



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

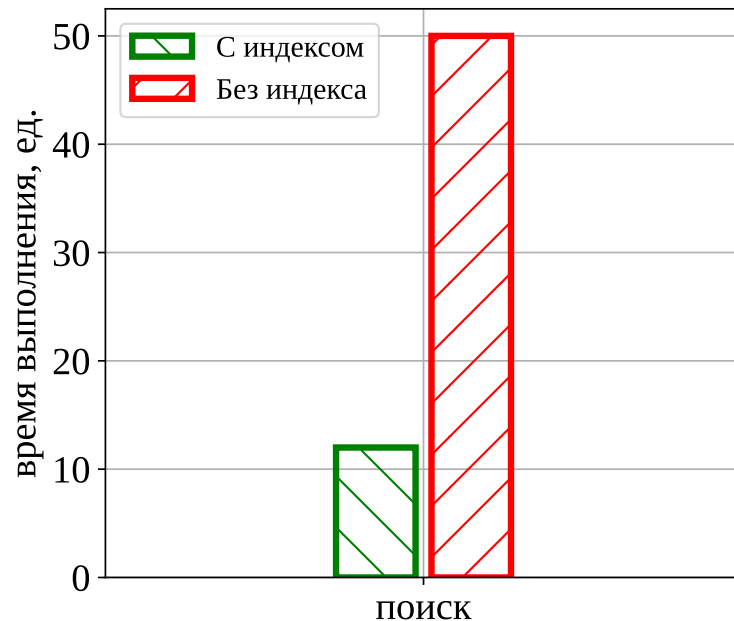
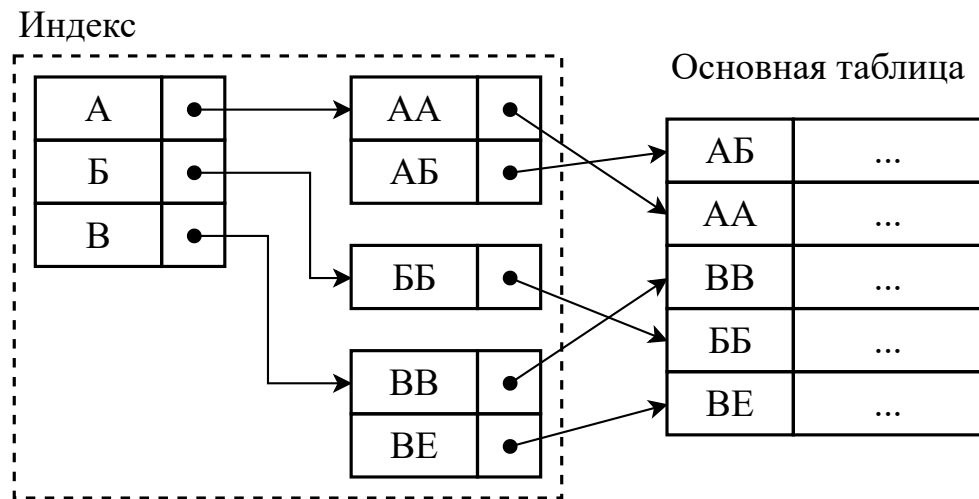
Метод построения поисковых индексов в реляционной базе данных на основе глубоких нейронных сетей

Студент: Маслова Марина Дмитриевна ИУ7-83Б
Руководитель: Оленев Антон Александрович

Москва, 2023

Индексы

Индекс — это некоторая структура, обеспечивающая быстрый поиск записей в базе данных за счет определения соответствия ключа конкретной записи положению этой записи.



Цель и задачи

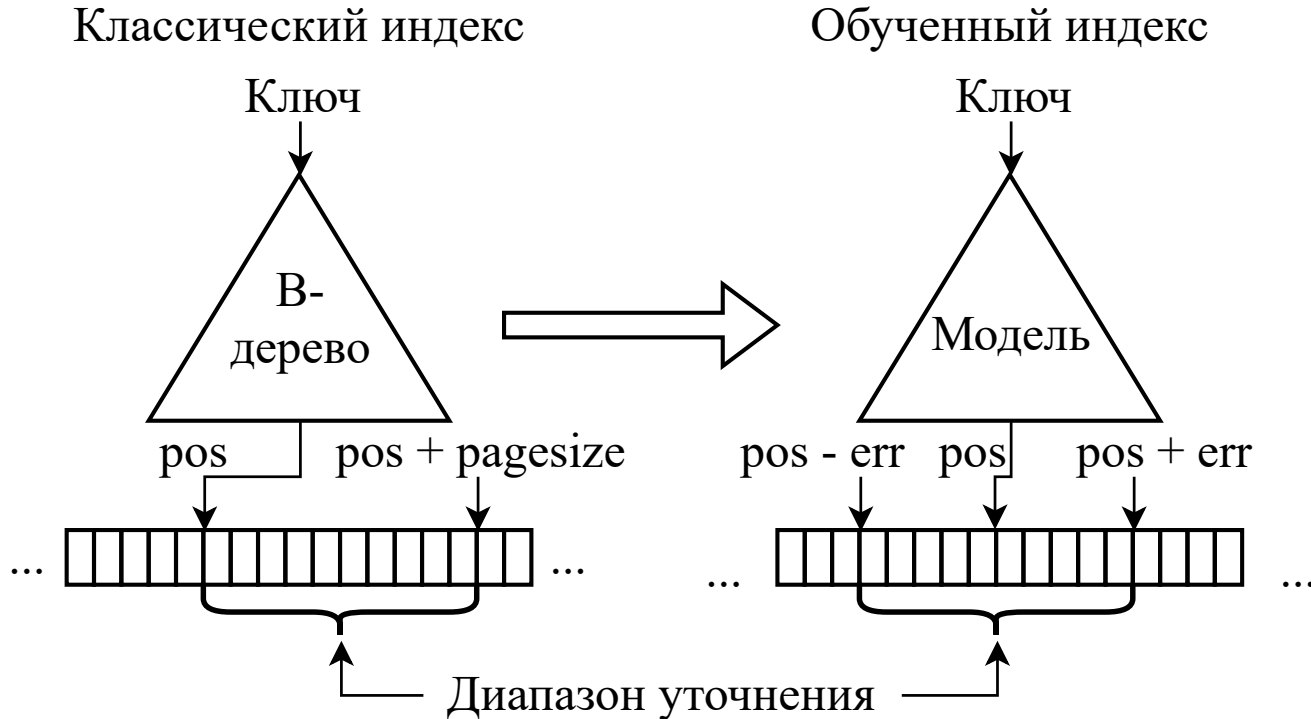
Цель: разработка метода построения поисковых индексов в реляционной базе данных на основе глубоких нейронных сетей и его программная реализация.

