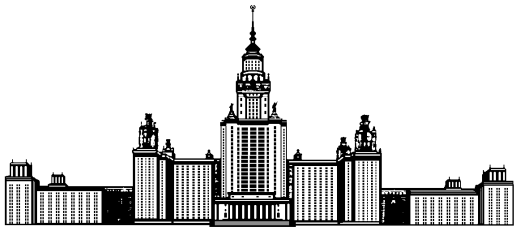
Московский государственный университет имени М. В. Ломоносова



Факультет Вычислительной Математики и Кибернетики
Кафедра Математических Методов Прогнозирования

Отчет по второму заданию курса "Практикум на ЭВМ"

"Градиентные методы обучения линейных моделей"

Выполнила:
студентка 3 курса 317 группы
Кривуля Полина Юрьевна

Москва, 2022

Содержание

1 Введение

Градиентный спуск относится к оптимизационным алгоритмам и используется для настройки параметров модели машинного обучения.

При стандартном (или «пакетном», «batch») градиентном спуске для корректировки параметров модели используется градиент. Градиент обычно считается как сумма градиентов, вызванных каждым элементом обучения. Вектор параметров изменяется в направлении антиградиента с заданным шагом. Поэтому стандартному градиентному спуску требуется один проход по обучающим данным до того, как он сможет менять параметры.

При стохастическом (или «оперативном») градиентном спуске значение градиента аппроксимируется градиентом функции стоимости, вычисленном только на одном элементе обучения. Затем параметры изменяются пропорционально приближенному градиенту. Таким образом параметры модели изменяются после каждого объекта обучения. Для больших массивов данных стохастический градиентный спуск может дать значительное преимущество в скорости по сравнению со стандартным градиентным спуском.

Между этими двумя видами градиентного спуска существует компромисс, называемый иногда «mini-batch». В этом случае градиент аппроксимируется суммой для небольшого количества обучающих образцов [1].

В данном исследовании изучается работа градиентного спуска и стохастического градиентного спуска для различных параметров на примере задачи биномиальной классификации текстов для определения токсичности комментария.

2 Теоретическая часть

2.1 Градиент функции потерь для задачи бинарной логистической регрессии

Функция потерь для задачи бинарной логистической регрессии представима в виде:

$$Q(X, w) = \frac{1}{l}\sum_{i=1}^{l}\mathcal{L}(M_i(w)) + \frac{\lambda}{2}\|w\|_2^2,$$

где $\mathcal{L} = \log\left(1 + exp(-M)\right),\ M = y_i\langle w, x_i\rangle$
Так как $d(1 + exp(-y_i\langle w, x_i\rangle)) = d(exp(-y_i\langle w, x_i\rangle)) = -exp(-y_i\langle w, x_i\rangle) * y_i\langle x_i, dw\rangle$,
$d(\|w\|_2^2) = \langle 2w, dw\rangle$,

$$dQ(X, w) = \frac{1}{l}\sum_{i=1}^{l}\frac{-exp(-y_i\langle w, x_i\rangle) * y_i\langle x_i, dw\rangle}{1 + exp(-y_i\langle w, x_i\rangle)} + \lambda\langle w, dw\rangle$$

$$\Rightarrow \bigtriangledown_w Q(X, w) = \frac{1}{l}\sum_{i=1}^{l}\frac{-exp(-y_i\langle w, x_i\rangle) * y_i * x_i}{1 + exp(-y_i\langle w, x_i\rangle)} + \lambda w$$

2.2 Градиент функции потерь для задачи многоклассовой логистической регрессии

Функция потерь для задачи мультиномиальной логистической регрессии (без регулязатора) представима в виде:

$$Q(X, w) = -\frac{1}{l}\sum_{i=1}^{l}\sum_{j=1}^{K}[y_i = j]\, log\frac{exp\langle w_i, x_i\rangle}{\sum_{z=1}^{K} exp\langle w_z, x_i\rangle}$$

Найдем ее градиент для некоторого w_q:

$$\bigtriangledown_{w_q} Q(w) = -\frac{1}{l}\sum_{i=1}^{l}\sum_{j=1}^{K}[y_i = j]\, \bigtriangledown_{w_q} log\frac{exp\langle w_i, x_i\rangle}{\sum_{z=1}^{K} exp\langle w_z, x_i\rangle}$$

Так как $log\frac{exp\langle w_i,x_i\rangle}{\sum_{z=1}^{K} exp\langle w_z,x_i\rangle} = log(exp\langle w_i,x_i\rangle) - log(\sum_{z=1}^{K} exp\langle w_z,x_i\rangle)$,

$$\nabla_{w_q} Q(w) = -\frac{1}{l}\sum_{i=1}^{l}\sum_{j=1}^{K}[y_i = j]\,(x_i\,[j = q] - \frac{x_i * exp\langle w_i,x_i\rangle}{\sum_{z=1}^{K} exp\langle w_z,x_i\rangle})$$

2.3 Задача многоклассовой логистической регрессии при двух классах

При двух классах задача многоклассовой логистической регрессии сводится к бинарной, при этом необходимо "переименовать" классы на -1 и 1:

$$Q(X,w) = -\frac{1}{l}\sum_{i=1}^{l}\sum_{j=1}^{2}[y_i = j]\,log\frac{exp\langle w_i,x_i\rangle}{\sum_{z=1}^{2} exp\langle w_z,x_i\rangle} =$$

$$= -\frac{1}{l}\sum_{i=1}^{l}([y_i = 1] * log\frac{1}{1+exp\langle w_2 - w_1, x_i\rangle} + [y_i = 2] * log\frac{1}{1+exp\langle w_1 - w_2, x_i\rangle}) =$$

$$= \frac{1}{l}\sum_{i=1}^{l}([y_i = 1] * log(1+exp\langle w_2 - w_1, x_i\rangle) + [y_i = 2] * log(1+exp\langle w_1 - w_2, x_i\rangle)) =$$

Воспользуемся переходом $\{1,2\} \to \{-1,1\}$

$$= \frac{1}{l}\sum_{i=1}^{l}([y_i = -1] * log(1+exp\langle w_{+1} - w_{-1}, x_i\rangle) + [y_i = +1] * log(1+exp\langle w_{-1} - w_{+1}, x_i\rangle)) =$$

$$= \frac{1}{l}\sum_{i=1}^{l} log(1+exp(-y_i * \langle w, x_i\rangle))$$

Функции потерь эквивалентны. Теперь докажем эквивалентность предсказаний моделей:

$$sign\langle w,x\rangle = (-1) * [\langle w,x\rangle < 0] + (+1) * [\langle w,x\rangle < 0] = \underset{y\in\{-1,+1\}}{SoftMax}\langle w_y,x\rangle = \underset{y\in\{1,2\}}{SoftMax}\langle w_y,x\rangle$$

2.4 Разностная проверка градиента

Для проверки корректности аналитической формулы градиента функции потерь для задачи бинарной логистической регрессии реализован модуль численного подсчета, вычисляющий градиент приближенно по формуле:

$$[\nabla f(w)]_i \approx \frac{f(x+\epsilon e_i) - f(x)}{\epsilon}$$

$e_i = (0,0,0,...,1,0,...,0)$ – базисный вектор, $\epsilon > 0$ – небольшое положительное число.

На примере используемой в задании выборке при $\epsilon = 10^{-8}$ квадрат нормы разности аналитического и численного подсчета вышел равным $1.4 * 10^{-12}$, что говорит о корректности аналитического расчета.

3 Преобразование выборки

В качестве предварительной обработки текста выполнены следующие действия:

- Все тексты приведены к нижнему регистру;
- В тексте все символы, не являющиеся цифрами и буквами, заменены на пробелы, а много раз подряд встречающиеся пробелы заменены на один;
- Выборка преобразована в разреженную матрицу, где значение x в позиции (i,j) означает, что в документе i слово j встретилось x раз;

- Для уменьшения размерности признакового пространства и ускорения проведения экспериментов применен параметр min_df=0.0001, уменьшивший размерность пространства с 89658 признаков до 16050.

