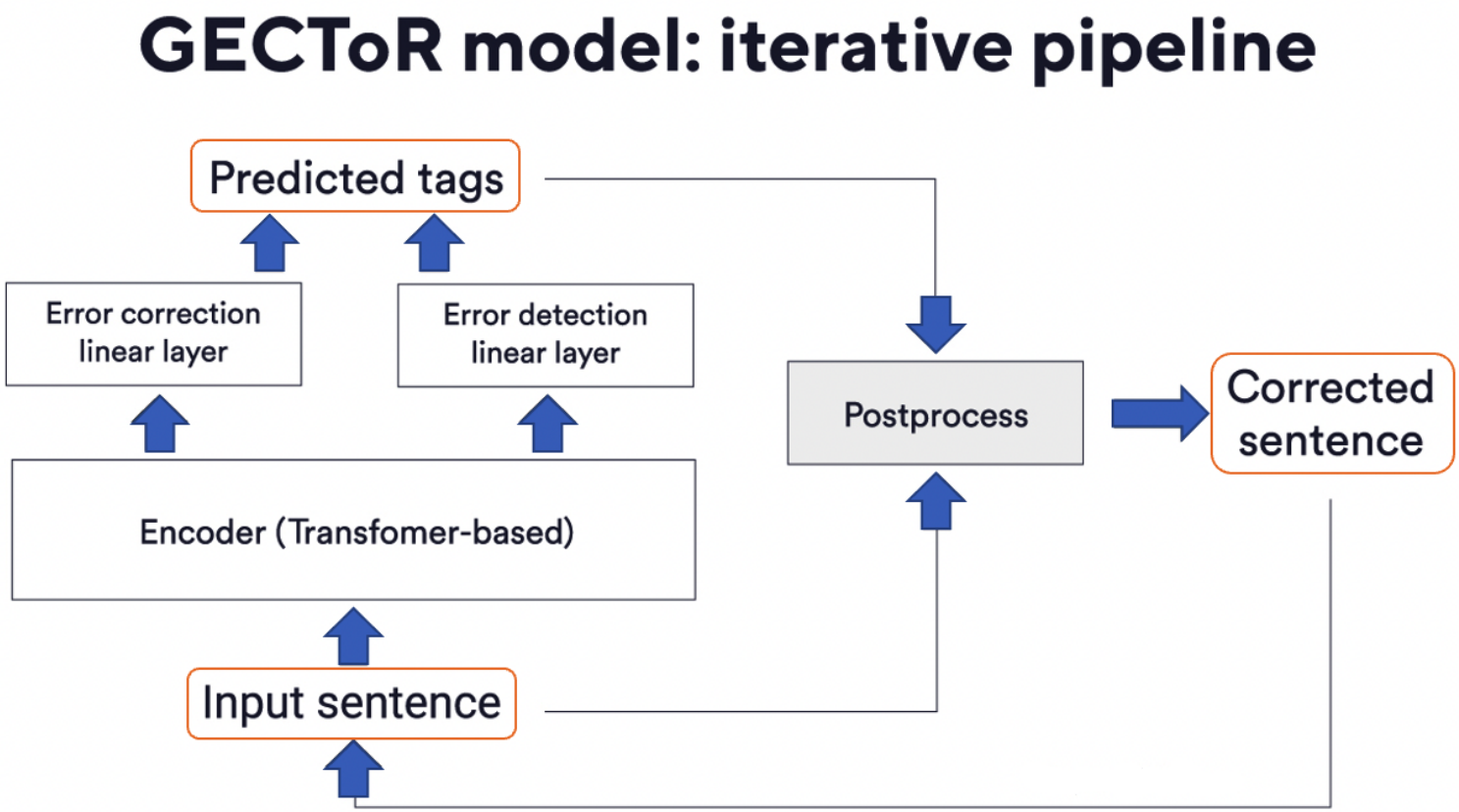
Утв. Редакционное предписание $e_l^* : l = argmin\{\sigma_1, \sigma_2, ..., \sigma_{o_k}\}$ является оптимальным.

Преимущества предложенного метода

Преимущества данного подхода:

1. нет необходимости в ручной разработке словаря грамматических правил, следовательно, может быть обобщено на любой низкоресурсный язык;
2. нет необходимости в ручной разработке словаря правил W, корректирующие преобразования могут получены путем нахождения редакционного предписания.

