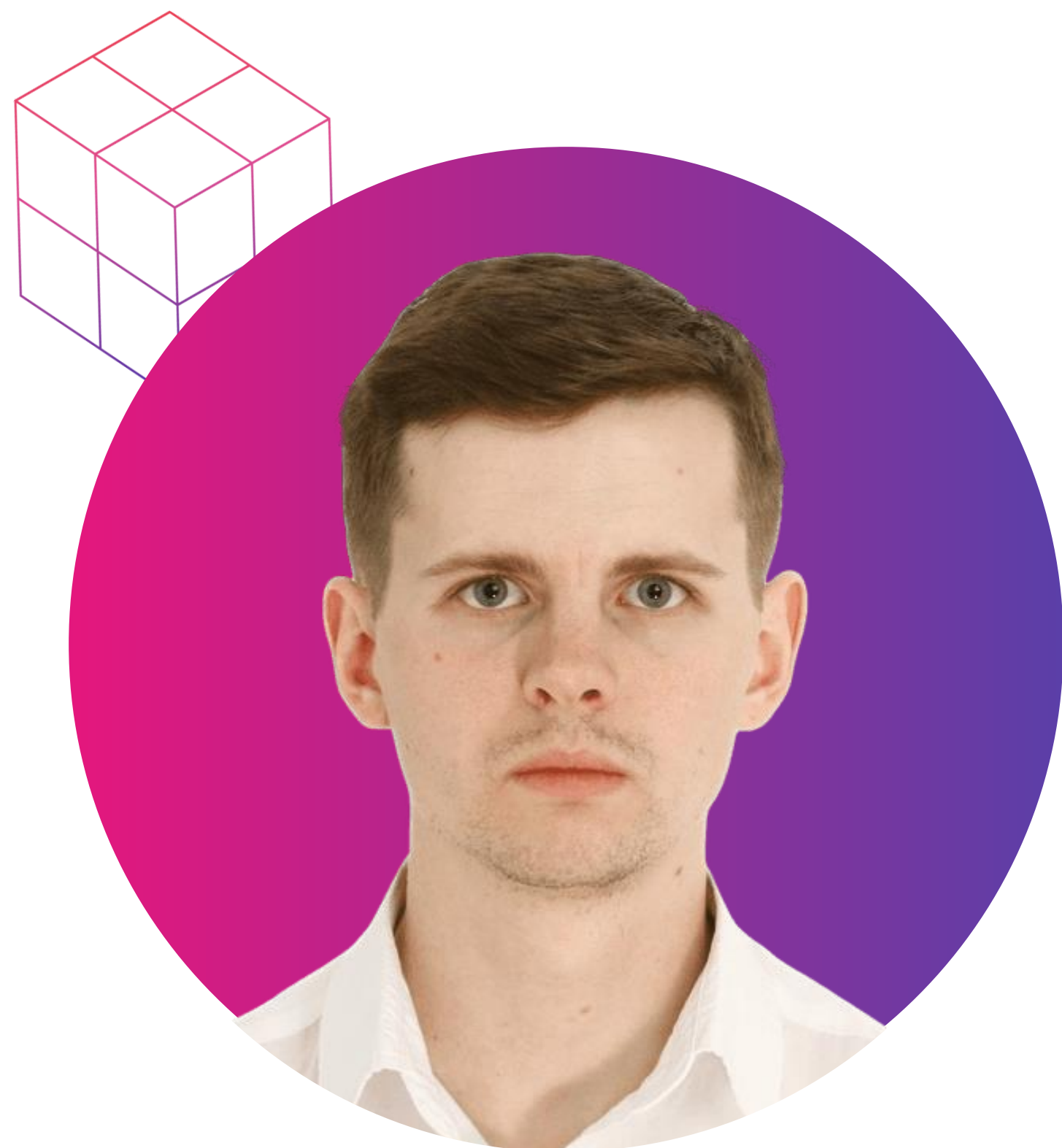


# Нейросетевой подход к решению задачи кредитного скоринга

**Фирстов Алексей**

**Альфа-Банк**





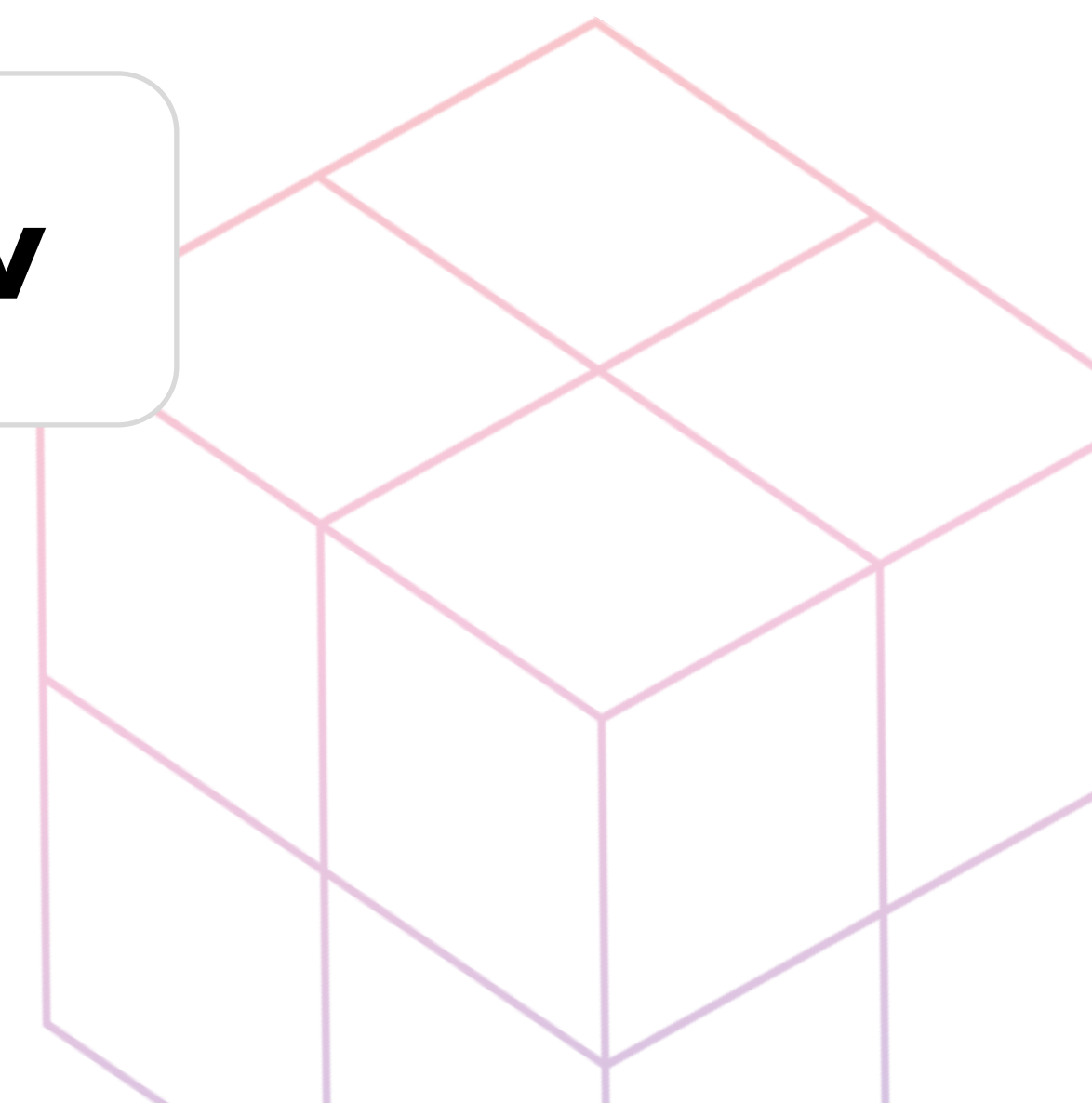
# Фирстов Алексей

Data Scientist

Центр продвинутой аналитики  
Лаборатория машинного обучения  
Альфа-Банк



**@Aleksey\_Firstov**



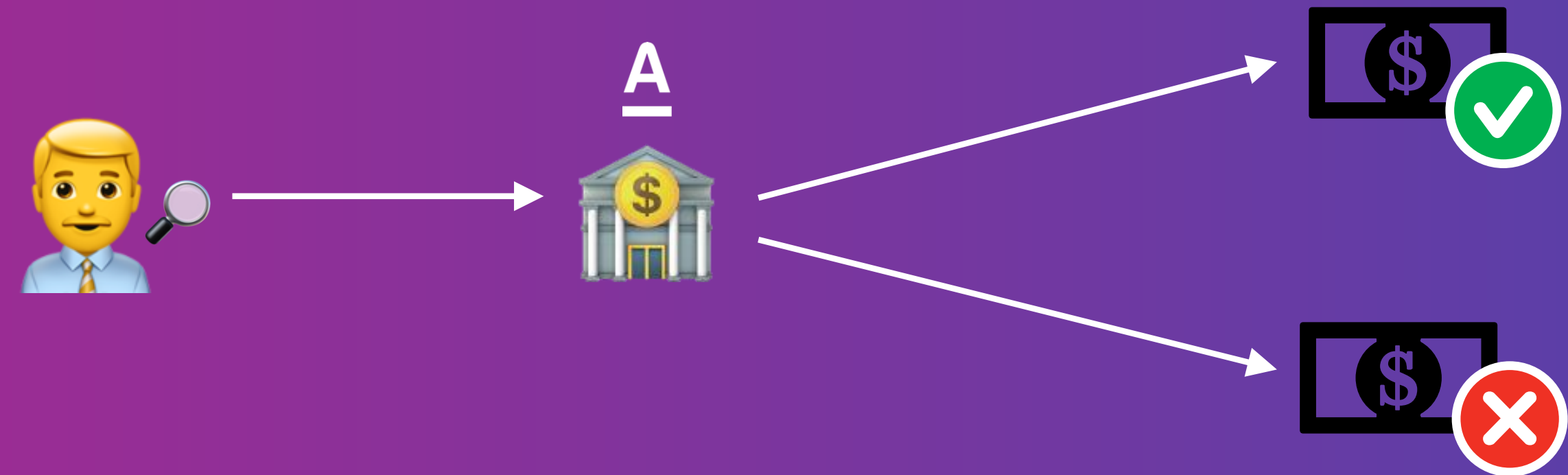
# О ЧЁМ БУДЕМ РАЗГОВАРИВАТЬ?

- 1. Постановка задачи и метрики**
2. Кредитный скоринг в Альфа-Банке
3. Источники данных и подходы к моделированию
4. Модели на последовательных данных
5. Обработка текстовых признаков
6. Техники обучения
7. Основные результаты
8. Продакшн и планы на будущее

# Задача кредитного скоринга

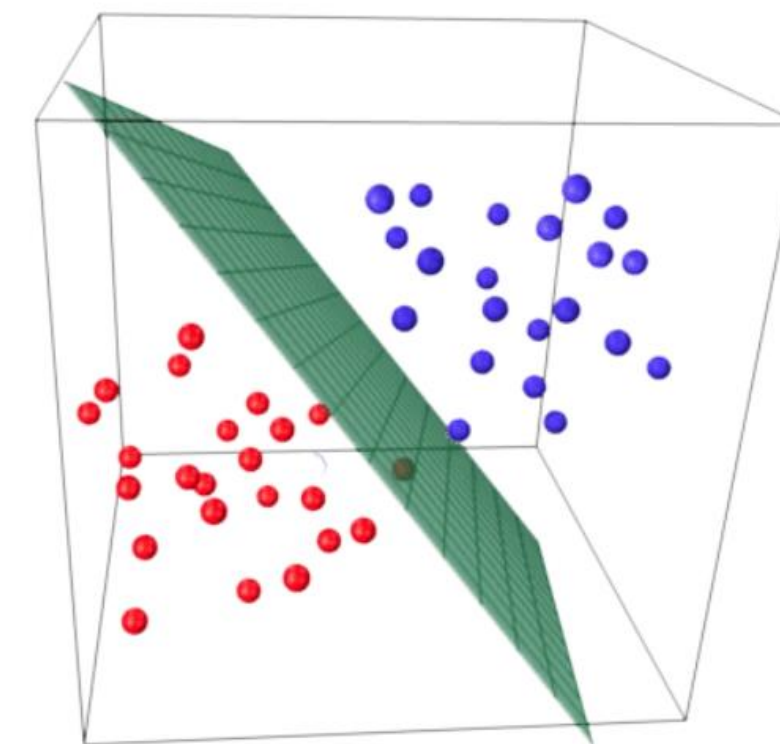
## Задача:

Определить, уйдёт клиент в дефолт\* по кредитному продукту или нет



Решаем задачу бинарной классификации

- Надёжный заемщик
- Не надёжный заемщик



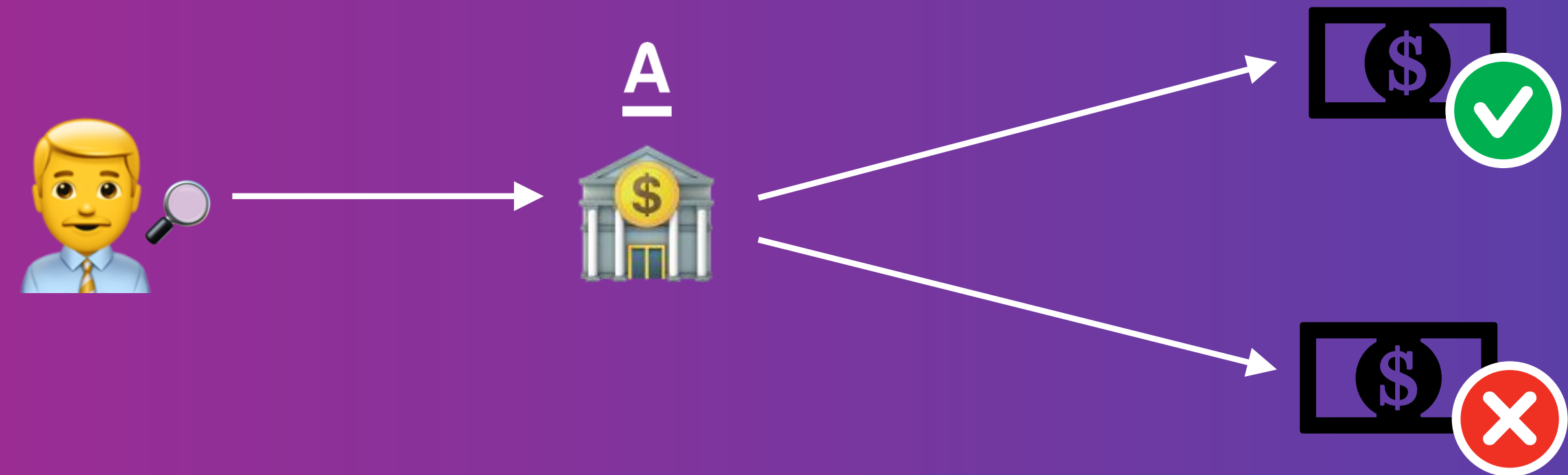
«Длинное» определение дефолта:

Просрочка на 90+ дней или реструктуризация по прошествии 12 месяцев со дня одобрения продукта

# Задача кредитного скоринга

## Задача:

Определить, уйдёт клиент в дефолт\* по кредитному продукту или нет



$fx$



Клиент	Вероятность невозврата кредита
Mike	0,78
Jack	0,51
Larry	0,13
Kate	0,06
William	0,03

Ненадежные

Надежные

«Длинное» определение дефолта:

Просрочка на 90+ дней или реструктуризация по прошествии 12 месяцев со дня одобрения продукта

**Коэффициент Джини** — метрика, показывающая качество ранжирования в задаче бинарной классификации

$$Gini = (ROCAUC - 0.5) * 2 * 100$$

Границы изменения — от 0 до 100 пунктов

Идеальная модель 

Клиент	Вероятность невозврата кредита	Вернул ли кредит?
Mike	0,78	Нет
Jack	0,51	Нет
Larry	0,13	Да
Kate	0,06	Да

Очень плохая модель 

Клиент	Вероятность невозврата кредита	Вернул ли кредит?
Mike	0,78	Да
Jack	0,51	Нет
Larry	0,13	Да
Kate	0,06	Нет

# Целевая метрика

**Коэффициент Джини** — метрика, показывающая качество ранжирования в задаче бинарной классификации

$$Gini = (ROCAUC - 0.5) * 2 * 100$$

**Границы изменения — от 0 до 100 пунктов**

0 – случайное ранжирование

100 – идеальное ранжирование

Почему Джини,  
а не precision/recall/F1?

