ИЗДАТЕЛЬСТВО ГРАМОТА

Филологические науки. Вопросы теории и практики
Philology. Theory & Practice

ISSN 2782-4543 (online)
ISSN 1997-2911 (print)

2022. Том 15. Выпуск 10. С. 3382-3386 | 2022. Volume 15. Issue 10. P. 3382-3386
Материалы журнала доступны на сайте (articles and issues available at): philology-journal.ru

RU

Экспериментальное моделирование базы данных сбалансированного лингвистического корпуса

Горожанов А. И.

Аннотация. Целью исследования является построение функционирующей экспериментальной модели реляционной базы данных для оперирования сбалансированным лингвистическим корпусом художественного произведения. Научная новизна заключается в том, что впервые в рамках гуманитарного исследования проводится моделирование базы данных лингвистического корпуса с тщательным описанием и учетом технических деталей и с опорой на положения авторской концепции профессионально ориентированного программирования. Работа состояла из трех этапов: формирования технического задания (разработана структура двух таблиц реляционной базы данных, выбран формат SQLite, предусмотрены дополнительные колонки таблиц для последующего расширения содержания исследований), написания программного кода создания и наполнения базы данных (использованы язык программирования Python, модуль обработки естественного языка spaCy) и его апробации на материале текстов трех романов Ф. Кафки «Замок», «Америка» и «Процесс» (получены три функционирующие базы данных). Результаты показали, что современные программные инструменты обработки естественного языка позволяют автоматически создавать полноценные базы данных для обработки запросов SQL, которые впоследствии возможно расширять в ручном или автоматическом режиме.

EN

Experimental Database Modelling of a Balanced Linguistic Corpus

Gorozhanov A. I.

Abstract. The research aims to build a functioning experimental model of a relational database for operating with a balanced linguistic corpus of a fiction work. Scientific novelty lies in the fact that for the first time within the framework of a humanities study, a database of a linguistic corpus is being modeled with a thorough description and taking into account technical details and based on the provisions of the author's concept of professionally oriented programming. The work involved three stages: forming a technical task (the structure of two tables of a relational database was developed, the SQLite format was selected, additional columns of the tables were provided for the subsequent expansion of the content of research), writing the source code for creating and filling the database (the Python programming language and the spaCy natural language processing module were used) and testing it based on the material of the texts of three F. Kafka's novels "The Castle", "Amerika" and "The Trial" (three functioning databases were created). The research findings have shown that modern natural language processing software tools allow one to create automatically full-fledged databases for processing SQL queries, which can be further expanded manually or automatically.

Введение

База данных (БД) является неотъемлемой частью любого лингвистического корпуса, поэтому выбор типа БД – ключевой вопрос при разработке программного обеспечения для корпусных исследований.

Обзор предметно-специальной литературы по проблеме показывает, что публикации в гуманитарных журналах затрагивают описания БД, однако при этом рассказывается не собственно о технических спецификациях, а об их языковом наполнении, использовании в рамках тех или иных проектов и т.п. (Хохлова, 2021; Mizrahi, Dickinson, 2022; Verma, Sikarvar, Yadav et al., 2022). Только единичные работы содержат технические детали, хотя и в ограниченном объеме (Лесников, 2021, с. 30; Писарик, 2021, с. 152-154).

Кроме того, в настоящее время корпусная лингвистика является динамично развивающейся областью языкознания, в частности благодаря тому, что современные компьютерные технологии позволяют оперировать большими объемами текстовых данных, для которых цифровой формат скорее первичен, чем вторичен.

Научная статья (original research article) | https://doi.org/10.30853/phil20220563

Использование готовых (типовых) технических решений уже не может в полной мере удовлетворить всех потребностей исследователей, так как лингвистические корпусы все чаще становятся мультимодальными, выходят за рамки исключительно текстовых данных и их характеристик, поэтому БД современных корпусов должны быть структурно гибкими, а процессы их построения и наполнения необходимо максимально автоматизировать, опираясь для этого на синергетический подход (здесь – равное вовлечение технического и гуманитарного компонентов на всех этапах работы).