Задачи:

- проанализировать методы построения индексов;
- разработать метод построения индексов в реляционной базе данных на основе глубоких нейронных сетей;
- разработать программное обеспечение, реализующее данный метод;
- исследовать операции поиска и вставки с использованием индекса, построенного разработанным методом.

Построение индексов и машинное обучение

Классические структуры индексов: В-деревья, хеш-таблицы, битовые карты, — могут быть заменены моделями машинного обучения.

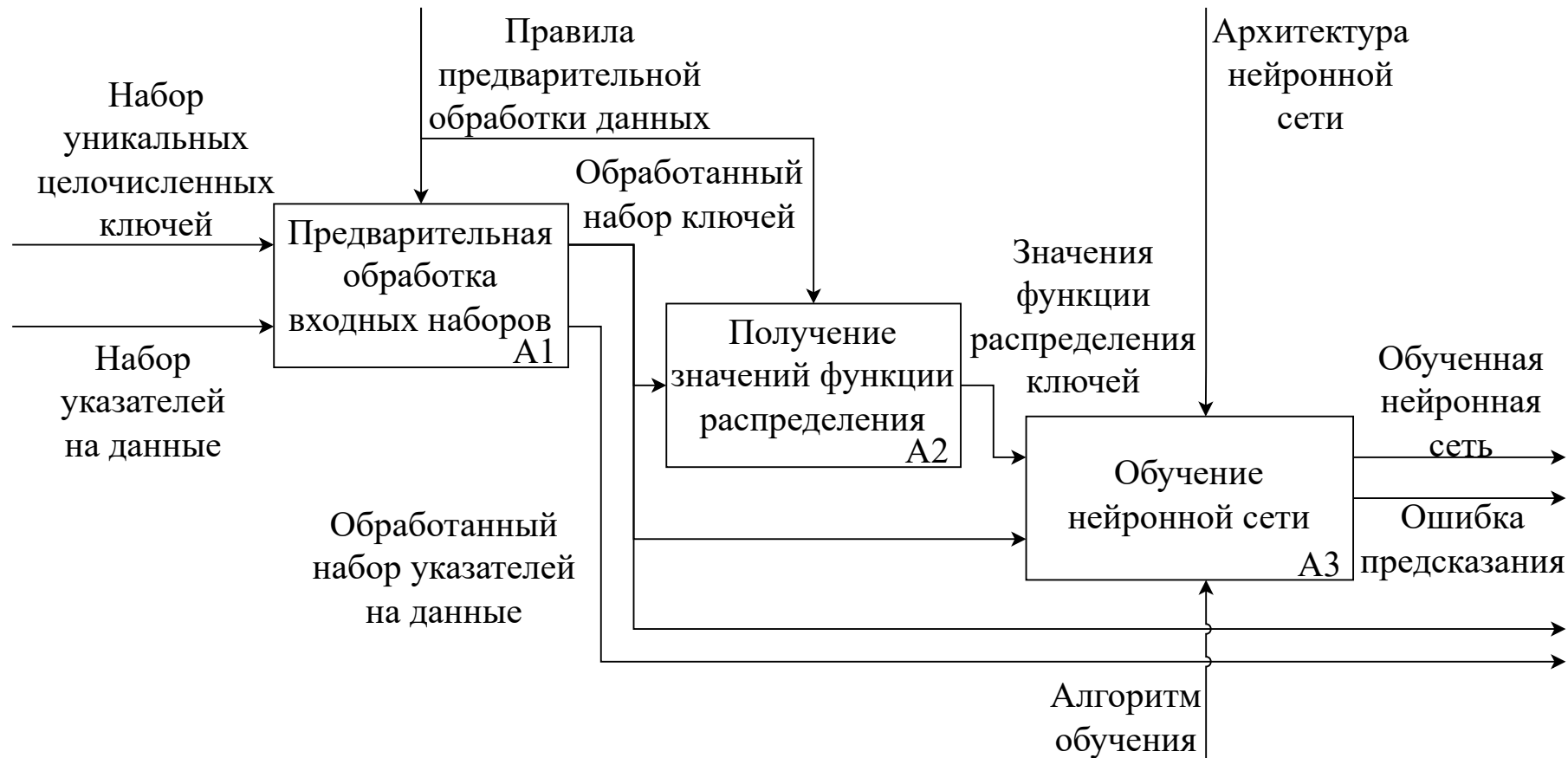


Сравнение методов построения индексов

Критерий сравнения		Классические индексы			Обученные индексы
		В-дерево	Хеш-таблица	Фильтр Блума	
Временная сложность	поиска	$O(\log N)$	$O(1) / O(N)$	$O(k)$	$O(1) / O(N)$
	вставки	$O(\log N)$	$O(1) / O(N)$	$O(k)$	(*)
Память		Высокая	Средняя	Низкая	Средняя
Поиск в диапазоне		+	-	-	+
