Министерство науки и высшего образования
Российской Федерации
«Московский физико-технический институт
(государственный университет)»
Физтех-школа прикладной математики и информатики
Факультет управления и прикладной математики
Кафедра «Интеллектуальные системы»

Кулага Роман Александрович

Классификация потока финансовых новостей с целью выявления динамики цен биржевых инструментов

03.04.01 — Прикладные математика и физика

Выпускная квалификационная работа
(магистерская диссертация)

Научный руководитель:
д. ф.-м. н. Воронцов К.В.

Москва
2018

Аннотация

Рассматриваемая в рамках данной работы задача классификации новостного потока имеет широкое практическое применение в области биржевой торговли. Однако, построение высококачественной модели классификатора в силу ряда обстоятельств является нетривиальной задачей. Для решения задачи создана модель классификации, в которой объекты формируются путем агрегации новостных заголовков, а их признаковое описание строится на основе битермов. Разметка для обучения классификатора производится автоматически и представляет из себя отдельную модель со своими параметрами. Полученные в результате исследования результаты удовлетворяют необходимым критериям качества и позволяют применять модель на практике.

Содержание

Введение

Нет никаких сомнений в том, что большинство крупных компаний, имеющих дело с торговлей на бирже, так или иначе используют новости для принятия решений о совершении сделок. Это касается как крупных инвестиционных фондов, поддерживающих актуальный финансовый портфель, так и менее крупных игроков, включая независимых трейдеров, занимающихся внутридневной торговлей. Лидирующие в области количественного анализа компании также закладывают использование новостей при написании некоторых торговых роботов, заключающих сделки в автоматическом режиме без непосредственного участия человека.

При этом результаты исследований и созданные для этого инструменты, как правило, находятся внутри каждой компании под NDA и найти в открытом доступе данные материалы возможности нет.

Целью данной работы является построение модели предсказания скачов цены финансовых инструментов по новостному потоку.

Недостижимый идеал с потенциально широкой областью применения — модель регрессии, которая, анализируя текущий поток новостей, могла бы с высокой точностью предсказать, как поведет себя цена в ближайшем будущем, причем как в количественном (насколько велико отклонение), так и в качественном (падение или рост) плане. Построить такую модель, разумеется, не выйдет хотя бы по той причине, что в ряде случаев отклонения цен либо вовсе не имеют, либо имеют обратную причинно-следственную связь с новостями (цена изменилась, после чего начали публиковаться соответствующие новости).

Другая сложность построения идеальной модели кроется в том, что правильно провести процедуру ее обучения весьма нетривиально. Во-первых, в случае, когда объекты строятся исключительно на текстовом описании новостей, имеет место ситуация, что похожие объекты могут в разных случаях как приводить к скачку цен, так и не приводить (пример — регулярные финансовые отчеты: если ожидания трейдеров совпадают с опубликованной информацией, реакции на новость не последует; в противном случае можно ожидать движений цены). Во-вторых, требуется на базе новостного потока каким-то образом формировать для алгоритмов объекты и, что более важно, размечать их (в случае регрессии это некоторая количественная характеристика, а для классификации — разметка на классы), причем принцип формирования этой разметки — отдельная модель со своими параметрами. Так как разметка производится автоматически, часть объектов неизбежно будет размечена с некоторой ошибкой.

Если же говорить про менее сложные модели (имеющие больший шанс на успешное построение), можно научить алгоритм выделять моменты времени, в которые начинается значительное движение цены. При достаточно хороших результатах такой инструмент можно использовать в различных целях, например:

- подсказывать трейдеру, что «сейчас стоит обратить внимание на определенный инструмент» (имеется возможность заработать)

- «предостеречь от торговли», так как цена может резко измениться (актуально для некоторых типов стратегий в высокочастотной торговле, а также в ряде других случаев при ручном заключении сделок, когда резкие изменения цены крайне нежелательны)
- использовать как часть более сложного торгового робота (результат классификации подается на вход как признак)

Структурно работа состоит из введения, двух основных глав, выводов и заключения. Основными являются главы:

1. «Постановка задачи». В главе описаны исходные данные и их базовая обработка, а также постановка задачи классификации, общая для дальнейших экспериментов.
2. «Модели и эксперименты». В первой секции главы описывается самая простая модель, в дальнейших секциях — модификации предыдущих моделей, позволяющие улучшить качество классификации. Для каждого эксперимента помимо основной идеи также приведены полученные результаты.

1. Постановка задачи

1.1. Исходные данные

В качестве исходных текстовых данных были использованы новостные заголовки из ряда крупных финансово-политических источников. Для всех новостей известны: источник, время публикации с точностью до секунды (приведенное к часовому поясу UTC), а также текст заголовка. Использовались новости, написанные исключительно на английском языке.

В качестве данных по биржевым инструментам использовались исторические данные по тикерам с «Yahoo Finance» [1]. Данные представлены в виде csv файлов, в которых для каждого момента времени (с точностью до секунды) и для каждого финансового инструмента записана информация о последней цене торговых сделок, их общий объем и количество в эту секунду. Стоит также иметь в виду, что многие инструменты торгуются на разных биржах и имеют различное расписание торгов.

При обучении и оценке качества моделей были использованы тексты новостных заголовков и данные по ценам биржевых инструментов за полгода — с 1 июля по 31 декабря 2016 года. Все дальнейшие результаты в данной работе были получены с использованием цен по инструменту «OilBrent» на бирже «ICE».

При обработке данных и построении моделей были использованы языки Python (и ряд инструментов, в том числе описанные в [2], [3], [4], [5]) и C++11 [6]

1.2. Признаковое описание документов

Все новостные заголовки предварительно обрабатываются следующим образом:

1. Текст заголовка разбивается на слова, числа и знаки пунктуации как отдельные единицы.
2. Вся пунктуация из текста удаляется. Практически никакой потери информации при этом не происходит, ведь новостные заголовки имеют небольшую длину и как правило содержат лишь одну мысль.
3. Все слова в тексте приводятся к одному регистру.
4. Все слова подвергаются процедуре лемматизации — приведения к словарной форме.
5. Производится процедура фильтрации стоп-слов по уже сформированному словарю.
6. Для всех оставшихся слов производится процедура токенизации. Единственная особенность: каждый раз, когда в тексте встречается частица «не», она объединяется с последующим словом, таким образом формируя единый токен «не» + «слово». Числа при этом не удаляются — все числа в дальнейшем представлены одинм отдельным токеном.

Таким образом, если обозначить за N_d число новостных заголовков, за N_{uni} количество различных токенов (униграмм), всю коллекцию новостных заголовков можно представить в виде сильно разреженной матрицы $D \in \mathbb{Z}_+^{N_d \times N_{uni}}$, где каждая отдельно взятая новость представлена разреженным вектором $d \in \mathbb{Z}_+^{N_{uni}}$. При этом $D_{i,j}$ представляет собой частоту встречаемости j-го терма в i-ом документе.

1.3. Автоматическая разметка документов на классы

Для обучения модели классификатора, оценивающего динамику цен биржевых инструментов, нам необходимо некоторым образом уметь размечать новостной поток. При этом специфика такова, что в случае простых объектов (отдельных новостей) их количество и разнообразие очень велико, а в случае менее тривиальных объектов их содержание слишком сложно для интерпретирования человеком, так что произвести ручную разметку на классы не представляется возможным. Поэтому в текущей работе предлагается несколькими альтернативными способами формировать разметку на базе биржевых данных по ценам.

