Отчет о практическом задании «Градиентные методы обучения линейных моделей».

Практикум 317 группы, ММП ВМК МГУ.

Алексеев Илья Алексеевич.

Ноябрь 2022.

Содержание

1 Введение.

Данное практическое задание посвящено исследованию градиентного спуска и стохастического градиентного спуска на примере обучения логистической регрессии в задаче распознавания токсичности текста [1]. Рассмотрена зависимость сходимости методов от величины шага (learning rate), степени затухания шага и начального приближения весов модели. Для векторизации были использованы методы `Bag of words` и `Tf-Idf`. К тексту были применены лемматизация и удаление стоп слов.

2 Пояснения к задаче.

2.1 Бинарная классификация.

Обучающая выборка $X = \{(x_i, y_i)\}_{i=1}^{\ell}$, где $x_i \in \mathbb{R}^d,\ y_i \in \mathbb{Y} = \{-1, +1\}$. Линейная модель классификации:

$$a(x) = \text{sign}(\langle w, x\rangle).$$

Отступ на i-ом объекте: $M_i(w) = y_i\langle w, x_i\rangle$. Сигмоида: $\sigma(z) = 1/(1 + e^{-z})$. Логарифмическая функция потерь: $\mathcal{L}(M) = -\log(\sigma(M))$, где $\log(x)$ – натуральный логарифм числа x. Функционал эмпирического риска:

$$Q(X, w) = -\frac{1}{\ell}\sum_{i=1}^{\ell}\log(\sigma(y_i\langle w, x_i\rangle)).$$

Производная сигмоиды по весу w_j:

$$\frac{\partial}{\partial w_j}\sigma(y\langle w, x\rangle) = \frac{\partial}{\partial w_j}\left[\frac{1}{1 + \exp(-y\langle w, x\rangle)}\right] = -\frac{-yx_j\exp(-y\langle w, x\rangle)}{(1 + \exp(-y\langle w, x\rangle))^2} = \sigma(1 - \sigma)yx_j.$$

Пусть $\sigma_i = \sigma(y_i\langle w, x_i\rangle)$. Производная по весу w_j:

$$\frac{\partial}{\partial w_j}Q(X, w) = -\frac{1}{\ell}\sum_{i=1}^{\ell}\frac{1}{\sigma_i}\frac{\partial\sigma_i}{\partial w_j} = -\frac{1}{\ell}\sum_{i=1}^{\ell}(1 - \sigma_i)y_ix_{ij}$$

Пусть $\sigma \in \mathbb{R}^\ell$ – вектор с компонентами σ_i. Пусть вектор $c = a \circ b$ таков, что $c_i = a_ib_i$, где $a, b, c \in \mathbb{R}^\ell$. Тогда градиент функционала эмпирического риска равен

$$\nabla_w Q(X, w) = -\frac{1}{\ell}X^T[y \circ (1 - \sigma)].$$

Вместе с $L2$-регуляризацией:

$$\nabla_w Q(X, w) = -\frac{1}{\ell}X^T[y \circ (1 - \sigma)] + \lambda w. \tag{1}$$

2.2 Многоклассовая классификация.

Обучающая выборка $X = \{(x_i, y_i)\}_{i=1}^{\ell}$, где $x_i \in \mathbb{R}^d,\ y_i \in \mathbb{Y} = \{1, 2, \ldots, K\}$. Линейные модели классификации:

$$a_k(x) = \text{sign}(\langle w_k, x\rangle), \quad k = \overline{1, K}.$$

Softmax-преобразование:

$$\mathbb{P}(y = j \,|\, x, w) = \frac{\exp\langle w_j, x\rangle}{\sum_{k=1}^{K} \exp\langle w_k, x\rangle}.$$

Минус логарифм правдоподобия является функционалом эмпирического риска:

$$Q(X, w) = -\frac{1}{\ell} \sum_{i=1}^{\ell} \log \mathbb{P}(y_i \,|\, x_i, w).$$

Раскроем логарифм:

$$Q(X, w) = -\frac{1}{\ell} \sum_{i=1}^{\ell} \langle w_{y_i}, x_i\rangle + \frac{1}{\ell} \sum_{i=1}^{\ell} \log \left(\sum_{k=1}^{K} \exp\langle w_k, x_i\rangle \right).$$