Указанная обработка является стандартной и обычно позволяет улучшить качество работы алгоритма. Выборка при этом представляется в виде «мешка слов» («Bag of words»). В этом представлении каждому тексту в соответствие ставится «мешок» его слов, в котором не учитывается расположение слов, а лишь их число вхождений в данный текст.

4 Поведение используемых методов в зависимости от параметров

Для минимизации функционала $Q(X, w)$ методом градиентного спуска выбирается начальное приближение для вектора весов w, затем запускается итерационный процесс, на каждом шаге которого вектор w изменяется в направлении антиградиента функционала $Q(X, w)$:

$$w^{(k+1)} = w^{(k)} - \eta_k \nabla_w Q(X, w), \qquad (1)$$

где $\eta_k = \frac{\alpha}{k^\beta} > 0$ - темп обучения, монотонно уменьшающийся с течением итераций.

При этом во всех экспериментах выбрано значение регулязатора $\lambda = 1$.

При стохастическом градиентном спуске градиент суммы оценивается градиентом подмножества слагаемых, на каждой итерации оно выбирается случайно. Это позволяет ускорить сходимость алгоритма.

В данном эксперименте проводится исследование поведения градиентного и стохастического градиентного спуска для задачи логистической регрессии при изменении следующих параметров:

- Размера шага α (step_alpha);
- Размера шага β (step_betta);
- Начального приближения весов.

Для стохастического градиентного спуска, помимо перечисленного, исследуется поведение метода при изменении размера подвыборки batch_size. Исследование поведения проводится при анализе зависимостей значения функции потерь и точности от итерации метода (эпохи в случае стохастического варианта).

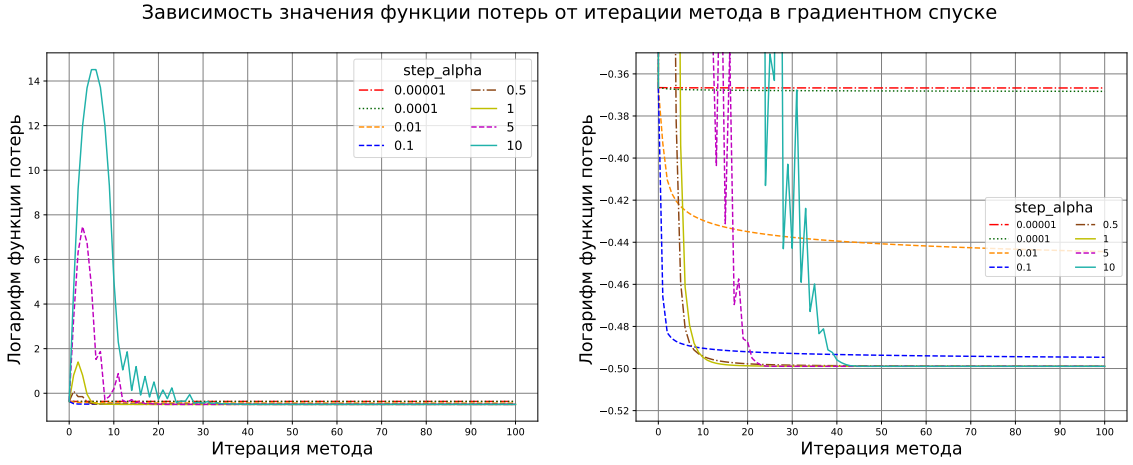


Рис. 1: Значения функции потерь в градиентном спуске при различных параметрах α

- Параметр α перебирается по сетке: {0.00001, 0.0001, 0.01, 0.1, 0.5, 1, 5, 10}. При исследовании данного параметра используется значение $\beta = 1$ и нулевое начальное приближение весов, а

для стохастического градиентного спуска batch_size=10000. При значениях $\alpha > 1$ алгоритм становится менее устойчивым и медленнее сходится. При небольших значениях параметра алгоритм, напротив, слишком устойчив (градиентный спуск: рис. 1, 2, стохастический градиентный спуск: рис. 3, 4). Стохастический градиентный спуск при этом показывает результаты в среднем лучше.

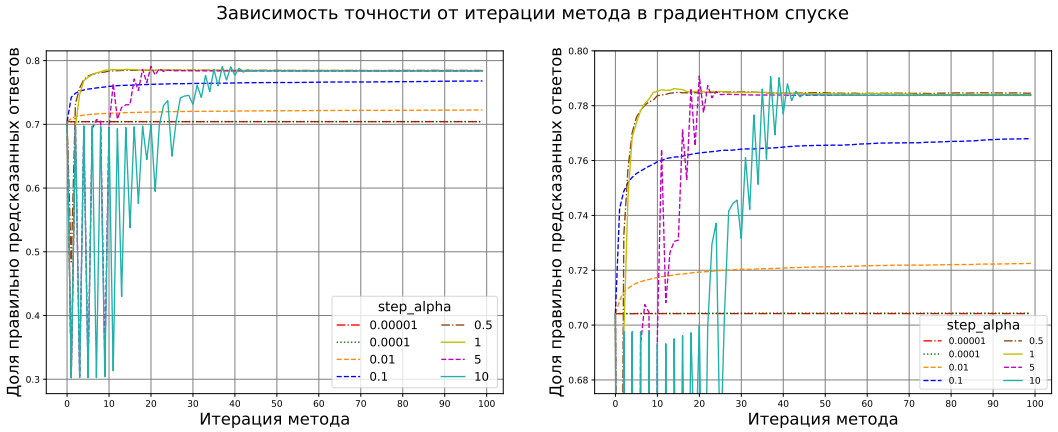


Рис. 2: Точность в градиентном спуске при различных параметрах α

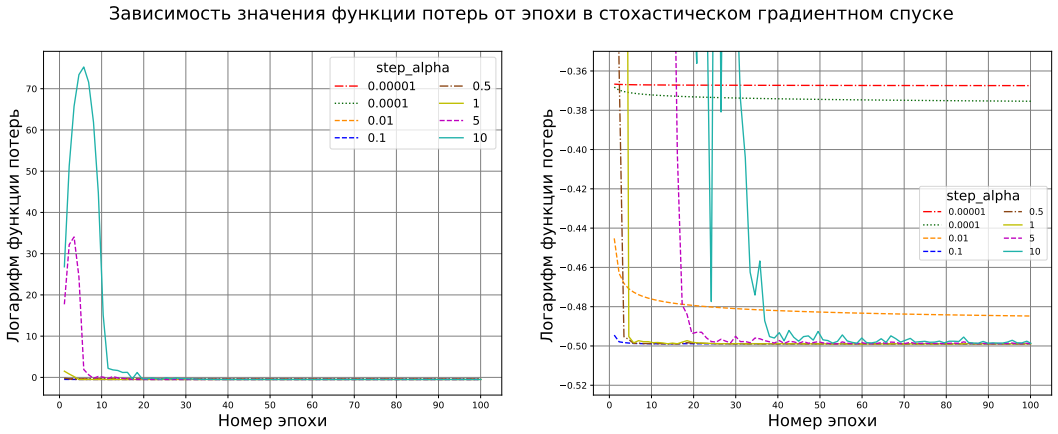


Рис. 3: Функция потерь в стохастическом градиентном спуске при различных параметрах α