Так как для всех новостей имеется точное время их публикации, его можно использовать для связи с изменениями цен.

Как правило, на длительных промежутках времени (порядка часа) у цены есть некоторый тренд (на спокойных участках можно в первом приближении считать, что он линейный). Так как нас интересуют относительно короткие интервалы (порядка минут), на которых цена меняется резко, требуется некоторым образом избавиться от трендовой составляющей в цене. Для этого предлагается в каждый момент времени вычитать из цены скользящее среднее за последний час.

Везде в дальнейшем под ценой инструмента будем подразумевать цену, из которой трендовая составляющая уже вычтена. Такой подход требует также помимо неторговых часов исключать из рассмотрения первый час торговой сессии, но позволяет лучше отлавливать неожиданные скачки цен даже в те дни, когда трендовая составляющая роста или падения цены велика.

Независимо от того, какие объекты на базе новостных заголовков мы в дальнейшем будем рассматривать, очень важно для каждого момента времени уметь оценивать некоторую величину, характеризующую движение цен в последующем за ним интервале времени. Для ясности обозначим текущий рассматриваемый момент времени за t_0, длительность последующего интервала за ΔT, цену в момент времени t за $P(t)$, объем сделок — за $V(t)$. В качестве функции, характеризующей движение цен $\Delta P(t_0, \Delta T)$ в интервале между t_0 и $t_0 + \Delta T$, можно использовать следующие функции:

- $\Delta P_1(t_0, \Delta T) = \max\limits_{\delta t = 1, \ldots, \Delta T} \frac{|P(t_0+\delta t) - P(t_0)|}{P(t_0)}$, т.е. максимальное по модулю относительное отклонение цены
- $\Delta P_2(t_0, \Delta T) = \left| \sum\limits_{\delta t=1}^{\Delta T} \frac{[P(t_0+\delta t) - P(t_0)]}{P(t_0)} \cdot V(t_0 + \delta t) \right|$, т.е. взвешенный объемами сделок модуль суммы относительных отклонений цены

С использованием цен, количеств и объемов сделок также было опробовано несколько других функций, но они так или иначе похожи на один из двух предложенных вариантов, поэтому не приведены в списке.

В дальнейшем для простоты будем называть выбранную функцию $\Delta P(t_0, \Delta T)$ «величиной скачка» цены.

Разметку на классы «0» и «1» предлагается производить пороговым правилом: если $\Delta P(t_0, \Delta T) > P_{threshold}$, то будем считать, что моменту времени соответствует класс «1» (будем говорить, что в такие моменты наблюдается «скачок» цены). В противном случае — класс «0» (т.е. «скачка» цены не наблюдается)

На практике был получен результат, что для двух конкретных реализаций величины скачка $\Delta P_1(t_0, \Delta T)$ и $\Delta P_2(t_0, \Delta T)$ разбиение моментов времени по классам имеет минимальные отличия (сходство $\approx 98\%$ для порогов, дающих одинаковое процентное соотношение классов «1» и «0»).

В то же время функцию вида $\Delta P_1(t_0, \Delta T)$ можно без труда пересчитывать при переходе от t_0 к $t = t_0 + 1$ за $O(1)$ в смысле амортизированной алгоритмической сложности. Альтернативная функция устроена сложнее, поэтому ее пересчет на каждом шаге требует $O(\Delta T)$ операций. Именно поэтому везде в дальнейшем предлагается использовать в качестве величины скачка функцию $\Delta P(t_0, \Delta T) = \Delta P_1(t_0, \Delta T)$

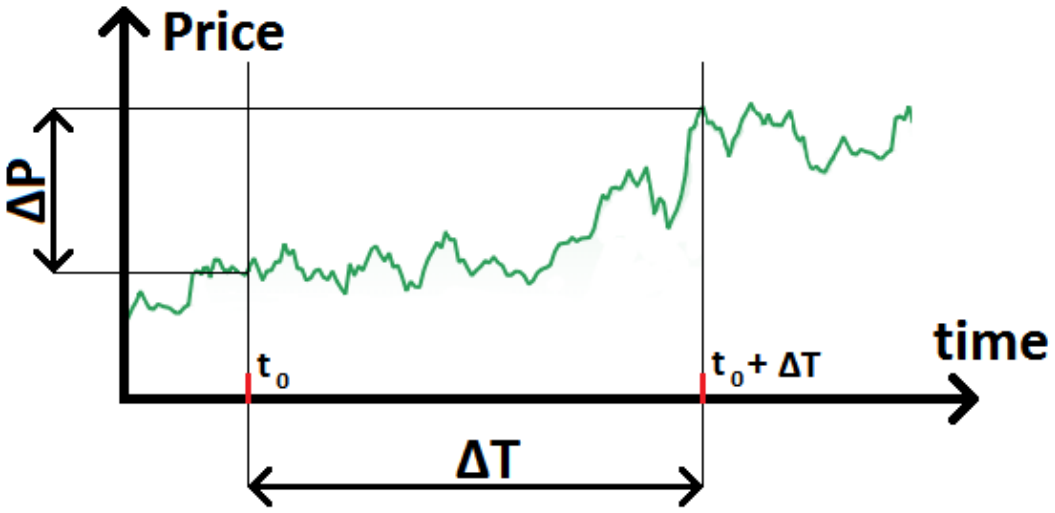


Рис. 1: Величина скачка цены $\Delta P(t_0, \Delta T)$

1.4. Задача классификации

Конечная задача классификатора — глядя на текущее состояние новостного потока оценить, как он соотносится с ценами.

В дальнейшем будет рассматриваться два разных типа объектов:

- Мешок слов соответствующего документа, взвешенный с использованием TF-IDF. Пусть N_{feat} — количество некоторых вещественных признаков, d — документ; тогда признаковым описанием объекта будем считать сильно разреженный вектор $d \in \mathbb{R}^{N_{feat}}$

- Набор самых важных категориальных признаков документа (понятие «важности» зависит от того, рассматриваем мы униграммы или битермы, подробнее в соответствующих разделах). Пусть N_{cat} — максимально разрешенное количество категориальных признаков для описания одного документа, d — документ; тогда его признаковым описанием будем считать вектор $d \in \mathbb{Z}^{N_{cat}}$

Итоговым ответом классификатора является один из двух классов: «1», если имеет место движение цен и «0» в противном случае. Формирование объектов и автоматическая разметка производятся несколькими способами, подробно описанными для каждого эксперимента.

1.5. Измерение качества решения

Предлагается в качестве метрик качества использовать ROC-кривую, площадь под ее графиком (ROC-AUC), а также PrecisionRecall-кривую и соответствующую площадь под графиком (PR-AUC, или же AP — Average Precision).

Оценка ROC-AUC [7] имеет смысл, если классификатор используется в качестве предупреждающей системы, которая сообщает, что, вероятно, в скором времени цена сильно изменится. При этом хочется избежать большого числа ложных положительных срабатываний (минимизируя False Positive Rate) и не пропустить настоящие срабатывания (максимизируя True Positive Rate). По требованиям это близко к задаче медицинской диагностики, где для оценки широко используется метрика ROC-AUC.

С другой стороны, если же использовать результат классификации как часть автоматической торговой стратегии, то именно точность вкупе с полнотой имеют гораздо большее значение, поэтому с практической точки зрения имеет смысл также оценивать PR-AUC. На этапе эксплуатации конкретную величину порога можно выбрать, максимизируя F1-меру.