▶ Наши модели не принимают финальное решение о выдаче/невываде кредита

▶ Важен не 0/1 ответ модели, а скор, оценивающий кредитную надежность клиента

# О ЧЁМ БУДЕМ РАЗГОВАРИВАТЬ?

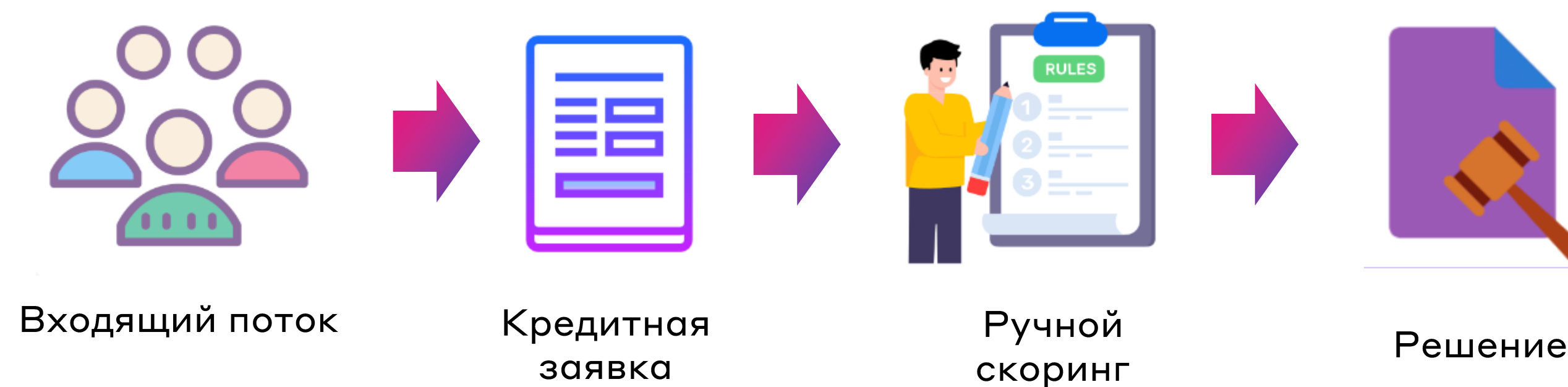
1. Постановка задачи и метрики
- 2. Кредитный скоринг в Альфа-Банке**
3. Источники данных и подходы к моделированию
4. Модели на последовательных данных
5. Обработка текстовых признаков
6. Техники обучения
7. Основные результаты
8. Продакшн и планы на будущее



# Эволюция кредитного скоринга в Альфа-Банке



## Принятие решение на основе бизнес-логики



# Эволюция кредитного скоринга в Альфа-Банке



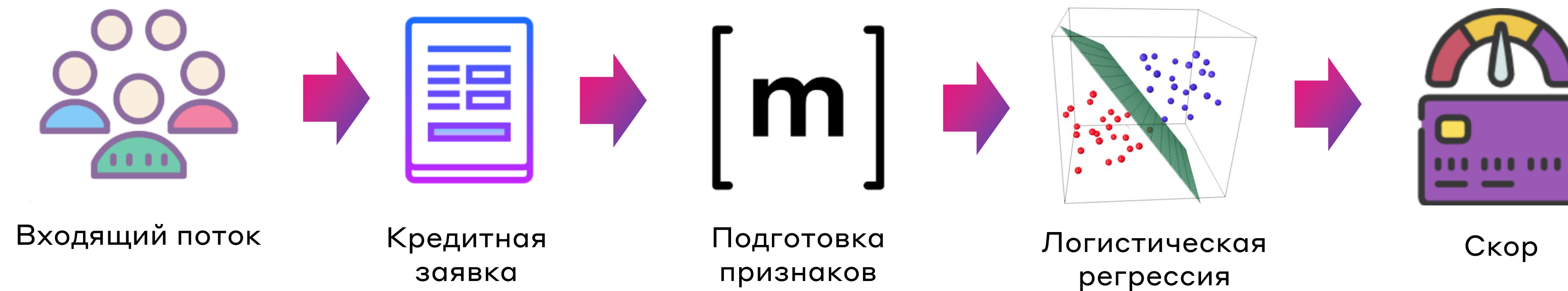
**Логистические регрессии** позволили подходить к каждому клиенту индивидуально

**Вызовы:**  
новый технологический стек

# Эволюция кредитного скоринга в Альфа-Банке



## Принятие решений на основе простых моделей



Иванов И.И.	[1 ... 30000]
⋮	⋮
Петров П.П.	[0 ... 80000]

# Эволюция кредитного скоринга в Альфа-Банке



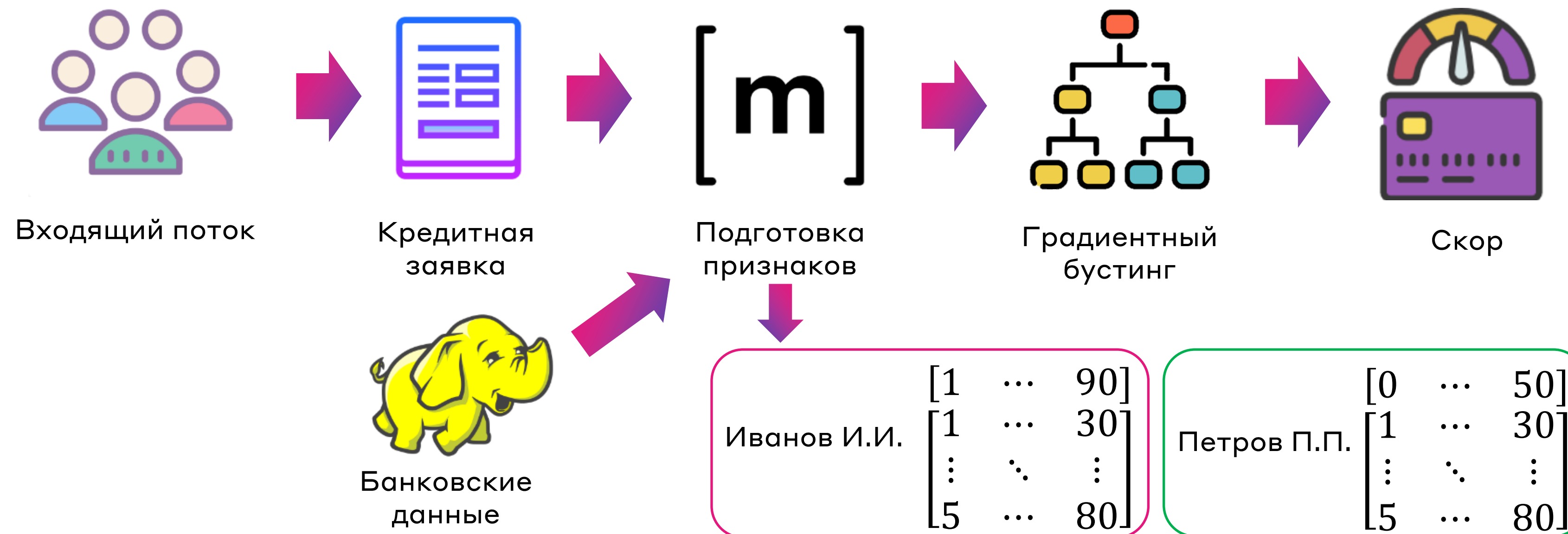
Переход на **градиентный бустинг** значительно повысил точность модельного прогноза

**Вызовы:**  
неинтерпретируемые модели; усложняются источники данных

# Эволюция кредитного скоринга в Альфа-Банке



## Принятие решений на основе сложных моделей



# Эволюция кредитного скоринга в Альфа-Банке

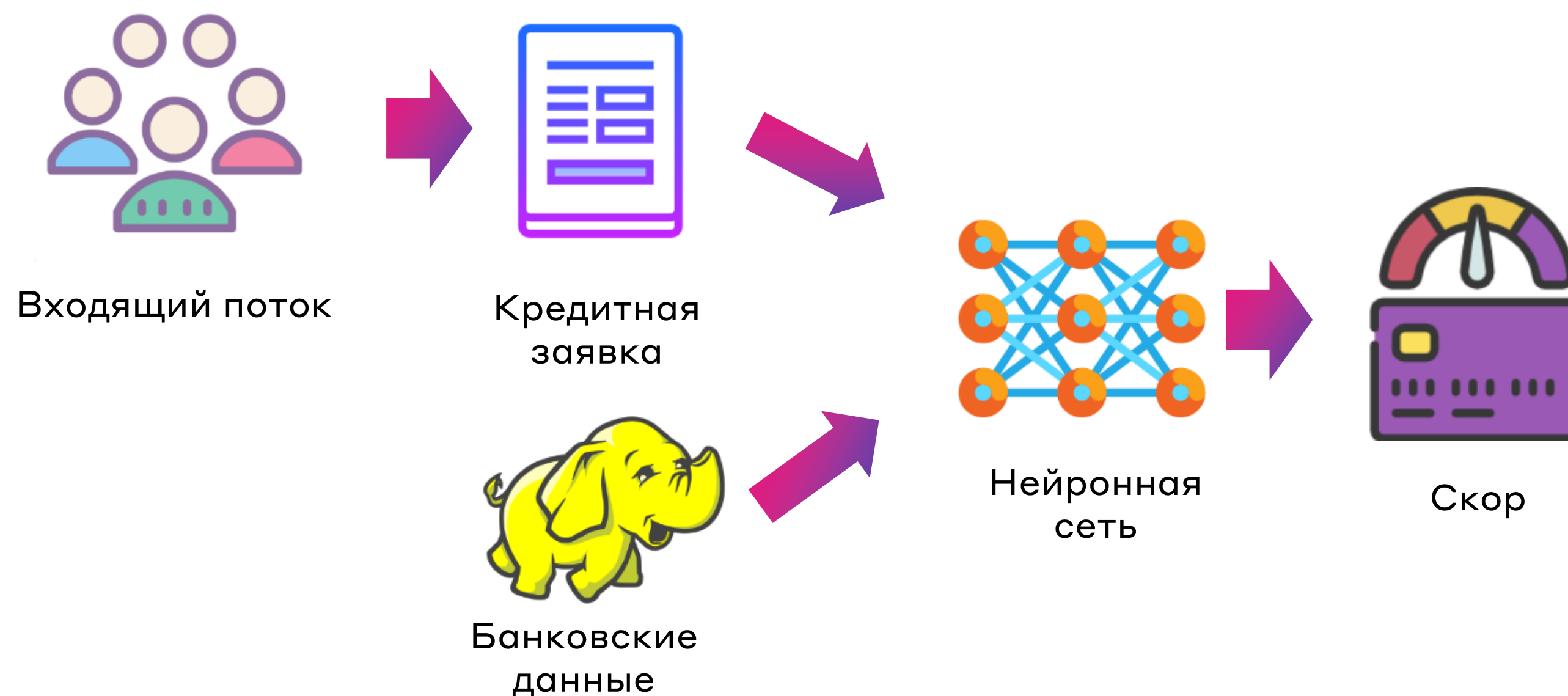


**Нейронные сети**  
впервые **значимо**  
**и стабильно**  
превзошли  
градиентный бустинг  
по точности прогноза

# Эволюция кредитного скоринга в Альфа-Банке



## Принятие решений на основе сложных моделей

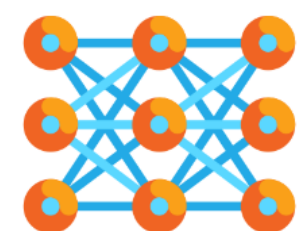


# Эволюция кредитного скоринга в Альфа-Банке

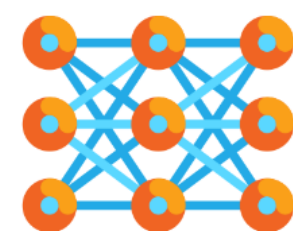


## Наши дни

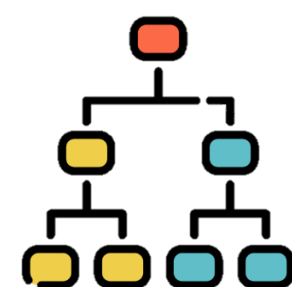
### Принятие решений на основе сложных моделей



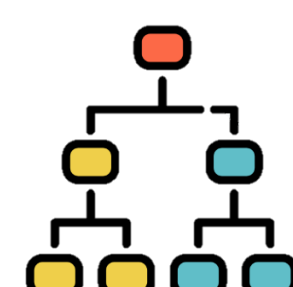
Нейронная сеть №1



Нейронная сеть №2



Градиентный бустинг №1



Градиентный бустинг №2



Бизнес-правила



Решение



# Как нейросети повлияли на бизнес и команду

- 01** Первыми внедрили нейросети в кредитный скоринг
- 02** Заработали миллиарды рублей чистой прибыли
- 03** Команда разработки выросла с 1 до 10 человек
- 04** Бизнес увидел потенциал нейросетей, команда активно расширяемся и штурмует новые направления

# О ЧЁМ БУДЕМ РАЗГОВАРИВАТЬ?

1. Постановка задачи и метрики
2. Кредитный скоринг в Альфа-Банке
- 3. Источники данных и подходы к моделированию**
4. Модели на последовательных данных
5. Обработка текстовых признаков
6. Техники обучения
7. Основные результаты
8. Продакшн и планы на будущее

# Банковские последовательные данные



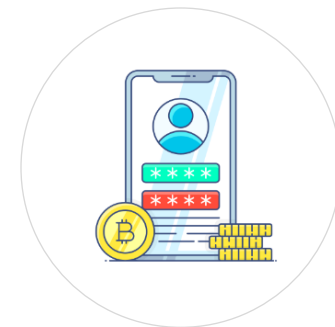
**Карточные транзакции**  
История транзакций клиента  
по кредитным / дебетовым картам

5 млрд



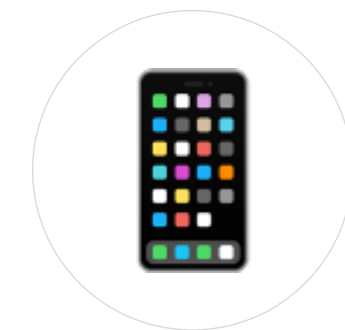
**Кредитная история\***  
Информация о прошлых кредитах  
клиента

50 млн



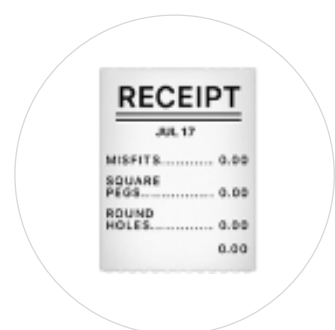
**Транзакции  
расчетного счета**  
История транзакций клиента  
по расчетному счету

5 млрд



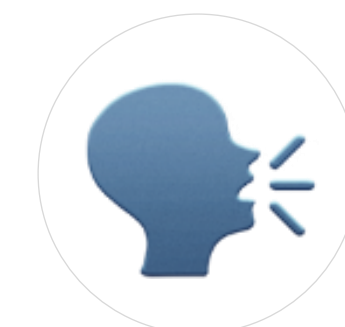
**Логи сайта  
и мобильного приложения**  
История взаимодействий клиента  
с веб-сайтом и мобильным  
приложением

>1млрд



**Чеки\***  
Детализация карточных  
транзакций

1 млрд



**История  
Коммуникаций с клиентом**  
Коммуникации банка с клиентов  
по различным продуктам

80 млн

# Подходы к моделированию последовательных данных

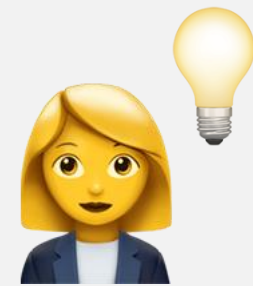
## Классический ml

Data Scientist вручную генерирует признаки для моделей классического ML

Последовательность



Генерация идей



Табличные признаки



Обучение классической модели



Предсказание



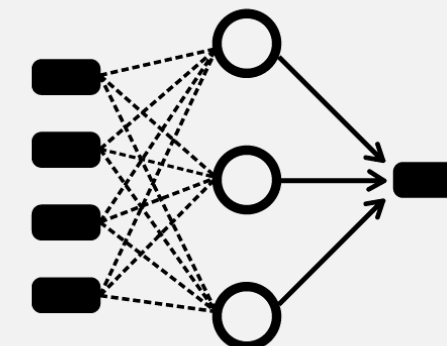
## Нейросети

Data Scientist подает на вход в нейросеть всю последовательность, нейросеть сама выявляет всю необходимую информацию в процессе обучения

Последовательность



Обучение нейронной сети



Предсказание

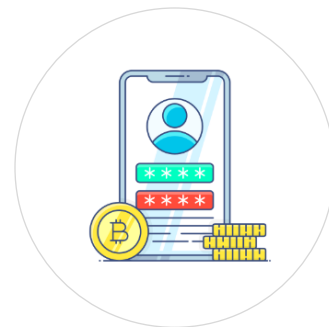


# Банковские последовательные данные



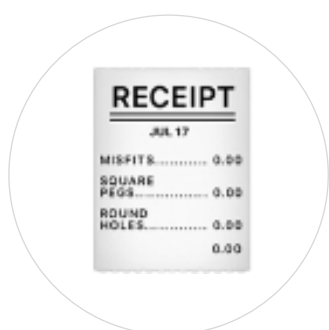
**Карточные транзакции**  
История транзакций клиента  
по кредитным / дебетовым картам

5 млрд



**Транзакции  
расчетного счета**  
История транзакций клиента  
по расчетному счету

5 млрд



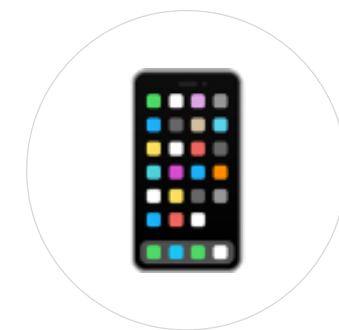
**Чеки\***  
Детализация карточных  
транзакций

1 млрд



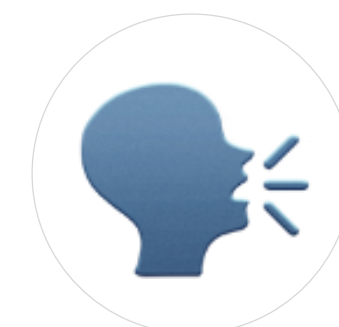
**Кредитная история\***  
Информация о прошлых кредитах  
клиента