Сказанное выше обуславливает актуальность нашего исследования.

Сформулируем задачи работы:

1. Составить техническое задание к БД, оптимальное с точки зрения как технических, так и лингвистических параметров.
2. Написать программный код для автоматической генерации БД с привлечением методов обработки естественного языка.
3. Апробировать программное приложение на аутентичном языковом материале трех романов Ф. Кафки «Замок», «Америка» и «Процесс».

Практическая значимость работы заключается в том, что ее результаты предоставят исследователям в области корпусной лингвистики широкие возможности для быстрой генерации БД сбалансированных корпусов интересующих их художественных произведений, а также могут быть использованы при подготовке учебно-методических материалов для чтения дисциплин «Цифровая трансформация в профессиональной деятельности» и «Информационно-коммуникационные технологии в научной деятельности».

Методами исследования послужили моделирование (в части разработки модели БД), эксперимент (в ходе апробации полученной БД) и анализ (в части оценки полученных результатов).

Теоретической базой исследования стали труды, посвященные применению библиотек обработки естественного языка (Ayre, Bittar, Kam et al., 2021; Jugran, Kumar, Tyagi et al., 2021; Okhapkin, Okhapkina, Iskhakova et al., 2021) и принципов концепции профессионально ориентированного программирования (Gorozhanov, Guseynova, 2020), в том числе для интерпретации художественного текста (Горожанов, Гусейнова, 2021).

Основная часть

1. Составление технического задания

Среди многообразия БД, которые можно использовать для хранилища лингвистического корпуса, выделим форматы «простого» текста (отсутствие какой-либо разметки), XML (с разметкой тегами и их атрибутами) и SQL (могут хранить текстовые данные как с разметкой, так и без нее).

В контексте нашего исследования остановимся на последнем варианте, т.к. он позволяет относительно удобно работать с большими объемами данных, достаточно легко интегрируется в оффлайн- и веб-приложения и совместим с различными языками программирования.

Из числа БД, которые позволяют работать с запросами SQL, выберем SQLite, а в качестве языка программирования – Python.

Более сложной задачей является разработка структуры БД: определение количества таблиц, характера записей в каждой таблице и установления «реляции», т.е. связующего компонента между этими таблицами.

Поскольку БД должна генерироваться автоматически, но тем не менее иметь разметку, мы воспользуемся модулем обработки естественного языка spaCy, который имеет встроенные методы разделения текста на предложения и элементарные единицы – токены, а также может вполне точно определять морфологические характеристики последних. Исходя из этого, включим в нашу модель две таблицы – для предложений и для токенов, а поскольку токены существуют только внутри предложений, то связующим элементом между таблицами станет порядковый номер предложения.

Первая таблица должна иметь следующую структуру (см. Рисунок 1).

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	
1	id	integer	🔑						*NULL*
2	sentnum	integer					✱		*NULL*
3	senttext	text					✱		*NULL*
4	sentoption01	text							'NONE'
5	sentoption02	text							'NONE'
6	sentoption03	text							'NONE'
7	sentoption04	text							'NONE'
8	sentoption05	text							'NONE'

Рисунок 1. *Структура таблицы предложений*

Из восьми колонок таблицы первая генерируется БД автоматически (id), а остальные должны быть добавлены программным способом. Колонка sentnum содержит номер предложения и представлена целочисленным

типом данных, senttext предназначена для текста предложения (это текстовые данные). Следующие опциональные колонки для текстовых данных зарезервированы для возможного расширения характеристик предложений и в ходе эксперимента оставлены пустыми.

Таблица токенов получила следующий вид (см. Рисунок 2).