Поиск единичного ключа		+	+	-	+
Проверка существования		+	+	+	+

(*) — вставка в обученный индекс требует переобучения, сложность которого зависит от архитектуры используемой модели машинного обучения.

Метод построения поисковых индексов



Предварительная обработка данных



Нормализация ключей производится в диапазон $[0, 1]$ по формуле:

$$x_{\text{норм}} = \frac{x - x_{\text{мин}}}{x_{\text{макс}} - x_{\text{мин}}},$$

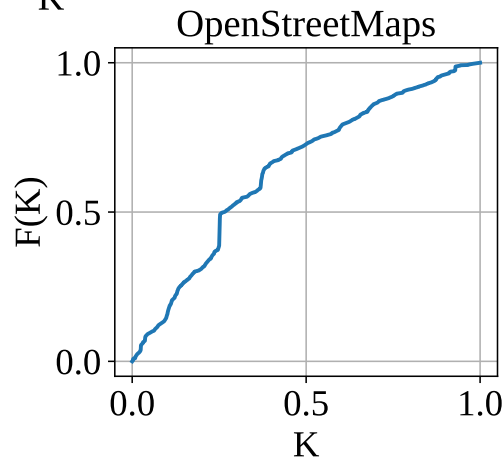
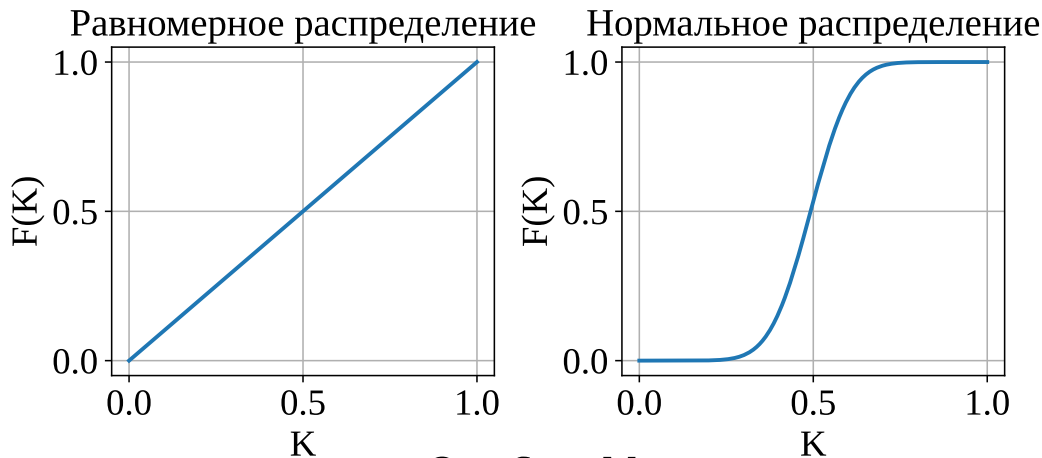
где $x_{\text{норм}}$ — нормализованное значение ключа;
 x — натуральное значение ключа;
 $x_{\text{мин}}$, $x_{\text{макс}}$ — минимальное и максимальное значение ключа в наборе соответственно.

Вычисление значений функции распределения

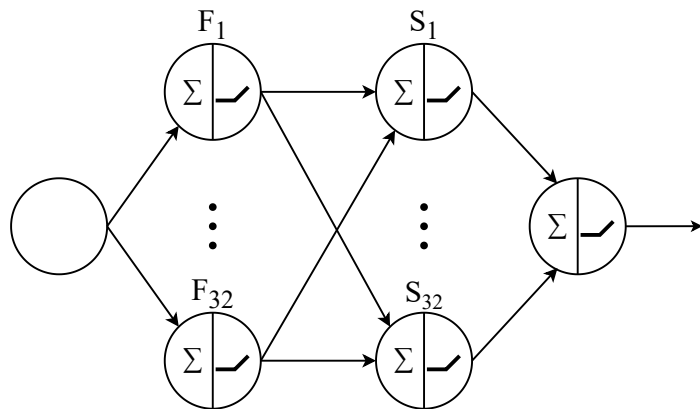
Значение функции распределения определяется по позиции ключа по формуле:

$$F(K) = \frac{p}{N},$$

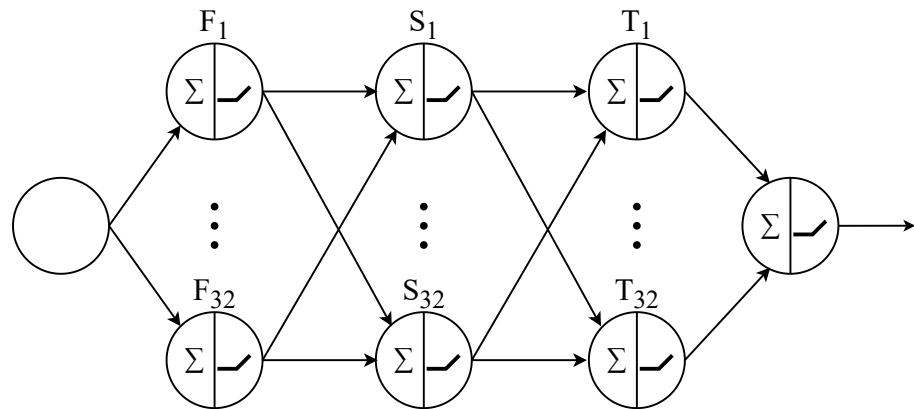
где $F(K)$ — функция распределения;
 K — нормализованный ключ;
 p — позиция ключа;
 N — количество ключей.



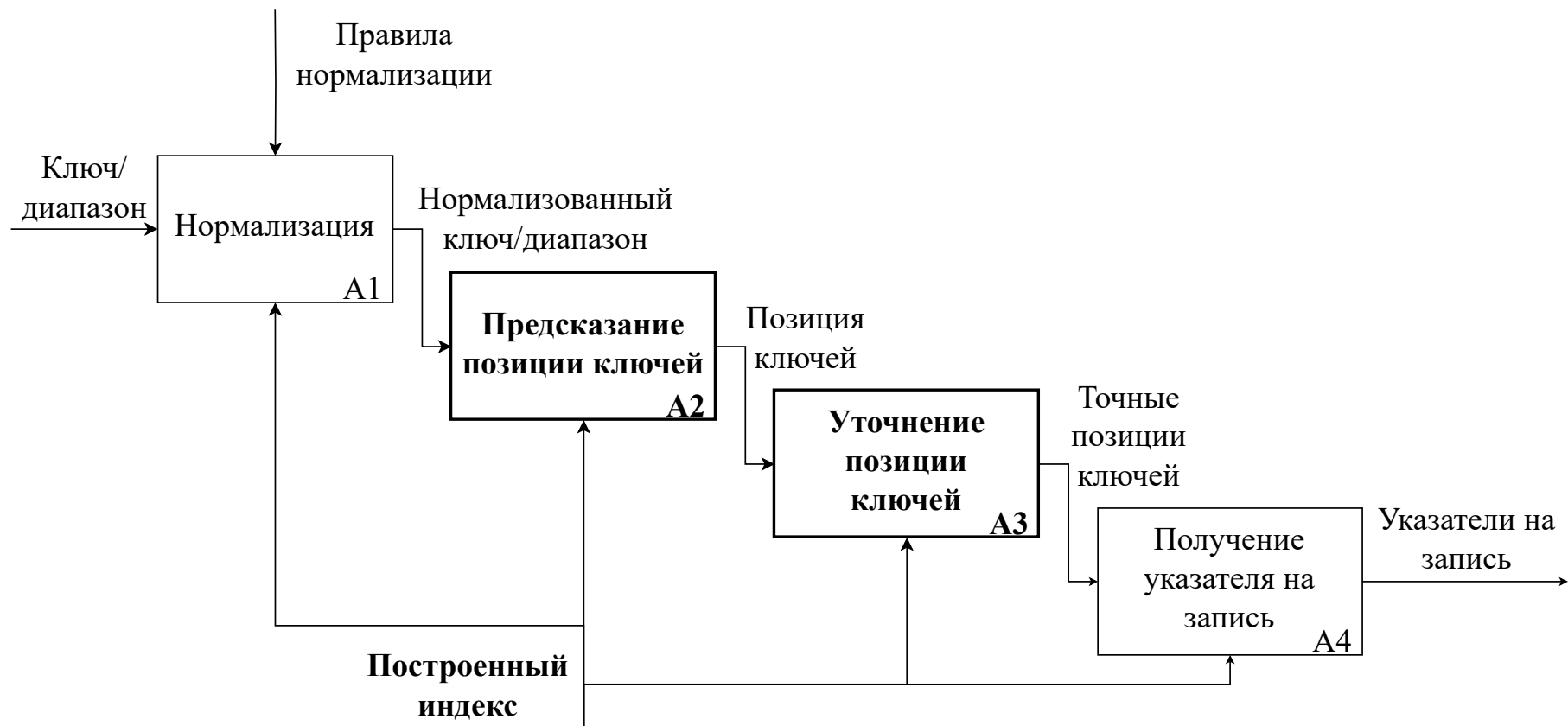
Архитектура нейронной сети



1 вход: нормализованный ключ K .
1 выход: значение функции
распределения $F(K)$.
2-3 скрытых слоя по 32 нейрона.
Функция активации: ReLU.