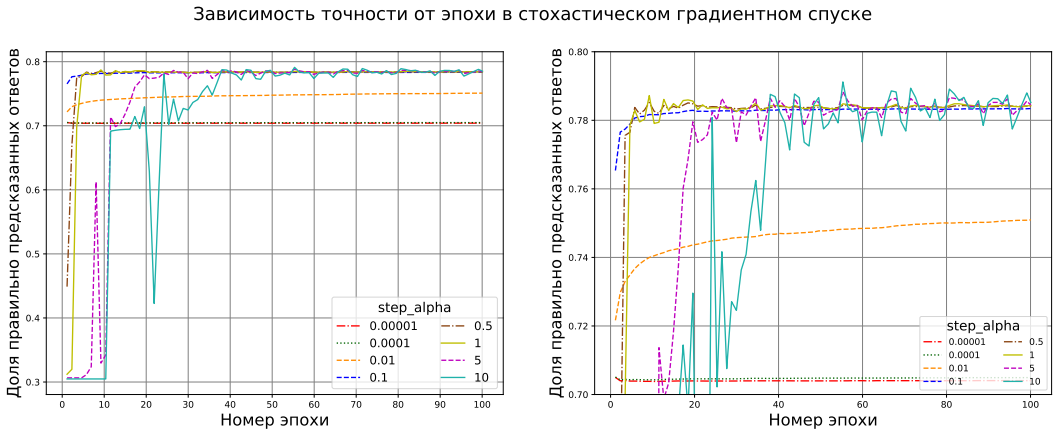


Рис. 4: Точность в стохастическом градиентном спуске при различных параметрах α

- Параметр β перебирается по сетке: {0.1, 0.25, 0.5, 1, 1.01, 1.1, 5, 10, 100}. При исследовании данного параметра используется значение $\alpha = 0.5$ и нулевое начальное приближение весов, а для стохастического градиентного спуска batch_size=10000. С уменьшением данного параметра алгоритм становится менее устойчивым, с увеличением, напротив, слишком устойчивым (градиентный спуск: рис. 5, 6, стохастический градиентный спуск: 7, 8). При этом стохастический градиентный спуск показывает результаты в среднем лучше, и меньше колеблется при небольших значениях β. Однако, как и в случае с параметром α, при "хороших" параметрах β, градиентный спуск ведет себя стабильно, в то время как стохастический градиентный спуск немного колеблется. Наиболее сильно это заметно на графиках точности (рис. 2 и 4, рис. 6 и 8).

  Влияние данных параметров (α и β) на устойчивость связано с формулой итерационного процесса (1). При увеличении α (уменьшении β) последующие значения весов сильнее начинают зависеть от значения градиента функционала, а при слишком малых α (больших β) значения градиента практически не меняют значения весов и алгоритм становится излишне устойчивым.

  Далее используем значения $\alpha = 0.5$ и $\beta = 1.1$, если не оговорено обратного, так как они показали лучшие результаты в данном эксперименте.

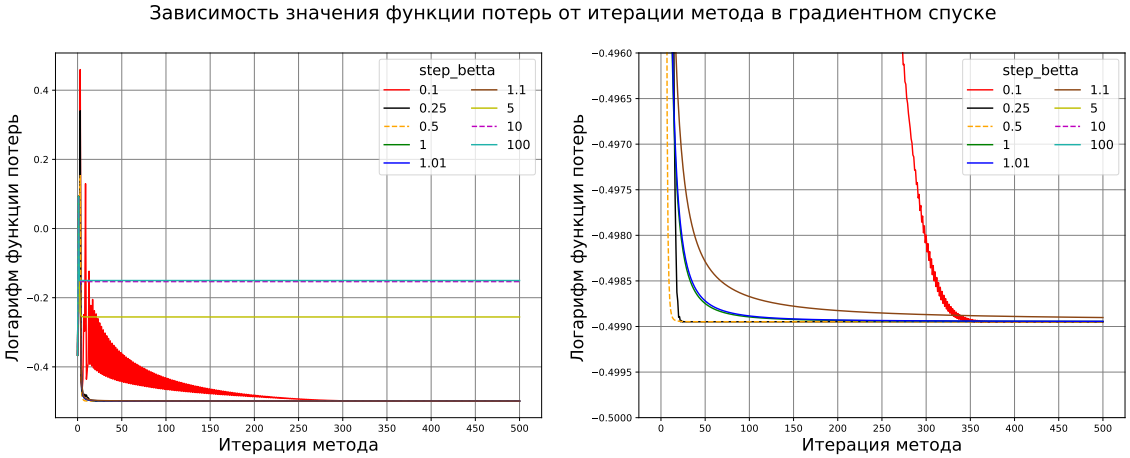


Рис. 5: Значения функции потерь в градиентном спуске при различных параметрах β

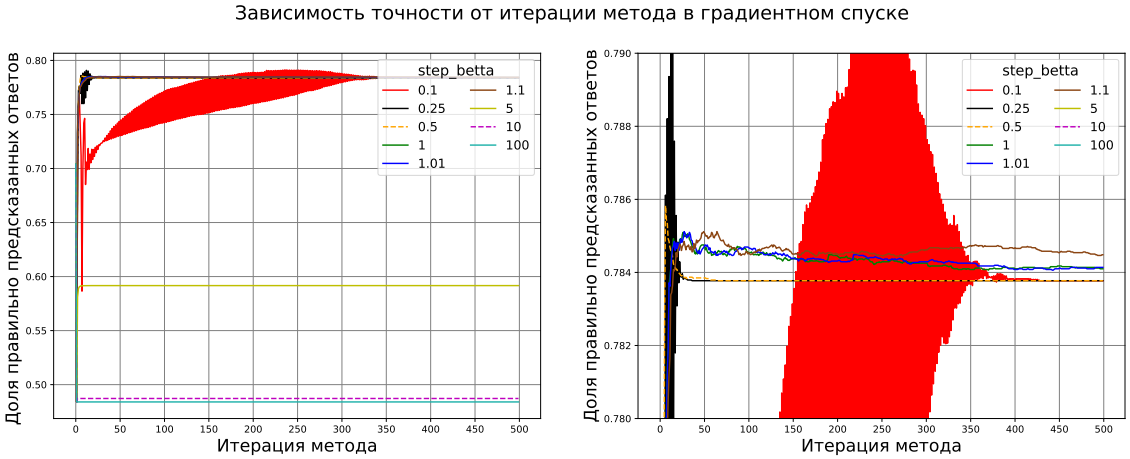


Рис. 6: Точность в градиентном спуске при различных параметрах β

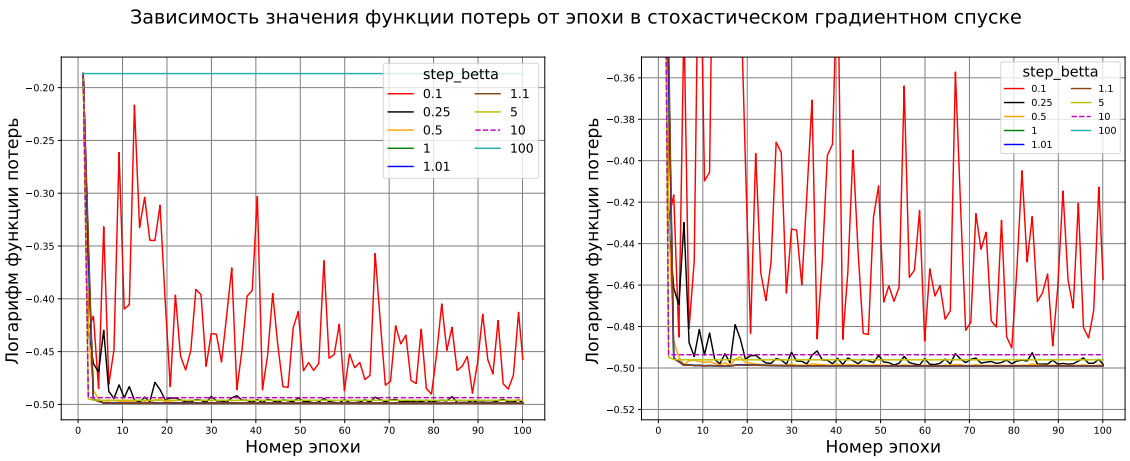


Рис. 7: Функция потерь в стохастическом градиентном спуске при различных параметрах β