50 млн



**Логи сайта  
и мобильного приложения**  
История взаимодействий клиента  
с веб-сайтом и мобильным  
приложением

>1млрд

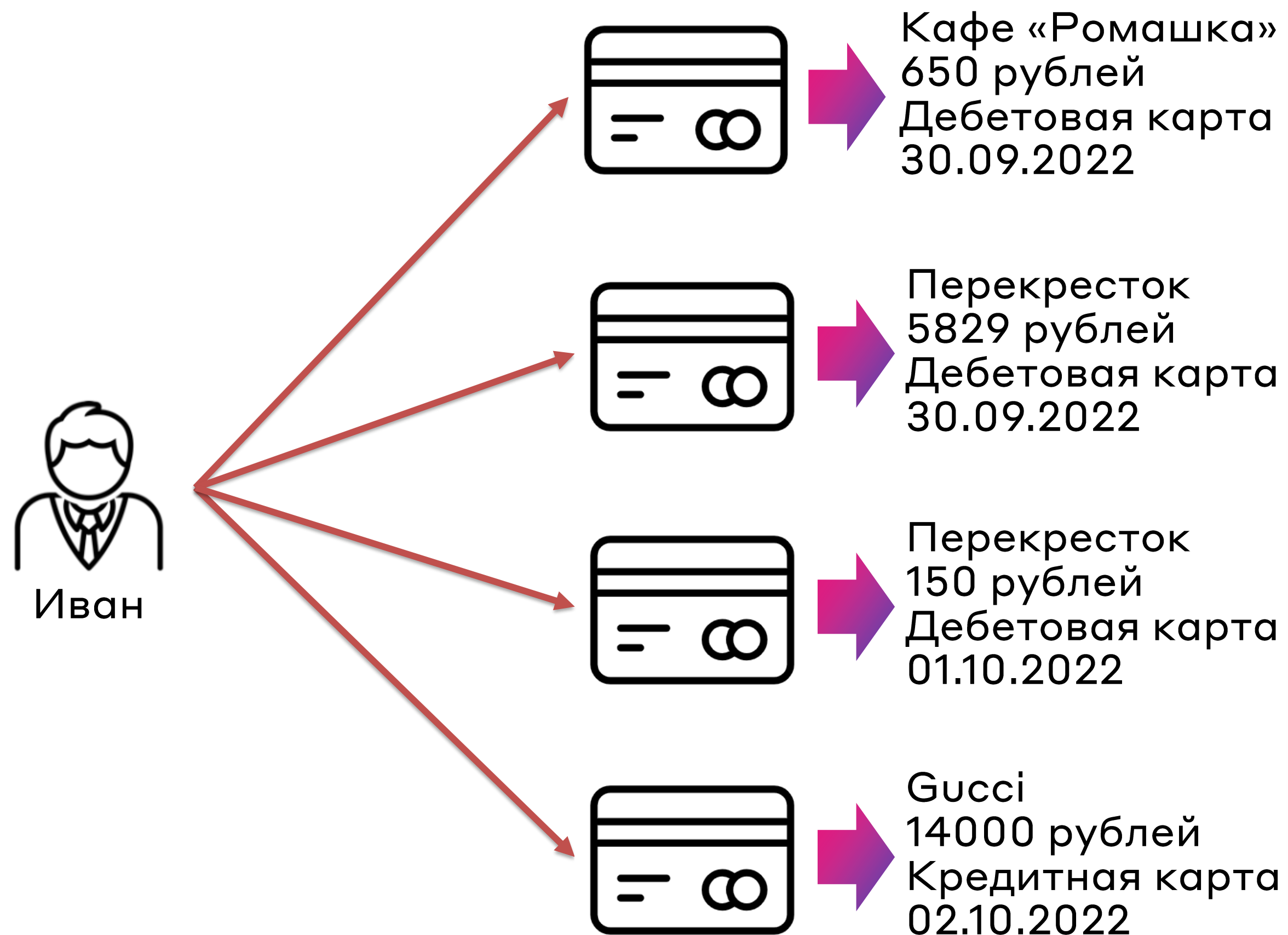


**История  
Коммуникаций с клиентом**  
Коммуникации банка с клиентов  
по различным продуктам

80 млн

\* При наличии действующего согласия со стороны клиента

# Карточные транзакции



## Формируем выборку для обучения

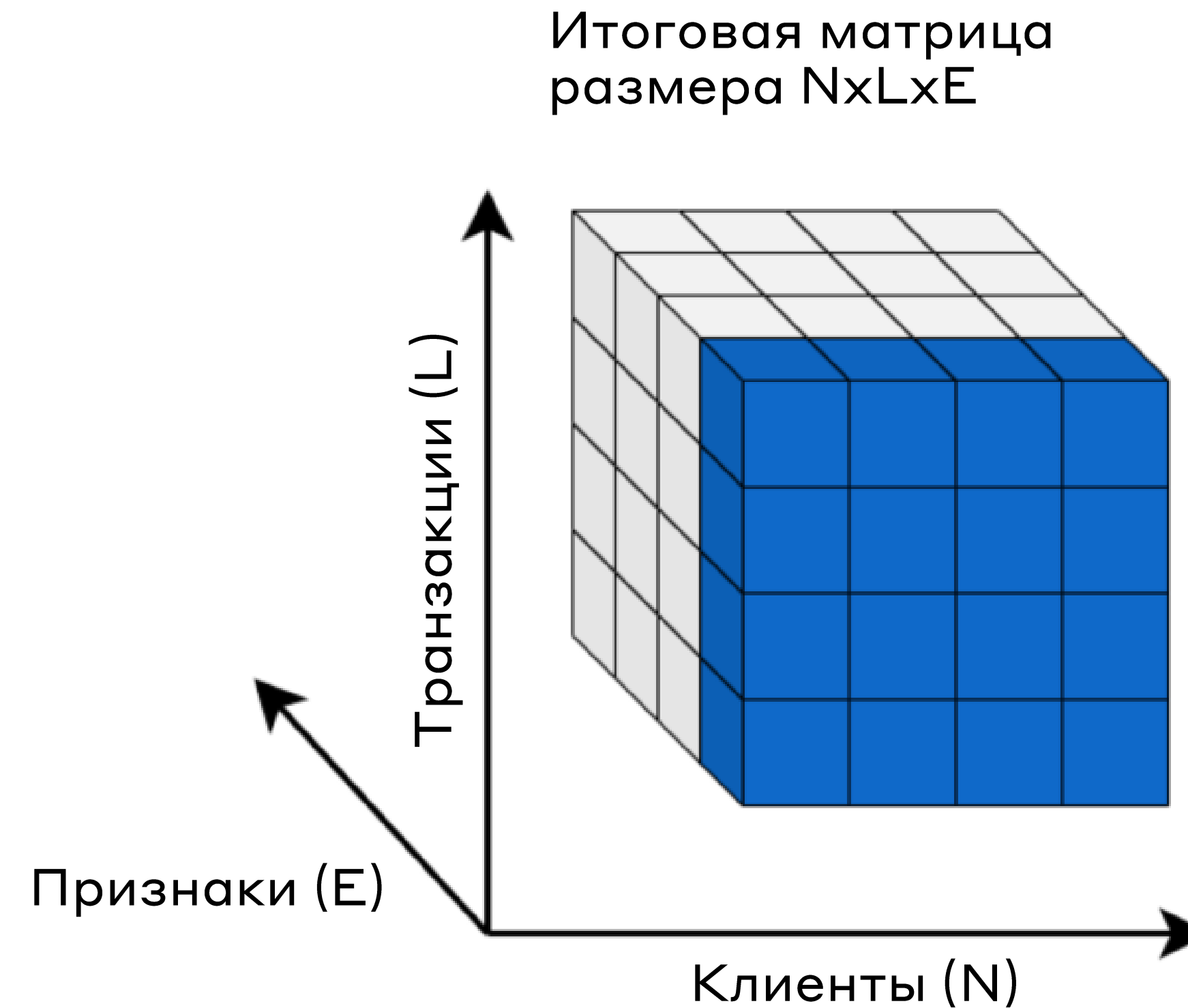
Кафе Ромашка	650	Дебетовая	30.09.2022
Перекресток	5829	Дебетовая	30.09.2022
Перекресток	150	Дебетовая	01.10.2022
Gucci	14000	Кредитовая	02.10.2022



Матрица двумерная, но клиент пока что-то только один, добавим больше клиентов

<sup>1</sup> L — количество транзакций  
<sup>2</sup> E — количество транзакционных признаков

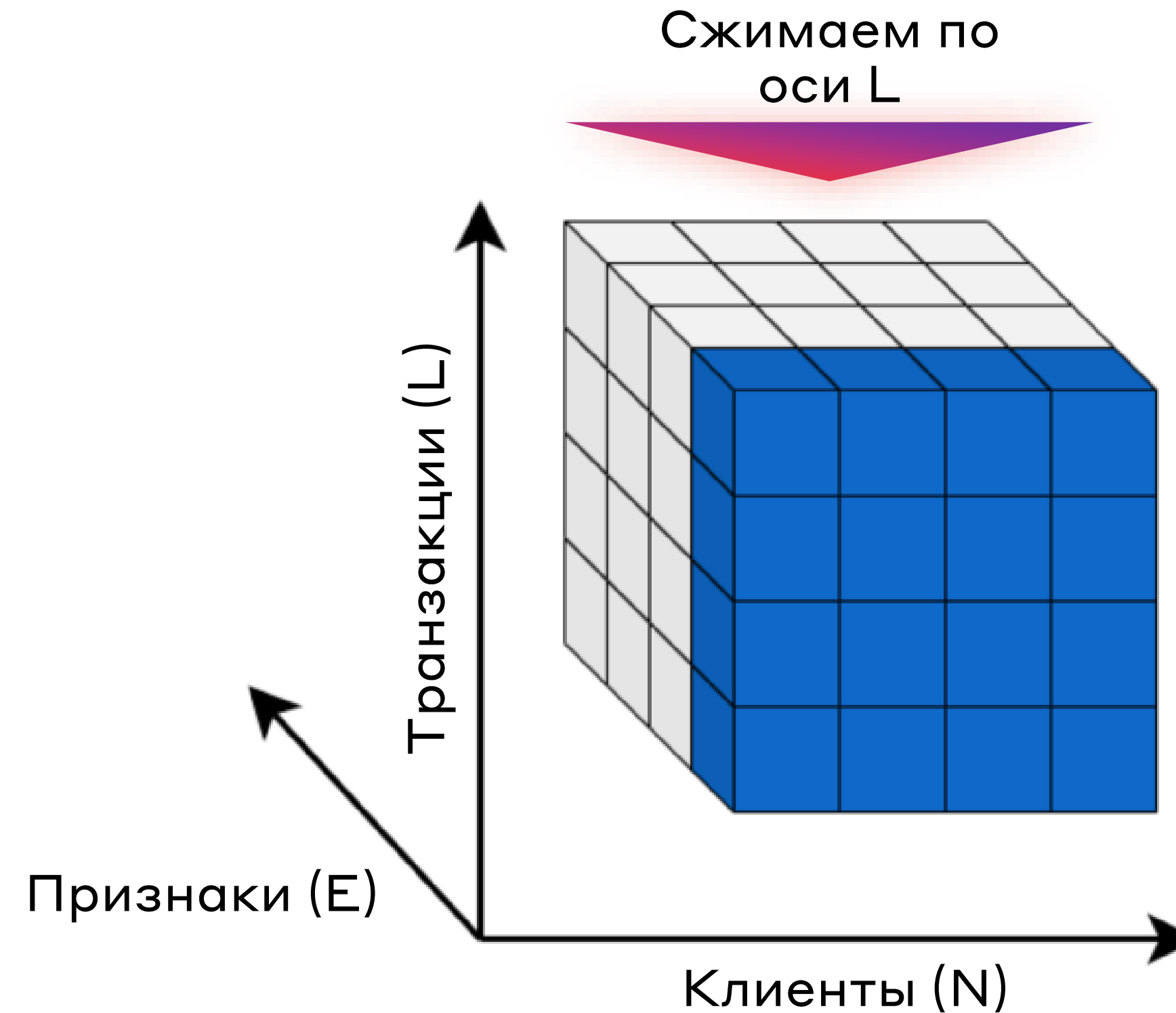
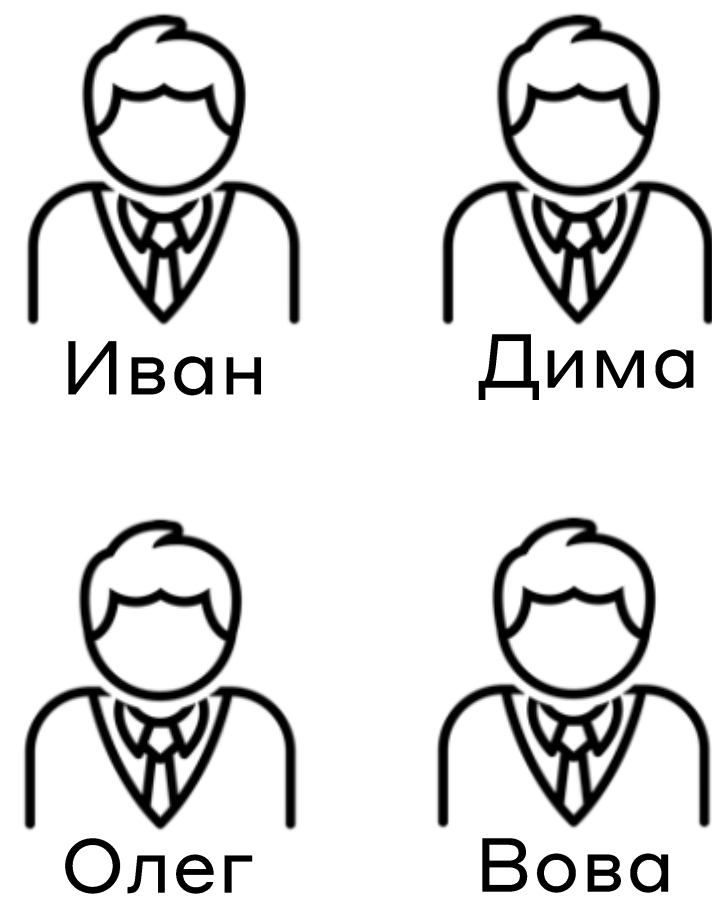
# Матрица карточных транзакций



## Проблема:

Классические модели машинного обучения принимают на вход матрицы  $N \times E$

# Матрица карточных транзакций

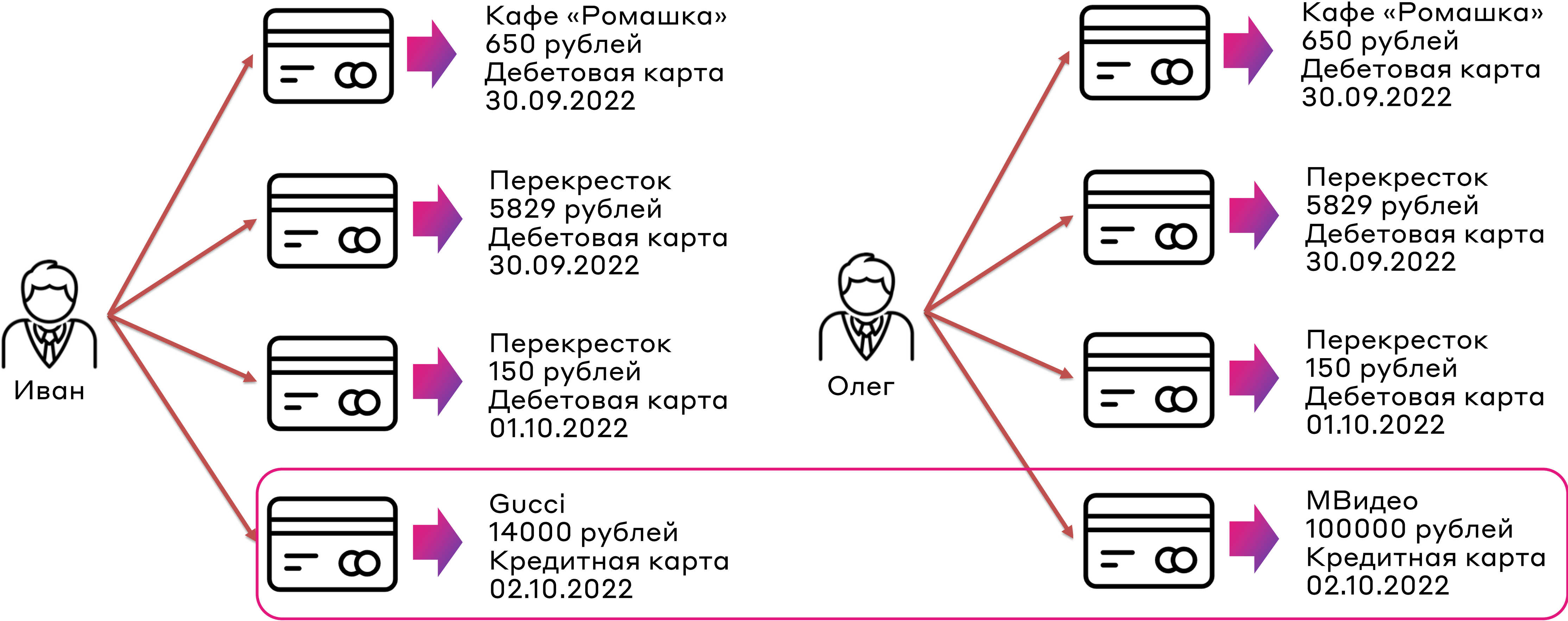


**Решение:**

Агрегация признаков по оси L



# Карточные транзакции



Что произойдет при агрегации?

# Агрегация транзакций

**01** Самый популярный магазин

**03** Самая популярная карта

**02** Средний чек

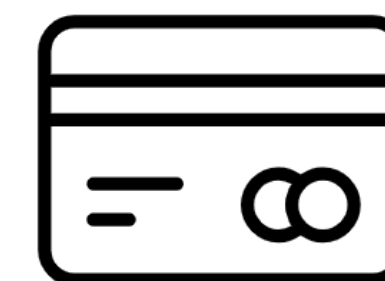
**04** Среднее количество транзакций



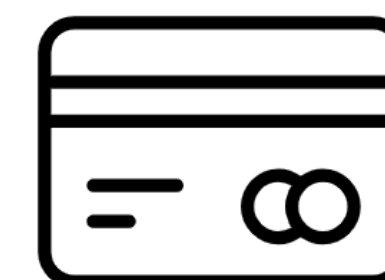
Перекресток 5157 Дебетовая 1,3



Перекресток 26657 Дебетовая 1,3



Gucci  
14000 рублей  
Кредитная карта  
02.10.2022



МВидео  
100000 рублей  
Кредитная карта  
02.10.2022

# Агрегация транзакций

**01** Самый популярный магазин

**03** Самая популярная карта

**02** Средний чек

**04** Среднее количество транзакций



Иван

Перекресток	5157	Дебетовая	1,3
-------------	------	-----------	-----



Олег

Перекресток	26657	Дебетовая	1,3
-------------	-------	-----------	-----



Теряем много важной информации  
Необходимо генерировать признаки

## Новые признаки

Численные признаки:

1. Средний чек
2. Медианный чек
3. Максимальный чек
4. И т.д.