Пусть $w = (w_1, w_2, \ldots, w_K) \in \mathbb{R}^{d\times K}$ – матрица весов. Градиентом по вектору w_p будет p-ый столбец матрицы $\nabla_w Q$:

$$[\nabla_w Q]_p = \nabla_{w_p} Q = -\frac{1}{\ell} \sum_{i:y_i=p} x_i + \frac{1}{\ell} \sum_{i=1}^{\ell} \frac{x_i \exp\langle w_p, x_i\rangle}{\sum_{k=1}^{K} \exp\langle w_k, x_i\rangle}, \quad p = \overline{1, K}.$$

Вместе $L2$-регуляризацией:

$$[\nabla_w Q]_p = -\frac{1}{\ell} \sum_{i:y_i=p} x_i + \frac{1}{\ell} \sum_{i=1}^{\ell} \frac{x_i \exp\langle w_p, x_i\rangle}{\sum_{k=1}^{K} \exp\langle w_k, x_i\rangle} + \lambda w_p, \quad p = \overline{1, K}.$$

Пусть $K = 2$. Softmax-преобразование:

$$\mathbb{P}(y = 1 \,|\, x, w_1, w_2) = \frac{\exp\langle w_1, x\rangle}{\exp\langle w_1, x\rangle + \exp\langle w_2, x\rangle} = \frac{1}{1 + \exp\langle \underbrace{w_2 - w_1}_{\widetilde{w}}, x\rangle} = \frac{1}{1 + \exp\langle \widetilde{w}, x\rangle},$$

$$\mathbb{P}(y = 2 \,|\, x, w_1, w_2) = \frac{\exp\langle w_2, x\rangle}{\exp\langle w_1, x\rangle + \exp\langle w_2, x\rangle} = \frac{1}{1 + \exp\langle \underbrace{w_1 - w_2}_{-\widetilde{w}}, x\rangle} = \frac{1}{1 + \exp\langle -\widetilde{w}, x\rangle}.$$

Видим, что $\mathbb{P}(y = 1 \,|\, x, w_1, w_2) = \sigma(-\langle \widetilde{w}, x\rangle)$, $\mathbb{P}(y = 2 \,|\, x, w_1, w_2) = \sigma(\langle \widetilde{w}, x\rangle)$. Если подставить это в функционал эмпирического риска мультиномиальной регрессии, то получим функционал эмпирического риска логистической регрессии для $\mathbb{Y} = \{-1, +1\}$. Значит многоклассовая классификация методом мультиномиальной регрессии при $K = 2$ эквивалентна бинарной классификации методом логистической регрессии.

2.3 Градиентный спуск (GD).

Градиентным спуском (gradient descent, GD) называют итерационный метод минимизации функционала $f : \mathbb{R}^d \to \mathbb{R}$, при котором $(k + 1)$-ый член итерационной последовательности строится следующим образом:

$$w^{k+1} = w^k - \eta_k \cdot \nabla f(w^k), \; k = 0, 1, \ldots$$

Параметрами этого метода являются шаг η_k и начальное значение w_0. Мы будем рассматривать шаг вида

$$\eta_k = \frac{\alpha}{k^\beta}, \tag{2}$$

где $\alpha > 0, \beta > 0$ – параметры. Будем считать, что спуск можно остановить, если было достигнуто предельное число итераций или $|f(w^k) - f(w^{k+1})| < \varepsilon$, где ε – некоторое малое число (tolerance).

2.4 Стохастический градиентный спуск (SGD).

Стохастический градиентный спуск (stochastic gradient descent, SGD) является модификацией метода градиентного спуска для функционала вида

$$Q(w) = \sum_{i=1}^{\ell} f_i(w).$$

$(k+1)$-ый член итерационной последовательности строится следующим образом:

$$I = \{i_1, \ldots, i_b\} \sim \text{Uniform}[0, \ell],$$
$$w^{k+1} = w^k - \eta_k \cdot \frac{1}{b} \sum_{i \in I} \nabla f_i(w^k),\ k = 0, 1, \ldots$$

где I – выборка уникальных индексов размера b (batch size).

2.5 Предобработка корпуса и векторизация текста.