При этом также стоит иметь в виду, как между собой соотносятся ROC и PR кривые [8], а именно: если одна ROC-кривая полностью доминирует вторую, то и соответствующая первой PR-кривая также будет доминировать соответствующую второй PR-кривую. Обратное утверждение тоже верно. В случае, когда строгого доминирования нет, никаких общих выводов сделать нельзя.

Для получения количественных оценок выбранных метрик (ROC-AUC и PR-AUC) используется кросс-валидация. Выборка разбивается на K блоков, затем на каждой итерации i-ый блок используется в качестве тестового (на нем вычисляются соответствующие метрики), а обучение производится на всех остальных. По окончанию процедуры в качестве итоговых значений ROC-AUC и PR-AUC берутся усредненные по всем итерациям значения. При этом в дальнейшем все графики ROC и PR кривых приведены для медианных с точки зрения AUC итераций.

2. Модели и эксперименты

2.1. Базовая модель

2.1.1. Идея эксперимента

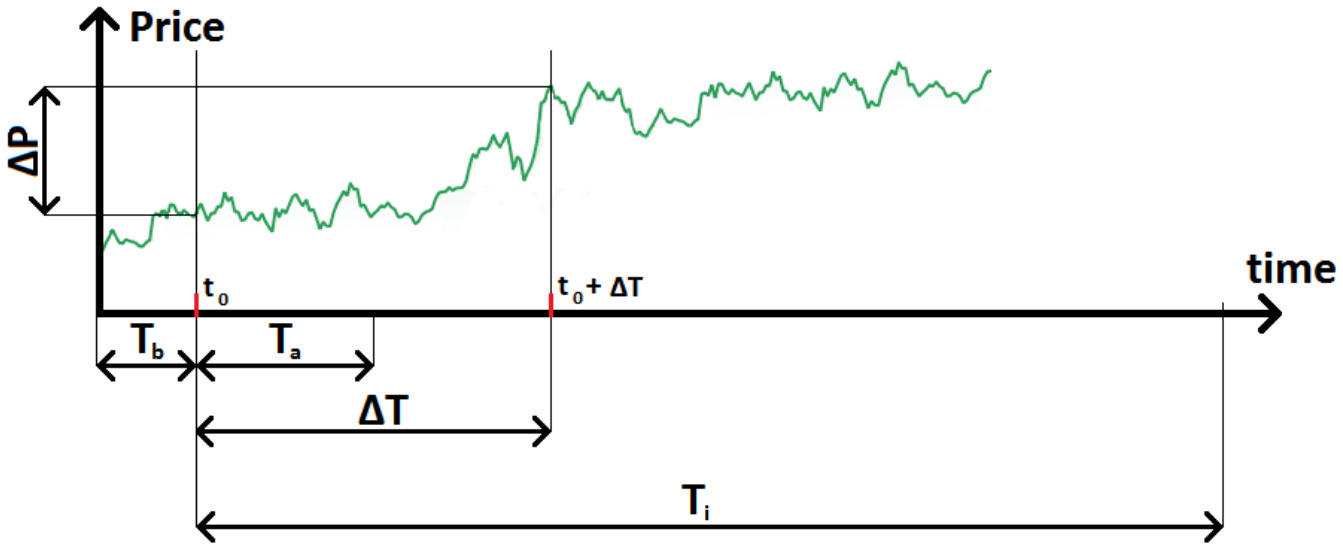


Рис. 2: Параметры автоматической разметки на классы: T_b, T_a — интервалы, соответствующие классу «1», ΔT — интервал, на котором вычисляется величина скачка цены ΔP, T_i — интервал игнорирования новостей после скачка

В текущей секции в качестве одного объекта предлагается рассматривать каждый отдельно взятый новостной заголовок.

Произведем автоматическую разметку имеющихся новостей на классы «0» и «1». Класс «0» в данном случае соответствует отсутствию движения цены, класс «1» — наличию значительного изменения цены.

Для каждой новости известно время ее публикации с точностью до секунды. Выборка содержит непрерывный поток новостей за полгода — будем двигаться вдоль этого диапазона окном, как показано на рисунке 2. Положение начала окна обозначим t_0, ширину окна ΔT, величину скачка ΔP, а также введем три параметра: T_b (от «before»), T_a (от «after») и T_i (от «ignore»).

Будем считать, что в случае, если величина скачка ΔP не превосходит заранее заданный порог $P_{threshold}$, то текущее значение t_0 для нас неинтересно и мы переходим к следующему значению.

Если же выполнено условие $\Delta P > P_{threshold}$, то обозначим все новости в интервале $[t_0 - T_b;\ t_o + T_a]$ классом «1», все новости в интервале $(t_o + T_a;\ t_o + T_i)$ удаляем из выборки, после чего переходим к очередному t_0, превосходящему $t_0 + T_i + T_b$ и продолжаем аналогичную процедуру.

После того, как обход завершен, все неудаленные и неразмеченными новости помечаем классом «0».

За таким алгоритмом разметки стоит следующее предположение: со скачком цены связана та часть новостного потока, которая непосредственно предшествует моменту начала скачка, а

также новости за некоторый интервал времени, сопровождающий скачок. Последующая часть новостей удаляется из обучающей выборки по той причине, что в случае более длительных скачков следом за размеченным интервалом может продолжаться значительное изменение цен, по факту относящиеся все к тому же скачку; для нас же важно обнаружить и выделить в класс 1 именно те новости, которые относятся к самому началу.

Перейдем к признаковому описанию объектов. Для каждой новости уже имеется описание в виде набора униграмм.

Пусть D — множество всех новостей-документов, причем каждый документ $d \in D$ состоит из некоторого набора слов-униграмм $\{w_i : w \in d\}$. Будем говорить, что $tf(w,d)$ — количество вхождений слова w в документ d, $N_d(w) = \frac{|D|}{|D_w|}$ — документная частота слова w, где $D_w = \{d : w \in d\}$, а также $idf(w) = \log(N_d(w))$

В первую очередь исключим из рассмотрения все те униграммы, документная частота которых $N_d(w)$ ниже заданного порога N_d^{min}, так как по этим словам не получится накопить достаточно статистики (величина порога выбирается небольшой). Ко всем оставшимся униграммам применим взвешивание с помощью TF-IDF [9]: будем рассматривать в качестве признакового описания документа d не вектор с частотой униграмм, а нормализованный вектор со значениями $tfidf(w,d) = tf(w,d) \cdot idf(w)$.

Особенность TF-IDF состоит в том, что наибольший вес получают слова, часто встречающиеся в текущем документе и редко — в остальных. Таким образом, TF-IDF можно рассматривать в качестве меры важности слова в документе. Стоит, однако, иметь в виду, что TF-IDF проявляет себя наилучшим образом именно на объемных документах, состоящих из большого числа слов. Документы, рассматриваемые в текущей секции — новостные заголовки и как правило имеют маленькую длину.

В качестве базовых алгоритмов классификации были выбраны Random Forest, Logistic Regression, SVM с линейным ядром (все три — из библиотеки scikit-learn [10, 11]), а также XGBoost [12, 13].

Помимо указанных ранее, здесь и далее будет использоваться алгоритм CatBoost [14, 15]. Его особенность заключается в том, что он позволяет использовать категориальные признаки. Категориальным называется признак, имеющий дискретное множество значений, причем на данном множестве может не быть отношения порядка.