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	
1	id	integer							*NULL*
2	tokennum	integer							*NULL*
3	sent_num	integer							*NULL*
4	tokentext	text							*NULL*
5	tokenpos	text							*NULL*
6	tokenlemma	text							*NULL*
7	tokenattr	text							*NULL*
8	tokenoption01	text							'NONE'
9	tokenoption02	text							'NONE'
10	tokenoption03	text							'NONE'
11	tokenoption04	text							'NONE'
12	tokenoption05	text							'NONE'

Рисунок 2. *Структура таблицы токенов*

Колонка id, так же как и в первой таблице, заполняется автоматически, tokennum – это абсолютный номер токена в БД, sent_num – это номер предложения, в которое входит текущий токен. Все три колонки имеют целочисленные данные. Колонка tokentext содержит текст токена, tokenpos – это код части речи токена, tokenlemma – это исходная форма токена (например, инфинитив для глагола), а tokenattr предназначается для морфологических характеристик токена. Остальные пять колонок потенциально могут быть заполнены какими-либо другими характеристиками.

С целью придания БД гибкости для большинства колонок не прописывается обязательное условие иметь наполнение, поскольку spaCy не предусматривает всех характеристик для всех частей речи, к которым, например, этот модуль относит и знаки препинания.

2. Разработка программного приложения

Перед преобразованием текстовых файлов романов Ф. Кафки с помощью программы на Python необходимо было преобразовать их к так называемому «нормальному» виду, чтобы парсер spaCy не выделил в качестве отдельных токенов скопления пробелов. С этой целью все тексты романов были превращены в один абзац, который не содержал знаков переноса строки и серий из двух и более пробелов.

После генерации БД и двух пустых таблиц согласно требованиям технического задания и подключения модуля spaCy программа инициализировала цикл for, в котором по очереди перебирались все токены заданного текста. Набор вложенных проверок if – elif – else следил за тем, чтобы соблюдалась нумерация текущего предложения и все предложения по очереди записывались в первую таблицу. Если очередной токен оказывался началом предложения или одновременно и началом, и концом предложения, то счетчик предложений прибавлял единицу.

Благодаря использованию модуля spaCy код программы занял немногим менее 100 строк (см. Листинг 1).

Листинг 1. *Фрагмент кода программы – цикл for*

```
    # Счетчик предложений
    counterSent = 1

    # Цикл для перебора каждого токена
    for token in doc:

        # Все кавычки в лемме заменяются на описательные слова
        if token.text == "'":
            lemma = "quote"
        elif token.text == '"':
            lemma = "doublequote"
        else:
            lemma = token.lemma_

        # Подключение к базе данных
        conn = sqlite3.connect("amerikasql.db")

        if token.is_sent_start:
            conn.execute("INSERT INTO sents VALUES(NULL, %d, '%s', NULL, NULL, NULL, NULL, NULL)" %
(counterSent, token.sent))
```

```
            conn.commit()
            conn.execute("INSERT INTO tokens VALUES(NULL, %d, %d, '%s', '%s', '%s', '%s', NULL, NULL,
NULL, NULL, NULL)" % (token.i, counterSent, token.text, token.pos_, lemma, token.morph))
            conn.commit()

            if token.is_sent_end:
                counterSent += 1

        elif token.is_sent_end:
            conn.execute("INSERT INTO tokens VALUES(NULL, %d, %d, '%s', '%s', '%s', '%s', NULL,
NULL, NULL, NULL, NULL)" % (token.i, counterSent, token.text, token.pos_, lemma, token.morph))
            conn.commit()

            counterSent += 1

        else:
            conn.execute("INSERT INTO tokens VALUES(NULL, %d, %d, '%s', '%s', '%s', '%s', NULL, NULL,
NULL, NULL, NULL)" % (token.i, counterSent, token.text, token.pos_, lemma, token.morph))
            conn.commit()

    conn.close()
```

3. Апробация программного приложения

Далее созданная программа была экспериментально апробирована на текстах трех романов Ф. Кафки. Приведем несколько первых записей обеих таблиц БД для романа «Замок» (см. Рисунки 3 и 4).