Пусть все слова всех документов (в нашем случае комментариев) образуют множество $\{w_1, w_2, \ldots, w_N\}$. Пусть слово w входит в документ d ровно $\text{tf}(w, d)$ раз. Тогда векторизацией методом `Bag of words` назовём представление документа d в виде вектора $v(d) \in \mathbb{R}^N$ такого, что

$$[v(d)]_i = \text{tf}(w_i, d),\ i = \overline{1, N}.$$

Пусть слово w встречается ровно в $\text{df}(w)$ документах. Пусть $\text{idf}(w) = \log(n/\,\text{df}(w)) + 1$, где n – общее число документов. Тогда векторизацией методом `Tf-Idf` назовём представление документа d в виде вектора $v(d) \in \mathbb{R}^N$ такого, что

$$[v(d)]_i = \text{tf}(w_i, d) \cdot \text{idf}(w_i),\ i = \overline{1, N}.$$

Лемматизацией текста будем называть приведение каждого его слова к начальной форме. Будем использовать стоп слова из `nltk.corpus.stopwords.words('english')` [2].

3 Эксперименты.

Все эксперименты проводились над датасетом контеста Toxic Comment Classification Challenge [1]. Тренировочный датасет был разбит на обучающую выборку размера 41649 и валидационную выборку размера 10412. Тестовая выборка имеет размер 20676.

3.1 Предобработка корпуса и векторизация текста.

Каждый комментарий из обучающей выборки был очищен от всех символов, которые не являются буквами или цифрами, и векторизован методом `Bag of words`. Чтобы сократить размер вокабулярия и время обучения, были отброшены все слова, доля которых от общего числа слов меньше 0.0001. В результате вокабулярий обучающей выборки сократился с 78694 до 15948 слов. Полученный трансформер был применен к валидационной и тестовой выборкам.

3.2 Численная проверка градиента.

Аналитическая формула градиента функции потерь логистической регрессии (1) была сравнена с численным подсчётом градиента методом конечной разности:

$$\frac{\partial f}{\partial x_i} \approx \frac{f(x + \varepsilon \cdot e_i) - f(x)}{\varepsilon},\ \varepsilon \ll 1,\ [e_i]_j = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases} \tag{3}$$

Были сгенерированы

- выборка $X \in \mathbb{R}^{\ell \times d},\ x_{ij} \sim \text{Exp}(10)$,
- ответы $y \in \mathbb{R}^{\ell},\ y_i \sim \text{Uniform}\{-1, 1\}$,
- веса $w \in \mathbb{R}^{d},\ w_i \sim \text{N}(0, 1)$.

На этих данных были найдены градиенты методами (1) и (3) при $\varepsilon = 10^{-3}$ и посчитана $L2$-норма разности между ними δ. Результаты приведены в табл. 1

d	ℓ	$\lg \delta$
50	100	-10
500	1000	-8
5000	10000	-6

Таблица 1: Десятичный порядок нормы разницы градиента, посчитанного аналитически (1) и численно (3)

3.3 Шаг градиентного спуска.

Модель логистической регрессии была обучена методом градиентного спуска для разных значений параметров α и β, используя правило обновления шага (2). По полученной итерационной последовательности весов была посчитана accuracy на валидационной выборке (рис. 1).

Заметим, что чем больше параметр α, тем большей accuracy удаётся достичь и тем больше метод совершает осцилляций; чем больше параметр β, тем меньшей accuracy удаётся достичь и тем меньше метод совершает осцилляций.

Из полученных результатов и формулы (2) следует, что параметр α отвечает за начальную величину шага: если он будет слишком мал, то метод будет сходиться долго, а если – велик, то метод будет сильно осциллировать. При этом параметр β отвечает за угасание шага: если шаг будет угасать слишком быстро, то метод не успеет сойтись, а если – медленно, то метод не перестанет осциллировать. Значит, оптимальная пара α и β способна достичь компромисс между осцилляциями и точностью.

