



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)

ФАКУЛЬТЕТ \_\_\_\_\_ «Информатика и системы управления»

КАФЕДРА \_\_\_\_\_ «Программное обеспечение ЭВМ и информационные технологии»

# РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

## К К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ

### НА ТЕМУ:

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

Студент:	<u>ИУ7-83Б</u> (группа)	_____ (подпись, дата)	<u>М. Д. Маслова</u> (И. О. Фамилия)
Руководитель:		_____ (подпись, дата)	<u>А. А. Оленев</u> (И. О. Фамилия)
Нормоконтролер:		_____ (подпись, дата)	_____ (И. О. Фамилия)

2023 г.

# **1 Конструкторская часть**

## **1.1 Требования и ограничения метода**

Метод построения поисковых индексов в реляционной базе данных на основе глубоких нейронных сетей (далее – метод построения индексов) должен:

1. получать из таблицы реляционной базы данных набор ключей и набор соответствующих указателей на записи в индексируемой таблице или иных значений, выполняющих роль указателей;
2. выполнять предварительную обработку полученных наборов, такую, как их совместную сортировку по значениям ключей, получение позиций ключей в отсортированном виде и нормализацию ключей и позиций;
3. обучать модель нейронной сети на подготовленном наборе ключей и позиций;
4. сохранять параметры обученной модели для каждой таблицы с целью возможности выполнять запросы поиска без переобучения;
5. обеспечивать поиск записи (диапазона записей) таблицы по ключу (диапазону ключей) с использованием обученной модели;
6. обеспечивать корректность операции поиска после вставки/удаления новых записей путем переобучения модели;

На разрабатываемый метод накладываются следующие ограничения:

- в качестве ключей на вход принимаются целые числа для исключения решения дополнительной задачи преобразования входных данных;
- ключи во входном наборе уникальны.

## **1.2 Особенности метода построения индекса**

### **1.2.1 Общее описание метода построения индекса**

Основные этапы метода построения индекса приведены на функциональной декомпозиции метода на рисунке 1.1.

На вход методу подается набор уникальных целочисленных ключей, которые перед обучением модели глубокой нейронной сети проходят предварительную обработку по определенным правилам, описанным далее. Отдельным этапом выделено получение значений функций распределения для каждого