id	sentnum	senttext	sentoption01	sentopt
1	1	Es war spät abends, als K ankam.	NULL	NUl
2	2	Das Dorf lag in tiefem Schnee.	NULL	NUl
3	3	Vom Schloßberg war nichts zu sehen, Neb...	NULL	NUl
4	4	Lange stand K auf der Holzbrücke, die von ...	NULL	NUl
5	5	Dann ging er, ein Nachtlager suchen;	NULL	NUl
6	6	im Wirtshaus war man noch wach, der Wirt...	NULL	NUl
7	7	K war damit einverstanden.	NULL	NUl
8	8	Einige Bauern waren noch beim Bier, aber ...	NULL	NUl
9	9	Warm war es, die Bauern waren still, ein we...	NULL	NUl
10	10	Aber kurze Zeit darauf wurde er schon gew...	NULL	NUl
11	11	Ein junger Mann, städtisch angezogen, mit...	NULL	NUl
12	12	Die Bauern waren auch noch da, einige hat...	NULL	NUl
13	13	Der junge Mensch entschuldigte sich sehr ...	NULL	NUl
14	14	Niemand darf das ohne gräfliche Erlaubnis.	NULL	NUl
15	15	Sie aber haben eine solche Erlaubnis nicht ...	NULL	NUl

Рисунок 3. *Первые 15 записей таблицы предложений для романа «Замок»*

Иллюстрация показывает, что spaCy определил предложения более-менее корректно. Видимым исключением является разделение предложения по точке с запятой (предложения 5 и 6). С другой стороны, встроенный анализатор spaCy выделил наличие различных основ и именно поэтому принял такое решение. Для нас важно, что на иллюстрации и во всех 5404 записях таблицы колонки id и sentnum синхронизированы по значениям. Это доказывает, что счетчик предложений сработал точно.

id	kennu	ıt_nu	tokentext	tokenpos	tokenlemma	tokenattr	to
1	0	1	Es	PRON	ich	Case=Nom|Gender=Neut|Number=Sing|P...	
2	1	1	war	AUX	sein	Mood=Ind|Number=Sing|Person=3|Tense...	
3	2	1	spät	ADV	spät	Degree=Pos	
4	3	1	abends	ADV	abends		
5	4	1	,	PUNCT	,		
6	5	1	als	ADP	als		
7	6	1	K	PROPN	K	Case=Nom|Number=Sing	
8	7	1	ankam	VERB	ankommen	Mood=Ind|Number=Sing|Person=3|Tense...	
9	8	1	.	PUNCT	.		
10	9	2	Das	DET	der	Case=Nom|Definite=Def|Gender=Neut|Nu...	
11	10	2	Dorf	NOUN	Dorf	Case=Nom|Gender=Neut|Number=Sing	
12	11	2	lag	VERB	liegen	Mood=Ind|Number=Sing|Person=3|Tense...	
13	12	2	in	ADP	in		
14	13	2	tiefem	ADJ	tief	Case=Dat|Degree=Pos|Gender=Masc|Num...	
15	14	2	Schnee	NOUN	Schnee	Case=Dat|Gender=Masc|Number=Sing	
16	15	2	.	PUNCT	.		

Рисунок 4. *Первые 15 записей таблицы токенов для романа «Замок»*

Таблица токенов также оказалась заполненной достаточно точно. Номера предложений, в которые входят токены, отобразились корректно. Морфологические признаки токенов (при их наличии) отобразились

как единый текст с разделителем между отдельными признаками. Нумерация токенов в тексте отличается от id по той причине, что spaCy нумерует токены от нуля, а не от единицы.

В результате для романа «Замок» БД составилась из 5404 записей таблицы предложений и 131943 записей таблицы токенов при общем объеме БД 8,35 МБ. Для романов «Процесс» и «Америка» эти показатели составили 3753, 86933, 5,57 МБ и 4025, 100230, 6,45 МБ соответственно. Автоматическая генерация каждой БД заняла около 45 минут.