Архитектура модели



Вычислительные эксперименты

Dataset	#sents		Training stage
	Token-level	Word-level	
PIE-synthetic	9,000,000	9,000,000	I
Lang-8	787,613	947,344	II
NUCLE	51,929	56,958	II
FCE	25,968	34,490	II
W&I+LOCNESS	21,828	34,304	II, III

Table 1: Training datasets at each stage with the corresponding number of sentences in the GECToR article (word-level) and in our research (token-level)

Model	CoNLL-2014 (test)			BEA-2019 (test)		
	P	R	$F_{0.5}$	P	R	$F_{0.5}$
GECToR (token-level + XLNet)	72.3	40.4	62.4	70.5	41.6	61.9
GECToR (word-level + BERT)	72.1	42.0	63.0	71.5	55.7	67.6
GECToR (word-level + RoBERTa)	73.9	41.5	64.0	77.2	55.1	71.5
GECToR (word-level + XLNet)	77.5	40.1	65.3	79.2	53.9	72.4

Table 3: The best epochs at each training stage of the XLNet model in the GECToR article (word-level) and in our research (token-level).

- Word-level — токенизация на уровне слов, есть необходимость в ручной разработке словаря правил
- Token-level — токенизация на уровне WT-токенов, нет необходимости в ручной разработке словаря правил

Вывод

Сделано:

- Предложен метод инъективного отображения из множества произвольных символьных последовательностей в множество наперед заданных целевых последовательностей на уровне WordPiece токенов.
- Метод является универсальным относительно языков, зависит лишь от токенизатора. Не требует разработки грамматических правил.
- В эксперименте показано, что переход на уровень токенов показывает сравнительное качество работы при том, что задача не требует наличия размеченных данных.

Планируется:

- Провести эксперименты для других языков.
- Провести эксперименты для определения зависимости между качеством работы алгоритма и используемым токенайзером.

Список работ по теме НИР

Публикации

1. **Khabutdinov, I.A., Chashchin, A.V., Grabovoy, A.V. et al.** RuGECToR: Rule-Based Neural Network Model for Russian Language Grammatical Error Correction. Program Comput Soft 50, 315–321 (2024). https://doi.org/10.1134/S0361768824700129
2. **K. Varlamova, I. Khabutdinov and A. Grabovoy**, "Automatic Spelling Correction for Russian: Multiple Error Approach,"2023 Ivannikov Ispras Open Conference (ISPRAS), Moscow, Russian Federation, 2023, pp. 169-175, doi: 10.1109/ISPRAS60948.2023.10508161.
3. **Gritsai, German & Khabutdinov, Ildar & Grabovoy, Andrey**. (2024). Multi-head Span-based Detector for AI-generated Fragments in Scientific Papers. 220-225. 10.18653/v1/2024.sdp-1.21.
4. **K. Grashchenkov, A. Grabovoy and I. Khabutdinov**, "A Method of Multilingual Summarization For Scientific Documents,"2022 Ivannikov Ispras Open Conference (ISPRAS), Moscow, Russian Federation, 2022, pp. 24-30, doi: 10.1109/ISPRAS57371.2022.10076852.
5. **Gritsai, German & Voznyuk, Anastasia & Khabutdinov, Ildar & Grabovoy, Andrey.** (2024). Advacheck at GenAI Detection Task 1: AI Detection Powered by Domain-Aware Multi-Tasking. 10.48550/arXiv.2411.11736.
6. **Khabutdinov, I.A., Krinitskiy, M.A. Belikov, R.A.** Identifying Cetacean Mammals in High-Resolution Optical Imagery Using Anomaly Detection Approach Employing Machine Learning Models. Moscow Univ. Phys. 78 (Suppl 1), S149–S156 (2023). https://doi.org/10.3103/S0027134923070147

Выступления с докладом

1. RuGECToR: нейросетевая модель на основе правил для исправления грамматических ошибок на русском языке «Открытая конференция ИСП РАН», 2022.
2. Multi-head Span-based Detector for AI-generated Fragments in Scientific Papers, SDP@ACL, 2024.
3. Анализ работы BERT-подобных моделей в задачах классификации грамматических ошибок на русском языке «65-я научная конференция МФТИ», 2023.