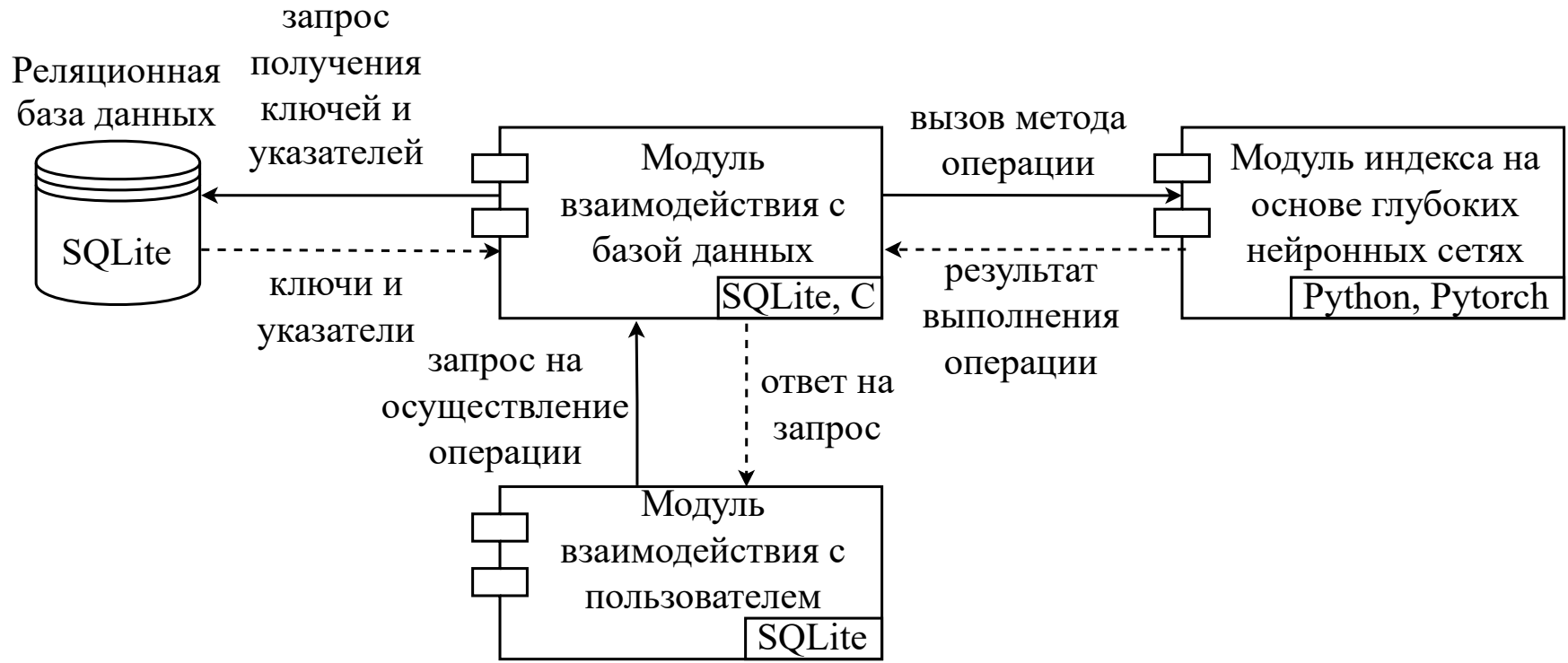
Применение метода: поиск



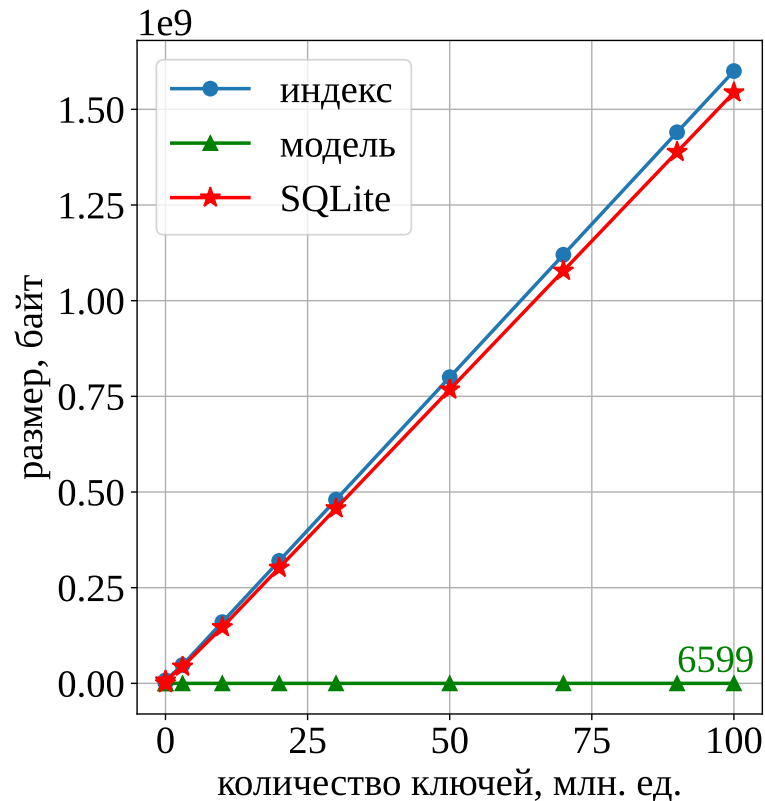
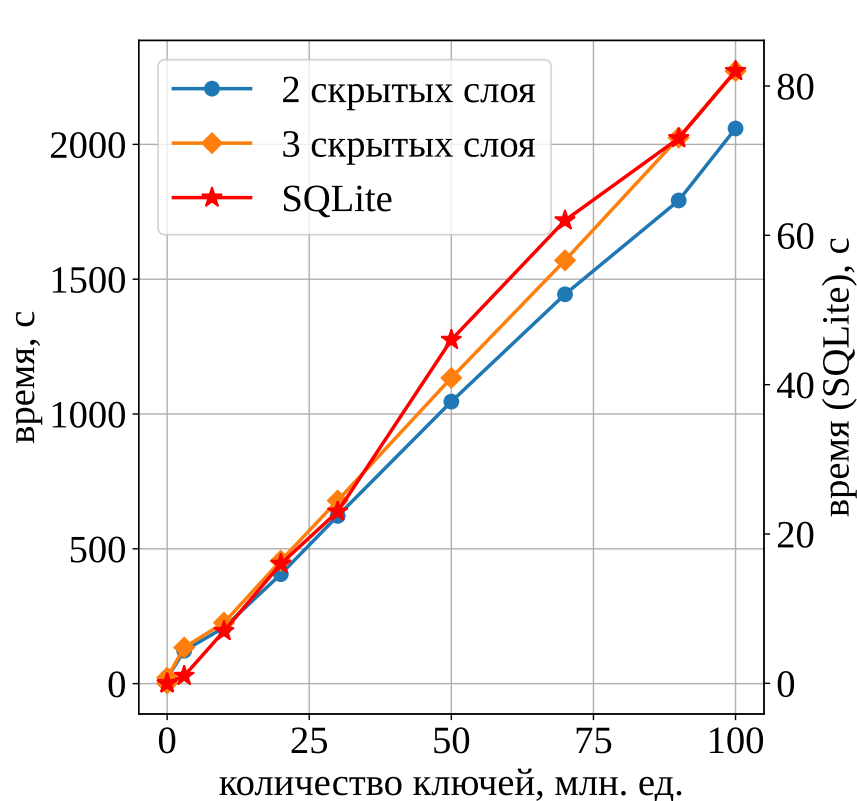
Применение метода: вставка