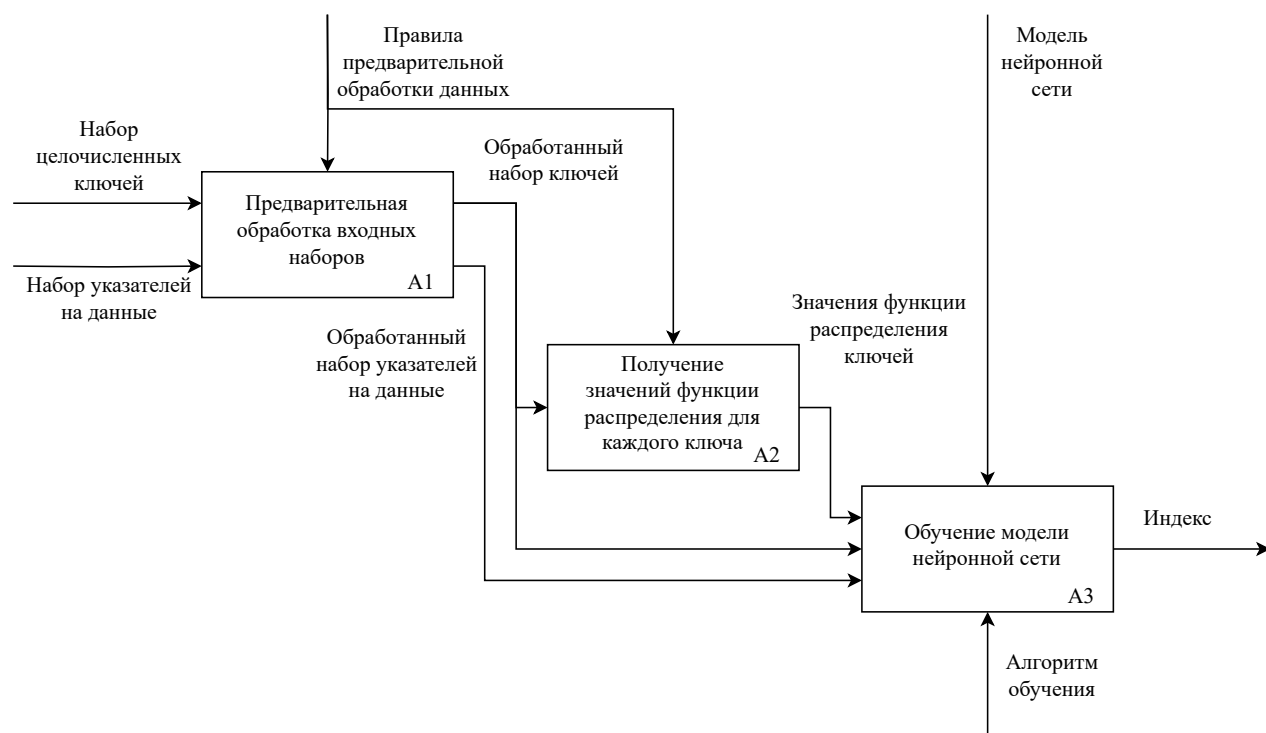


Рисунок 1.1 – Функциональная схема метода построения индекса

ключа, относящееся к предварительной обработке, но представляющее собой ее ключевой момент. Полученные после первых двух этапов обработанные ключи и соответствующие значения функций используются для обучения модели глубокой нейронной сети в качестве признаков и меток соответственно.

Ключевым моментом метода является представление в отсортированном (по ключам) виде наборов ключей и набора соответствующих указателей на данные. Именно отсортированный вид позволяет использовать закономерность распределения ключей по позициям для обучения модели, предсказывать позиции ключей и уточнять их.

Результатом работы метода является структура данных, представляющая собой индекс на основе глубокой нейронной сети и имеющая следующие поля:

- отсортированный массив ключей, поданных на вход;
- отсортированный по значениям ключей массив указателей на данные, соответствующие ключам;
- модель обученной глубокой нейронной сети, с помощью которой будет предсказываться положение ключа в отсортированном массиве;
- средняя и максимальная абсолютные ошибки предсказания позиции ключа, для ее уточнения и возврата верного указателя на данные.

Краткое описание индекса, являющегося результатом работы метода, как структуры данных представлено на рисунке 1.2.

Индекс	
- model	: модель нейронной сети
- keys	: массив целых чисел
- data	: массив указателей
- max_err	: целое число
- mean_err	: целое число

Рисунок 1.2 – Индекс как структура данных

Подробное описание каждого этапа приведено в следующих пунктах данного подраздела.

### 1.2.2 Предварительная обработка данных

Разрабатываемый метод построения индекса предполагает предварительную обработку набора целочисленных ключей, схема алгоритма которой представлена на рисунке 1.3.

