САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Направление: 02.03.02 «Фундаментальная информатика и информационные технологии
ООП: Программирование и информационные технологии

ОТЧЕТ О НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

Тема задания: Разработка системы скоринга доверия к клиентам хостинговой компании

Выполнил: Докиенко Денис Александрович 17.Б-13-пу
Фамилия И. О. номер группы

Руководитель научно-исследовательской работы: кандидат физ.-мат. наук Корхов Владимир Владиславович
ФИО, должность, ученая степень

Санкт-Петербург
2020

Содержание

Введение

На данный момент одной из важнейших задач хостинговых компаний является пресечение недобросовестного использования предоставляемых ими ресурсов. Используя арендуемые ресурсы клиенты могут:

- Устраивать атаки на сторонние веб-ресурсы, чаще всего это атаки типа DDOS;
- Рассылать спам-сообщения на e-mail.

При этом компания может понести следующие издержки:

- Урон репутации компании;
- Ресурсы на ручную обработку жалоб пострадавших пользователей сети Internet;
- Блокировка сторонними сервисами IP-адресов, принадлежащих компании. Например, проблемой становится попадание IP-адреса в почтовые blacklist-ы, а так же блокировка на уровне отдельных узлов или сегментов сети.

В связи с этим, компании, предоставляющие услуги аренды разделяемой инфраструктуры, должны реализовывать анализ и мониторинг исходящего трафика своих клиентов. После детектирования аномальной активности сервисов клиента необходимо принять решение о дальнейших действиях по отношению к клиенту. Можно применить один из следующих подходов:

- Уведомление клиента о происходящем;
- Мгновенная блокировка ресурсов клиента.

При этом, ручной разбор каждого инцидента, как показывает практика, может стоить слишком больших ресурсов. Это означает, что необходимо максимально возможным образом автоматизировать процесс принятия решений о мгновенной блокировке ресурсов клиента.

Логично предположить, что на данное решение может влиять не один параметр и в том числе можно использовать степень доверия к клиенту (score). В рамках данной работы будет рассмотрена именно задача скоринга доверия к клиенту.

Формальная постановка задачи

Для решения поставленной задачи используются данные, предоставленные одной из компаний об использовании их сервиса аренды облачных серверов.

На данный момент описанная система уже реализована, в том числе реализован и алгоритм скоринга клиентов, который будет описан ниже.

В целом, упрощая, схема обработки случаев недобросовестного использования ресурсов выглядит следующим образом (Рис. 1):

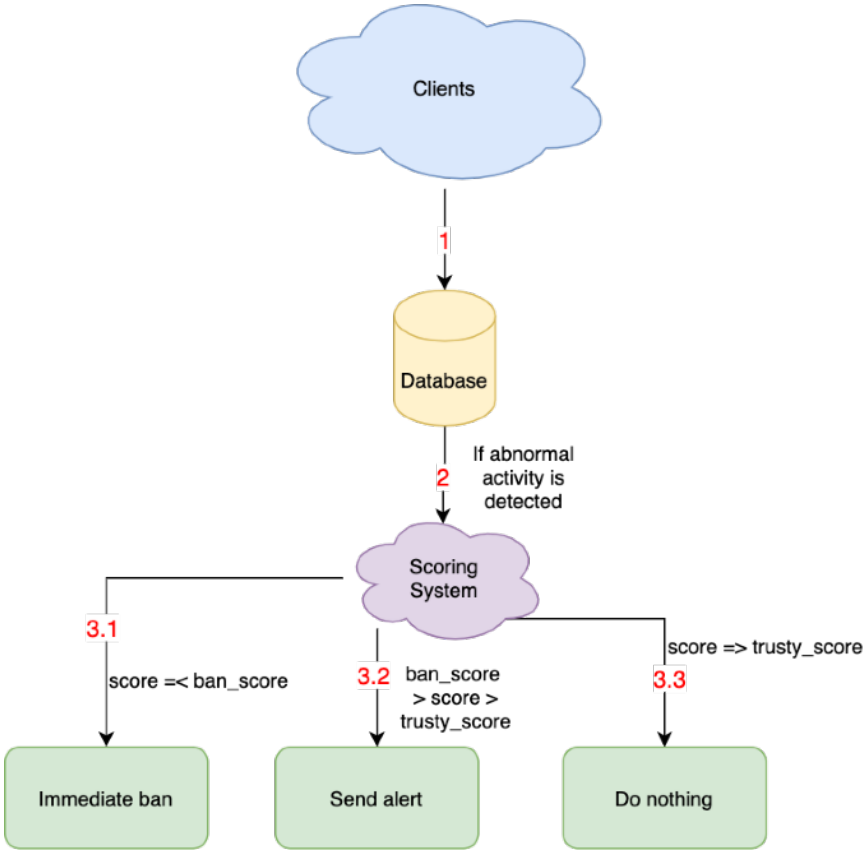

Рис.1 - Схема обработки

Поэтапно на данной схеме отражено следующее:

1. Получение и запись данных об активности системы клиента
2. Если обнаружена аномальная активность, то запрашивается информация о скоринге клиента (на самом деле реализована более сложная схема и информация о скоринге запрашивается не всегда)
3. После получения скоринга клиента
    - 3.1. Если скоринг меньше ban_score, то ресурсы клиента немедленно блокируются
    - 3.2. Если скоринг средний между пороговых значений ban_score и trusty_score, то отправляется уведомление ответственным специалистам
    - 3.3. Если скоринг больше trusty_score, то никаких действий не производится

Система скоринга доверия к клиентам на данный момент реализована в виде стандартного экспертного алгоритма. Используется набор булевых параметров, описывающих клиента (раскрыть которые в рамках данной работы не представляется возможным), взвешенная сумма которых и формирует скоринг. Таким образом, существующую систему можно представить следующей формулой:
$score = \sum_i w_i * p_i$, где $w_i = [0,1]$ - вес, и $p_i = \{0,1\}$ - параметр $(0 = false, 1 = true)$

Веса w_i были подобраны на основе эмпирического опыта экспертами компании.