Построить категориальные признаки на основе уже имеющихся можно следующим образом. Пусть каждый документ $d \in D$ описывается конечным числом N_{cat} категориальных признаков. Возьмем уже имеющееся признаковое описание документов в виде вектора значений $tfidf(w,d)$. Затем внутри каждого документа упорядочим слова $w \in d$ по убыванию в соответствии со значениями $tfidf(w,d)$, возьмем не более чем первые N_{cat} слов и запишем в качестве категориальных признаков значения токенов для этих слов w. В случае, если документ содержит менее N_{cat} слов, оставшиеся категориальные признаки заполним специальным значением «-1».

2.1.2. Результаты эксперимента

Все приведенные далее в этой секции значения метрик — усредненные результаты кросс-валидации на пяти различных блоках данных. Блоки формируются из подряд идущих новостей (без перемешивания) таким образом, что для любых двух блоков выполняется условие: все документы из одного блока имеют времена либо строго меньшие, чем времена документов из второго блока, либо строго большие. При этом граница между блоками не должна приходиться на интервал времени, соответствующий скачку цены (новости, размеченные классом «1»). Данные требования необходимы, чтобы множество похожих новостей, относящихся к одному и тому же событию и близких по времени, не оказались в разных блоках и тем самым не завысили оценки кросс-валидации.

Что касается параметров автоматической разметки — выбор был остановлен на двух конфигурациях:

1. $\Delta T = 10$ мин, $T_b = 3$ мин, $T_a = 5$ мин, $T_i = 60$ мин. При этих параметрах алгоритм должен прогнозировать скачок на десятиминутном интервале, основываясь на новостях не более чем за пять последующих от начала скачка минут. Носит предсказательный характер (оценивает будущее состояние цен).

2. $\Delta T = 3$ мин, $T_b = 3$ мин, $T_a = 5$ мин, $T_i = 60$ мин. В данной конфигурации алгоритм оценивает наличие скачка, основываясь на новостях в том числе после скачка. Классификация не имеет предсказательной силы (оценивает текущее состояние цен).

В обеих конфигурациях порог для величины скачка цены выбран по следующему принципу: $P_{threshold} = 3.5\sigma(\Delta T)$, где $\sigma(T)$ — волатильность данного инструмента на интервале времени T.

Результаты базовых алгоритмов на основе униграмм:

ΔT, мин	N_{obj}	N_1	N_1/N_{obj}	N_{feat}	Алгоритм	ROC-AUC	PR-AUC
10	1186095	194463	0.164	54890	Random Forest	0.58	0.24
					LogRegression	0.58	0.23
					Linear SVM	0.58	0.23
					XGBoost	0.61	0.26
3	937689	242985	0.259	54890	Random Forest	0.58	0.40
					LogRegression	0.59	0.40
					Linear SVM	0.59	0.40
					XGBoost	0.63	0.44

Результаты CatBoost на основе категориальных признаков:

ΔT, мин	N_{obj}	N_1	N_1/N_{obj}	N_{cat}	Алгоритм	ROC-AUC	PR-AUC
10	1186095	194463	0.164	25	CatBoost	0.60	0.31
3	937689	242985	0.259	25	CatBoost	0.64	0.48

В таблицах: ΔT — параметр автоматической разметки, N_{obj} — количество документов в выборке после всех преобразований, N_1 — количество документов, помеченных классом «1», N_{feat} —

количество различных признаков (слов) после фильтрации по частоте, N_{cat} — количество категориальных признаков.

Из таблиц видно, что результаты классификации всех алгоритмов в текущей модели не намного лучше, чем прогноз константой «1» (в случае чего метрики ROC-AUC и PR-AUC имели бы, соответственно, значения 0.5 и N_1/N_{obj}). Однако можно заметить, что даже в таких условиях использование категориальных признаков дает небольшое преимущество по отношению к XGBoost'у, что следует из результатов PR-AUC.

Низкие показатели метрик качества можно объяснить следующим образом. Во-первых, в те интервалы времени, которые мы размечаем классом «1», в потоке новостей помимо относящихся к скачку цены имеется большое количество нерелевантных новостей. При этом автоматическая разметка все-равно помечает их классом «1». Во-вторых, количество слов в каждом отдельно взятом новостном заголовке, как правило, небольшое, а значения TF-IDF на коротких документах имеют недостаточно высокую статистическую значимость.

2.2. Модель с агрегированием документов

2.2.1. Идея эксперимента

Для того, чтобы избавиться от недостатков предыдущего подхода, необходимо пересмотреть концепцию того, что мы считаем одним объектом.

Вернемся к рисунку 2. В такой модели автоматической разметки единичным событием в действительности является не пара [новость, скачок цены], а пара [временной интервал, скачок цены]. Поэтому предлагается в качестве объекта рассматривать не набор слов, встречающихся в тексте одной новости, а агрегированный набор слов, встречающихся в композиции всех новостей за некоторый интервал времени. Таким образом, вместо большого количества маленьких документов мы получим меньшее количество объемных документов.

Объект по временному интервалу формируется следующим образом. Сперва берутся все новости, время которых попадает в заданный временной интервал. Затем создается новый объект, представляющий собой вектор той же размерности, что и отдельно взятая новость. В новый объект записывается сумма векторов всех новостей за текущий интервал. Стоит помнить, что каждая новость описывается разреженным вектором встречаемости униграмм.

Опишем алгоритм построения агрегированных объектов на основе униграмм. Так же, как и ранее, будем двигаться по оси времени окном размера ΔT. Начало окна обозначим за t_0, максимальное в окне по модулю отклонение от цены в момент t_0 обозначим за ΔP, все прочие обозначения с рисунка 2 также сохраняются. В случае, если максимальное отклонение цен превосходит заданный порог ($\Delta P > P_{threshold}$), формируем из всех новостей в интервале $[t_0 - T_b;\ t_0 + T_a]$ единый объект и помечаем его классом «1». После этого переходим к новому значению t_0, превосходящему $t_0 + T_i + T_b$. Если же отклонение цен не превосходит порога, просто переходим к следующему за текущим значению t_0. Затем процедура повторяется.

После того, как обход завершен, у нас уже имеется множество всех объектов, размеченных

классом «1». Для того, чтобы сформировать объекты нулевого класса, из всех оставшихся неразмеченными временных диапазонов сформируем интервалы длительностью $T_b + T_a$ (что соответствует длительностям уже размеченных классом «1» объектов), возьмем из них некоторое количество N_0 случайных и по тем же правилам сформируем агрегированные объекты, размечая их классом «0». Параметр N_0 выбирается таким образом, чтобы итоговая выборка не была сильно несбалансированной (удовлетворительным считается соотношение классов между $1:1$ и $1:4$).

Что касается дальнейшей обработки, предлагается использовать аналогичный предыдущему метод взвешивания факторов с использованием TF-IDF и последующим формированием категориальных признаков для каждого объекта. В текущей модели мы имеем дело с объемными документами, состоящими из большого количества слов, поэтому использование TF-IDF обоснованно.