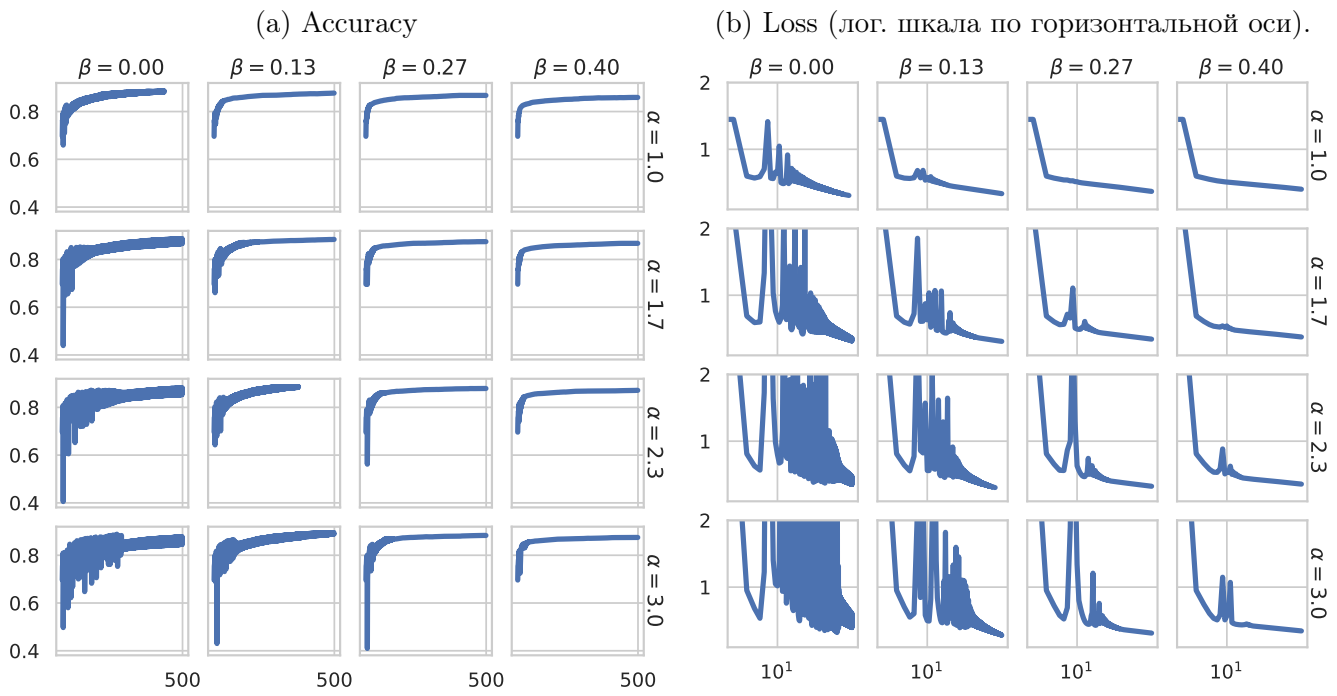


Рис. 1: Точность на валидационной выборке (по вертикали) в зависимости от шага градиентного спуска (по горизонтали).

3.4 Начальное значение градиентного спуска.

Обучим логистическую регрессию методом градиентного спуска с параметрами $\alpha = 1.93$, $\beta = 0.21$ и посмотрим на распределение получившихся весов, предварительно убрав выбросы (рис. 2a).

Данное распределение можно соотнести с распределением Гаусса, распределением Лапласа и равномерным распределением (рис. 2). Гистограмма показывает (рис. 2a), что веса сконцентрированы вокруг нуля, но с большим числом выбросов «вправо». Уберем выбросы и построим QQ-графики. По ним видно, что веса хорошо описываются указанными семействами распределений. Вычислим оценки максимального правдоподобия для параметров данных распределений. Сгенерируем начальные значения w_0 из полученных распределений.

При обучении модели мы не знаем этих оценок и распределений. Но сейчас у нас не стоит цель придумать эвристику для подбора начальных значений. Мы будем исследовать скорость сходимости градиентного спуска. Были изучены следующие начальные значения: инициализация константой (в том числе 0.0 и 1.0), инициализация выборкой из Uniform$[-1, 1]$, инициализация выборкой из Uniform$[0.2, 0.8]$, инициализация выборкой из Norm$(0, \sigma_{\text{clMML}})$, инициализация выборкой из Norm$(0.5, \sigma_{\text{clMML}})$, где σ_{clMML} – оценка максимального правдоподобия дисперсии.

- Инициализация единицами дала худший результат. Метод не успел проделать большой «путь» от единиц до нуля, вокруг которого должны концентрироваться веса модели.
- Инициализация Uniform$[0.2, 0.8]$ дала немного меньшую точность, чем инициализация Uniform$[-1, 1]$. Это тоже объясняется тем, что в первом случае метод должен проделать больший «путь» до нуля.
- Аналогично с Norm$(0, \sigma_{\text{clMML}})$ и Norm$(0.5, \sigma_{\text{clMML}})$.