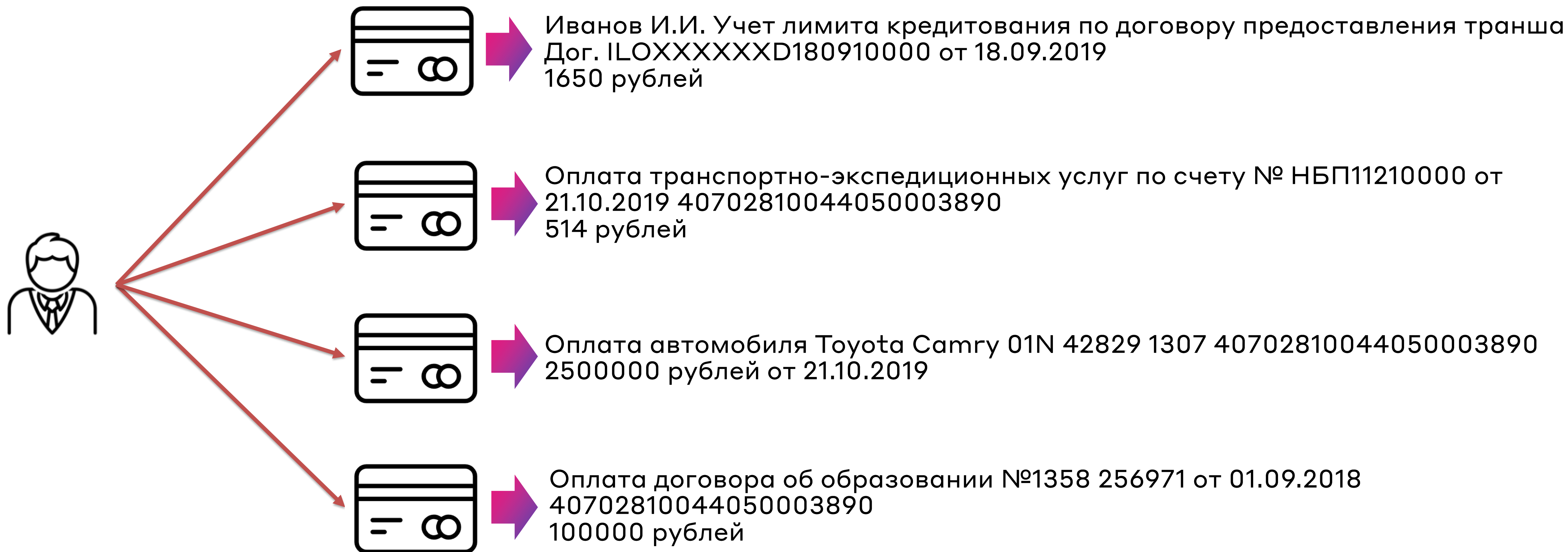
Категориальные признаки:

1. Самый популярный магазин
2. Самая популярная карта
3. И т.д.

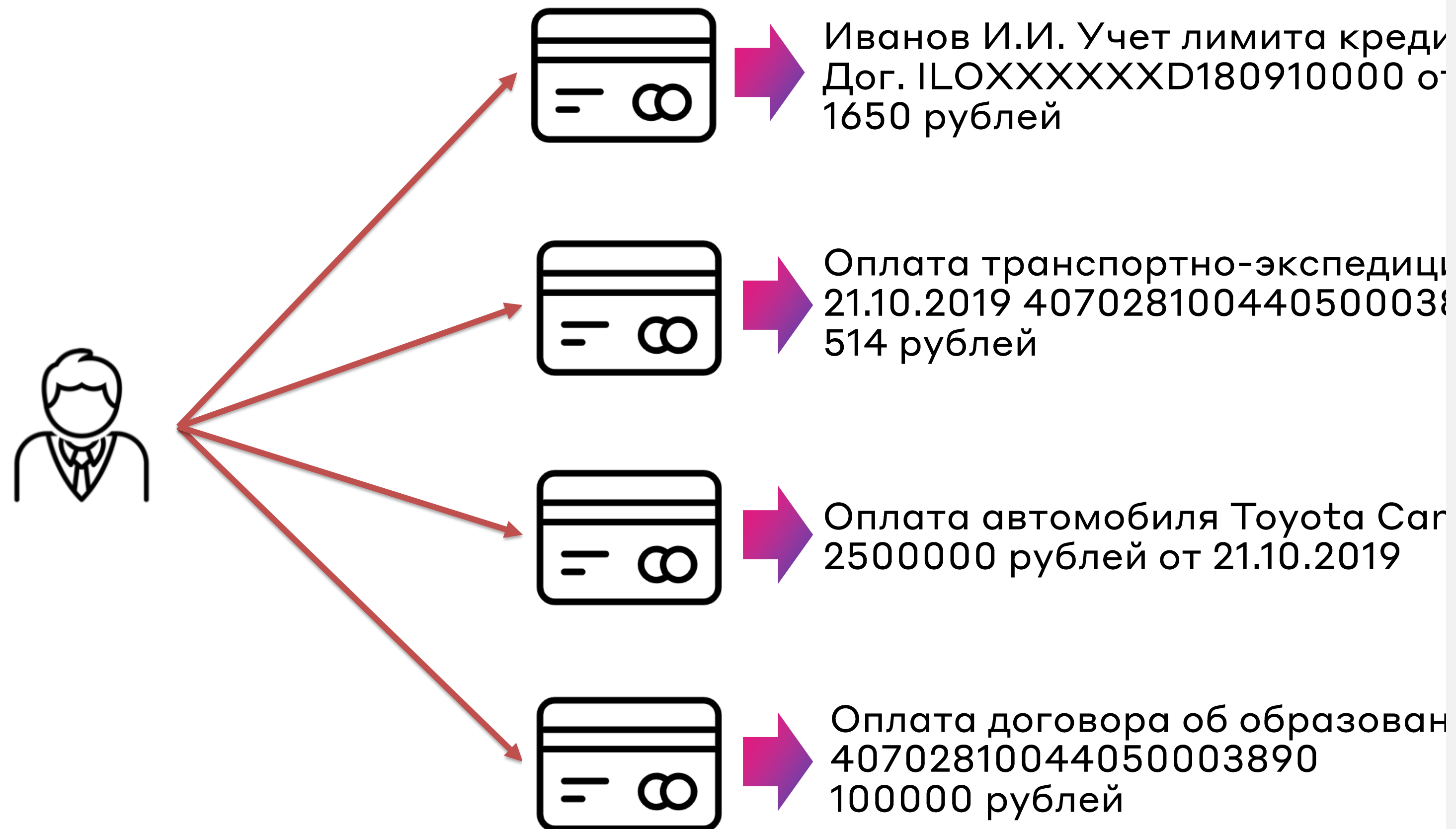
Временные признаки:

1. Среднее кол-во транзакций в день
2. Среднее время между транзакциями
3. И т.д.

# Транзакции расчетного счета



# Транзакции расчетного счета



**Так, и что с этим делать?**

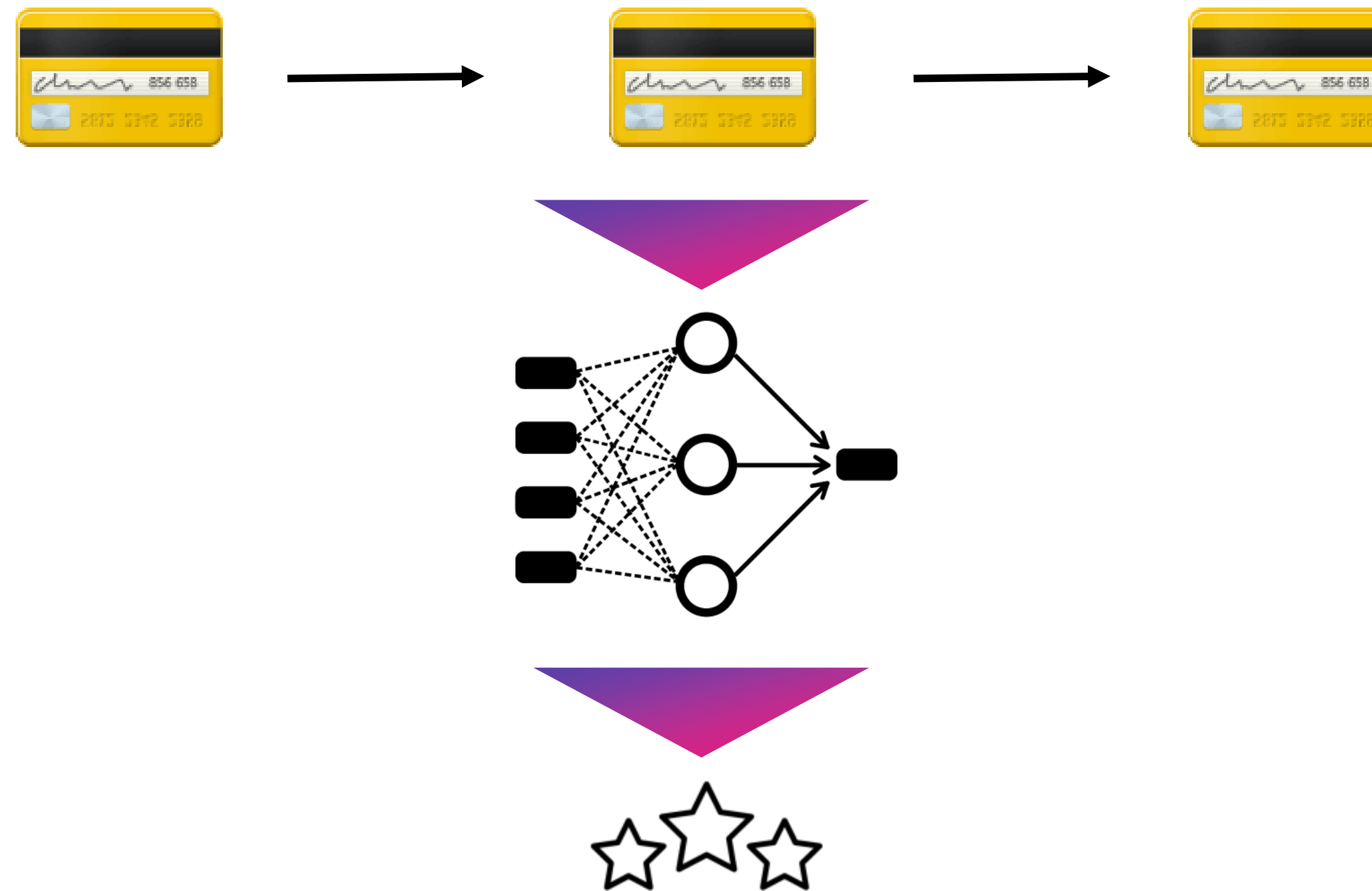


# О ЧЁМ БУДЕМ РАЗГОВАРИВАТЬ?

1. Постановка задачи и метрики
2. Кредитный скоринг в Альфа-Банке
3. Источники данных и подходы к моделированию
4. Модели на последовательных данных
5. Обработка текстовых признаков
6. Техники обучения
7. Основные результаты
8. Продакшн и планы на будущее

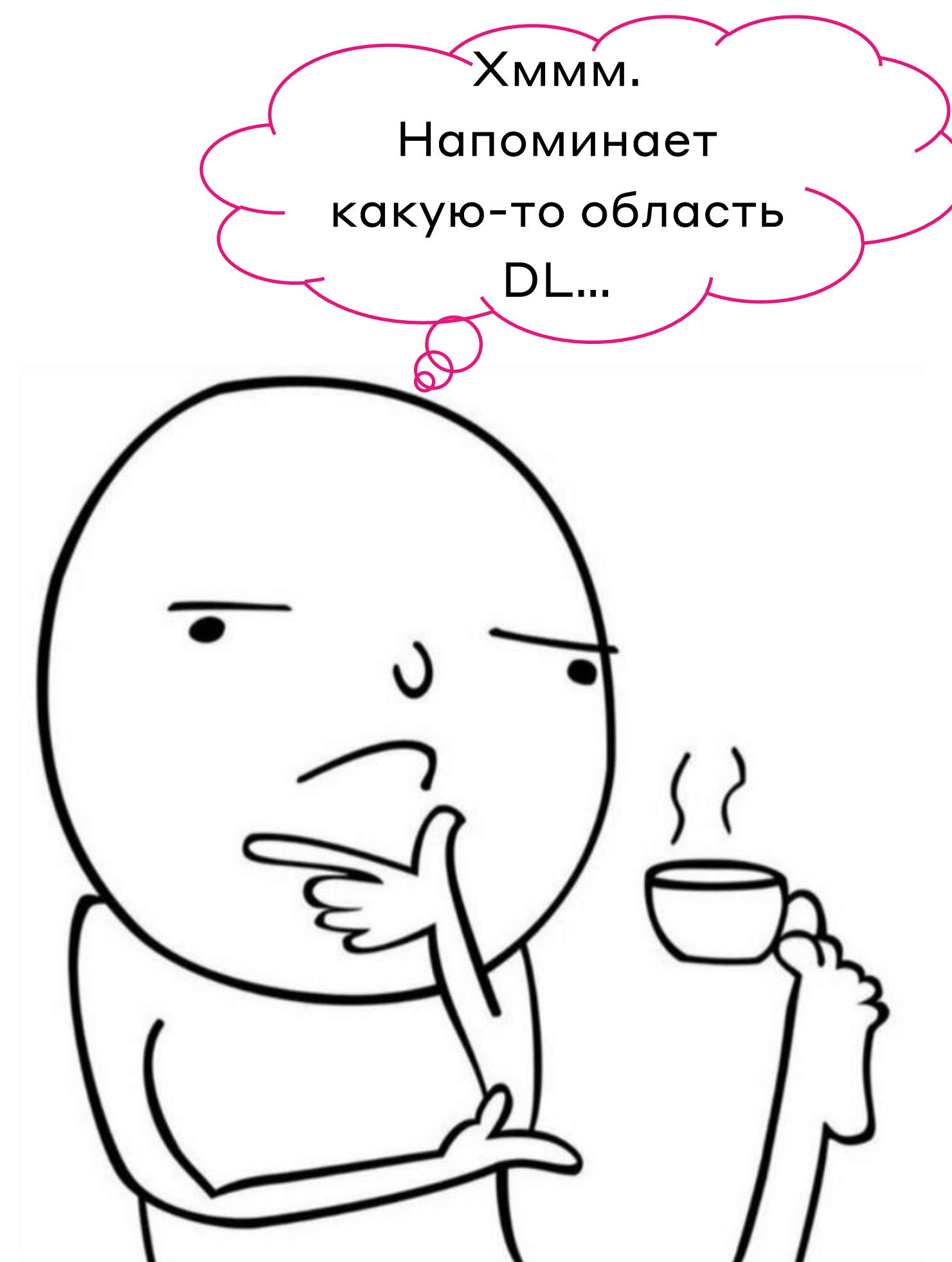
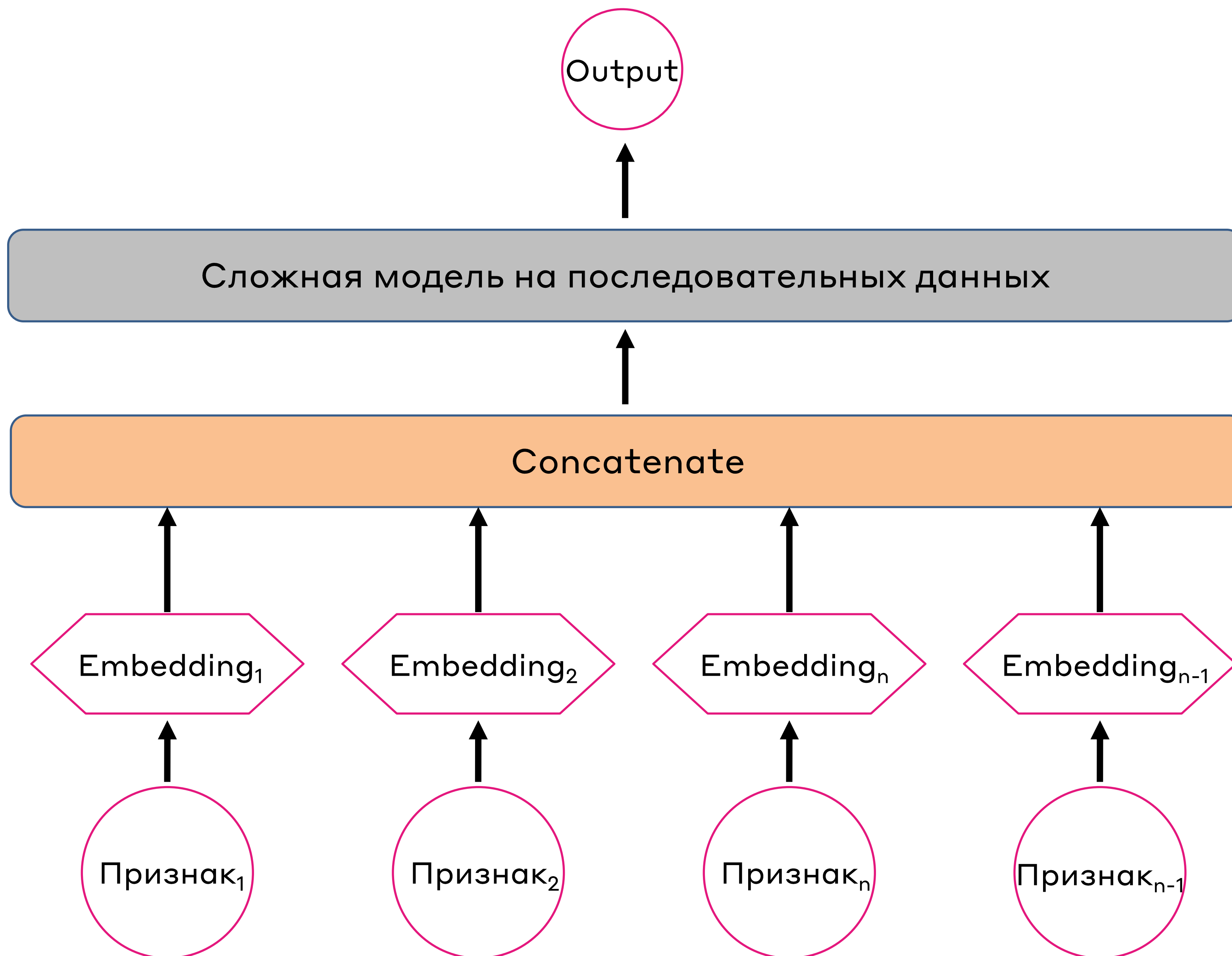
# Общая постановка задачи