Экспертные системы, как правило, обладают рядом недостатков, в том числе:

- Сложность подбора коэффициентов;
- Статичность - система не может обучаться на основе случаев ошибочного срабатывание, модификация системы должна производиться вручную и требует ручного анализа результатов работы экспертом.

В данной работе необходимо построить систему, устраняющую указанные недостатки. Для этого будут использоваться алгоритмы машинного обучения. Далее будет описан процесс получения и предобработки данных, а так же анализ существующих алгоритмов и их последующее применение для решения задачи.

Результаты исследований и код расположены в открытом доступе по ссылке [3]. Полученные датасеты из-за ограничений накладываемых законом о персональных данных и коммерческой тайной выложить в открытый доступ не представляется возможным.

Цели исследования

Целями работы являются:

- Исследование и анализ имеющихся данных клиентов компании, выбор потенциальных признаков и целевой переменной, формальное формулирование задачи
- Исследование и анализ подходящих алгоритмов машинного обучения
- Сбор и подготовка данных, проведение препроцессинга
- Применение выбранных алгоритмов на данных
- Анализ полученных результатов, сравнение полученных решений между собой и с имеющимся экспертным алгоритмом, а так же выявление перспектив дальнейшей работы

Выбор данных о клиентах

Система скоринга доверия к клиентам в данном случае должна корректно принимать решение о необходимости блокировки аккаунта клиента. В связи с этим логично выбрать в качестве целевой переменной для построения модели информацию о том, заблокирован ли в данный момент аккаунт (blocked).

Если аккаунт на данный момент заблокирован, то это значит, что клиент не смог доказать правомерность использования ресурсов компании. Если же аккаунт не заблокирован, то, даже несмотря на возможные подозрительные инциденты, клиент в итоге оказался добросовестным.

В качестве признаков, в том числе решено использовать следующие параметры клиента:

- is_phone_bad - принадлежит ли телефон аккаунта клиента к числу «подозрительных». Для расчёта данного параметра используется существующий алгоритм компании.
- email_domain - домен прикрепленного к аккаунту e-mail адреса. Можно сказать, что почтовые аккаунты, созданные на распространенных сервисах (yandex, mail, google и т.п.), являются более доверенными. Это связано с тем, что идентифицировать личность владельца такого аккаунта проще, чем, к примеру, владельца почты на любом из сервисов one-time-email, а следовательно злоумышленник скорее воспользуется вторым типом почтовых адресов.
- is_email_bad - принадлежит ли почтовый адрес аккаунта клиента к числу «подозрительных». Для расчёта данного параметра так же используется существующий алгоритм компании.
- is_first_paymant_by_paypal - произведен ли первый платеж аккаунта через PayPal. Сервис PayPal, в отличии от других сервисов, доступных для оплаты услуг компании, обеспечивает высокий уровень анонимности, поэтому злоумышленник скорее воспользуется им для оплаты. Рассматривается только первый платеж в связи со сложностью получения полной билинговой информации по клиентам.

Для сбора данных использовались существующие API компании. Так же на данном этапе был произведен первый этап препроцессинга. Такие данные, как данные об ИНН, БИК и подобные параметры клиентов были преобразованы в булевы переменные типа is_inn_exist, is_bik_exist и подобные.

Выбор алгоритмов

Поскольку целевая переменная принимает всего два значения: 0 (клиент не заблокирован) или 1 (клиент заблокирован), задачу можно формализовать как задачу бинарной классификации. Так же важно отметить, что фактически все собранные признаки являются категориальными. Из этого следует, что необходимо выбирать алгоритмы хорошо работающие именно с признаками данного типа.

Как правило, оптимальное соотношения качества и сложности реализации обеспечивают различные алгоритмы ансамблирования более простых алгоритмов. Именно такие алгоритмы и будут рассмотрены в рамках данной работы. В качестве «простого алгоритма» будем рассматривать решающие деревья. Несмотря на все их минусы, в том числе сильную подверженность переобучению, их обьединение часто дает очень хорошие результаты.

Градиентный бустинг

Первым рассмотрим алгоритм градиентного бустинга. Важной особенностью данного алгоритма является то, что для него не нужно очищать и нормализовать датасет. К тому же, он вполне неплохо работает с категориальными признаками. В том числе, существуют реализации, работающие с категориальными признаками «из коробки».

Градиентный бустинг — это техника машинного обучения для задач классификации и регрессии, которая строит модель предсказания в форме ансамбля слабых

предсказывающих моделей (в данном случае - из деревьев). Алгоритм последовательно строит деревья так, что каждое следующее дерево улучшает качество всего ансамбля. Таким образом, градиентный бустинг можно представить в следующем виде:

$$f(x) = h_0 + v \sum_{j=1}^{M} h_j(x), \text{ где}$$

h_0 - некоторая константная модель (начальное приближение);

$v \in (0,1]$ - параметр, регулирующий скорость обучения и влияние отдельных деревьев на всю модель;

$h_j(x)$ - деревья решений.