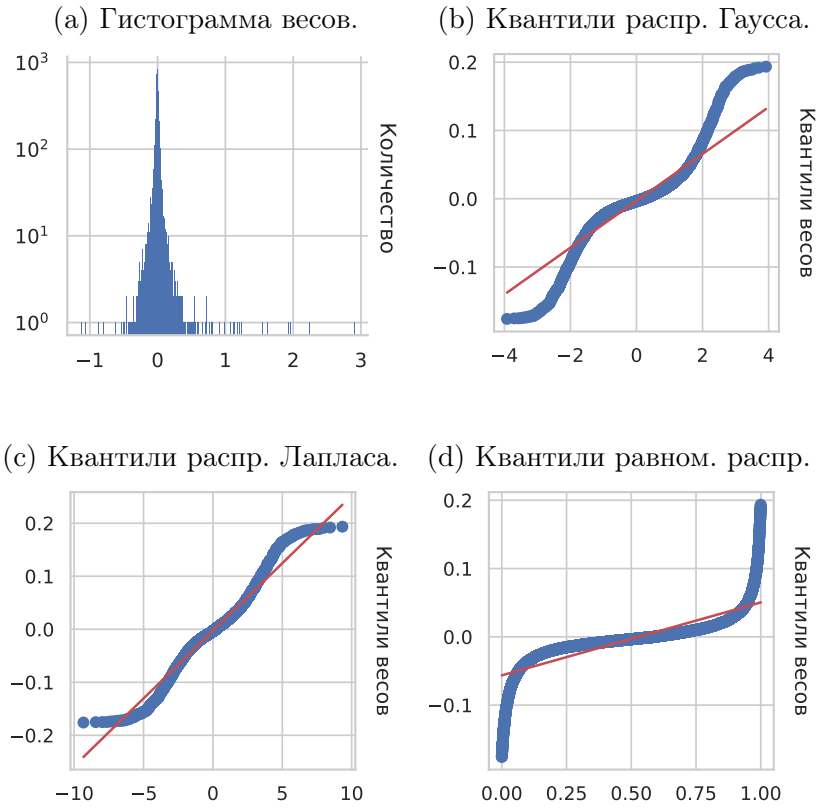


Рис. 2: Распределение весов обученной модели.

- Инициализация нулями показала результат, совпадающий с инициализацией `clean MML norm`. Значит, инициализация теоретическим средним значением весов – хорошая эвристика.

3.5 Шаг и размер батча стохастического градиентного спуска.

Эксперимент 3.3 был повторен для метода стохастического градиентного спуска. Кроме α и β исследовалась зависимость от размера батча. Результаты представлены на рис. 3.

Видим ту же тенденцию с α, β: их оптимальная пара способна достичь компромисс между осцилляциями и точностью. Заметим, что чем больше размер батча, тем большей точности методу удаётся достичь. Это объясняется тем, что с увеличением размера батча градиент, считаемый на каждой итерации, даёт более точную оценку градиента. В качестве компромисса между точностью и скоростью подсчёта будем использовать размер батча, равный 500.

3.6 Начальное значение для стохастического градиентного спуска.

Эксперимент 3.4 был повторен для метода стохастического градиентного спуска. Оценки максимального правдоподобия не подсчитаны заново, а взяты прежние. Результаты предтавлены на рисунке

Все результаты аналогичны результатам 3.4, за исключением того, что метод `zeros` дал меньшую точность, чем Uniform$[-1, 1]$; метод Norm$(0, \sigma_{\text{clMML}})$ дал большую точность, чем `clean MML lapl`.