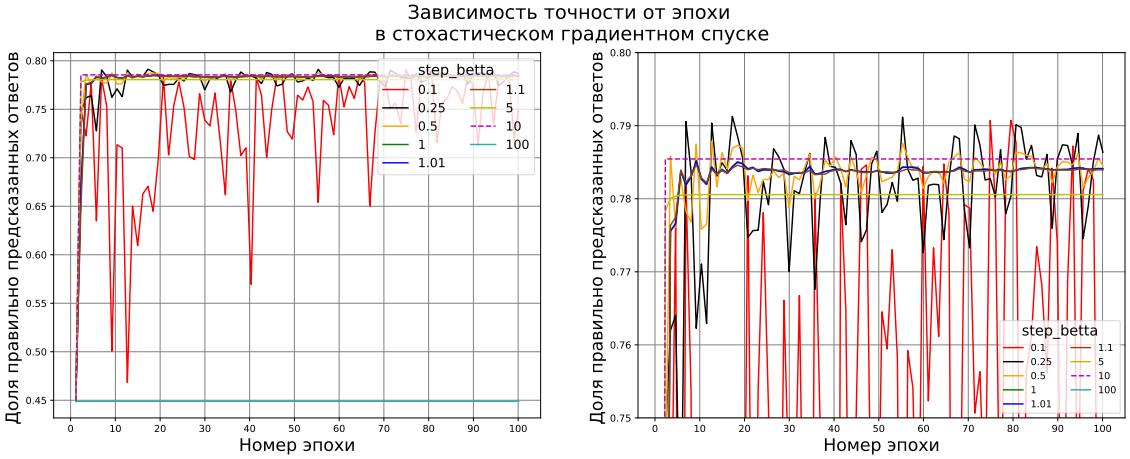


Рис. 8: Точность в стохастическом градиентном спуске при различных параметрах β

- Исследуем зависимость от начального приближения.

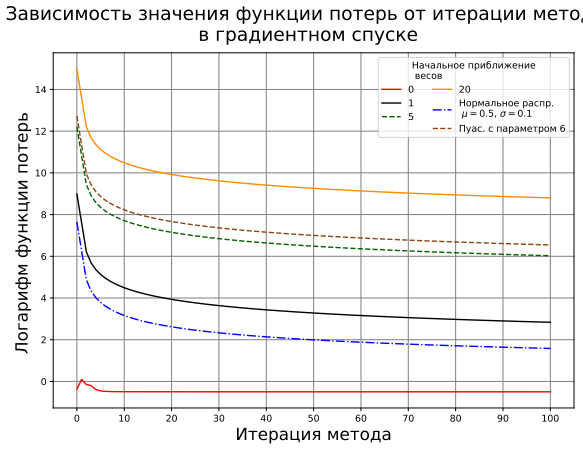


Рис. 9: Функция потерь при различных начальных приближениях

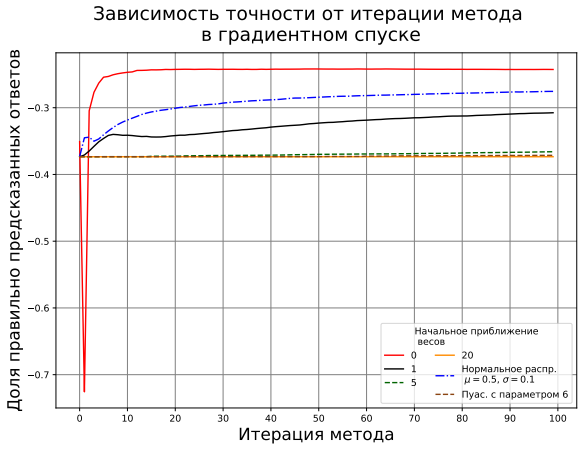


Рис. 10: Точность при различных начальных приближениях

Начальные приближения весов исследованы в следующих вариантах:

– Одинаковые веса со значениями 0, 1, 5, 20;

– Нормальное распределение с параметрами $\mu = 0.5$, $\sigma = 0.1$;

– Пуассоновское распределение с параметром 6.

Лучший результат дает начальное нулевое приближение весов (градиентный спуск: рис. 9, 10, стохастический градиентный спуск: рис. 11, 12). При этом видно, что начальное приближене весом на поведение стохастического градиентного спуска не оказывает такого влияния, как на градиентный спуск. Далее используется такое приближение, если не оговорено другое.

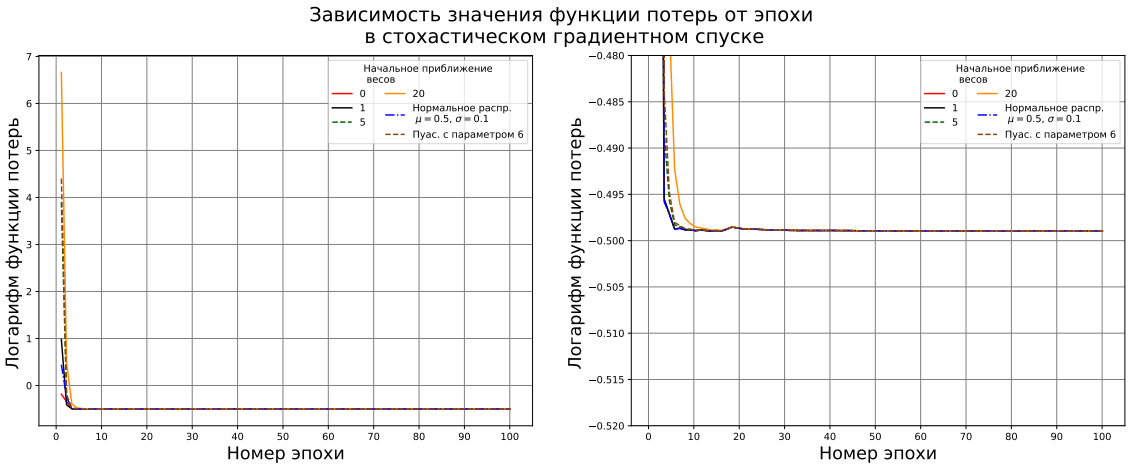


Рис. 11: Функция потерь в стохастическом градиентном спуске при различных начальных приближениях

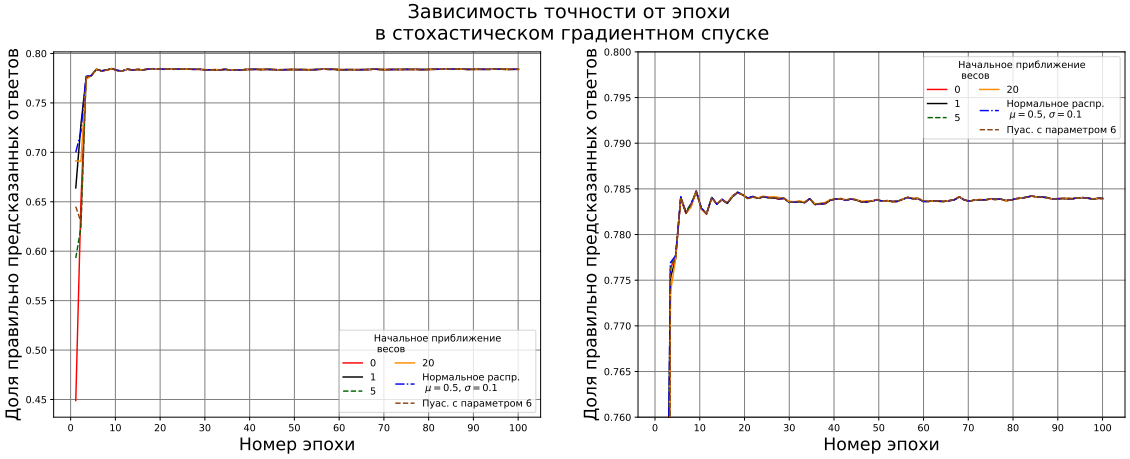


Рис. 12: Точность в стохастическом градиентном спуске при различных начальных приближениях

- Для стохастического градиентного спуска исследуем его поведение в зависимости от параметра batch_size.

  Значения размера подвыборки batch_size перебираются по сетке: {500, 1000, 5000, 10000, 25000, 50000, 100000}. При меньших размерах подвыборки заметны колебания функции потерь и точности, а при больших – более низкие значения точности (рис. 13, 14). Далее, если не оговорено другое, используется batch_size = 10000.