Новые слагаемые-деревья добавляются в сумму путем жадной минимизации эмпирического риска, заданного некоторой функцией потерь $L(y, y') = L(y, f(x))$.

Random forest

В отличии от рассмотренного ранее алгоритма градиентного бустинга, в данном алгоритме ассемблирование решений осуществляется через бэггинг:

Пусть обучающая выборка состоит из *N* образцов, размерность пространства признаков равна *M*, и задан параметр *m* (в задачах классификации обычно $m \approx \sqrt{M}$) как неполное количество признаков для обучения.

1. Сгенерируем случайную подвыборку с повторениями размером из обучающей выборки;
2. Построим решающее дерево, классифицирующее образцы данной подвыборки, причём в ходе создания очередного узла дерева будем выбирать набор признаков, на основе которых производится разбиение (не из всех M признаков, а лишь из m случайно выбранных);
3. Дерево строится до полного исчерпания подвыборки и не подвергается процедуре прунинга.

Оптимальное число деревьев подбирается таким образом, чтобы минимизировать ошибку классификатора на тестовой выборке.

Таким образом, в отличии от метода бустинга, деревья строятся не на основании данных о предыдущих деревьях, а на некоторой подвыборке признаков, за счет чего деревья в ансамбле сильно отличаются друг от друга.

Изучение данных и обработка данных

Для обучения моделей была получена информация о 70000 клиентах. На данном этапе были произведены следующие действия:

- email_domain был закодирован с использованием частотного кодирования. Практически для решения поставленной задачи не важно, на каком именно домене клиент создал свою почту. Важная часть содержащейся в признаке информации - это распространенность используемого домена.
- В булевых признаках is_phone_bad, is_email_bad, is_first_paymant_by_paypal, is_ip_bad наблюдалась сильная разреженность. Решено было использовать кодирование по

трем меткам - информации нет, True и False. В данном случае, отсутствие конкретной информации по признакам является так же важными данными о клиенте.

Исследование алгоритмов

Исследуем работу существующего экспертного алгоритма.

На приведенных ниже гистограммах (Рис. 2.1) и (Рис. 2.2) указаны значения скоринга, которые выдает алгоритм для заблокированных и не заблокированных клиентов.

Значения скоринга здесь и далее были округлены до значений от 0 до 1 с одним десятичным знаком.