**ВХОД** Последовательность данных о клиенте до определенной даты



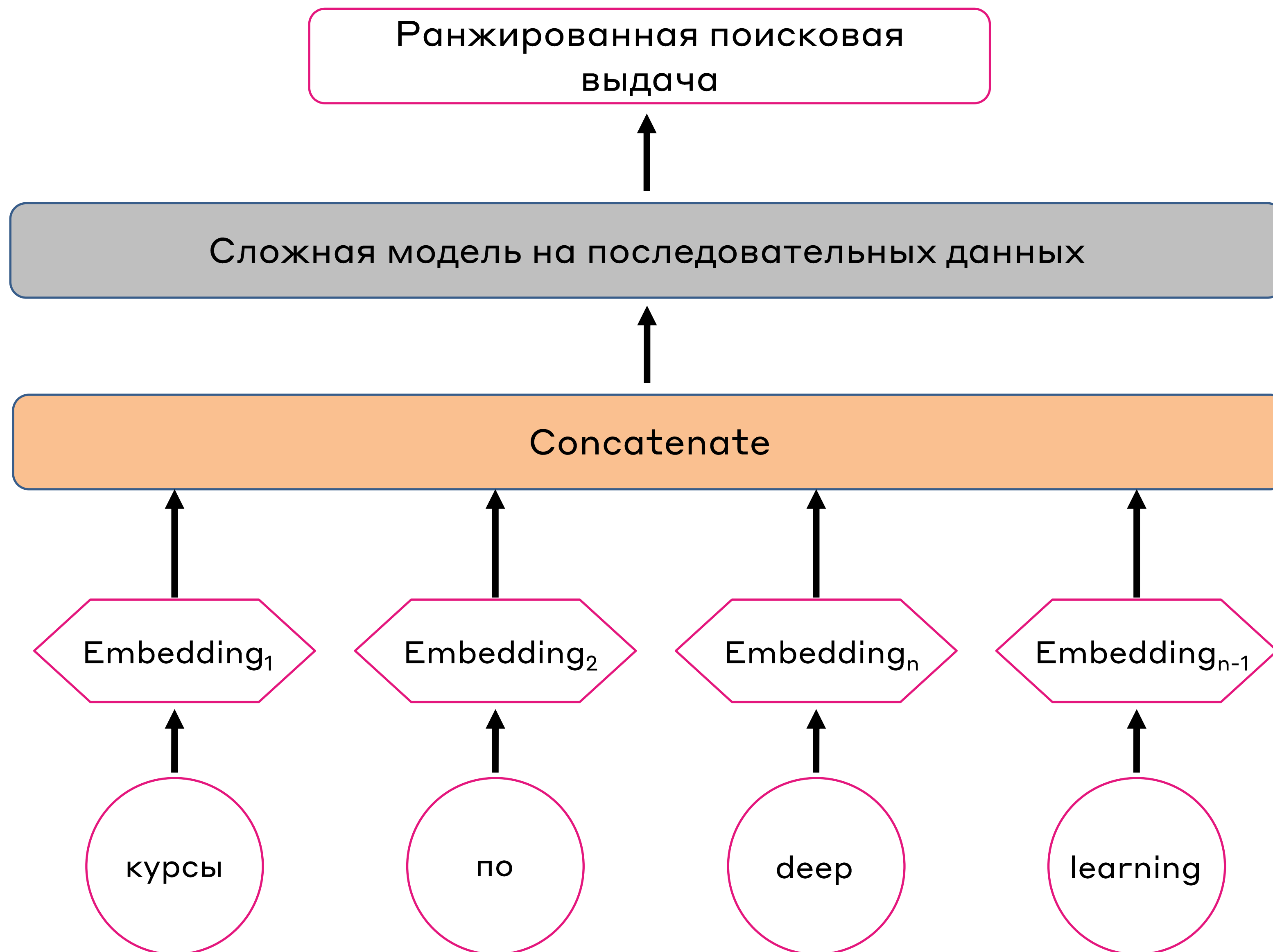
**ВЫХОД** Предсказание целевой переменной для конкретной решаемой задачи

# Общая схема модели



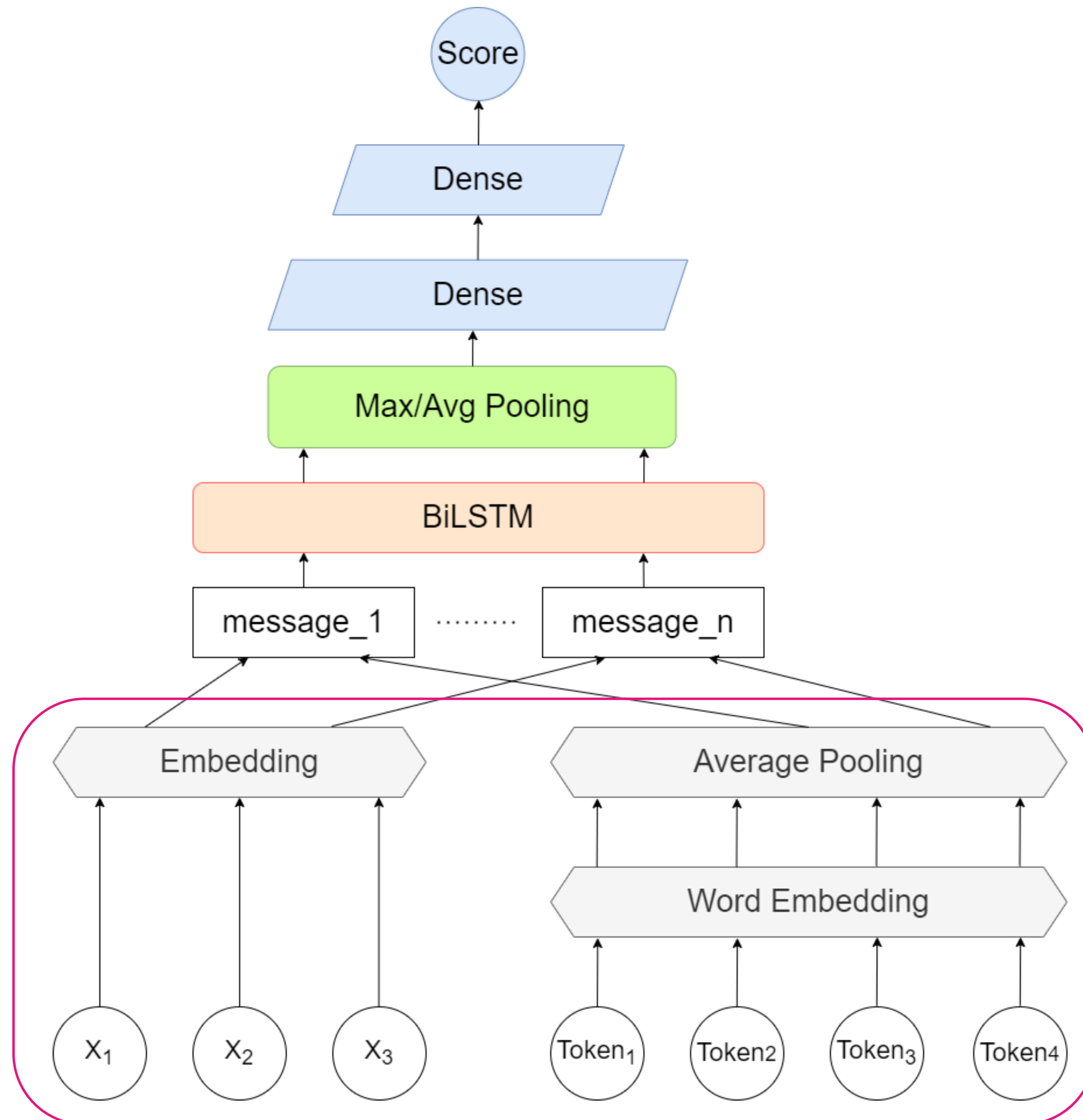


# Общая схема модели



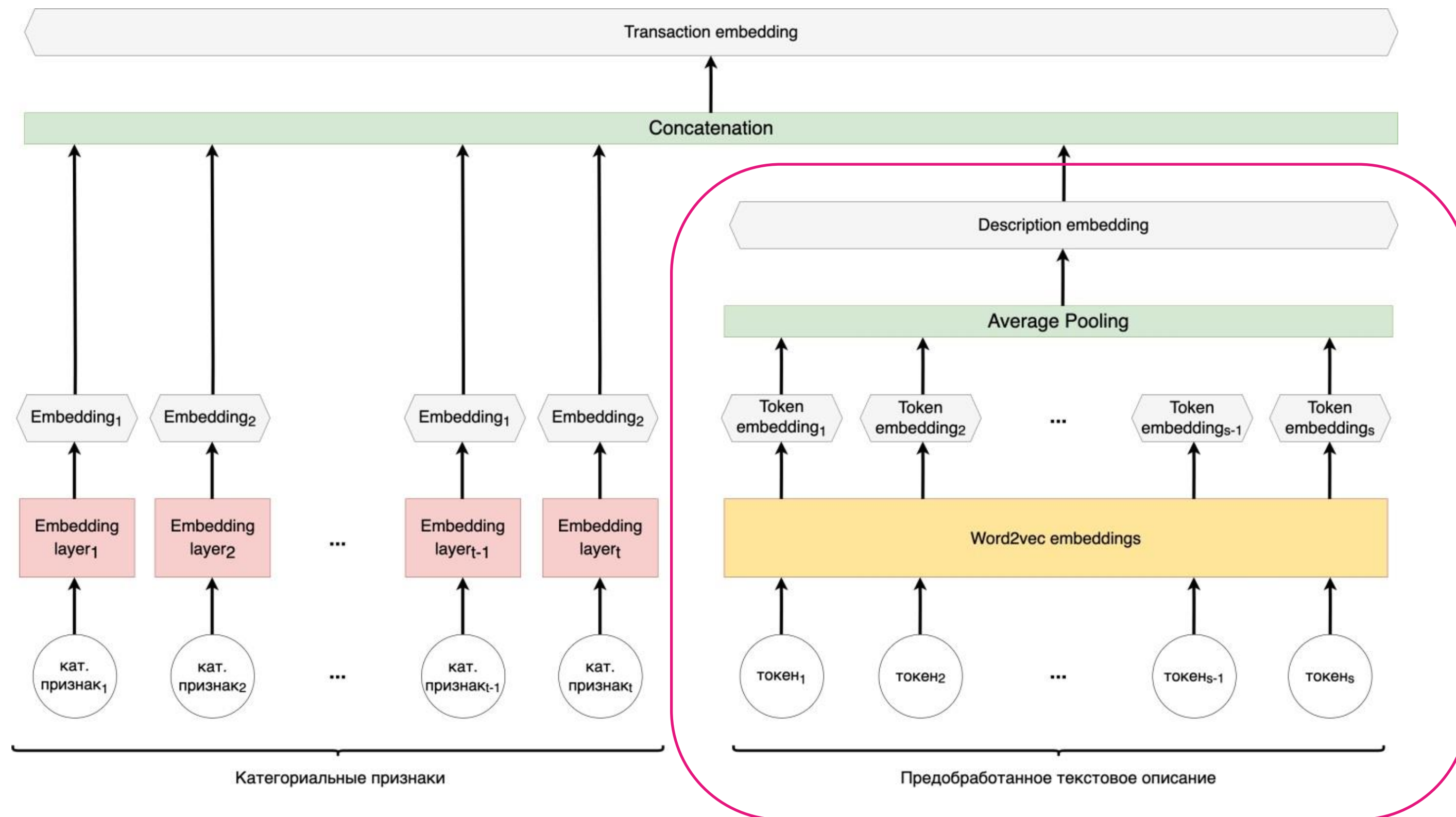
- Как и задача обработки естественного языка
- Вместо слов – элементы последовательности

# Архитектура модели



- Как и в NLP, но вместо слов – элементы последовательности
- Вещественные признаки приводятся к категориальным путём бинаризации (на 20 бакетов)
- Embedding-слой для каждого категориального признака

# Обработка признаков



- Как и в NLP, но вместо слов – элементы последовательности
- Вещественные признаки приводятся к категориальным путём бинаризации (на 20 бакетов)
- Embedding-слой для каждого категориального признака

# О ЧЁМ БУДЕМ РАЗГОВАРИВАТЬ?

1. Постановка задачи и метрики
2. Кредитный скоринг в Альфа-Банке
3. Источники данных и подходы к моделированию
4. Модели на последовательных данных
- 5. Обработка текстовых признаков**
6. Техники обучения
7. Основные результаты
8. Продакшн и планы на будущее

# Описания в транзакциях расчетного счета

- 01** Оплата ЖКХ ЛС 460000 Иванов Иван Иванович г. Иваново, ул. Кара01N 36 2100 407028107312000000000000000000  
3010181000000000000000
  
- 02** Снятие, ремонт и установка топливного бака на Мерседес А000АА11 НДС 200 2000 407028107312000000000000000000  
3010181000000000000000
  
- 03** Оплата за дизайн рекламной продукции (календари, ручки, пакеты, кружки01N)
  
- 04** Оплата за отопление за Октябрь 2021 г. по счету N 1 от 01.10.2021 (01N280000 2200 407028107312000000000000000000  
3010181000000000000000

## Обработка описаний

Замена всех токенов, содержащих цифру, на специальный «unknown» токен <unk\_tok>

Выделение токенов с фамилией и замена на <fio\_tok>

Лемматизация оставшихся токенов

Замена непопулярных токенов на специальный токен <unpop\_tok>

# Примеры обработки описаний

## Было

Оплата ЖКХ ЛС 460000 Иванов Иван  
Иванович г. Иваново, ул. Кара01N 36 2100  
40702810731200000000000000000000  
301018100000000000000000

Снятие, ремонт и установка топливного  
бака на Мерседес А000АА11 НДС 200 2000  
40702810731200000000000000000000  
301018100000000000000000

Оплата за отопление за Октябрь 2021  
г. по счету N 1 от 01.10.2021 (01N280000  
2200 40702810731200000000000000000000  
301018100000000000000000

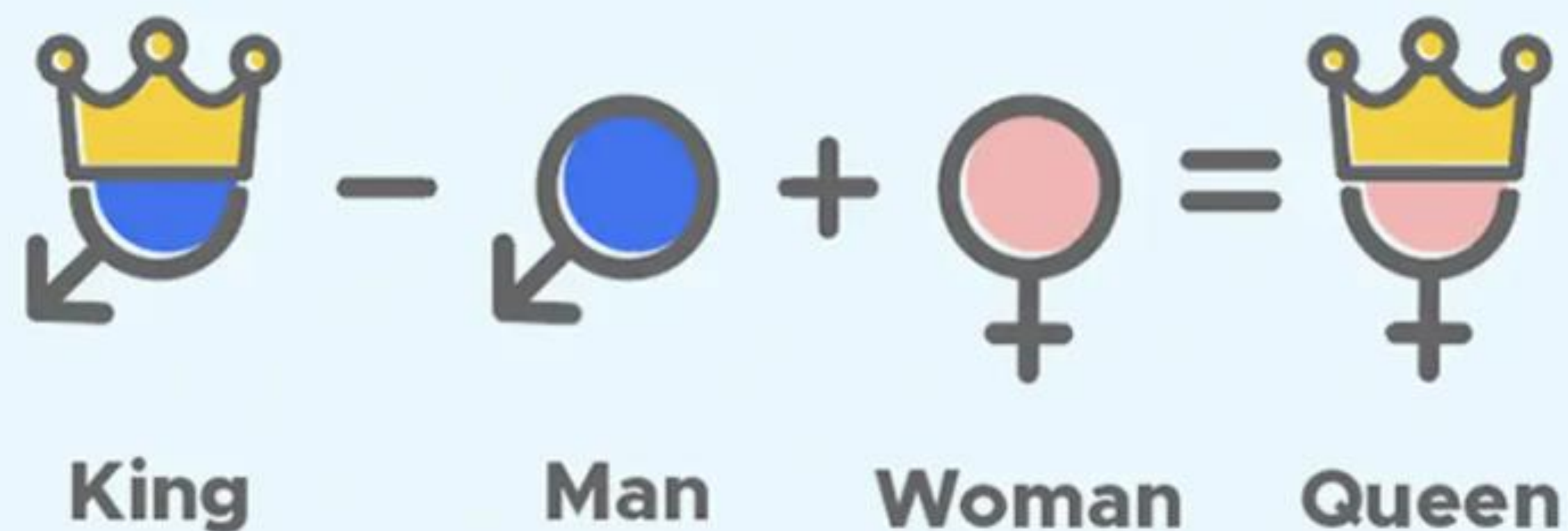
## Стало

Оплата жкх unprop\_tok unk\_tok fio\_tok  
fio\_tok fio\_tok иваново unprop\_tok кара  
unk\_tok

Снятие ремонт установка топливный  
unprop\_tok unprop\_tok мерседес unk\_tok  
unprop\_tok unk\_tok ндс unk\_tok

Оплата unprop\_tok отопление unprop\_tok  
октябрь unk\_tok unprop\_tok счет unk\_tok  
unprop\_tok unk\_tok