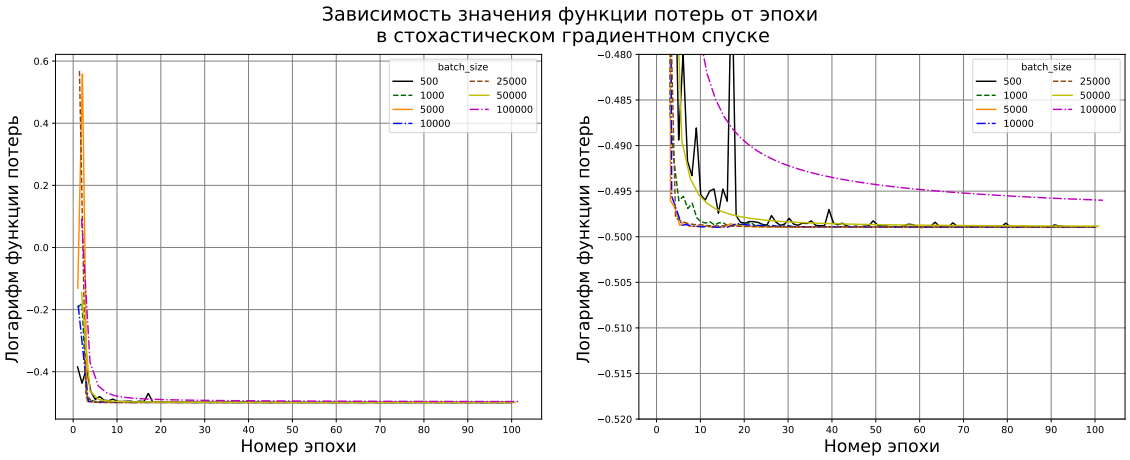


Рис. 13: Функция потерь в стохастическом градиентном спуске при различных batch_size

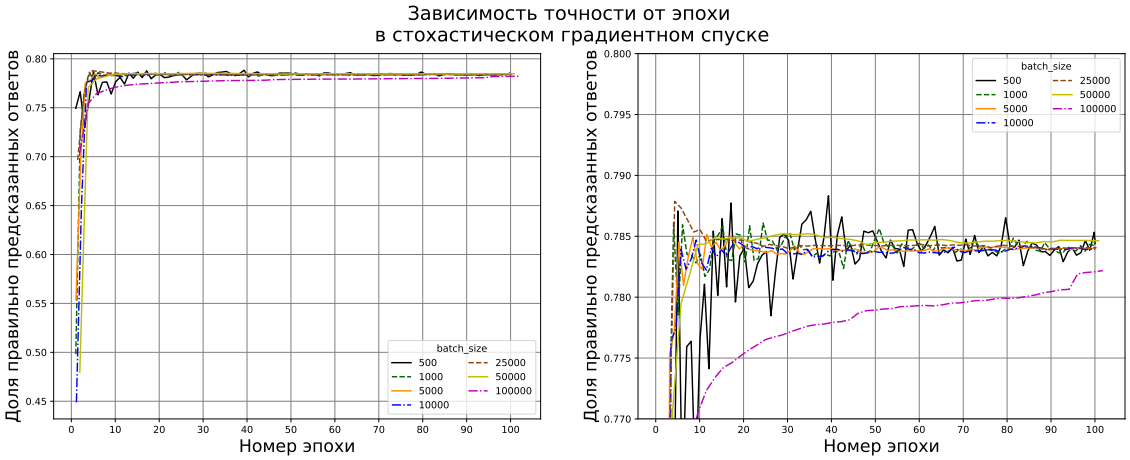


Рис. 14: Точность в стохастическом градиентном спуске при различных batch_size

Стохастический градиентный спуск в среднем показывает результаты лучше, чем градиентный спуск и меньше колеблется при "плохих" используемых параметрах. Особенно это заметно при изменении начального приближения. При большом значении batch_size стохастический градиентный спуск ведет себя как градиентный спуск, при небольших значениях данного параметра сильно колеблется.

5 Расширенная предобработка

Применим к коллекции дополнительную предобработку. Воспользуемся алгоритмом лемматизации. Данный алгоритм приводит слова к лемме — нормальной форме. После этого уберем из коллекции стоп-слова (предлоги, междометия, частицы, часто встречающиеся общеупотребительные слова). Параметр min_df остается равным 0.0001.

После данной обработки размерность признакового пространства уменьшилась с 16050 до 14329. На графиках 15, 16 видно, что благодаря данной обработке работа алгоритмов ускорилась без негативного влияния на точность.

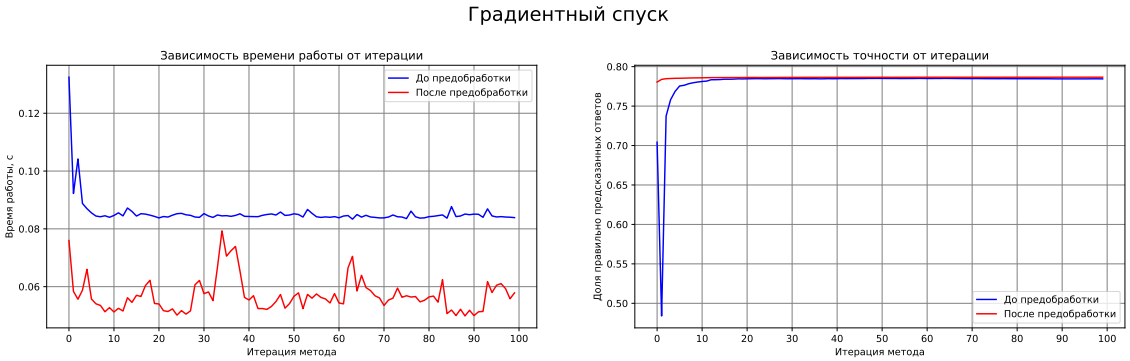


Рис. 15: Влияние дополнительной обработки на градиентный спуск

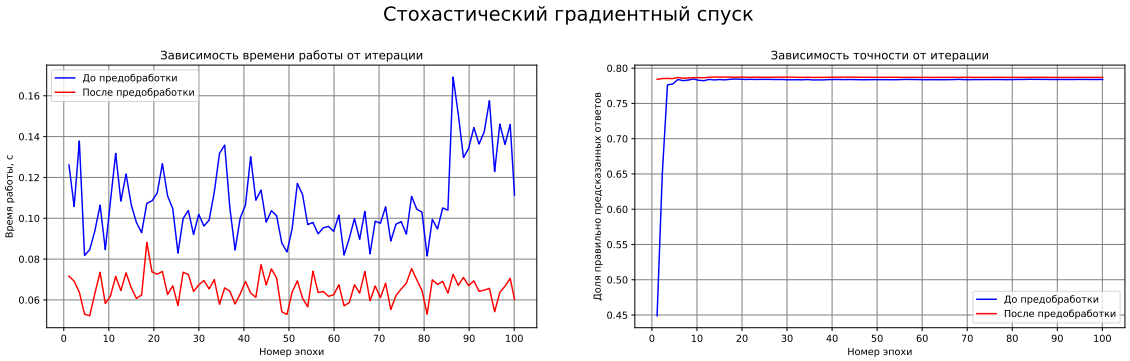


Рис. 16: Влияние дополнительной обработки на стохастический градиентный спуск

6 Исследование влияния способа представления коллекции на алгоритмы

В предыдущем пункте было выяснено, что дополнительная предобработка положительно влияет на поведение методов, поэтому здесь и далее используем ее, если не оговорено другое.

В предыдущих экспериментах выборка была представлена в виде «мешка слов» («Bag of words»). В данном эксперименте используется и другое представление: Tf-Idf, учитывающее распределение слов во всём корпусе текстов. При этом каждый документ представляется вектором длины $|V|$, то есть, размерность признакового пространства остается такой же (что будет подтверждено далее графиками). Проводится сравнение скорости работы и точности методов при этих двух представлениях, при этом используется выборка с расширенной обработкой.