2.2.2. Результаты эксперимента

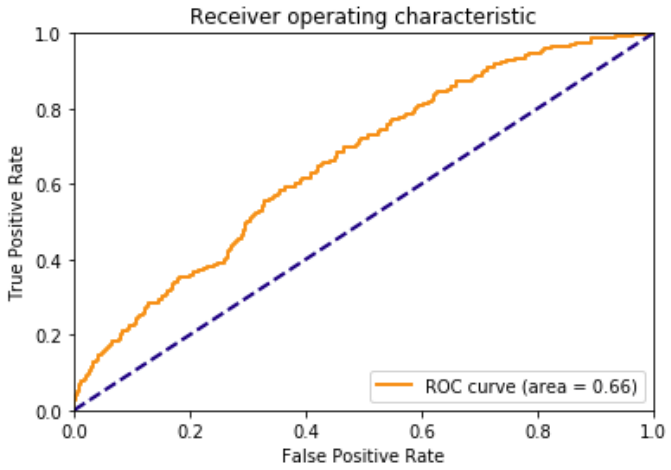
Для оценки использовались все те же две конфигурации параметров, что и в предыдущей секции. Единственное отличие — количество категориальных факторов N_{cat} увеличено до 60, так как теперь мы имеем дело с более объемными документами.

Оценки метрик получены на кросс-валидации K-Fold со случайным перемешиванием (что в силу независимости объектов теперь уже не приводит к завышению оценок). Разбиение выборки производится на 5 частей.

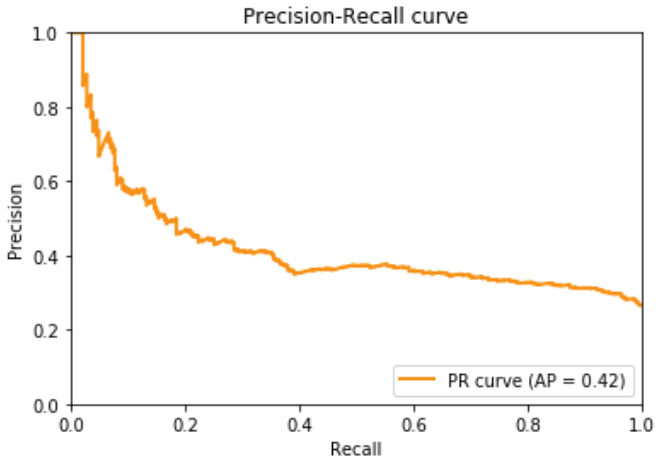
Результаты CatBoost в виде таблицы для двух различных конфигураций:

ΔT, мин	N_{obj}	N_1	N_1/N_{obj}	N_{cat}	ROC-AUC	PR-AUC
10	4838	1338	0.277	60	0.66	0.42
3	5696	1696	0.298	60	0.72	0.54

Также приведем графики ROC и PR кривых для обеих конфигураций (см. рисунки 3 и 4)

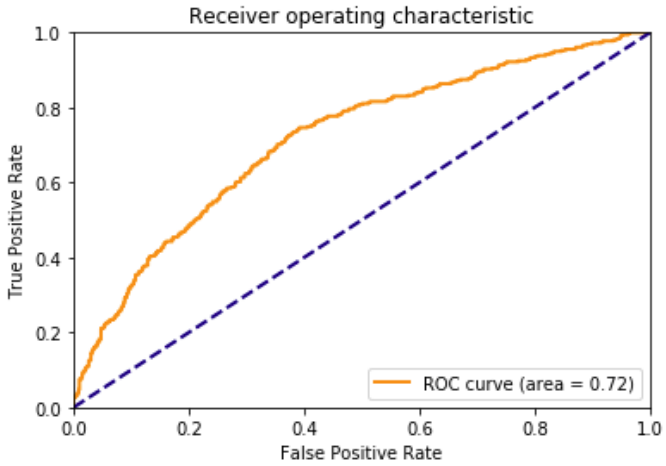


(a) ROC-curve

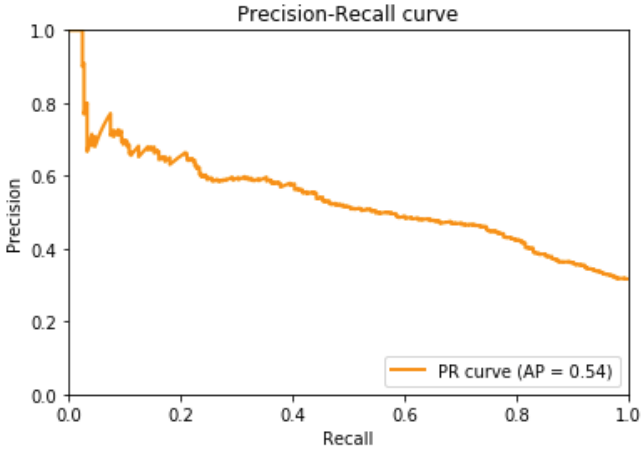


(b) PR-curve

Рис. 3: Метрики качества с использованием документов-композиций при $\Delta T = 10$ мин



(a) ROC-curve



(b) PR-curve

Рис. 4: Метрики качества с использованием документов-композиций при $\Delta T = 3$ мин

Как мы видим, качество классификации по сравнению с предыдущим подходом значительно улучшилось. Однако, разница в результатах между двумя конфигурациями все еще достаточно существенная, а соответствующие значения метрик недостаточно хороши, чтобы использовать модель на практике.

Полученный результат можно объяснить тем, что текущий подход к формированию признаков никак не использует информацию о взаимосвязи слов внутри новостного заголовка — агрегированный объект формируется исключительно из частот всех входящих в него униграмм.

2.3. Модель с использованием битермов

2.3.1. Идея эксперимента

Для того, чтобы учитывать информацию о взаимосвязи слов внутри новостных заголовков, необходимо перейти к рассмотрению признакового описания, отличного от мешка слов. В качестве альтернативы униграммам предлагается использовать биграммы, либо битермы.

Биграммы представляют собой пары подряд идущих слов. Например, из предложения вида «A B C D» можно выделить биграммы «A;B», «B;C», «C;D».

Битермы же представляют собой все различные пары слов в рамках одной фразы (в нашем случае - коминации слов внутри одного новостного заголовка). Например, предложение вида «A B C D» содержит в себе битермы «A;B», «A;C», «A;D», «B;C», «B;D», «C;D».

Битермы — более общая конструкция, чем биграммы, поэтому в дальнейшем модель предлагается строить именно на их основе. Тем не менее, те же самые идеи можно использовать и с биграммами. Также стоит отметить, что при формировании битермов мы не учитываем порядок слов внутри новости, таким образом битермы «A;B» и «B;A» считаются эквивалентными.

Для заданной коллекции текстов, как правило, число различных битермов значительно превосходит количество униграмм, что приводит к необходимости еще более строгого отбора призна-

ков. Для фильтрации битермов, по которым не накоплено достаточное количество статистических данных, предлагается использовать фильтрацию по документной частоте, а для оставшихся — Significance Score, аналогичный использованному в статье [16].

Пусть A и B — две униграммы, образующие битерм $(A;B)$, $f(A,B)$ — случайная величина, характеризующая документную частоту битерма $(A;B)$, $\mu_0(A,B)$ — мат ожидание величины $f(A,B)$ в предположении нулевой гипотезы, что термы A и B встречаются в тексте независимо. Будем говорить, что $sig(A,B) = \frac{f(A,B)-\mu_0(A,B)}{\sqrt{f(A,B)}}$ — значение Significance Score для битерма $(A;B)$.

Чем выше значение $sig(A,B)$, тем более правдоподобна гипотеза, что термы A и B встречаются не независимо и образованный ими битерм имеет высокую смысловую значимость. Близость же данного значения к нулю или отрицательность означает, что данный битерм с высокой вероятностью образован случайно и сочетание термов практически не несет в себе никакого смысла.