# Обучение Word2Vec



**100 КК**

Размер  
корпуса

**50**

Размер  
эмбединга

**5 000**

Порог «популярности»  
токена

**40 000**

Количество  
уникальных токенов



# Words to Embeddings

## Корпус

word	idx	Feat. 1	Feat. 2	Feat. 3
мама	1	0,03	-0,15	0,9
мыла	2	0,39	0,51	-0,13
раму	3	0,17	0,36	-0,16
...	...	...	...	...
word <sub>n</sub>	n	0,17	0,19	-0,16

Вход: Мама мыла раму



Word2idx: 1, 2, 3



Извлекаем вектор по  
индексу

0,03 -0,15 0,09  
0,39 0,51 -0,13  
0,17 0,36 -0,16



Усредняем и подаём в  
модель

0.2, 0.24, -0.07

# Entity Embeddings

Категориальные признаки / Текстовые признаки →  
→ Look Up Matrix → Вектор

## Преимущества:

На выходе  
слоя получается  
векторное  
представление  
признака, вместо  
одного числа

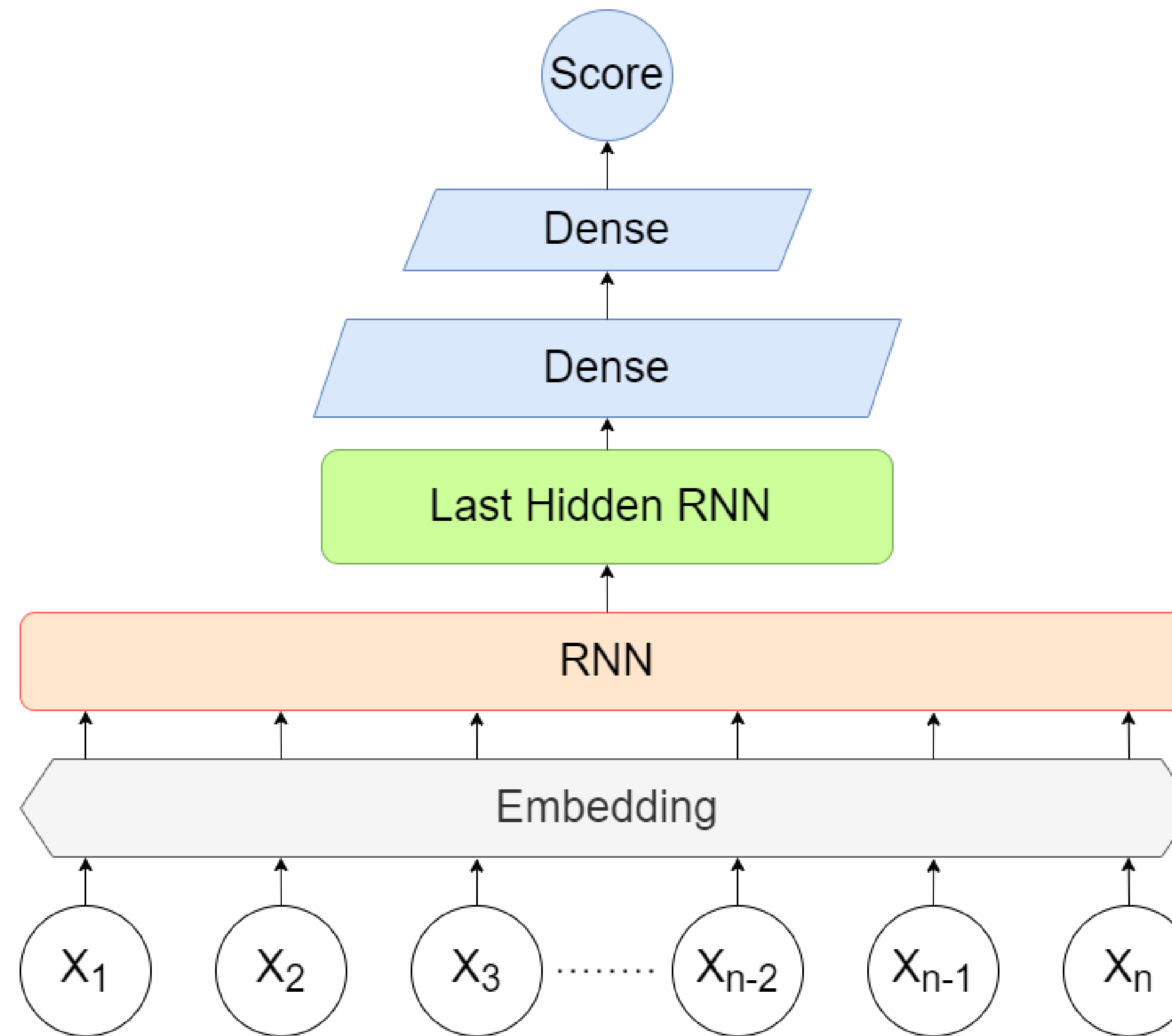
Эмбеддинги  
учатся вместе  
с остальными  
весами сети

Выученные  
эмбеддинги  
могут  
использоваться  
в других задачах

# О ЧЁМ БУДЕМ РАЗГОВАРИВАТЬ?

1. Постановка задачи и метрики
2. Кредитный скоринг в Альфа-Банке
3. Источники данных и подходы к моделированию
4. Модели на последовательных данных
5. Обработка текстовых признаков
- 6. Техники обучения**
7. Основные результаты
8. Продакшн и планы на будущее

# Архитектура модели



Слишком просто,  
давайте усложнять

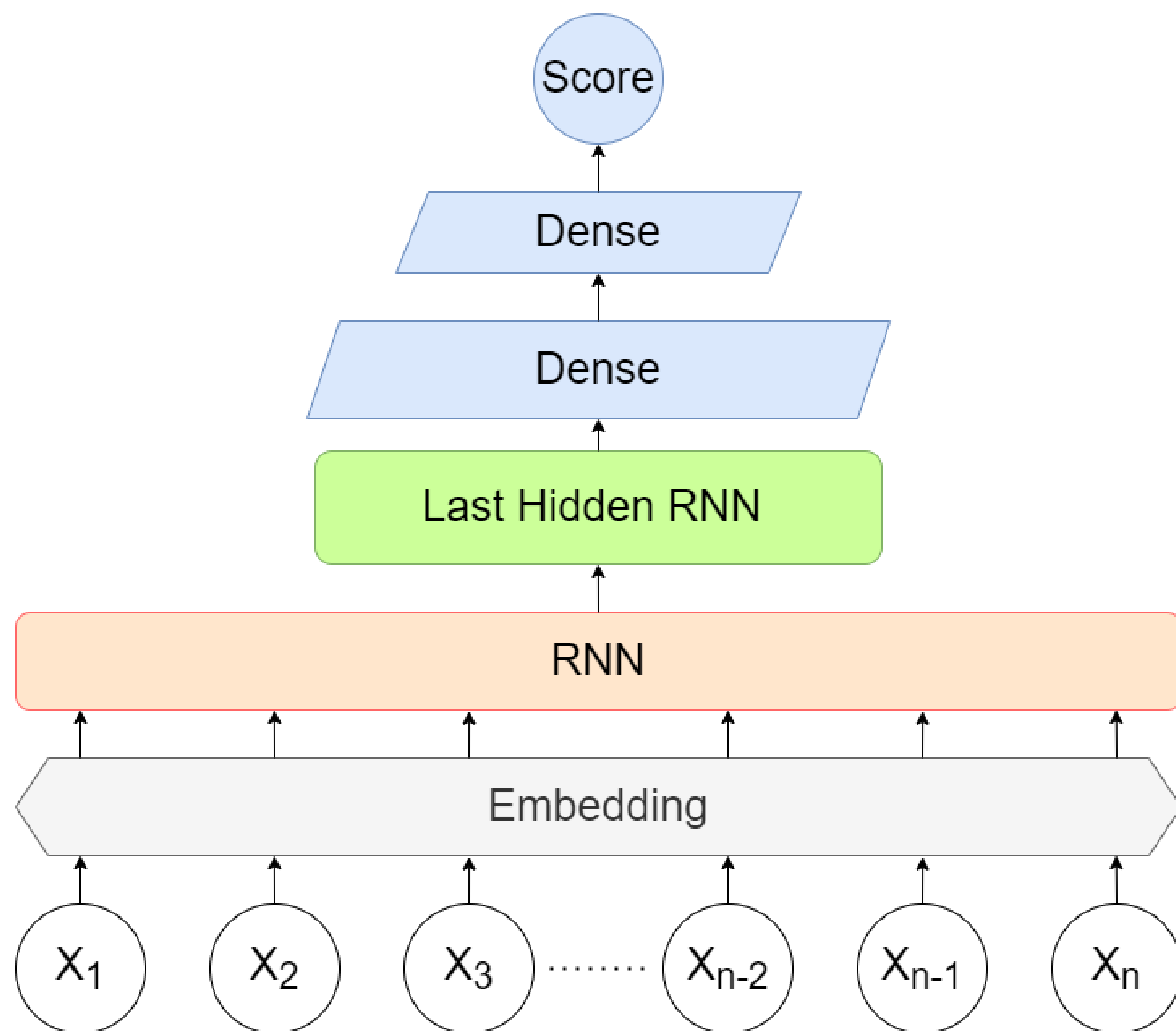
# Baseline

МОДЕЛЬ

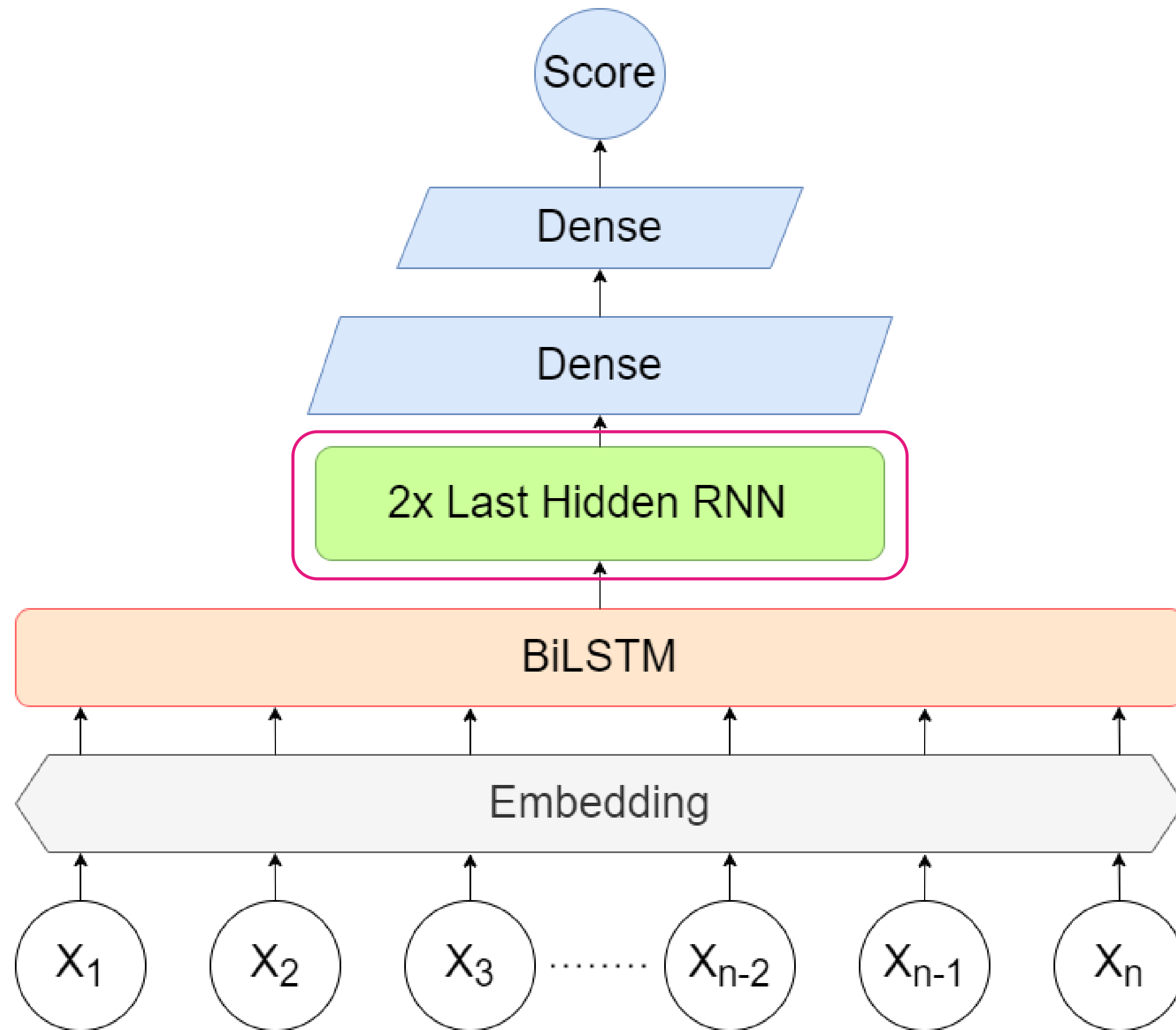
ROC-AUC

● Baseline LSTM

74,8



# LSTM → BiLSTM



МОДЕЛЬ

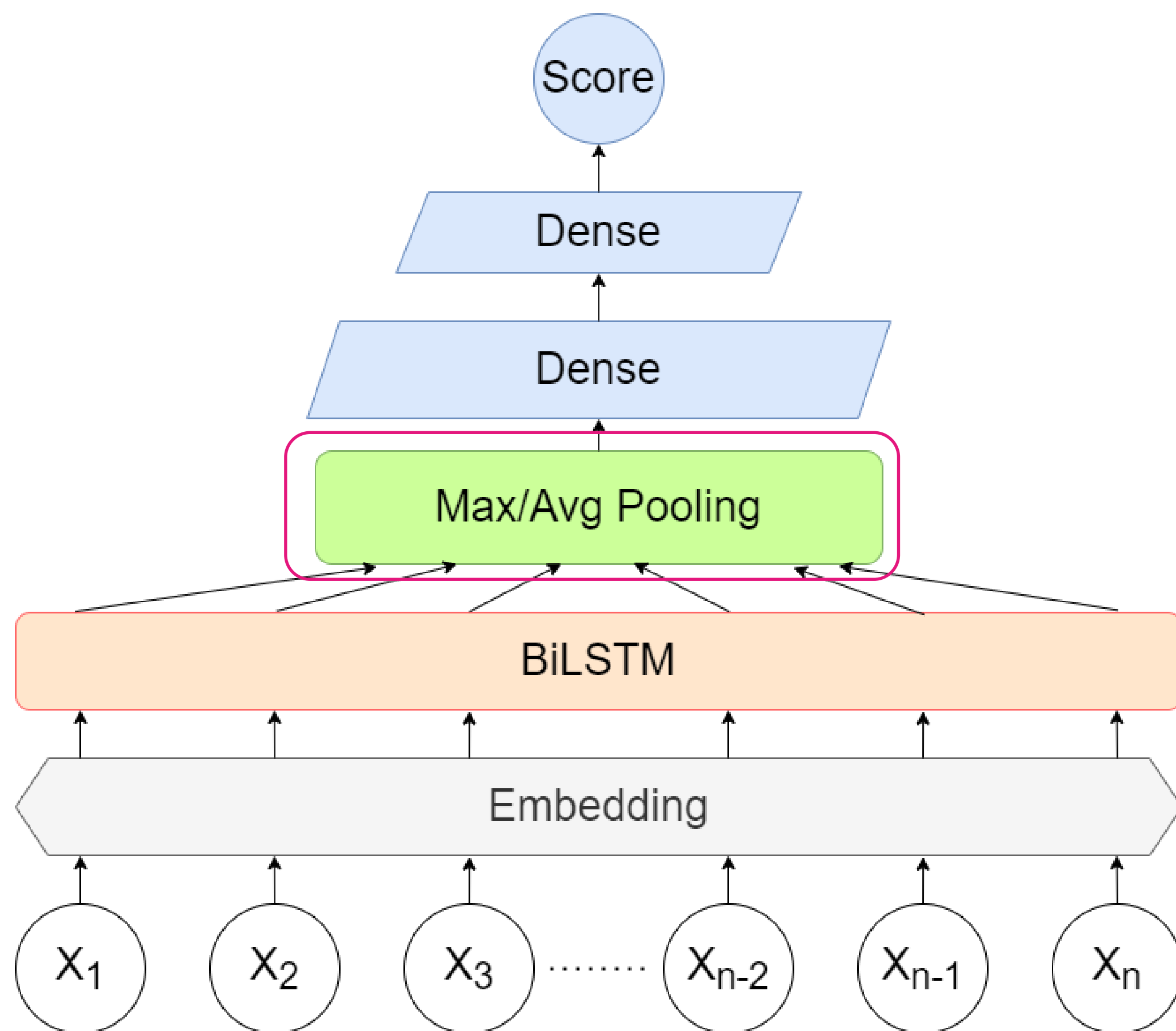
- Baseline LSTM
- ✚ BiLSTM

ROC-AUC

74,8

75,5

# Regularization



## МОДЕЛЬ

- Baseline LSTM
- + BiLSTM
- + max/avg pool (RNN out)