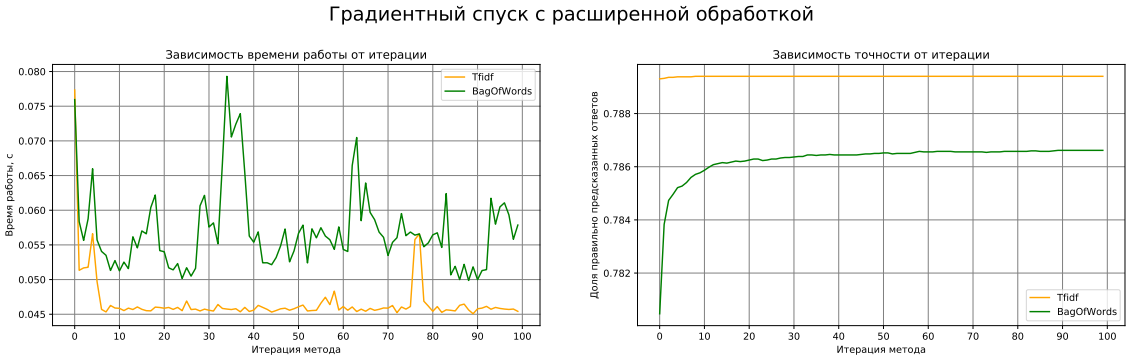


Рис. 17: Влияние способа представления на градиентный спуск

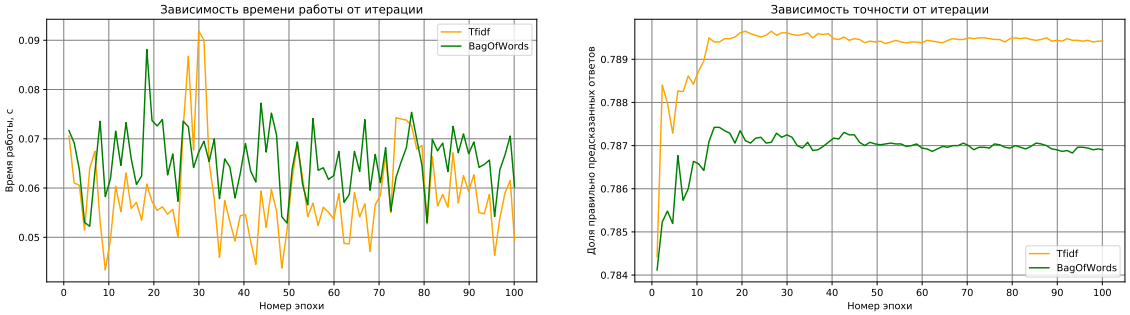


Рис. 18: Влияние способа представления на стохастический градиентный спуск

При сравнении этих двух методом получены результаты, представленные на графиках (рис. 17, 18). Представление в виде Tf-Idf дает лучшую точность и меньшее время работы, чем представление в виде Bag of words.

7 Изменение параметров min_df и max_df конструкторов

Изменение параметров min_df и max_df конструкторов позволяет уменьшить размерность признакового пространства, и, как следствие, уменьшить время работы алгоритма, а также в некоторых случаях улучшить качество работы.

Параметр max_df используется для удаления терминов, которые появляются слишком часто, также называемых "фиксированными словами". Например, max_df = 0.50 означает "игнорировать лексемы, которые появляются в более чем 50% документов". Значение max_df умолчанию 1.0, что означает "игнорировать лексемы, которые появляются в более чем 100% документов". Параметр min_df используется для удаления терминов, которые появляются слишком редко. Например, min_df = 0.01 означает "игнорировать лексемы, которые появляются менее чем в 1% документов". Значение min_df умолчанию 1, что означает "игнорировать термины, которые появляются менее чем в одном документе". Таким образом, значения по умолчанию не игнорируют никакие лексемы [3].

Ранее использовался параметр min_df = 0.0001, уменьшивший размерность пространства с 89658 признаков до 16050. В данном эксперименте исследуется влияние параметров min_df и max_df на размерность признакового пространства, скорость работы алгоритмов, точность и размерность признакового пространства. При этом используется представление и Bag of words, и Tf-Idf, и используется расширенная обработка текстов.

- Изменение параметра min_df

  Параметр max_df перебирается по сетке: {0, 0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.05, 0.1}. При этом значение min_df остается значением по умолчанию. При увеличении данного параметра скорость работы естественным образом уменьшается (прежде всего, за счет уменьшения размерности признакового пространства) (рис. 19, 20, 21, 22). Размерность признакового пространства не меняется заметно при трех минимальных перебранных min_df (рис. 23), а на четвертом минимальном (min_df=0.0001) скачок в размерности уже заметен, при этом, точность на всех четырех минимальных остается одинаково высокой. При дальнейшем увеличении параметра точность ухудшается. Получаем, что min_df=0.0001 дает максимальную точность и наименьшее из других параметров, дающих такую же точность, время работы. Заметим также, что как и было сказано ранее, размерности при обоих способах представления выборки (Tf-Idf и Bag of words) одинаковы (рис. 23).

Рис. 19: Поведение градиентного спуска при представлении Bag Of Words и изменении параметра min_df

Рис. 20: Поведение стохастического градиентного спуска при представлении Bag Of Words и изменении параметра min_df

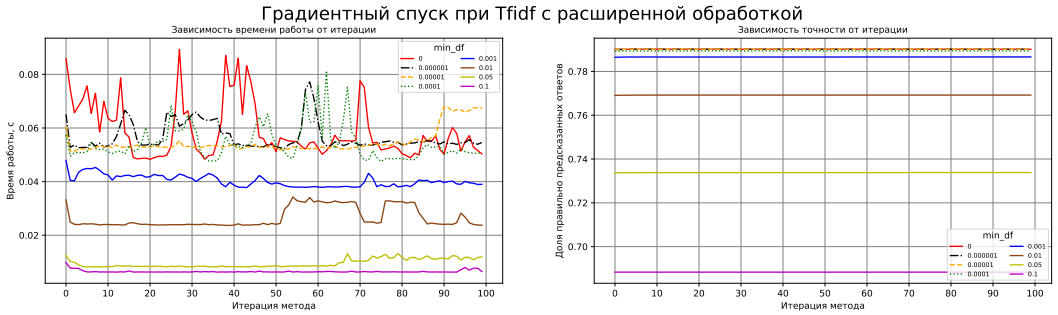


Рис. 21: Поведение градиентного спуска при представлении Tf-Idf и изменении параметра min_df

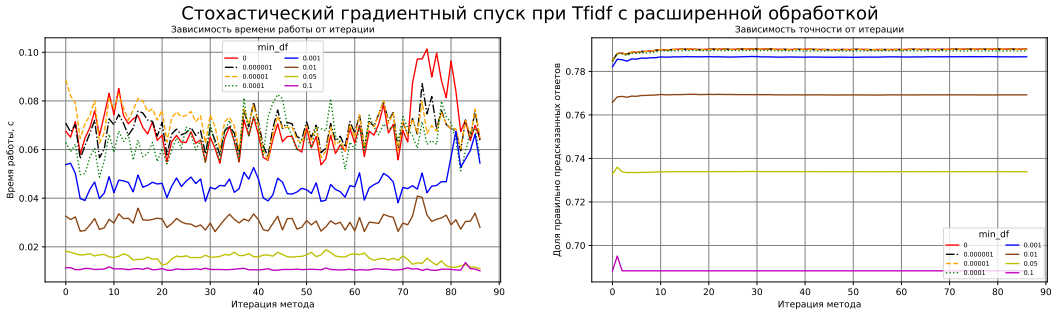


Рис. 22: Поведение стохастического градиентного спуска при представлении Tf-Idf и изменении параметра min_df