После того, как произведено формирование битермов для коллекции имеющихся документов и вычислены значения $sig(A,B)$ для всех встреченных битермов, необходимо осуществить фильтрацию признаков.

Как было сказано выше, первый этап фильтрации производится по документной частоте пороговым правилом с величиной порога N_d^{min}. А именно: удалению подлежит любой такой битерм b, для которого документная частота $N_d(b)$ удовлетворяет неравенству $N_d(b) < N_d^{min}$.

Второй этап фильтрации нужен, чтобы исключить неинформативные битермы. Пусть $b = (A,B)$ — битерм, задаваемый двумя термами A и B. Удалим из признакового описания все такие битермы b, для которых $sig(b) = sig(A,B) < S^{min}$.

Формирование агрегированных объектов из отдельных новостей производится способом, описанным в предыдущей секции с той лишь разницей, что вместо униграмм объект теперь представляет собой вектор битермов.

При формировании категориальных признаков для агрегированных объектов предлагается сортировать их внутри каждого объекта по убыванию в соответствии со значениями $sig(b)$ и брать не более N_{cat} первых.

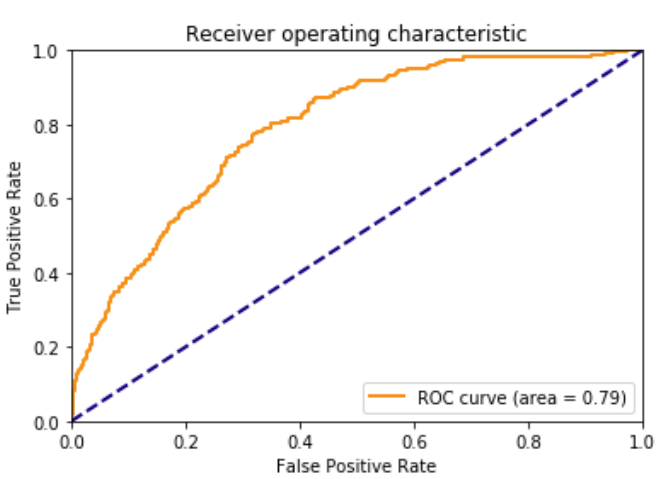
2.3.2. Результаты эксперимента

В следующей таблице представлены результаты для CatBoost в двух ранее использованных конфигурациях параметров. Оценки, как и в предыдущих экспериментах, получены с использованием кросс-валидации.

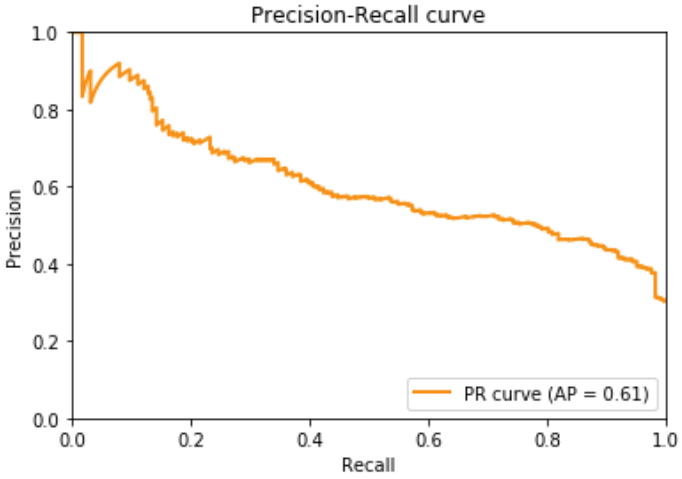
ΔT, мин	N_{obj}	N_1	N_1/N_{obj}	N_{cat}	ROC-AUC	PR-AUC
10	4838	1338	0.277	60	0.79	0.61
3	5696	1696	0.298	60	0.82	0.72

Также приведем графики ROC и PR кривых для обеих конфигураций (см. рисунки 5 и 6)

Из таблицы видно, что использование битермов в связке с отбором по Significance Score позволяет значительно улучшить результаты по сравнению с моделью, использующей униграммы и TF-IDF для отбора наиболее важных признаков.

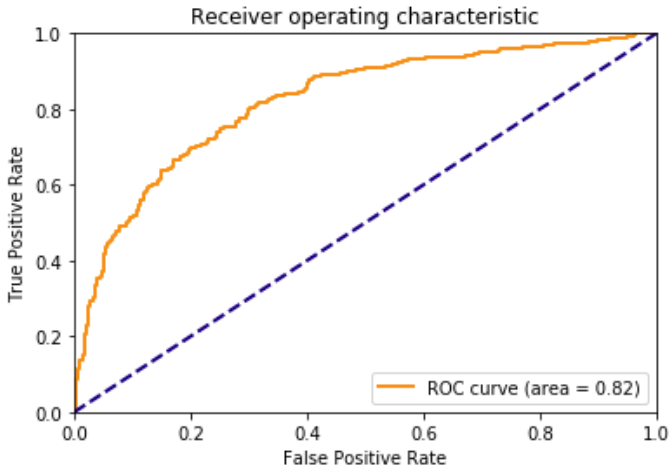


(a) ROC-curve

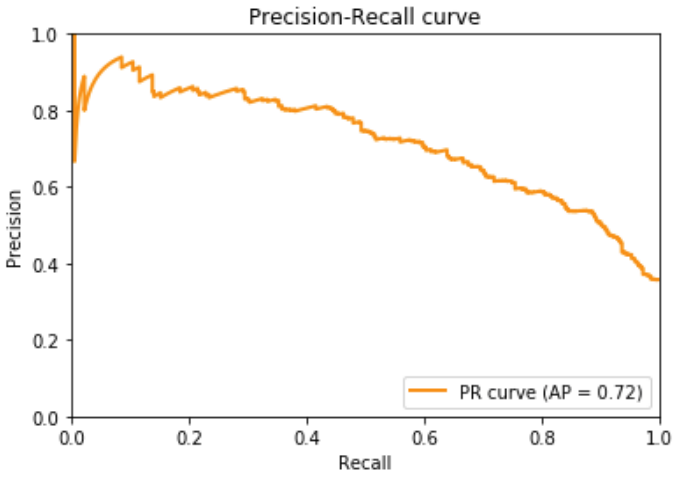


(b) PR-curve

Рис. 5: Метрики качества с использованием битермов при $\Delta T = 10$ мин



(a) ROC-curve



(b) PR-curve

Рис. 6: Метрики качества с использованием битермов при $\Delta T = 3$ мин

2.4. Модель регрессии

2.4.1. Идея эксперимента

До текущего момента мы строили модель классификации, которая обучается на автоматически размеченных по двум классам данных, где классы назначались пороговым правилом по величине модуля скачка цены. В данной секции предлагается обучать модель регрессии, которая на первом этапе предсказывает непосредственно сам модуль скачка цены, а не класс; на втором этапе полученное число с помощью порогового правила можно превратить в оценку класса «1» или «0». При этом в качестве объектов снова выступают агрегированные документы, а их признаковое описание построено на битермах и качественно полностью совпадает с описанным в предыдущей секции.