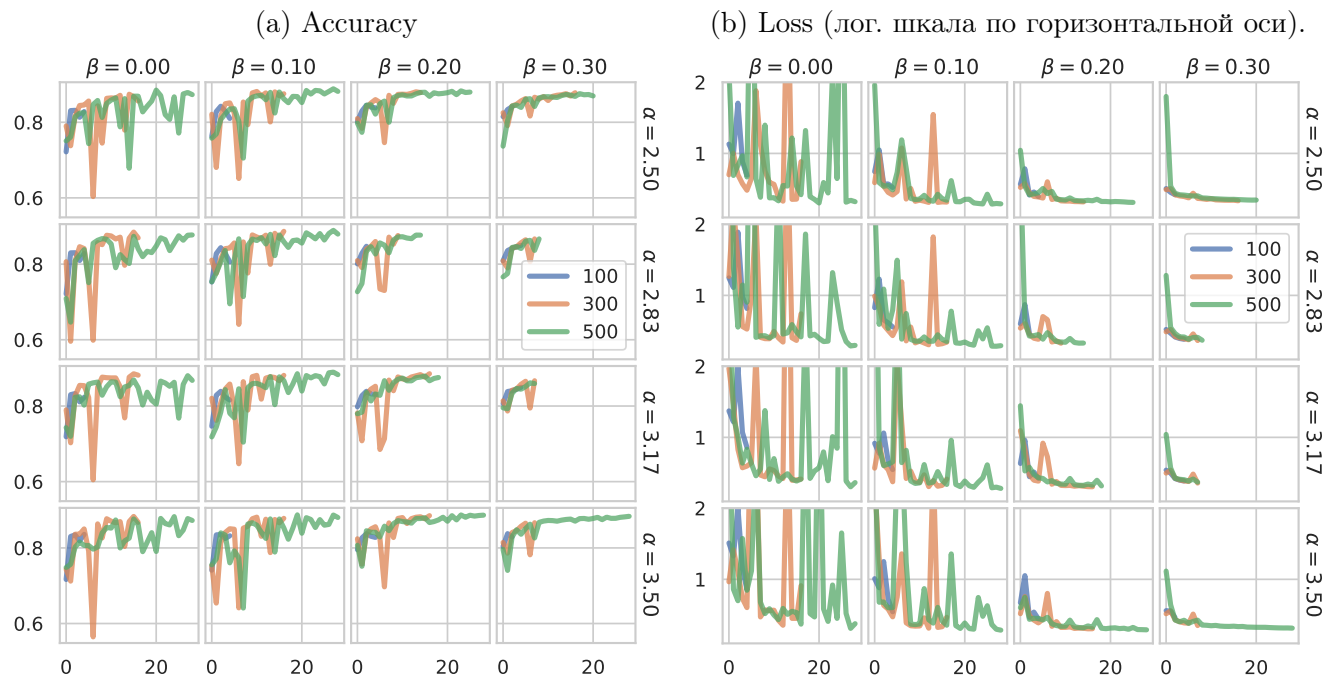


Рис. 3: Точность моделей на валидационной выборке (по вертикали) в зависимости от эпохи стохастического градиентного спуска (по горизонтали).