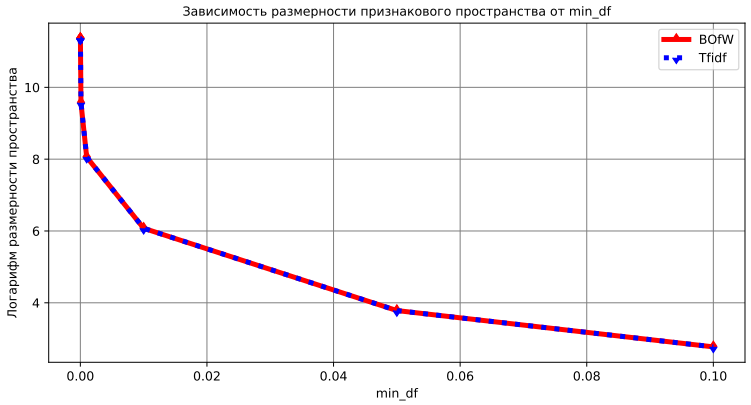


Рис. 23: Размерность признакового пространства при различных min_df

- Изменение параметра max_df

  Параметр max_df перебирается по сетке: {1.0, 0.2, 0.1, 0.05, 0.001, 0.0005, 0.0001, 0.00005}. При этом значение min_df изменено на значение по умолчанию. В отличие от случая с изменением параметра min_df, при уменьшении размерности признакового пространства в данном случае заметно улучшение качества работы модели, кроме последнего значения в сетке (рис. 24, 25, 26, 27). Таким образом, наибольшую точность дает max_df=0.0001. Заметим, что при трех максимальных перебранных max_df (1.0, 0.2, 0.1) точность не изменяется, но это связано с тем, что не меняется размерность признакового пространства (рис. 28). При уменьшении размерности признакового пространства (то есть, уменьшении max_df) время работы программы уменьшается.

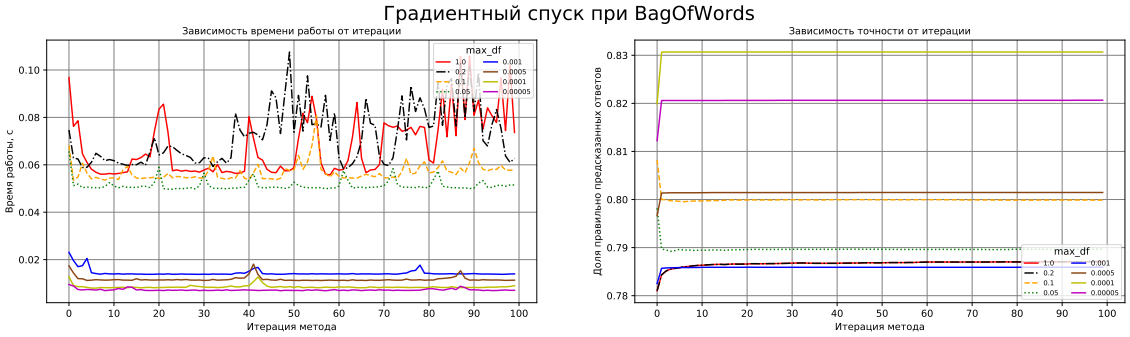


Рис. 24: Поведение градиентного спуска при представлении Bag Of Words и изменении параметра max_df

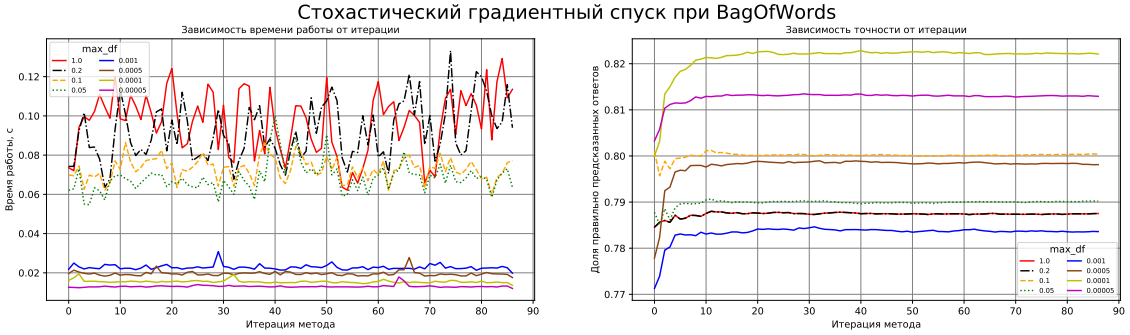


Рис. 25: Поведение стохастического градиентного спуска при представлении Bag Of Words и изменении параметра max_df

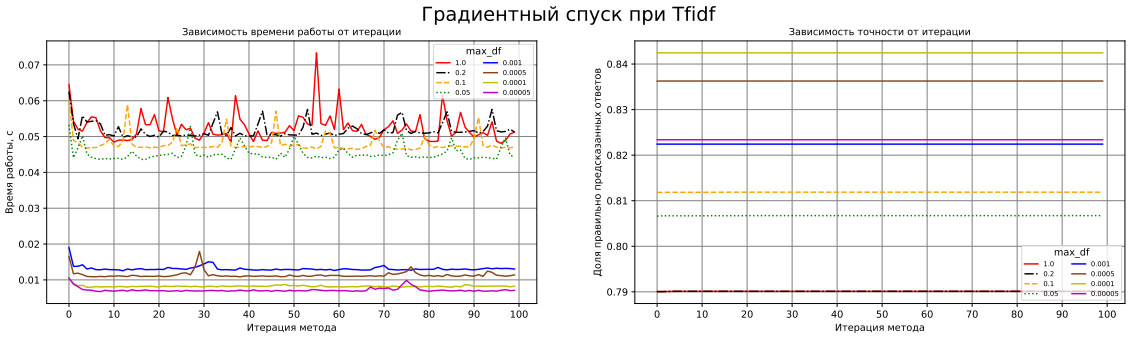


Рис. 26: Поведение градиентного спуска при представлении Tf-Idf и изменении параметра max_df

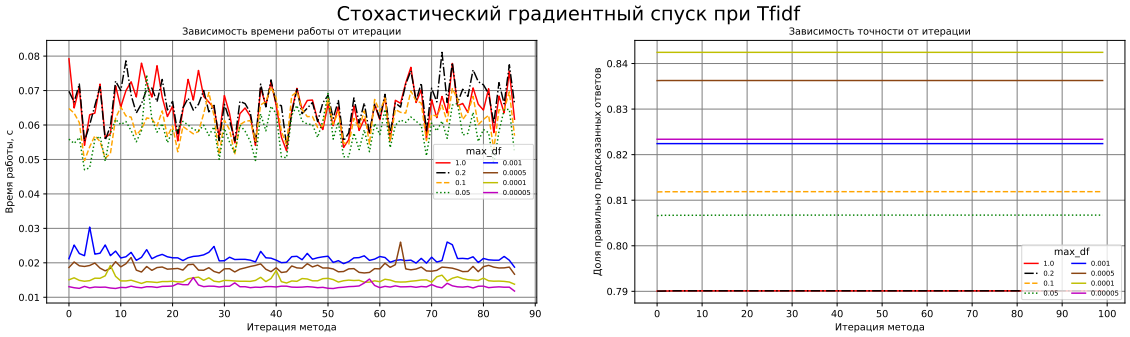


Рис. 27: Поведение стохастического градиентного спуска при представлении Tf-Idf и изменении параметра max_df

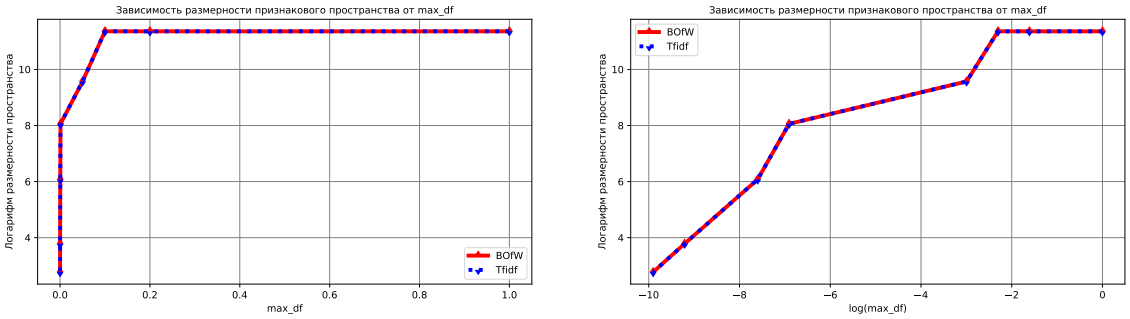


Рис. 28: Размерность признакового пространства при различных max_df

При уменьшении размерности признакового пространства увеличивается скорость работы алгоритма. При этом параметры min_df и max_df конструкторов можно подобрать такими, чтобы точность алгоритма не ухудшилась или даже улучшилась.