Оценку качества предлагается производить с точки зрения классификации (ROC-curve, PR-

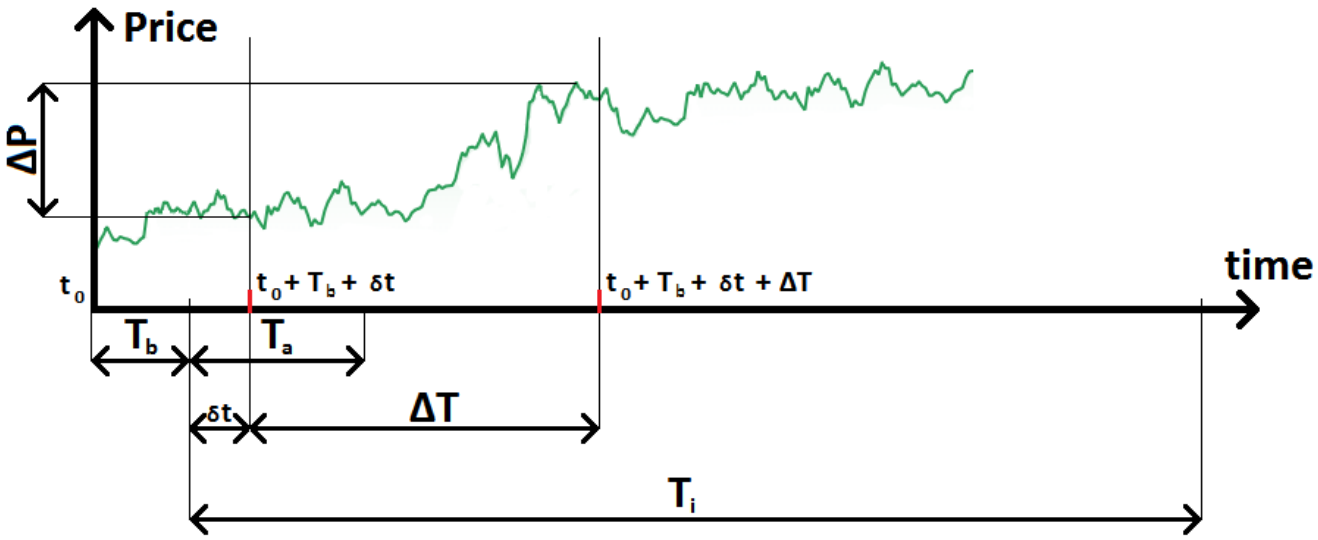


Рис. 7: Параметры автоматической разметки величины скачка цены; новый параметр δt

curve) путем кросс-валидации с разбиением выборки на 5 блоков.

Опишем процедуру формирования объектов. Введем временные параметры T_a, T_b, T_i, ΔT, а также принципиально новый параметр τ. В этот раз при обходе времен новостей предлагается двигаться не окном, а интервалами длительности $T_b + T_a$. В качестве значения параметра τ предлагается взять 20 секунд.

Начнем обход времен новостей, для наглядности см. рис. 7. Пусть t_0 - время начала текущего временного интервала. Будем считать величиной скачка цены для формируемого объекта значение $\delta P = \max\limits_{\delta t \in [-\tau, \tau]} [\Delta P(t_0 + T_b + \delta t, \Delta T)]$. Текущий объект формируем, агрегируя новости в интервале $[t_0, t_0 + T_b + T_a]$ и записывая соответствующий ему ответ δP. В случае, если $\delta P > P_{threshold}$, начало очередного рассматриваемого интервала переносим на время не менее $t_0 + T_b + T_i$, иначе - на следующий момент времени после $t_0 + T_b + \max(T_a, \tau + \Delta T)$ (тем самым обеспечивая, что для разных объектов интервалы, на которых наблюдается цена, не пересекаются), где производим аналогичную процедуру формирования объекта.

У такого подхода есть два основных отличия по сравнению с формированием объектов для классификатора в предыдущих секциях. Во-первых, величина скачка для каждого объекта выбирается как максимум по небольшому интервалу длительности 2τ (ранее бралось значение скачка цены в один момент времени). Во-вторых, здесь мы не производим отдельный поиск участков с наибольшими скачками цены, а формируем все объекты (как с малыми, так и с превышающими порог скачка цены) за один проход.

Для упрощения процедуры обучения также предлагается убрать из выборки наиболее сложные объекты, величина скачка цены на которых одновременно не является достаточно близкой к нулю и не превосходит порог для классификации классом «1». С целью добиться относительно сбалансированной выборки в качестве нижней границы удаляемых объектов берется 30-ый процентиль по значениям скачков цены на всех объектах в обучении (таким образом, от общего исходного

количества объектов остается 30% близких к нулю элементов). Верхняя граница фильтрации для обеих выбранных конфигураций параметров располагается близко к 90-му процентилю.

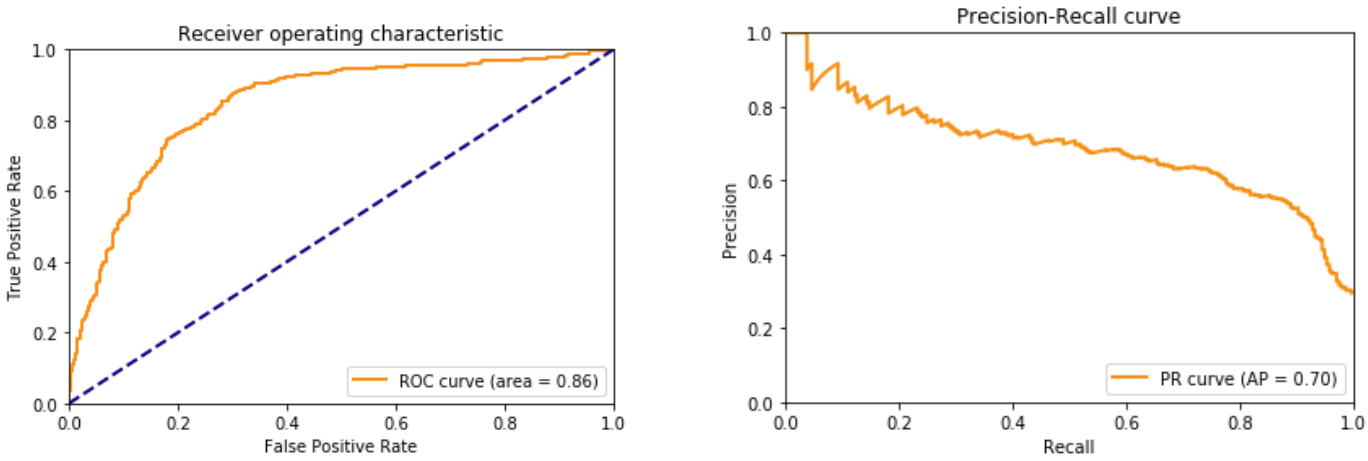
Категориальные признаки на базе битермов предлагается формировать тем же способом, что и в предыдущей секции - на основе Significance Score. В качестве обучаемого алгоритма регрессии был выбран CatBoost Regressor, так как имеет поддержку категориальных признаков, на которых строится модель.

2.4.2. Результаты эксперимента

В следующей таблице представлены результаты CatBoost Regressor с точки зрения классификации. После обучения модели производится оценка алгоритмом величины скачка цены и пороговым правилом переводится в классы «0» и «1»: в случае, если предсказанная величина превосходит порог $P_{threshold}$, оцениваем объект классом «1», иначе — классом «0». При оценке метрик результат сравнивается с классом, выставленным при генерации объекта на основе реальной величины скачка цены.