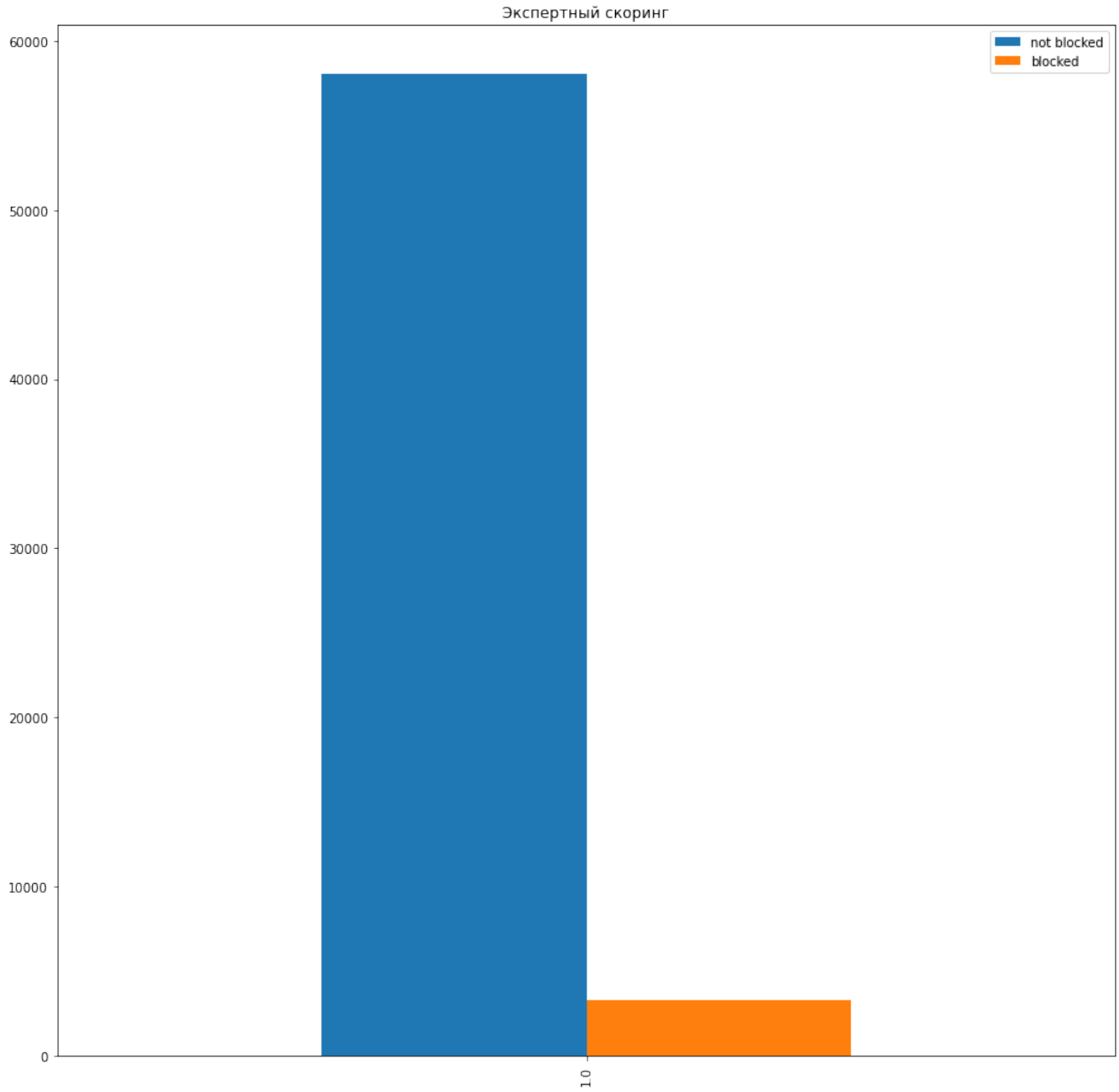

Рис 2.1 - экспертный алгоритм

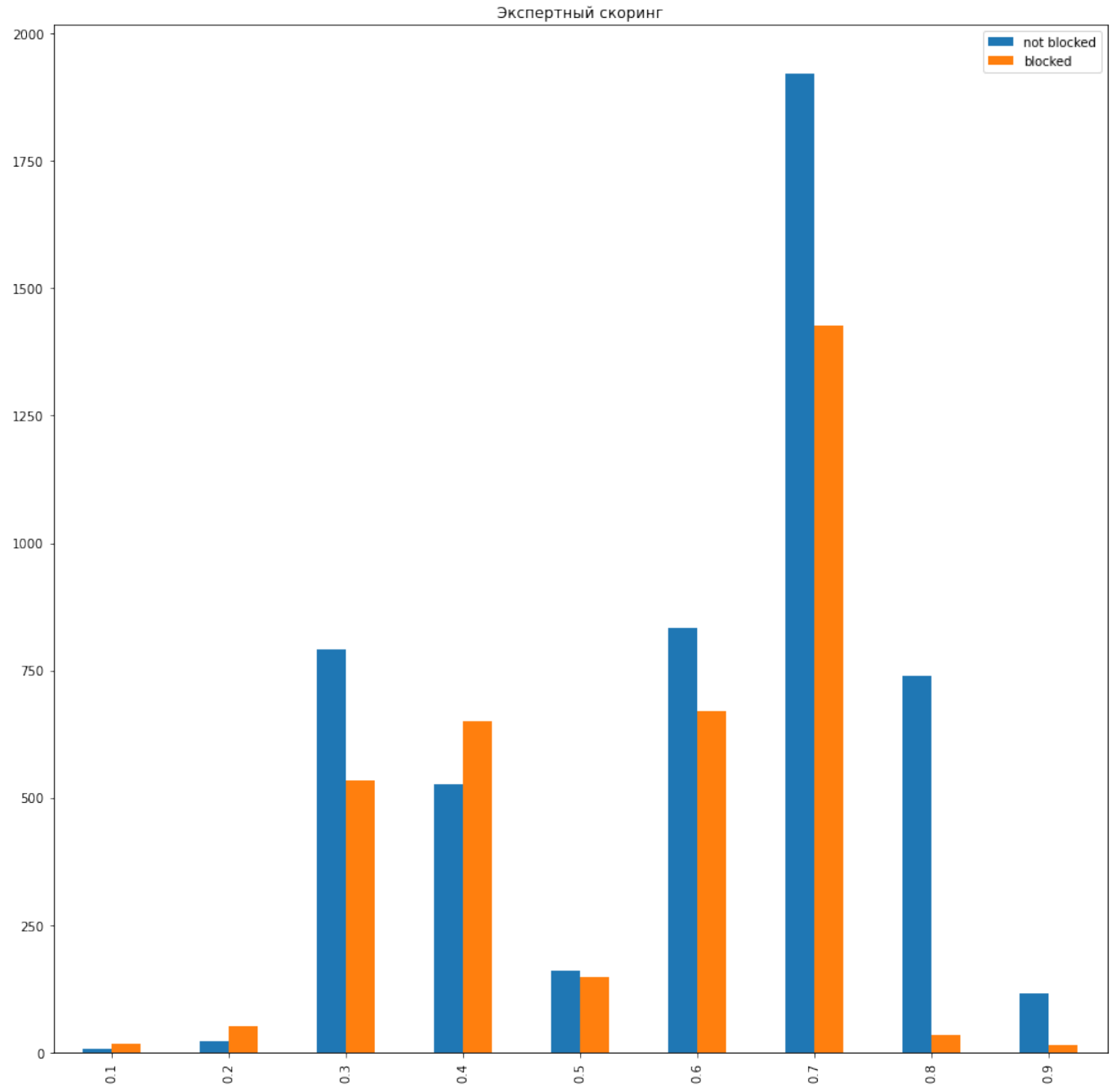

Рис. 2.2 - экспертный алгоритм

Далее рассмотрим результаты алгоритма градиентного бустинга.