8 Проверка лучшего алгоритма на тестовой выборке

Параметры, лучше всего показавшие себя на обучающей выборке:

- step_alpha=0.5;
- step_beta=1.1;
- Одинаковые веса со значениями 0;

- Для стохастического градиентного спуска batch_size=10000;
- Представление выборки в виде Tf-Idf с расширенной обработкой;
- min_df = 0.0001;
- max_df = 0.0001.

В таблице 1 представлены значения точности, полученные на тестовой выборке при данных параметрах и изменениях параметров min_df и max_df. Здесь GD – градиентный спуск, SGD – стохастический градиентный спуск.

Изменения параметров min_df и max_df связаны с тем, что при одновременном применении min_df = 0.0001 и max_df = 0.0001 размер выборки уменьшился настолько, что представление тестовой выборки при TfidfVectorizer стало невозможным. Поэтому параметр min_df был уменьшен до 0.00005. При одновременном применении параметров min_df и max_df точность показала достаточно низкий результат (0.6959 и 0.6885 при градиентном и стохатическом градиентном спуске соответственно). Это можно связать с переобучением: при расширенной обработке выборки из нее и так были удалены часто встречающиеся слова. Следующие столбцы таблицы показывают, что низкие значения точности на тестовой выборке связаны именно с применением параметра max_df со значением не по умолчанию. Также на всякий случай проверено, что min_df = 0.0001 дает лучший результат, чем min_df = 0.00005.

	max_df = 0.0001 min_df = 0.0001	max_df = 0.0001 min_df = 0.00005	max_df = 0.0001 min_df = 1	max_df = 1.0 min_df = 0.0001	max_df = 1.0 min_df = 0.00005
GD	–	0.6959	0.6901	0.8159	0.8153
SGD	–	0.6885	0.6810	0.8138	0.8132

Таблица 1: Зависимость точности на тестовой выборке от параметров конструкторов

Ниже представлена матрица ошибок алгоритма на тестовой выборке при градиентном спуске и параметре min_df = 0.0001 (рис. 29). Чаще всего алгоритм допускает ошибку на комментариях, которые на самом являются негативными (True label = -1).

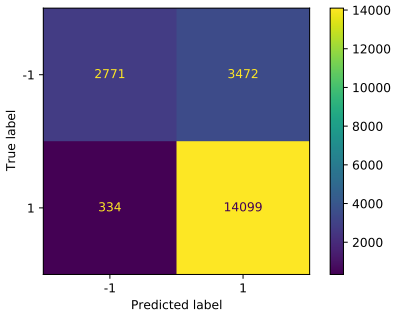


Рис. 29: Матрица ошибок при выбранных параметрах

При просмотре ошибочно определенных комментариях было замечено, что ошибки регулярно встречаются в комментариях, где употребляются слова в переносном значении и поэтому они могут перестать быть негативными (например, слово "hell" использовалось как в негативных комментариях, так и нет – зависело от контекста). Также в обучающей выборке слова, связанные с национальной принадлежностью и религиозными взглядами могли использоваться и в качестве оскорбления, и нет, поэтому на тестовой выборке при классификации таких комментариев были допущены ошибки. В целях сохранения формального стиля отчета тексты комментариев приводиться не будут.

9 Использование n-грамм

В данном эксперименте исследуется влияние размера максимальных добавленных n-грамм на качество и скорость работы градиентного спуска при лучших параметрах (обозначенных в предыдущем пункте).

С увеличением максимально добавленной n-граммы растет точность как на обучающей, так и на тестовой выборке (рис. 30, 31). При этом точность на тестовой выборке увеличилась на 0.25%. Однако, добавление n-грамм приводит к увеличению времени. Рост максимальной n-граммы ведет к росту времени. Например, для максимально исследованной n-граммы (15) время выполнения увеличилось в два раза.

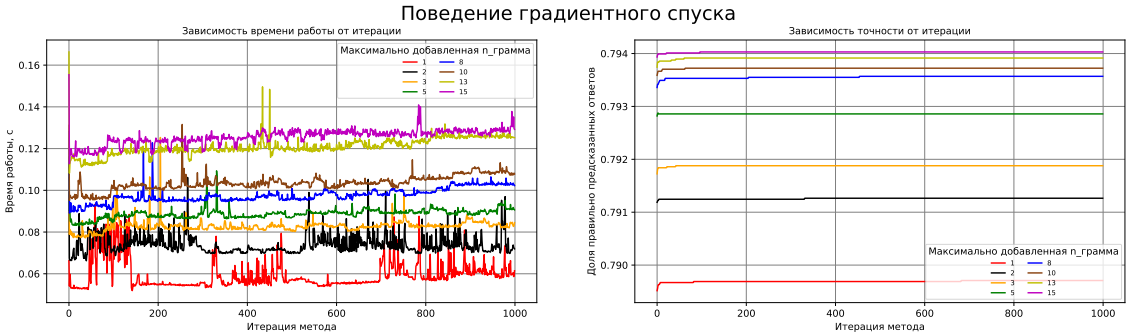


Рис. 30: Влияние максимально добавленной n-граммы на поведение градиентного спуска на обучающей выборке

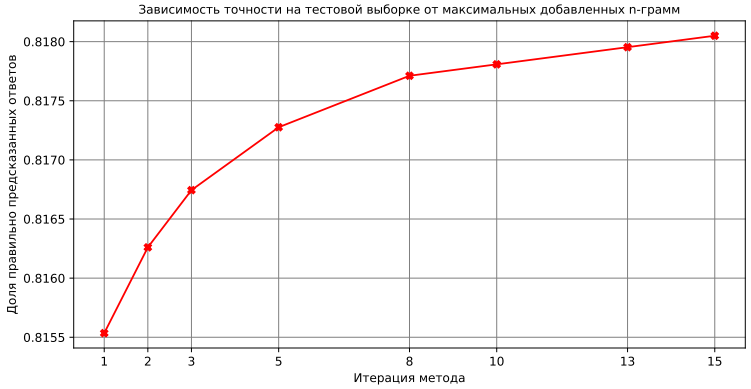


Рис. 31: Влияние максимально добавленной n-граммы на точность на тестовой выборке

10 Дополнительные улучшения

Так как негативные комментарии часто пишут с восклицательными знаками, принято решение преобразовать тестовую и обучающую выборку без удаления этих символов. Используется градиентный спуск при параметрах, обозначенных ранее, а также с использованием n-граммы (1, 15). При этом получена точность на тестовой выборке 0.8188, что немного улучшило качество модели.

Далее принято решение удалить восклицательные знаки из обучающей выборке только в не токсичных комментариях. При этом данной обработке получена точность на тестовой выборке 0.8189.

11 Выводы

Изучены методы градиентного и стохастического градиентного спуска в задаче бинарной классификации текстов. Исследована зависимость скорости работы и точности моделей в зависимости от различных параметров. Исследованы два способа представления выборки: Bag of words и Tf-Idf, а также способы уменьшения признакового пространства и их влияние на работу алгоритма.

Список используемой литературы

[1] Machinelearning – стохастический градиентный спуск

[2] Ноутбук с семинара

[3] Overcoder – про min_df и max_df

[4] Найдено абсолютно случайно