ΔT, мин	τ, с	N_{obj}	N_1	N_1/N_{obj}	N_{cat}	ROC-AUC	PR-AUC
10	20	3799	1014	0.267	60	0.86	0.70
3	20	4016	1160	0.289	60	0.88	0.74

Также приведем соответствующие графики ROC и PR кривых (см. рисунки 8 и 9)



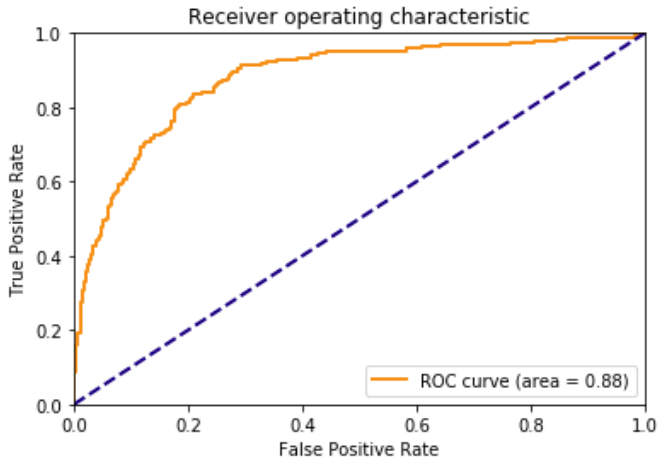
(a) ROC-curve

(b) PR-curve

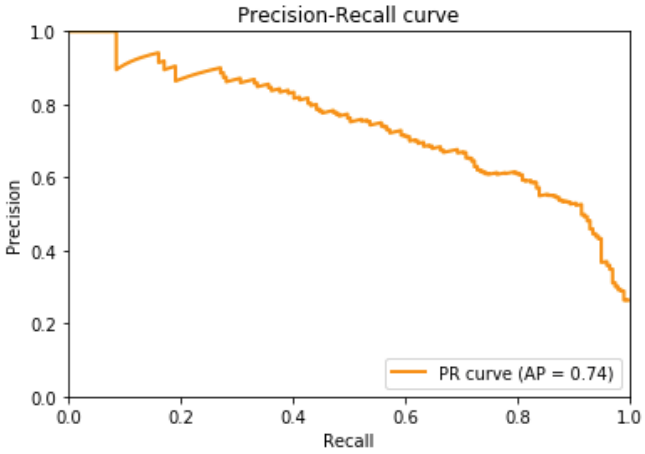
Рис. 8: Метрики качества CatBoost Regressor при $\Delta T = 10$ мин

Сравнивая полученные оценки качества с результатами непосредственного обучения классификатора CatBoost Classifier, можно сделать вывод, что полученный на основе модели регрессии классификатор лучше. Основной особенностью является тот факт, что модель в конфигурации, имеющей предсказательную силу ($\Delta T = 10$ мин), не так сильно теряет в качестве классификации.

Объяснить улучшение результатов можно следующим образом. Во-первых, модель обучается на более точных цифрах, так как предсказываемые величины - вещественные числа, а не огрубленные



(a) ROC-curve



(b) PR-curve

Рис. 9: Метрики качества CatBoost Regressor при $\Delta T = 3$ мин

значения классов; само огрубление в модели производится уже на этапе предсказания. Во-вторых, из выборки были убраны наиболее спорные с точки зрения классификации объекты.

Выводы

В результате исследования было подтверждено, что в задаче прогнозирования динамики цен биржевых инструментов на основе анализа потока финансовых новостей с использованием машинного обучения можно добиться положительных результатов.

В то же время при проведении ряда экспериментов было установлено, что задача выявления движения цен «в ближайшем прошлом» решается легче, чем задача прогнозирования движения цен «в будущем», хотя именно последняя представляет наибольший интерес с эксплуатационной точки зрения.

Итоговые оценки метрик качества удовлетворяют необходимым требованиям для практического применения модели в целях, предложенных во введении, а именно:

- Вспомогательная система, сигнализирующая о скорых движениях цены по конкретному финансовому инструменту. Полученные ROC-кривые позволяют выбрать оптимальный баланс между чувствительностью и специфичностью классификатора.
- Часть более сложной автоматической торговой системы. В этом сценарии важно найти баланс между точностью и полнотой классификации, что возможно в рамках полученных PR-кривых.

Тем не менее, на практике гораздо большую ценность имеют инструменты классификации потока новостей, которые умеют предсказывать не только сам факт движения цены, но также и его направление. Эта задача в рамках текущего исследования не рассматривалась. Можно, однако, для ее решения предложить следующий двухэтапный подход, основанный на результатах данной работы:

1. Производится бинарная классификация на классы «1» и «0», где «1» означает наличие движения цен, «0» — его отсутствие.
2. Для объектов, оцененных на первом шаге классом «1», снова производится бинарная классификация, но на этот раз целевыми классами являются «+1» и «-1», где «+1» означает рост цены, а «-1» — падение.

Подводя итоги, стоит также упомянуть основные трудности, возникшие во время исследования:

1. Отсутствие полного текста новостей. Используются исключительно новостные заголовки, что усложняет поиск статистических закономерностей, особенно без применения агрегации.
2. Необходимость автоматической разметки объектов на основе цен. При любых значениях параметров некоторая часть выборки будет заведомо размечена некорректно, поскольку между новостным потоком и графиком цены, вообще говоря, нет строгой и однозначной причинно-следственной связи. Более подробное описание проблемы приведено во введении.

Заключение

Результаты, полученные в данной работе:

1. Предложена формализация задачи предсказания скачка цены финансового инструмента по новостному потоку.
2. Предложены три модели (на основе агрегирования документов, на основе битермов, а также модель регрессии), показано их преимущество по сравнению с базовой моделью.
3. Сделана реализация, пригодная для практической эксплуатации.

Список литературы

[1] https://finance.yahoo.com/

[2] K. J. Millman and M. Aivazis, *Python for Scientists and Engineers* // Computing in Science & Engineering, vol. 13, no. 2, pp. 9–12, March-April 2011.

[3] J. D. Hunter, *Matplotlib: A 2D Graphics Environment* // Computing in Science & Engineering, vol. 9, no. 3, pp. 90–95, May-June 2007.

[4] Wes McKinney. *Data Structures for Statistical Computing in Python* // Proceedings of the 9th Python in Science Conference, pp. 51–56, 2010

[5] Travis E, Oliphant, *A guide to NumPy* // USA: Trelgol Publishing, 2006

[6] http://www.cplusplus.com/

[7] Tom Fawcett, *An introduction to ROC analysis* // Pattern Recognition Letters vol. 27, pp. 861—874, 2006.

[8] J. Davis, M. Goadrich, *The Relationship between Precision-Recall and ROC Curves* // Proc. Int'l Conf. Machine Learning, 2006.

[9] Ramos, Juan, *Using TF-IDF to determine word relevance in document queries* // ICML 2003.

[10] http://scikit-learn.org/

[11] Pedregosa F. et al. *Scikit-learn: Machine learning in Python* // Journal of Machine Learning Research. – 2011. – Т. 12. №. Oct. – pp. 2825–2830

[12] https://github.com/dmlc/xgboost

[13] https://xgboost.readthedocs.io/

[14] Anna Veronika Dorogush, Vasily Ershov, Andrey Gulin, *CatBoost: gradient boosting with categorical features support* // Yandex, 2017

[15] https://tech.yandex.com/catboost/

[16] Ahmed El-Kishky , Yanglei Song , Chi Wang , Clare R. Voss , Jiawei Han, *Scalable topical phrase mining from text corpora* // Proceedings of the VLDB Endowment, v.8 n.3, pp. 305–316, November 2014