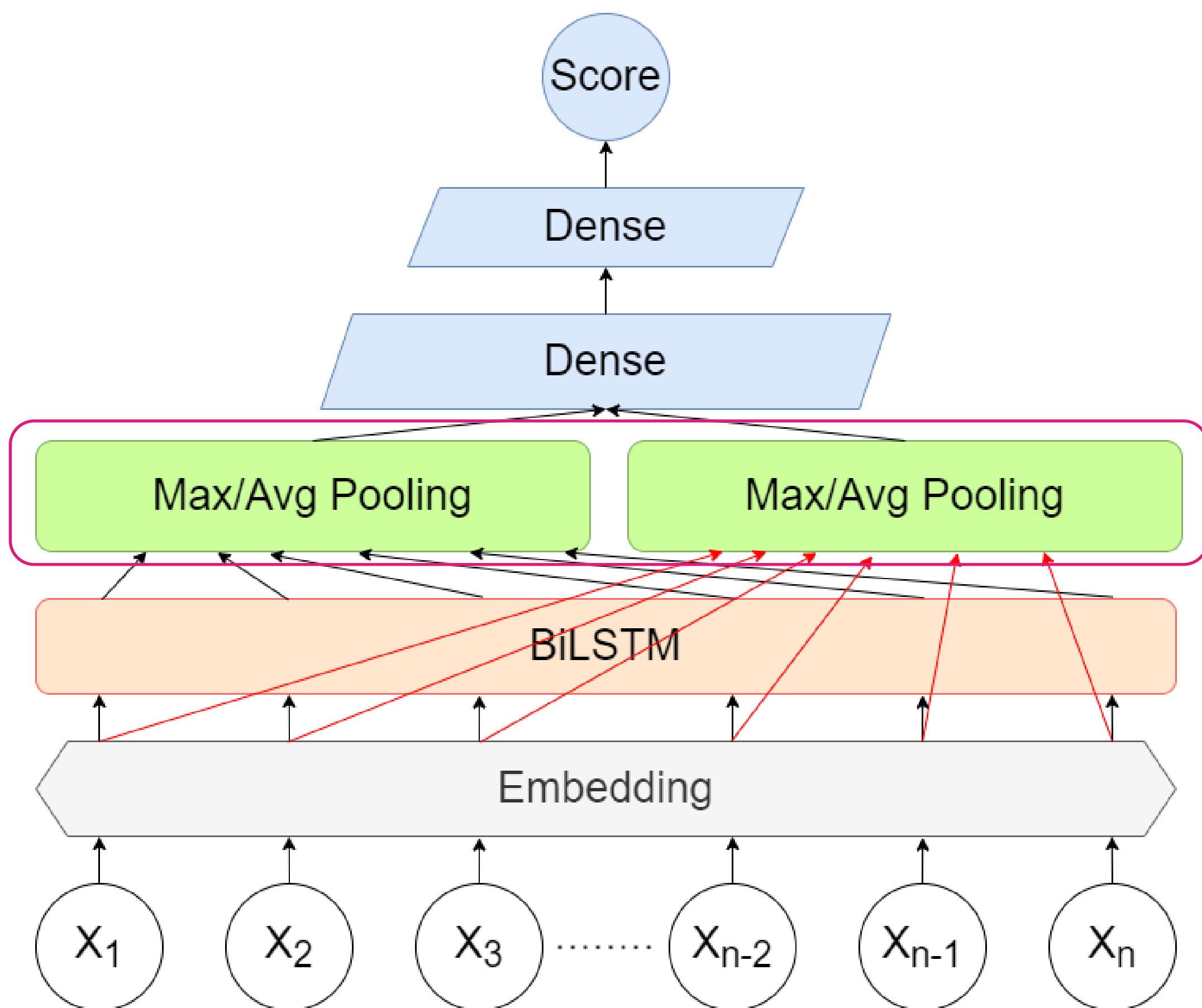
## ROC-AUC

74,8

75,5

76,0

# Regularization



## МОДЕЛЬ

- Baseline LSTM
- + BiLSTM
- + max/avg pool (RNN out)
- + max/avg pool (embeddings)

## ROC-AUC

74,8

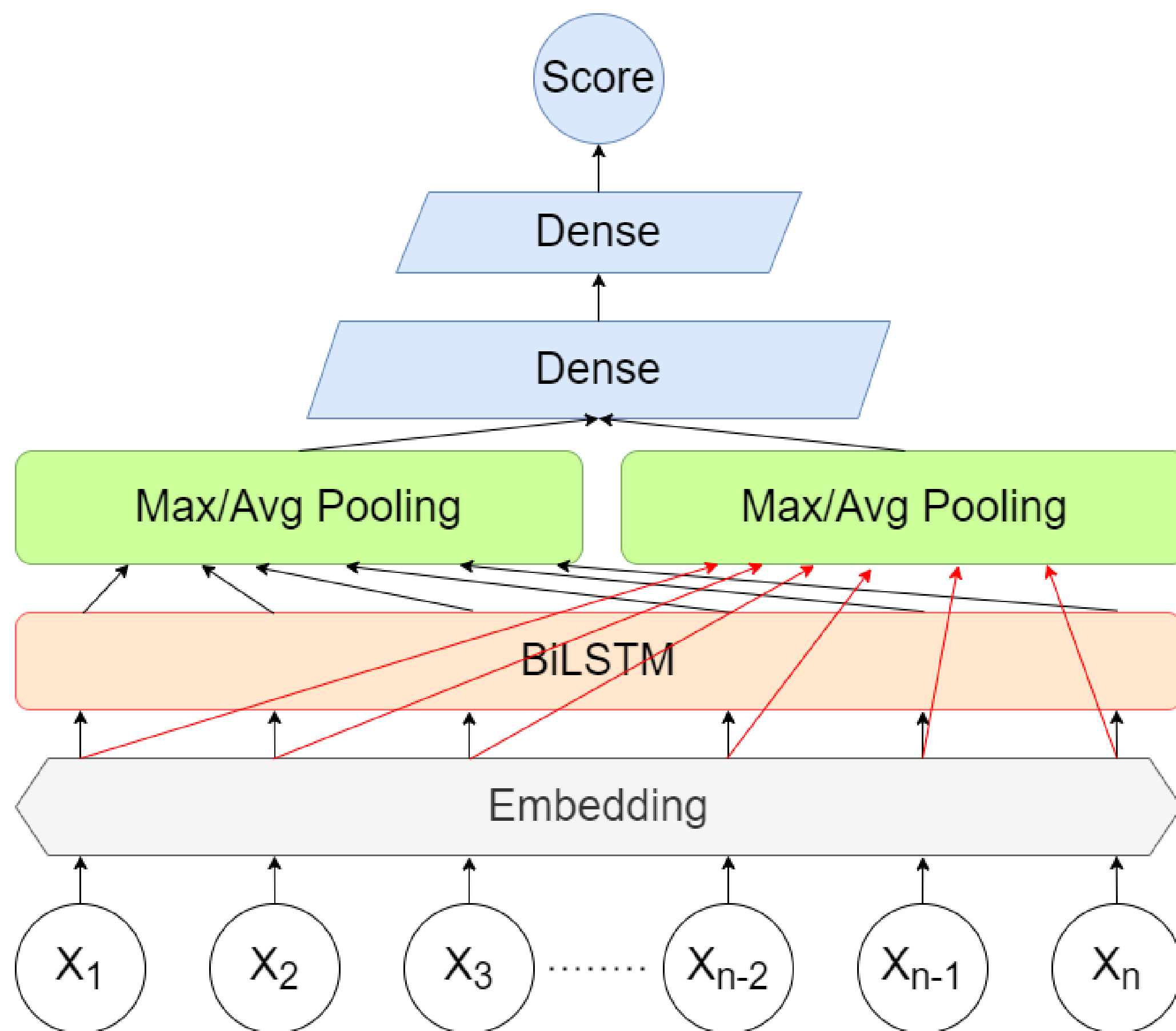
75,5

76,0

76,3



# Regularization



## МОДЕЛЬ

## ROC-AUC

●	Baseline LSTM	74,8
+	BiLSTM	75,5
+	max/avg pool (RNN out)	76,0
+	max/avg pool (embeddings)	76,3
+	Regularization (L1+dropout2d)	77,4
+	Sample weight	77,9

# Улучшаем пайплайн обучения

- ✓ Данные и их обработка
- ✓ Архитектура модели
- ✓ Формирование признаков
- Обучение модели



# Регуляризация модели

Техники обучения, позволяющие бороться с переобучением моделей

Dropout и Spatial Dropout слои  
L1- и L2- регуляризации:

$$L1_{reg} = \lambda \sum_{i=1}^n |w_i|$$

$$L2_{reg} = \lambda \sum_{i=1}^n w_i^2$$

где  $w_i$  - веса модели

- Аддитивно добавляется к общему лоссу
- Независимо применяется на слои сети

# L1/L2 Регуляризация

$$L1_{reg} = \lambda \sum_{i=1}^n |w_i|$$

```
for i, params in enumerate(self._model.parameters(), 0):  
    if i != 0: # not embedding layer  
        weights_norm_1 += torch.norm(params, 1)  
loss = loss + l1_weight * weights_norm_1
```

$$L2_{reg} = \lambda \sum_{i=1}^n w_i^2$$

```
for i, params in enumerate(self._model.parameters(), 0):  
    if i != 0: # not embedding layer  
        weights_norm_2 += torch.norm(params, 2) ** 2  
loss = loss + l2_weight * weights_norm_2
```

# nn.Dropout vs nn.Dropout2d для RNN

shape = 2x3x2

```
tensor([[[[1., 1.],
          [1., 1.],
          [1., 1.]],

        [[1., 1.],
          [1., 1.],
          [1., 1.]]]])
```

nn.Dropout

```
tensor([[[[0., 2.],
          [0., 0.],
          [0., 0.]],

        [[0., 2.],
          [2., 2.],
          [0., 2.]]]])
```

Почему бы не занулять случайно целиком какой-то эмбединг, тем самым «выключая» подпространство какого-нибудь признака?

**nn.Dropout2d**

```
dropout2 = nn.Dropout2d()
batch = batch.permute(0, 2, 1).unsqueeze(3) # N*L*E --> N*E*L-->N*E*L*1
batch = dropout2(batch) # randomly set to zero E axis with some P
batch = batch.squeeze(3).permute(0, 2, 1)
```

```
tensor([[[[0., 2.],
          [0., 2.],
          [0., 2.]],

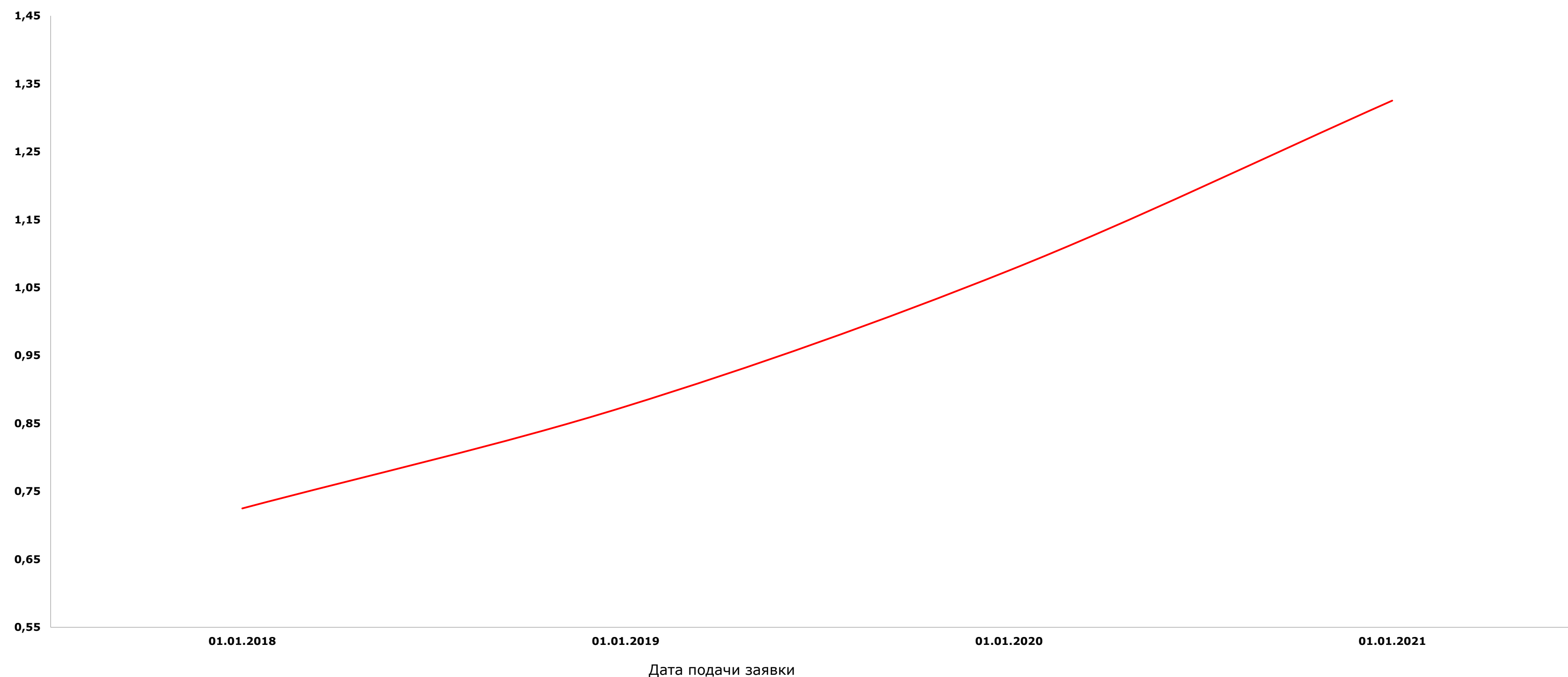
        [[2., 0.],
          [2., 0.],
          [2., 0.]]]])
```



# Sample Weight по времени

В данных есть временная ось,  
хотелось бы эффективно использовать данную информацию

Вес заявки  
при обучении



Регуляризация

Sample Weight

Оптимизация

# Формирование батчей для обучения



Иван

1000 транзакций  
за последний год



[1, 19, 1000]

**Проблема**

Необходимо сформировать Batch,  
но у всех клиентов разное  
количество транзакций



Олег

100 транзакций  
за последний год



[1, 19, 100] → +900 нулевых транзакций → [1, 19, 1000]

**Решение #1**

Отсутствующие транзакции у  
клиентов заполняем нулями



Дима

10 транзакций  
за последний год



[1, 19, 10] → +990 нулевых транзакций → [1, 19, 1000]

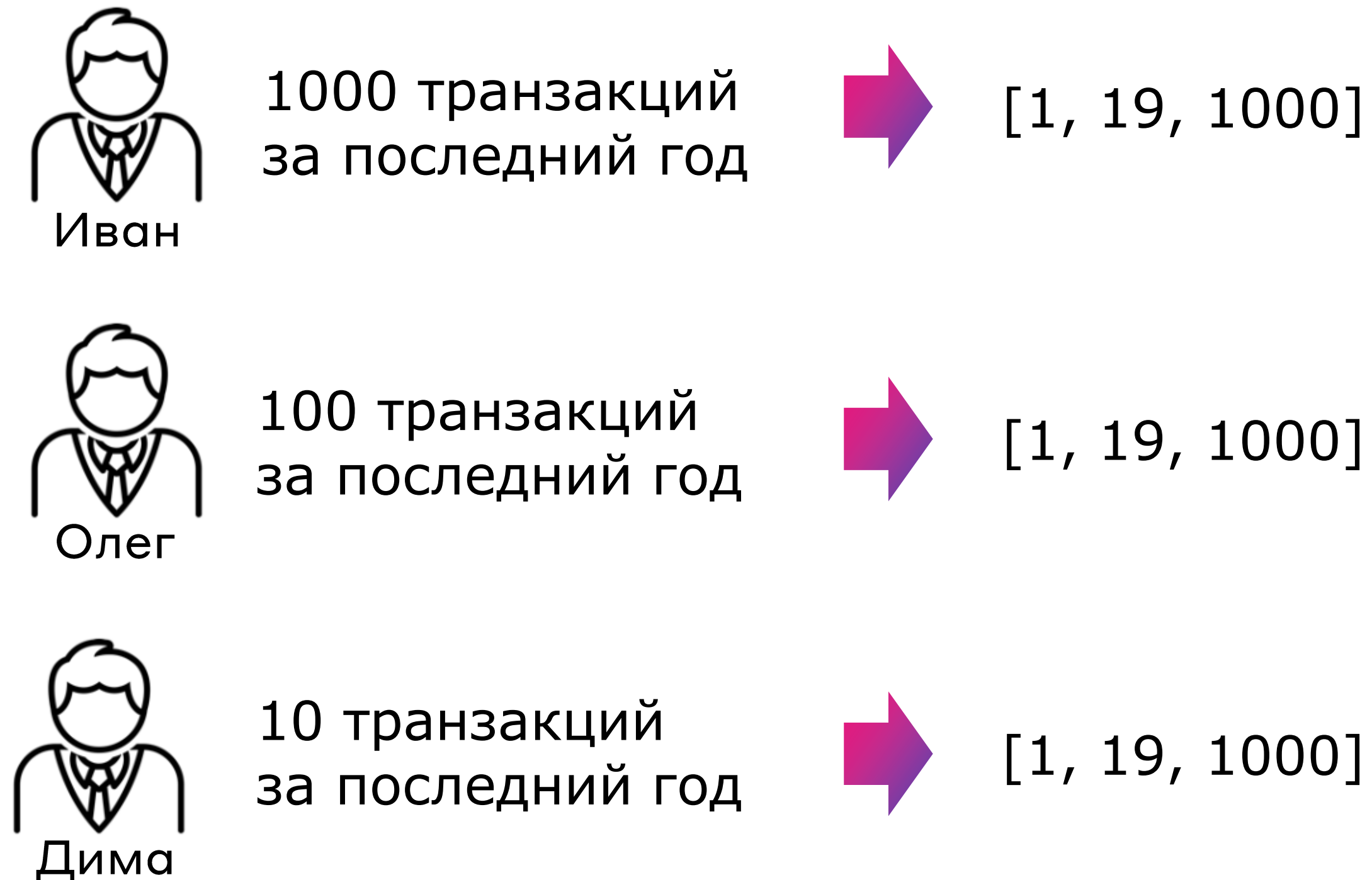
Batch\_size = [3, 19, 1000]

Регуляризация

Sample Weight

Оптимизация

# Формирование батчей для обучения



## Проблема

Необходимо сформировать Batch, но у всех клиентов разное количество транзакций

## Решение #1

Отсутствующие транзакции у клиентов заполняем нулями

## Следствие

В результате очень много нулей, а следовательно и много бесполезных вычислений=(



Очень плохой вариант

Batch\_size = [3, 19, 1000]

Регуляризация

Sample Weight

Оптимизация



# Формирование батчей для обучения



1000 транзакций  
за последний год



[1, 19, 1000]

**Проблема**

Необходимо сформировать Batch, но у всех клиентов разное количество транзакций



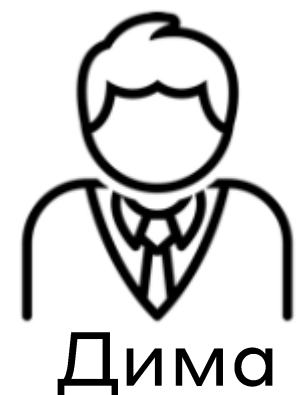
100 транзакций  
за последний год



[1, 19, 100]

**Решение #2**

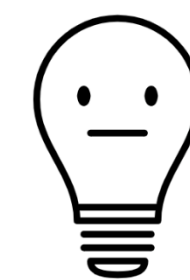
Используем pack\_sequence



10 транзакций  
за последний год



[1, 19, 10]



**Следствие**

Не можем использовать cuDNN реализацию LSTM/GRU, обучение будет медленнее

Уже лучше. Можно ещё лучше?