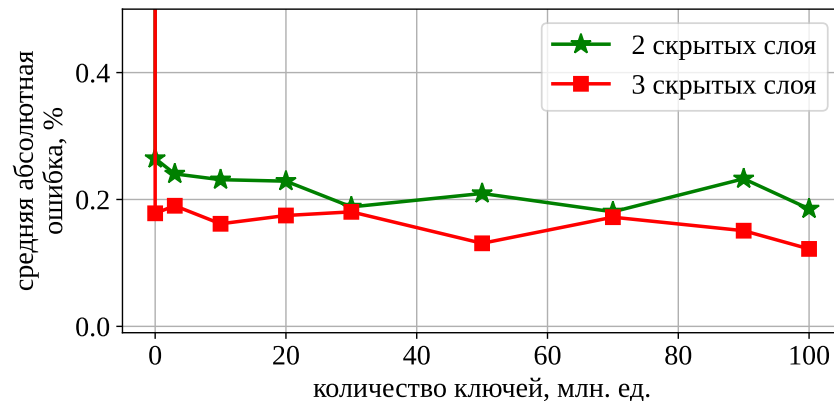
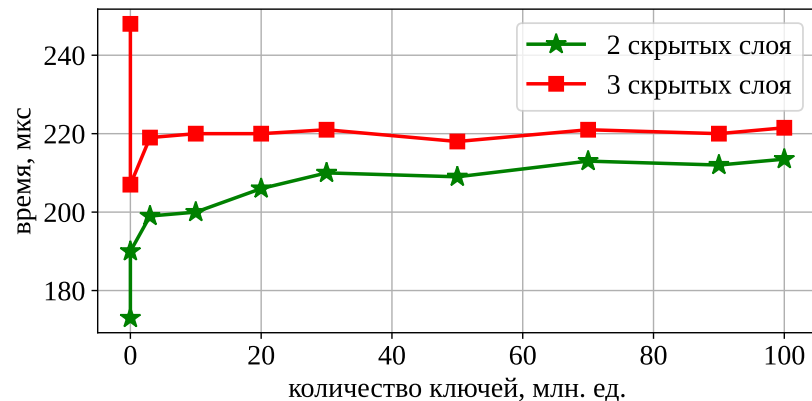
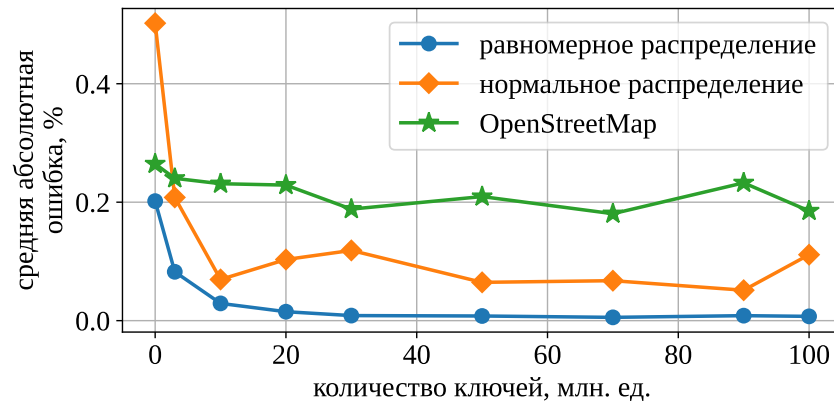
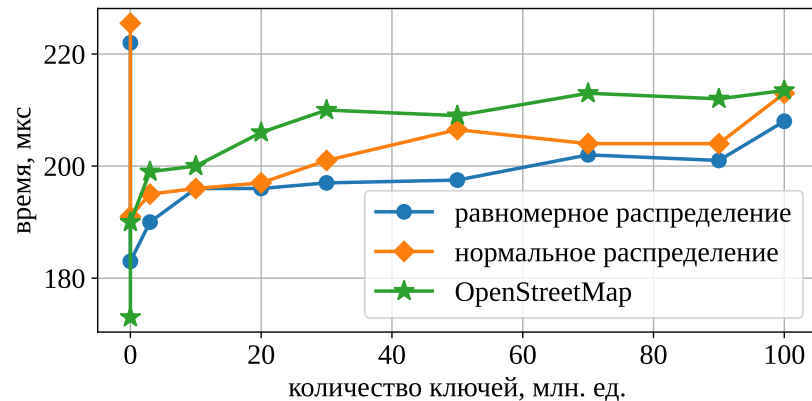
Структура программного обеспечения