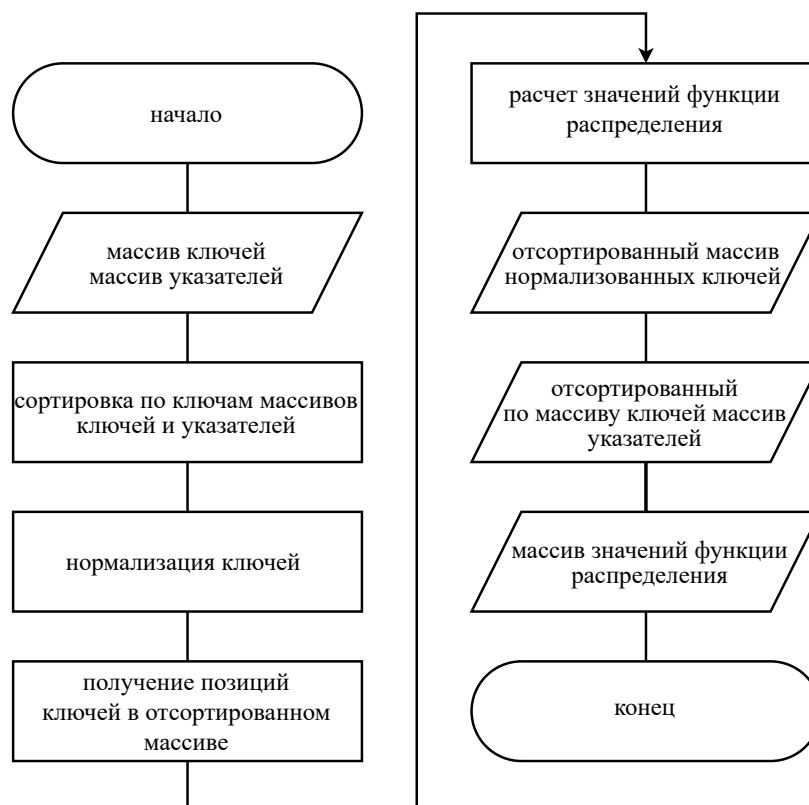


Рисунок 1.3 – Схема алгоритма предварительной обработки данных

На вход подаются согласованные массивы ключей и указателей на данные, то есть считается, что ключ, стоящем на первой позиции в массиве ключей, идентифицирует данные по указателю, стоящем на первой позиции в массиве указателей; ключ, стоящий на второй, — указатель, стоящий на второй, и так далее. С учетом этого происходит согласованная сортировка двух массивов по значениям ключей.

Далее для последующего обучения модели глубокой нейронной сети производится нормализация ключей, выступающих в качестве входных данных сети, в диапазон  $[0, 1]$ , для чего используется метод минимакс-нормализации, при котором нормализованное значение вычисляется по формуле:

$$x_{\text{норм}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}, \quad (1.1)$$

где  $x_{\text{норм}}$  — нормализованное значение ключа;

$x$  — натуральное значение ключа;

$x_{\min}$ ,  $x_{\max}$  — минимальное и максимальное возможное значение ключа в наборе соответственно.

Далее полученный набор ключей размечается путем вычисления для каждого ключа  $K$  значения функции распределения  $F$  по его позиции  $P$  в отсортированном массиве и количества индексируемых ключей  $N$  с помощью формулы:

$$F(K) = \frac{P}{N} \quad (1.2)$$

На этом предварительная обработка завершается и полученные отсортированные массивы ключей и указателей, а также соответствующие значения функции распределения передаются в качестве входных данных на этап обучения модели глубокой нейронной сети.

Полное описание алгоритма предварительной обработки представлено на листинге 1.1.

## Листинг 1.1 – Предварительная обработка данных

### Вход:

*keys* : массив целочисленных ключей;  
*data* : массив указателей на данные, соответствующие ключам;  
*N* : длина массивов.

### Выход:

*keys* : отсортированный массив нормализованных ключей;  
*data* : отсортированный по массиву ключей массив указателей;  
*cdf* : массив значений функции распределения.

```
1 begin
2   сортировать keys и data по keys;
   ▷ здесь и далее под операцией к вектору (массиву) и числу понимается
   ▷ применение данной операции с данным числом к каждому элементу вектора
3    $keys \leftarrow \frac{keys - keys[0]}{keys[N-1] - keys[0]}$ ;
4    $positions \leftarrow [0, 1, \dots, N - 1]$ ;
5    $cdf \leftarrow \frac{positions}{N-1}$ ;
6   return keys, data, cdf;
7 end
```

## 1.2.3 Разработка архитектуры глубокой нейронной сети

...

Получаемая позиция требует уточнения, происходящего за счет получаемого в результате обучения модели максимального отклонения от истинного расположения. (Алгоритм уточнения???)

## 1.3 Функциональная схема использования индекса

Основной операцией, выполняемой с помощью индекса, является поиск, функциональная схема выполнения которого представлена на рисунках 1.4-1.5.

Для реализации вставки происходит добавление новых значений ключа и указателя в существующие массивы, и повторяется алгоритм построения индекса (??? не сначала, а со значений параметров уже обученной модели ???).