Для реализации градиентного бустинга используется библиотека catboost [4], поскольку она «из коробки» поддерживает работу с категориальными данными. Для подбора гиперпараметров использовалась кросс-валидация. Далее будут рассмотрены результаты алгоритма на тестовой части выборки (Рис. 3.1 и Рис. 3.2).

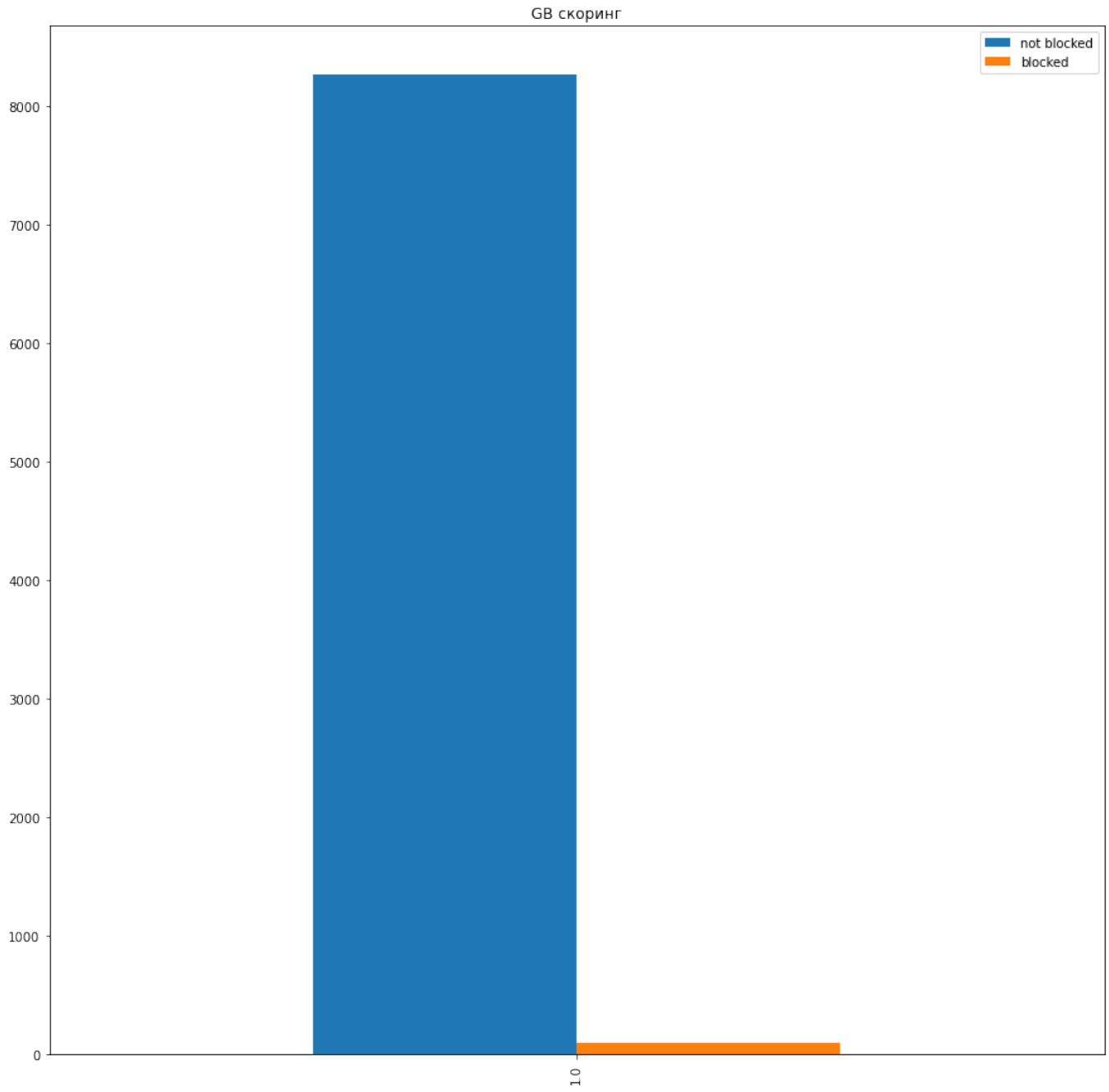

Рис. 3.1 - градиентный бустинг

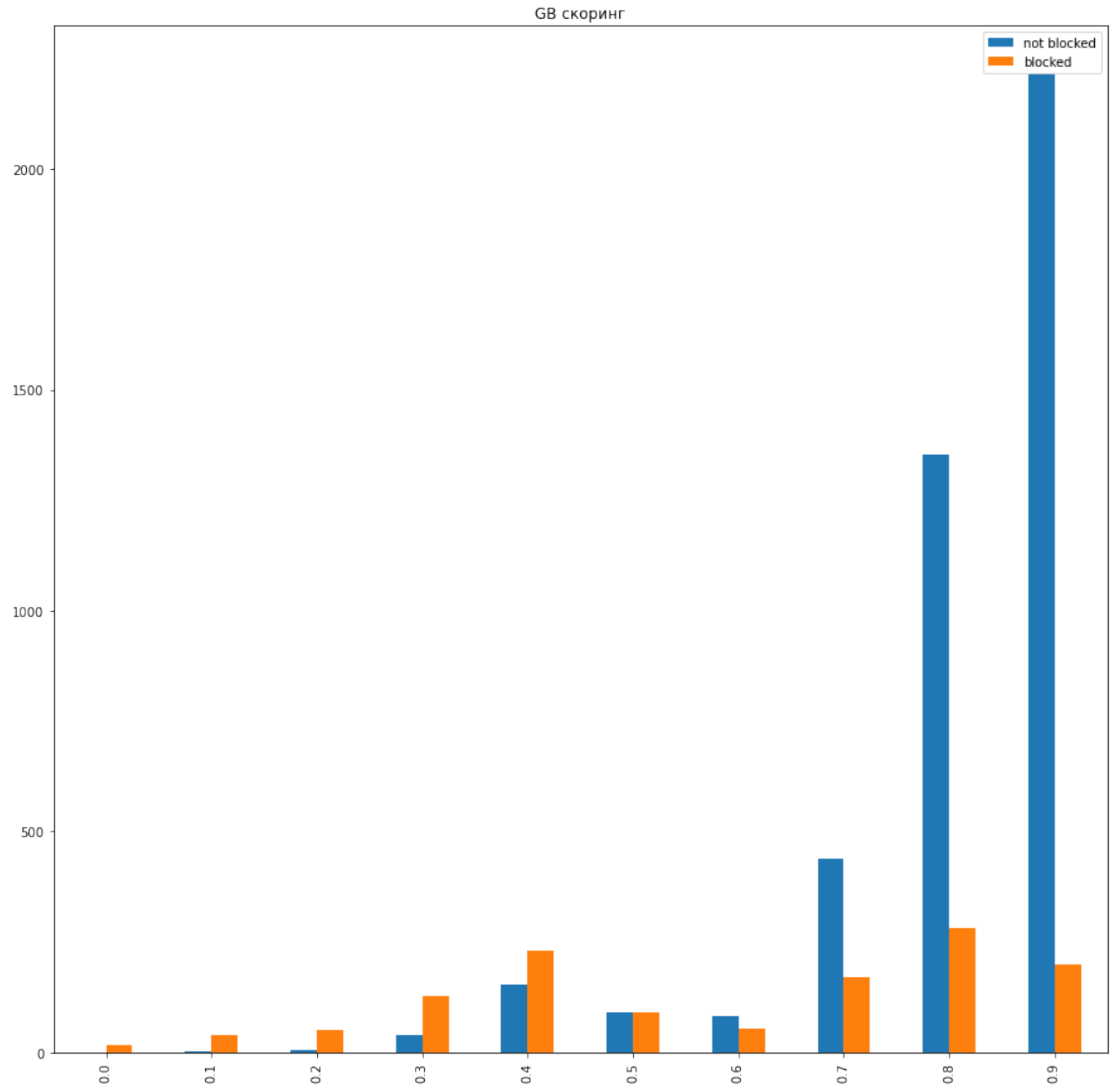

Рис. 3.2 - градиентный бустинг

Для реализации рандомного дерева используется библиотека sklearn [5]. Для поиска оптимального числа деревьев так же использовалась кросс-валидация (Рис. 4.1). Поскольку, данная модель не может работать с категориальными признаками, было произведено кодирование по методу one-hot-encoding с дополнительным столбцом для отсутствующих значений. Далее приведены результаты так же в виде гистограмм (Рис. 4.2 и Рис. 4.3).