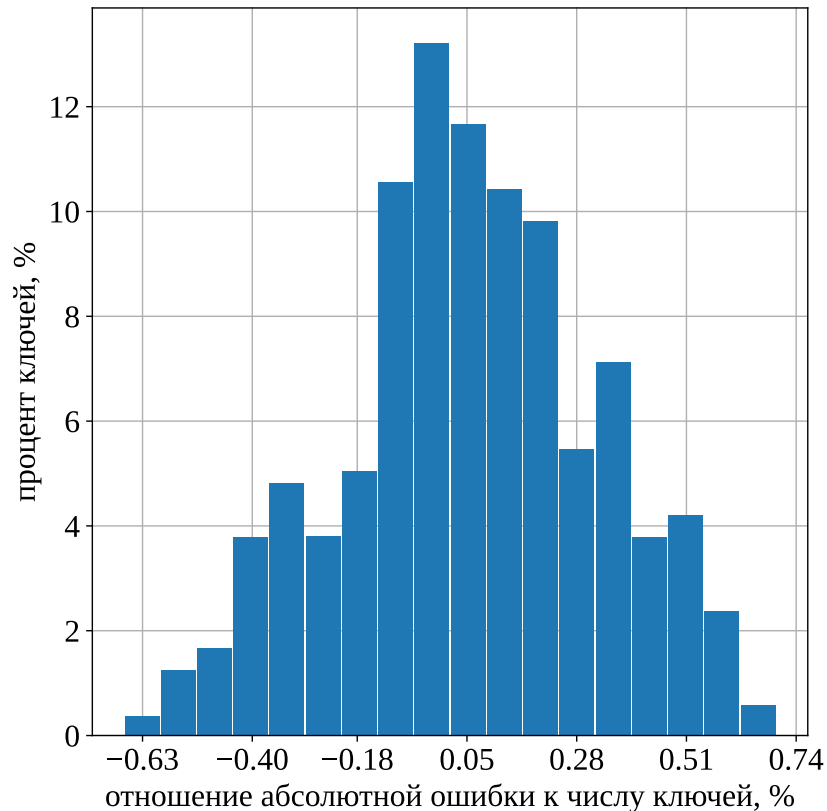
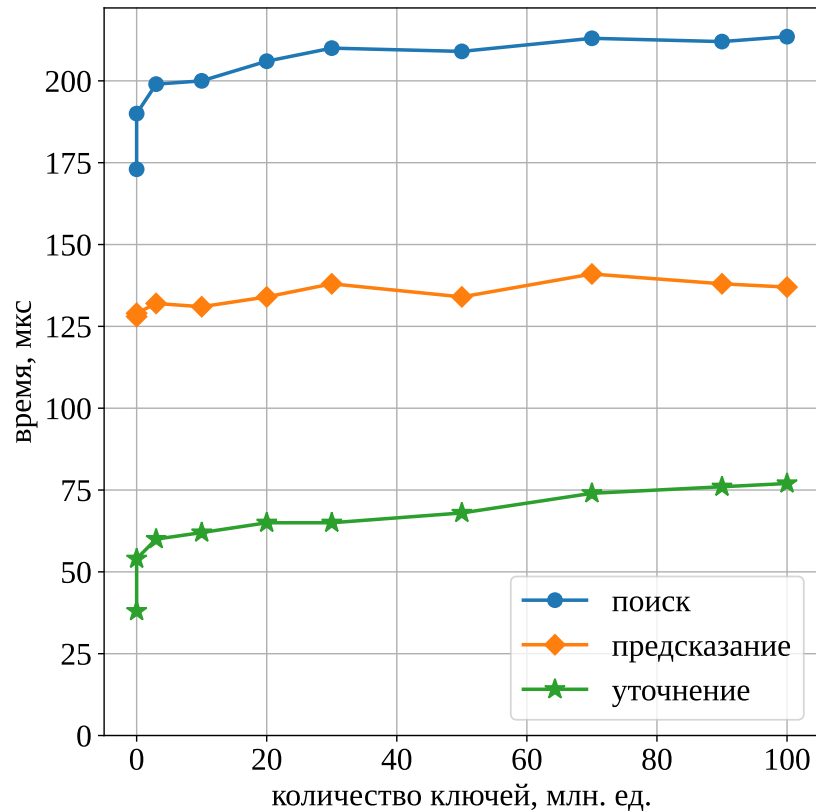
Зависимость времени построения и размера индекса от количества ключей