## 1.4 Данные для обучения и тестирования индекса

Так как в основе индекса на основе глубоких нейронных сетей лежит аппроксимация функции распределения ключей, работу разработанного метода.

- Равномерное
- Нормальное

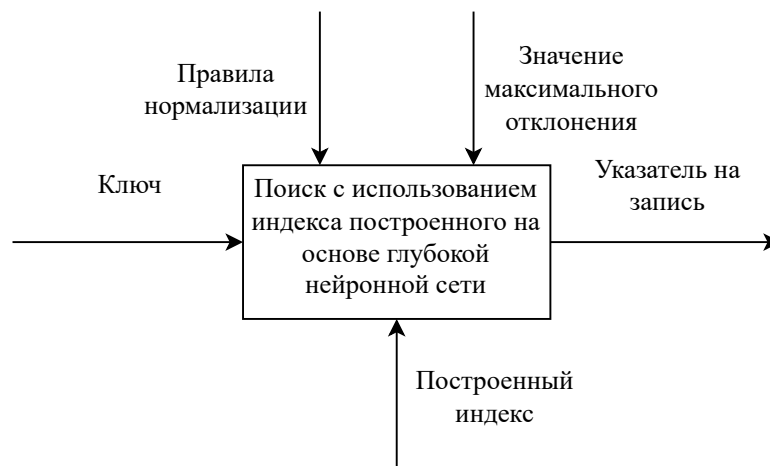


Рисунок 1.4 – Функциональная схема нулевого уровня поиска

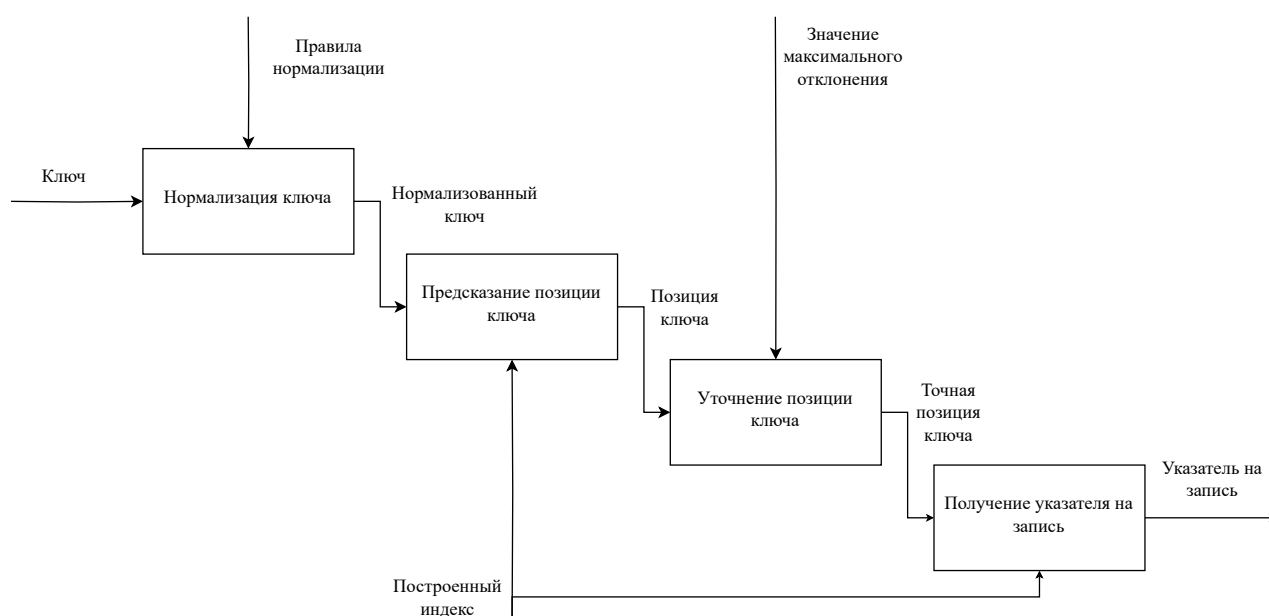


Рисунок 1.5 – Функциональная схема первого уровня поиска

- Экспоненциальное
- Реальные данные (osm, face, wiki ???)