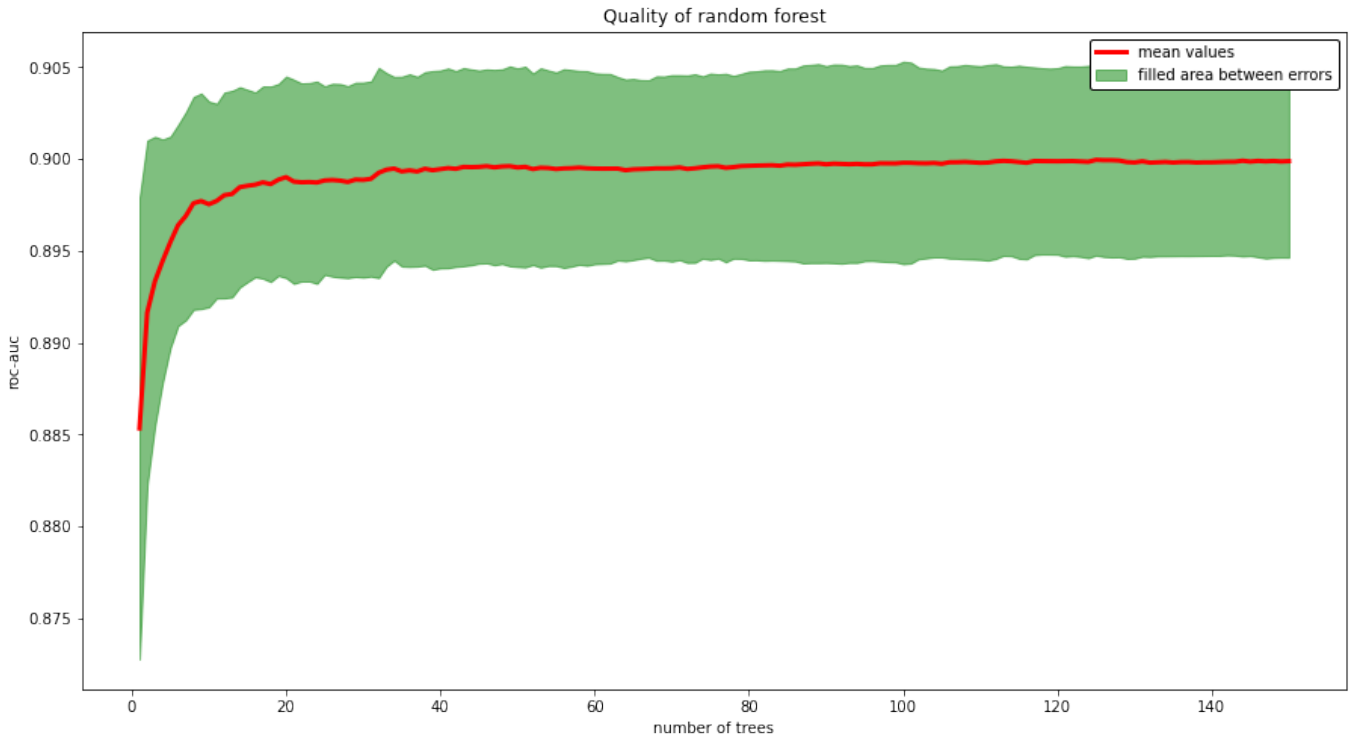

Рис. 4.1 - подбор оптимального числа деревьев

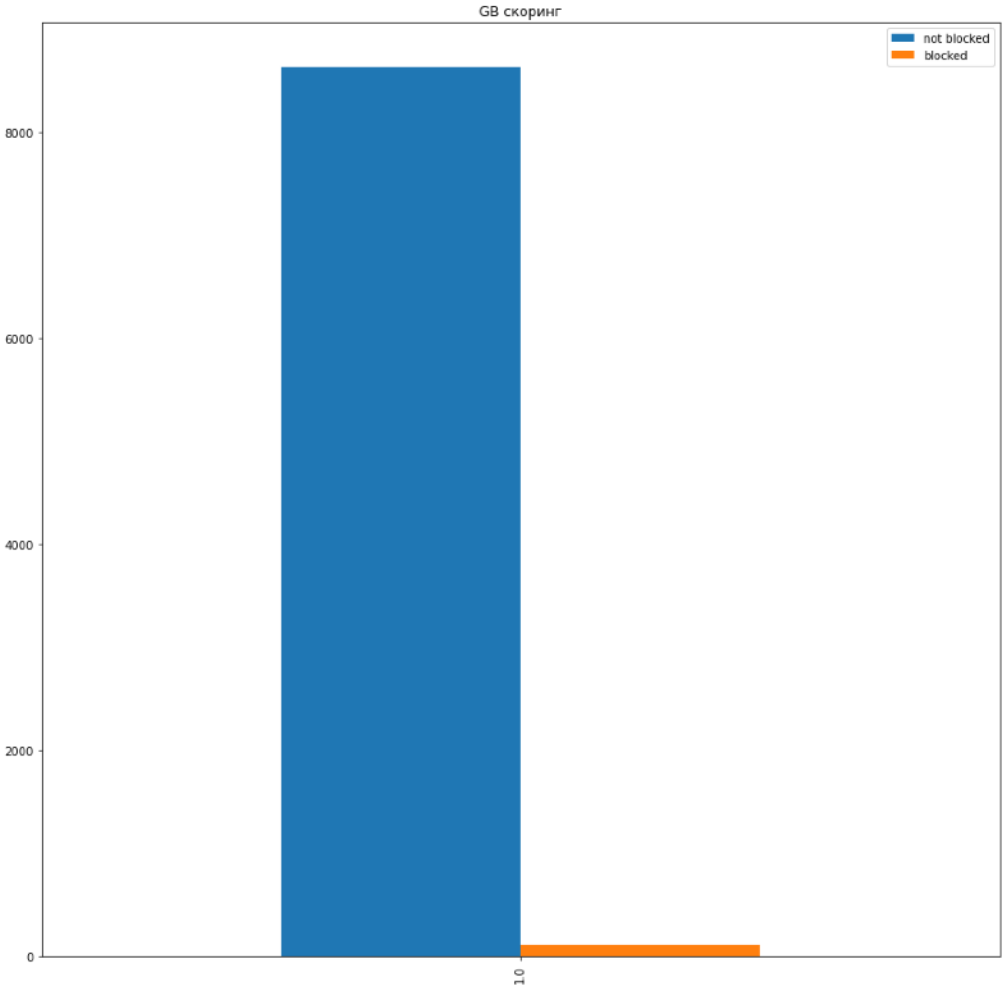

Рис 4.2 - рандомный лес

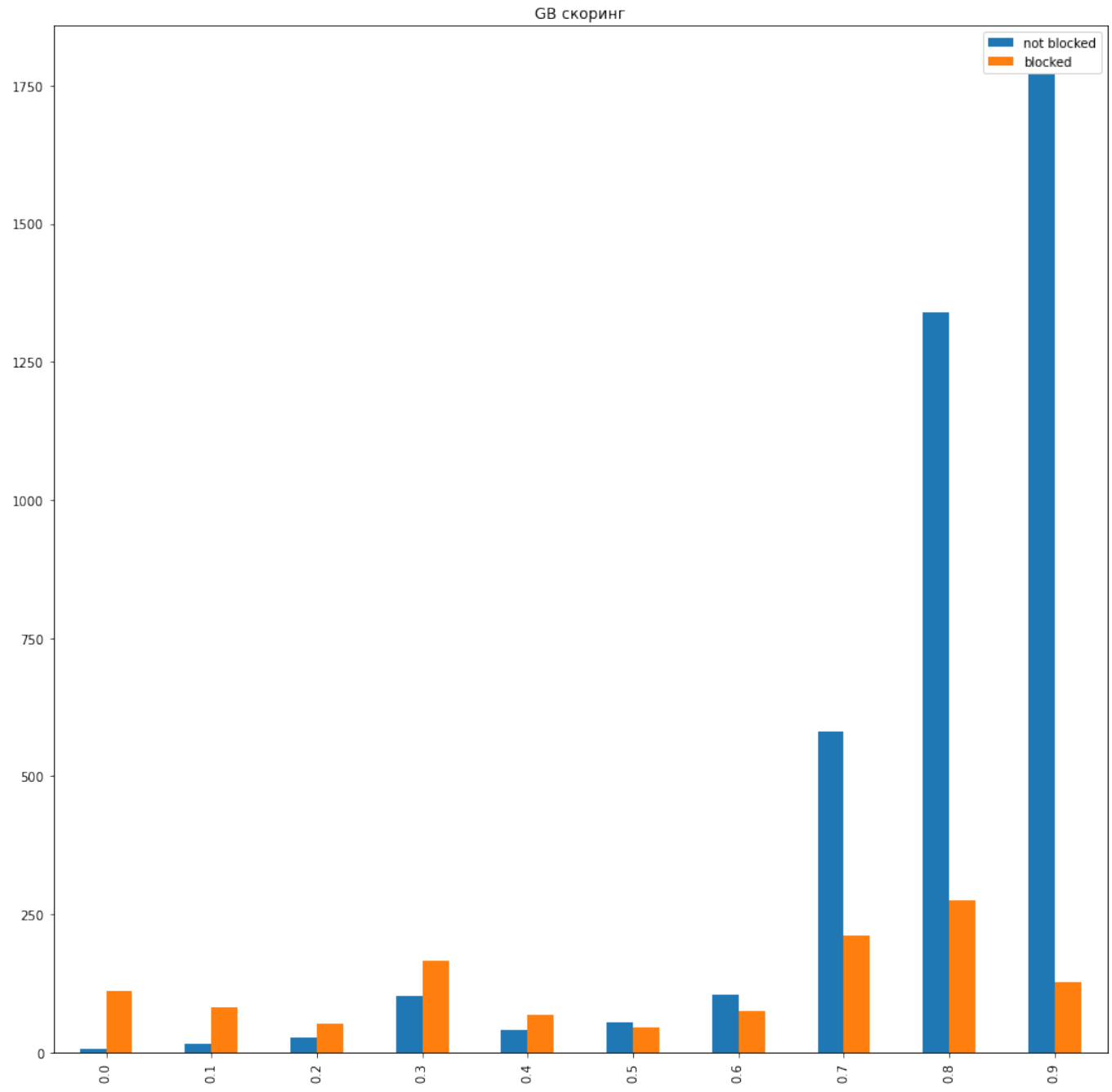

Рис 4.3 - рандомный лес

Так же можем сравнить средние результаты моделей на тестовых данных по некоторым метрикам при проведении пяти итераций обучения:

Модель	**ROC-AUC**	**Logloss**	**accuracy**
Градиентный бустинг	0.901	0.2	0.923
Случайный лес	0.894	0.274	0.925

Выводы

Видно, что результаты обеих моделей получились относительно близкими и сильно превосходят в качестве предсказания экспертную модель. При этом важно учесть, что для построения моделей не пришлось вручную выставлять веса признаков, как необходимо было сделать при использовании экспертной модели. Так же важно отметить, что у моделей построенных с применением методов Machine Learning куда больше возможностей к дальнейшему масштабированию по количеству используемых параметров.

В дальнейшем необходимо провести исследование с использованием большего числа параметров и самих рассматриваемых методов.