Регуляризация

Sample Weight

Оптимизация

# Sequence Bucketing



## Проблема



- Количество транзакций у клиентов сильно различается, надо добавлять много нулей
- Pack\_sequence не позволяет использовать cuDNN реализацию LSTM

## Решение



- Транзакции примерно одинаковой длины объединяем в бакеты и сэмплируем из них при обучении

# Sequence Bucketing



1000 транзакций  
за последний год



[1, 19, 1000]

**Проблема**

Необходимо сформировать Batch,  
но у всех клиентов разное  
количество транзакций



990 транзакций  
за последний год



[1, 19, 990] → +10 нулевых транзакций → [1, 19, 1000]

**Решение #3**

Использовать Sequence Bucketing



935 транзакций  
за последний год



[1, 19, 935] → +75 нулевых транзакций [1, 19, 1000]

Batch\_size = [3, 19, 1000]

Регуляризация

Sample Weight

Оптимизация

# Sequence Bucketing



Иван

1000 транзакций  
за последний год



Олег

990 транзакций  
за последний год



Дима

935 транзакций  
за последний год

## Проблема

Необходимо сформировать Batch,  
но у всех клиентов разное  
количество транзакций

## Решение #3

Использовать Sequence Bucketing



Вооо, вот так вообще отлично!

Регуляризация

Sample Weight

Оптимизация

# Формирование батчей для обучения

Было

- ~60 мин на эпоху обучения
- добавляем ~ 95% нулевых транзакций

Стало

- ✓ ~20 мин на эпоху обучения
- ✓ добавляем ~ 10% нулевых транзакций
- ✓ можем тратить больше времени на эксперименты

Регуляризация

Sample Weight

Оптимизация

# Что не зашло?

Как вы думаете, какой самый популярный вопрос про наши нейронки нам задают?



Небольшая подсказочка



# Что не зашло?

Как вы думаете, какой самый популярный вопрос про наши нейронки нам задают?

А пробовали ли вы трансформеры?

BiLSTM	78,3
Transformer Encoder	77,4

# Что не зашло?

Как вы думаете, какой самый популярный вопрос про наши нейронки нам задают?

А пробовали ли вы трансформеры?

Пробовали и не только своими силами  
Провели 3 соревнования на последовательных данных  
Получили >2000 решений  
В ~99% RNN>Transformer



# Что не зашло?

Что насчёт других функций потерь?

BCE	77,8
BCE + sample weight	78,3
Focal Loss	77,6
Focal Loss + sample weight	78,1

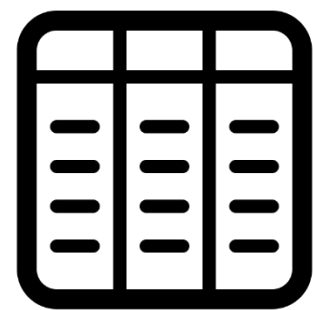
 На соревнованиях другие лоссы тоже не зашли=)

# О ЧЁМ БУДЕМ РАЗГОВАРИВАТЬ?

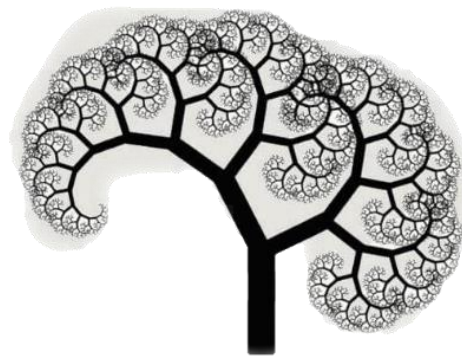
1. Постановка задачи и метрики
2. Кредитный скоринг в Альфа-Банке
3. Источники данных и подходы к моделированию
4. Модели на последовательных данных
5. Обработка текстовых признаков
6. Техники обучения
7. Основные результаты
8. Продакшн и планы на будущее

# Как подружить бустинг с нейронной сетью?

## Классический подход



Табличные  
данные

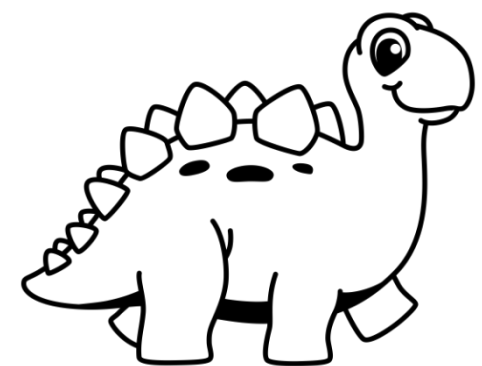


Градиентный  
бустинг

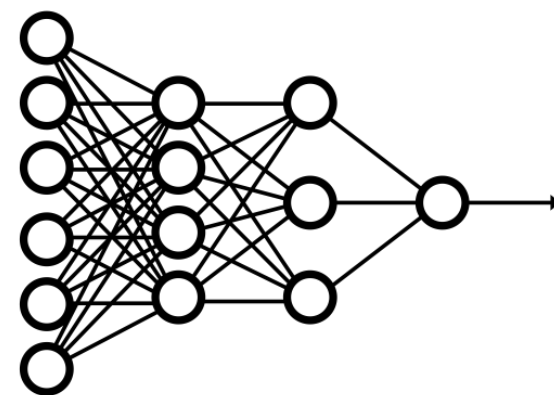


Текущий  
скор

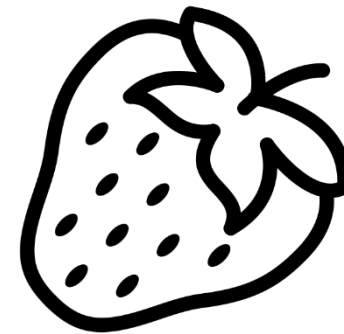
## Нейросетевой подход



Событийные  
данные

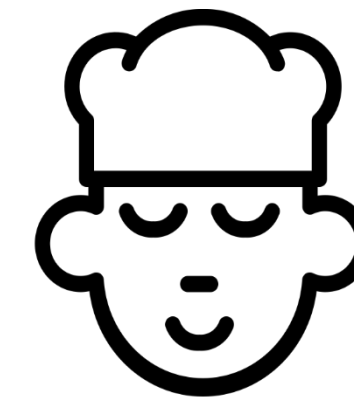


Нейронная  
сеть



Нейросетевой  
скор

## Объединяем подходы



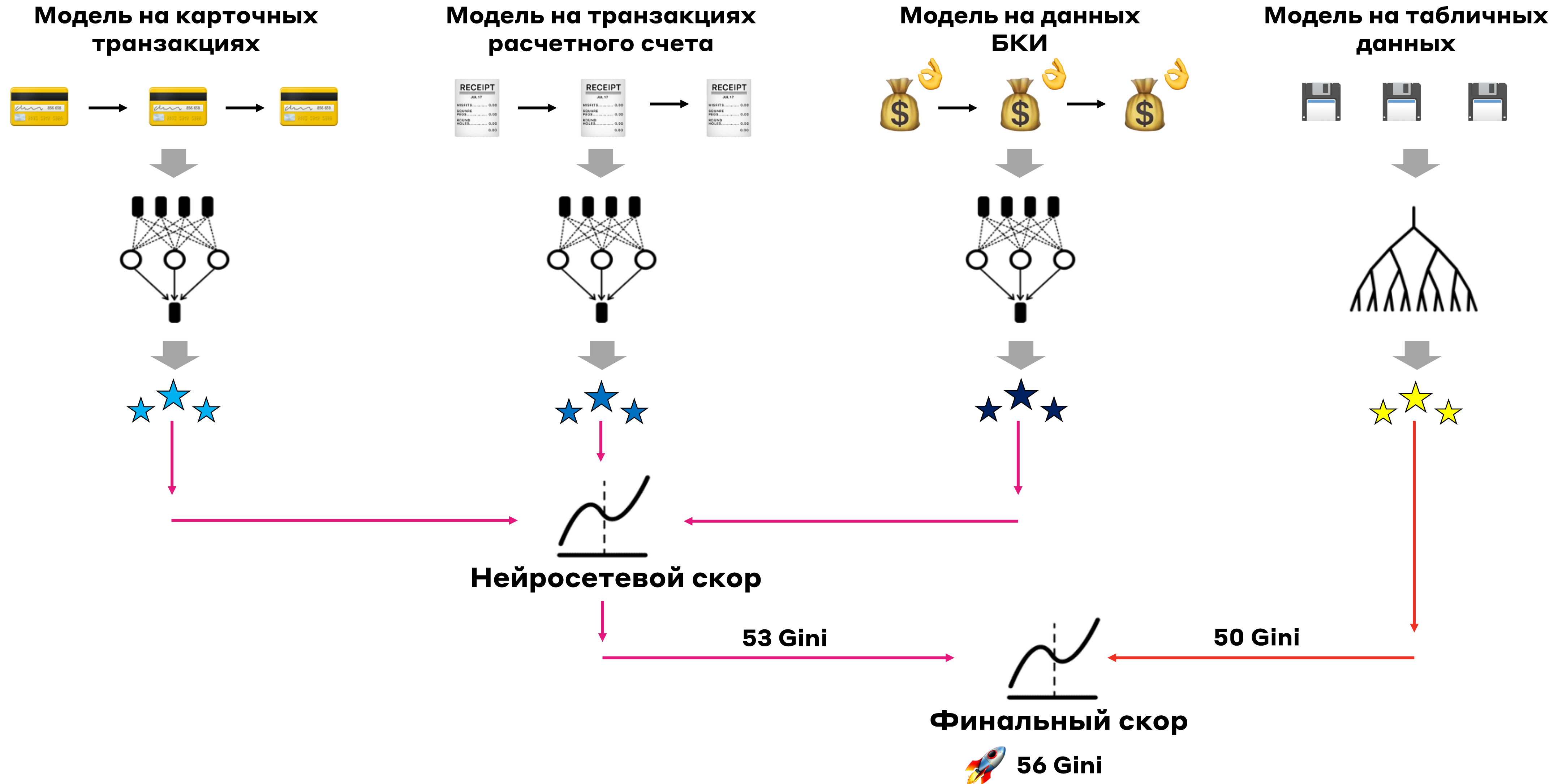
Логистическая  
регрессия



Финальный  
скор

- + Независимая разработка компонент кредитного скоринга
- + Прозрачный процесс мониторинга компонент финального сора

# Uplift от нейросетей



# Основные результаты



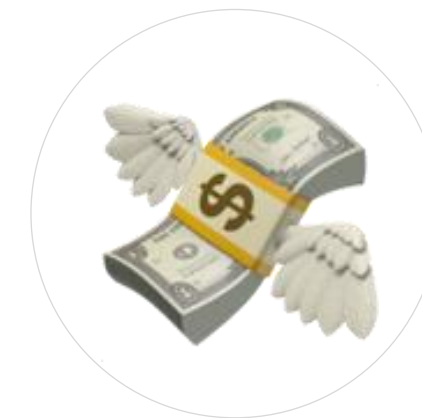
**Повышена ранжирующая способность**  
при неизменном риске можем выдавать больше кредитов



**Переиспользовали подход в других задачах**  
в задачах склонности и оттока нейросети также дают существенный uplift



**Отказались от части внешних данных**  
экономия на закупке скоров и данных из внешних источников



**Получен бизнес-эффект**  
внедрение нейростей в скоринг приносит >600 млн рублей годовой чистой прибыли

# О ЧЁМ БУДЕМ РАЗГОВАРИВАТЬ?

1. Постановка задачи и метрики
2. Кредитный скоринг в Альфа-Банке
3. Источники данных и подходы к моделированию
4. Модели на последовательных данных
5. Обработка текстовых признаков
6. Техники обучения
7. Основные результаты
8. Продакшн и планы на будущее

# Объемы данных по транзакциям

## Обучение модели

5 млн

Заявок

200 GB

Сырых данных

~2 млрд

Карточных транзакций /  
Транзакций расчетного счета

10 ч

На обучение модели

## Инференс модели

10 млн

Клиентов

500 GB

Сырых данных

~4 млрд

Карточных транзакций /  
Транзакций расчетного счета

2 ч

Инференс

# Объемы данных по БКИ

## Обучение модели

**5 млн**

Заявок

**500 млн**

Исторических кредитов

**200 GB**

Сырых данных

**10 ч**

На обучение модели

## Инференс модели

**25 млн**

Клиентов

**1.5 млрд**

Исторических кредитов

**500 GB**

Сырых данных

**2 ч**

Инференс



# Этапы продакшн расчета

**01**

**Сбор сырых данных  
из Hadoop**

**02**

**Предобработка данных  
с помощью PySpark и Python**

**03**

**Инференс  
модели**



Apache  
**Airflow**

# Планы. Единая модель на нескольких доменах

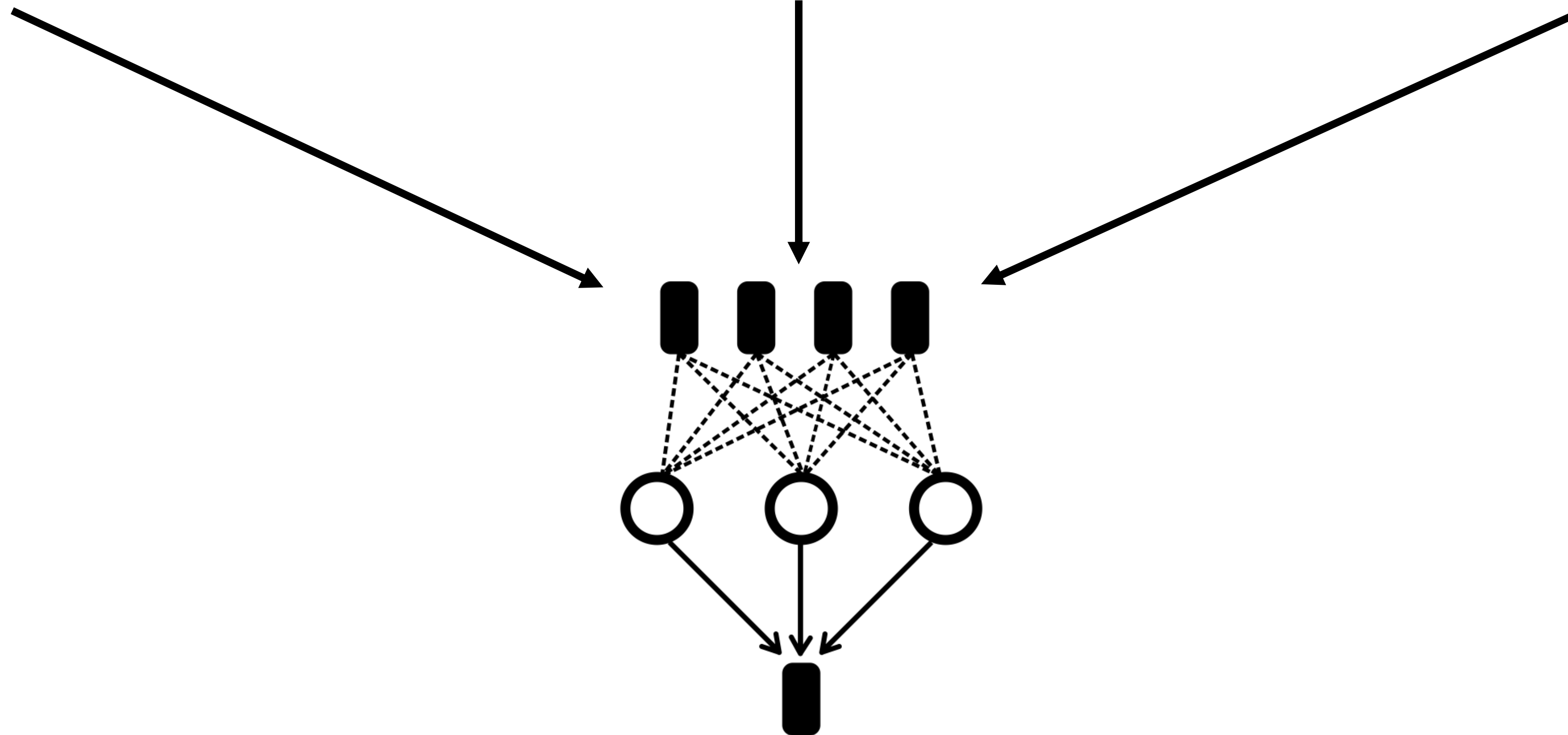
Карточные  
транзакции



Транзакции расчетного  
счета



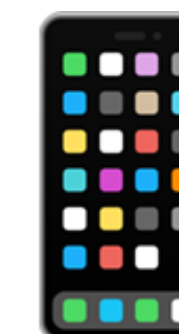
Данные  
БКИ



# Проработка новых данных и задач

## Данные

- Чеки
- История коммуникаций
- Логи мобильного приложения и сайта



## Задачи

- Модели склонности ко всевозможным продуктам
- Модели для Юридических Лиц



# ОСТАВАЙТЕСЬ НА СВЯЗИ

**Подписывайтесь**  
на наш tg-канал  
«Нескучный Data Science»



# СПАСИБО! ВОПРОСЫ?

**Фирстов Алексей**  
Data Scientist  
Альфа-Банк



**@Aleksey\_Firstov**