Заключение

Проанализировав результаты исследования, мы пришли к выводу о том, что разработанная экспериментальная модель реляционной БД, во-первых, была успешно реализована и, во-вторых, структура базы данных проявляет себя как гибкая в плане последующей работы.

Погрешности spaCy в отношении разделения текста на предложения могут быть нивелированы выводом при поиске последовательностей предложений, а не одиночных записей первой таблицы, для того чтобы пользователь смог «вручную» зафиксировать текстовый массив и работать с широким контекстом.

Предусмотренные дополнительные колонки таблиц могут быть использованы для внесения пользователями неформализованных характеристик (стилистических, семантических, прагматических, просодических и пр.).

В качестве перспективы исследования обозначим апробацию созданной модели на материале других языков, а также разработку и апробацию графического интерфейса – корпусного менеджера – для удобной обработки запросов БД.

Источники | References

1. Горожанов А. И., Гусейнова И. А. Прикладные аспекты анализа и интерпретации текстов (на материале немецкого и русского языков). Казань: Бук, 2021.
2. Лесников С. В. Формирование гипертекстового корпуса учебных словарей русского языка // Филологические науки. Научные доклады высшей школы. 2021. № 4. DOI: 10.20339/PhS.4-21.027
3. Писарик О. И. Принципы разработки базы данных подъязыка предметной области «Строительство» // Вестник Московского государственного лингвистического университета. Гуманитарные науки. 2021. № 5 (847). DOI: 10.52070/2542-2197_2021_5_847_150
4. Хохлова М. В. Атрибутивные коллокации в золотом стандарте сочетаемости русского языка и их представление в словарях и корпусах текстов // Вопросы лексикографии. 2021. № 21. DOI: 10.17223/22274200/21/2
5. Ayre K., Bittar A., Kam J., Verma S., Howard L. M., Dutta R. Developing a Natural Language Processing Tool to Identify Perinatal Self-Harm in Electronic Healthcare Records // PLoS ONE. 2021. No. 16 (8). DOI: 10.1371/journal.pone.0253809
6. Gorozhanov A. I., Guseynova I. A. Programming for Specific Purposes in Linguistics: A New Challenge for the Humanitarian Curricula // Training, Language and Culture. 2020. Vol. 4. No. 4. DOI: 10.22363/2521-442X-2020-4-4-23-38
7. Jugran S., Kumar A., Tyagi B. S., Anand V. Extractive Automatic Text Summarization Using SpaCy in Python NLP // 2021 International Conference on Advance Computing and Innovative Technologies in Engineering, ICACITE 2021. Greater Noida, 2021. DOI: 10.1109/ICACITE51222.2021.9404712
8. Mizrahi M., Dickinson M. A. Philosophical Reasoning about Science: A Quantitative, Digital Study // Synthese. 2022. Vol. 200. No. 2. DOI: 10.1007/s11229-022-03670-6
9. Okhapkin V. P., Okhapkina E. P., Iskhakova A. O., Iskhakov A. Y. Constructing of Semantically Dependent Patterns Based on SpaCy and StanfordNLP Libraries // Communications in Computer and Information Science (in Books). 2021. Vol. 1395. DOI: 10.1007/978-981-16-1480-4_45
10. Verma A., Sikarvar V., Yadav H., Jaganathan R., Kumar P. Shabd: A Psycholinguistic Database for Hindi // Behavior Research Methods. 2022. Vol. 54. No. 2. DOI: 10.3758/s13428-021-01625-2

Информация об авторах | Author information

RU

Горожанов Алексей Иванович[1], д. филол. н., доц.
[1] Московский государственный лингвистический университет

EN

Gorozhanov Alexey Ivanovich[1], Dr
[1] Moscow State Linguistic University

[1] *a_gorozhanov@mail.ru*

Информация о статье | About this article

Дата поступления рукописи (received): 04.09.2022; опубликовано (published): 10.10.2022.

Ключевые слова (keywords): реляционная база данных; корпусная лингвистика; профессионально ориентированное программирование; SQLite; spaCy; relational database; corpus linguistics; professionally oriented programming.