В заключение хочется отметить, что модель основанная на методе градиентного бустинга показала себя лучше, чем модель на случайном лесе. Она дает чуть лучшие результаты при нескольких итерациях обучения, меньше подвержена переобучению и проще в использовании за счет поддержки категориальных признаков без необходимости дополнительного кодирования.

Список источников

[1] Результаты исследования и исходный код в открытом доступе [Электронный ресурс] //URL: https://github.com/sncodeGit/cl_score

[2] Библиотека для реализации градиентного бустинга [Электронный ресурс] //URL: https://catboost.ai/

[3] Библиотека для реализации случайного леса [Электронный ресурс] //URL: https://xgboost.ai/

[4] Дипломная работа - «Machine learning with categorical features», автор - Фонарев Александр Юрьевич, научный руководитель - д.ф.-м.н., профессор Дьяконов Александр Геннадьевич [Электронный ресурс] // URL: http://www.machinelearning.ru/wiki/images/9/99/Diploma_fonarev.pdf

[5] Machine learning mastery [Электронный ресурс] // URL: https://www.machinelearningmastery.ru

[6] Хабр [Электронный ресурс] // URL: https://habr.com/

[7] Pierre Geurts, Damien Ernst, and Louis Wehenkel. Extremely randomized trees.Machine Learning, 63(1):3–42, 2006.

[8] А.Г. Дьяконов. Методы решения задач классификации с категориальными признаками.Прикладная математика и информатика, 46, 2014