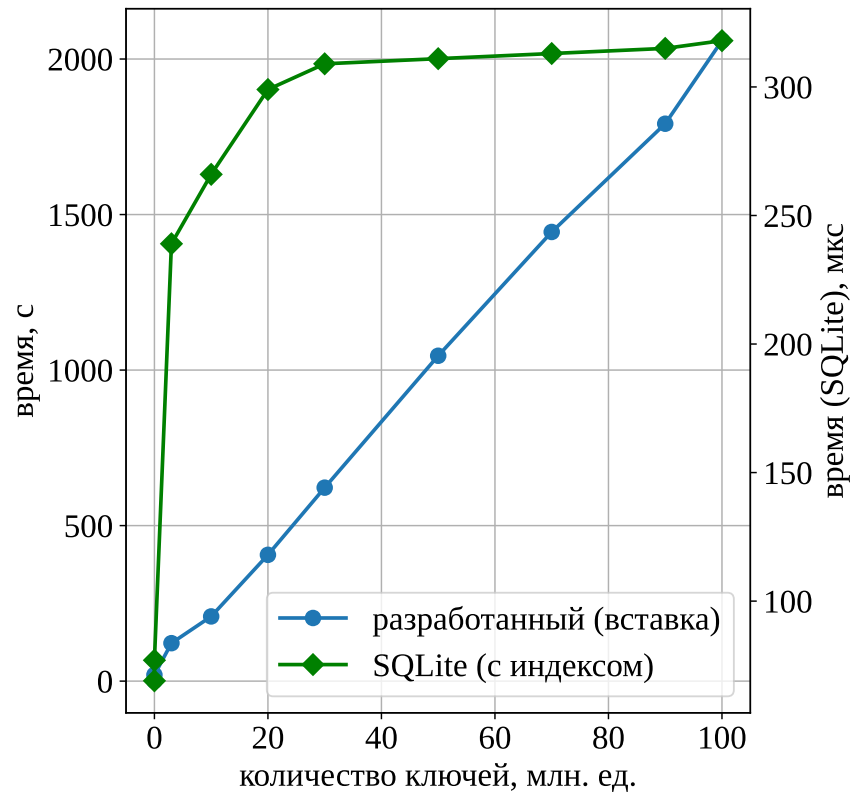
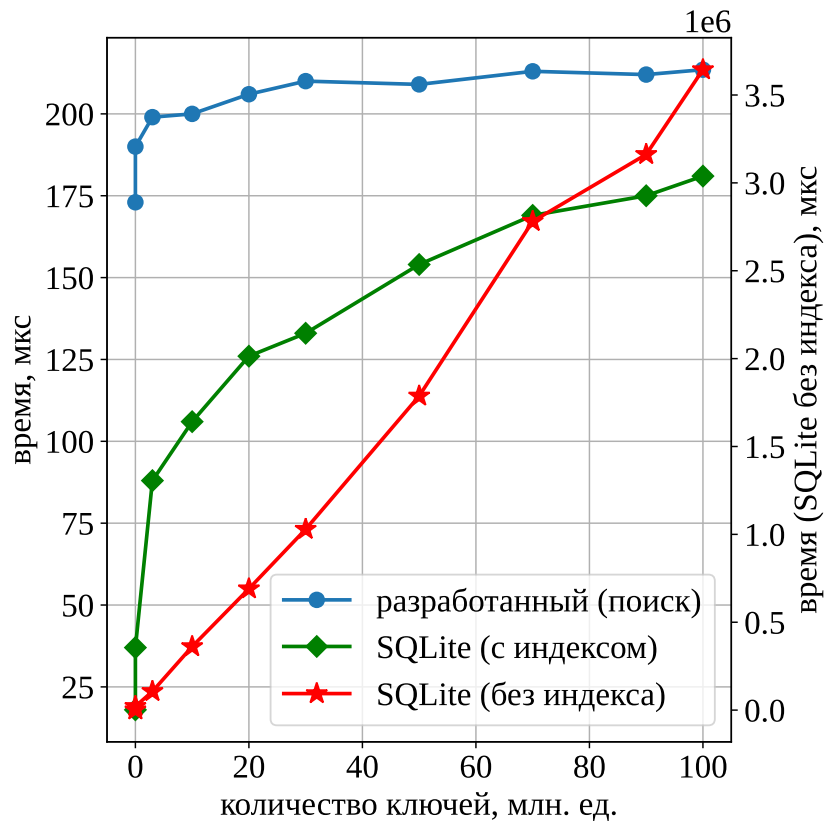
Зависимость времени поиска и средней абсолютной ошибки от количества ключей



Зависимость времени этапов поиска от количества ключей и распределение средней абсолютной ошибки



Зависимость времени поиска и вставки от количества ключей в сравнении с классическим индексом



Заключение

Разработан и реализован метод построения поисковых индексов в реляционной базе данных на основе глубоких нейронных сетей для чего:

- проанализированы методы построения индексов;
- разработан метод построения индексов в реляционной базе данных на основе глубоких нейронных сетей;
- разработано программное обеспечение, реализующее данный метод;
- проведено исследование операций поиска и вставки с использованием индекса, построенного разработанным методом.

Дальнейшее развитие

1. Оптимизация алгоритма вставки с учетом распределения ключей.
2. Добавление возможности построения индекса по ключам других типов данных.
3. Построение многомерных обученных индексов.