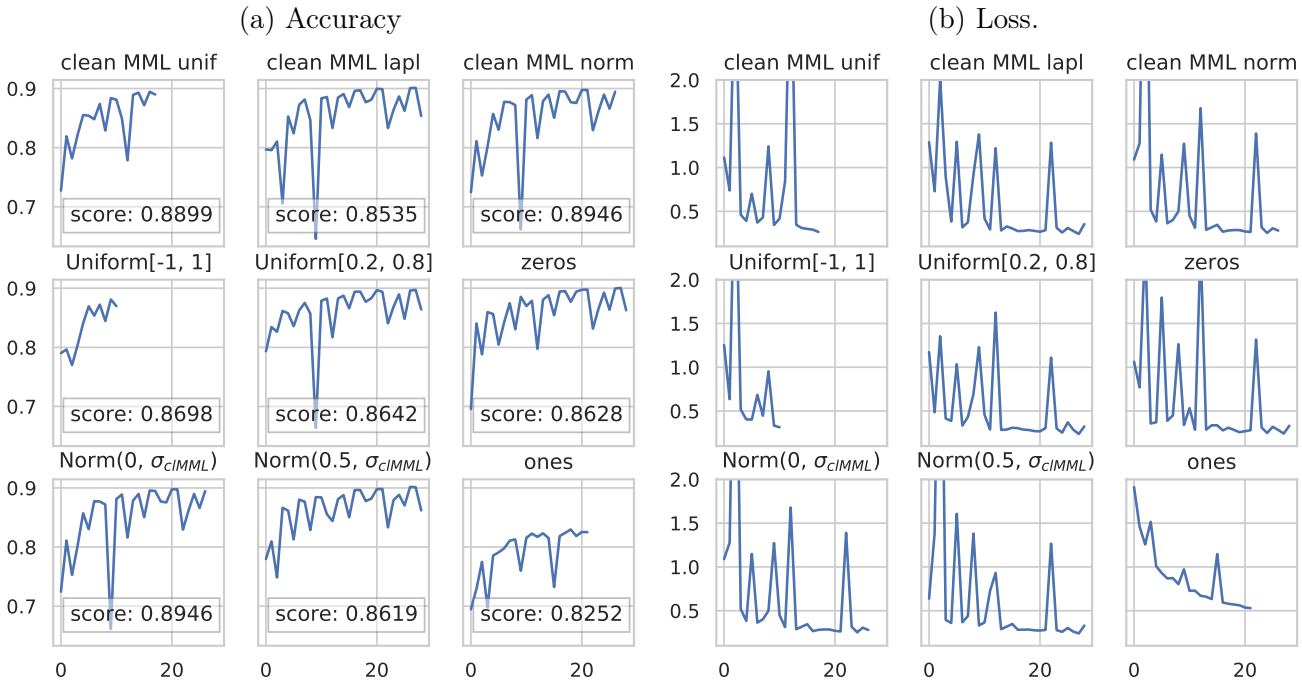


Рис. 4: Точность моделей на валидационной выборке (по вертикали) в зависимости от шага стохастического градиентного спуска (по горизонтали). Демонстрация зависимости скорости сходимости градиентного спуска от начального значения.

3.7 Сравнение GD и SGD.

Методы GD и SGD были обучены с оптимальными параметрами α, β и batch size (рис. 5). Accuracy на валидационной выборке в методе SGD имеет такой же тренд, как и в методе GD. Отличие в том, что изменение происходит немонотонно, со «скачками». Loss на обучающей выборке тоже отличается тем, что совершает «скачки». Достигнутая точность SGD оказывается больше, чем точность GD.

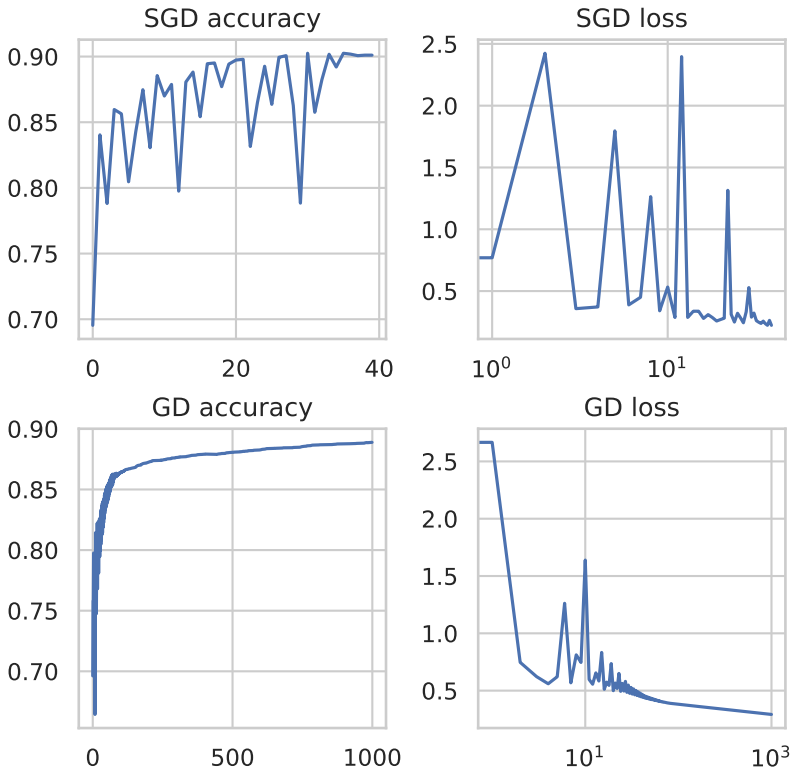


Рис. 5: Сравнение методов GD и SGD. Зависимость accuracy и loss от номера итерации спуска.

Заметим, что оптимальные параметры для GD отличаются от оптимальных параметров для SGD: $\alpha_{\text{SGD}} > \alpha_{\text{GD}}$ и $\beta_{\text{SGD}} < \beta_{\text{GD}}$. Также для метода SGD использовалось ограничение в 1500 итераций, а для GD – в 1000.

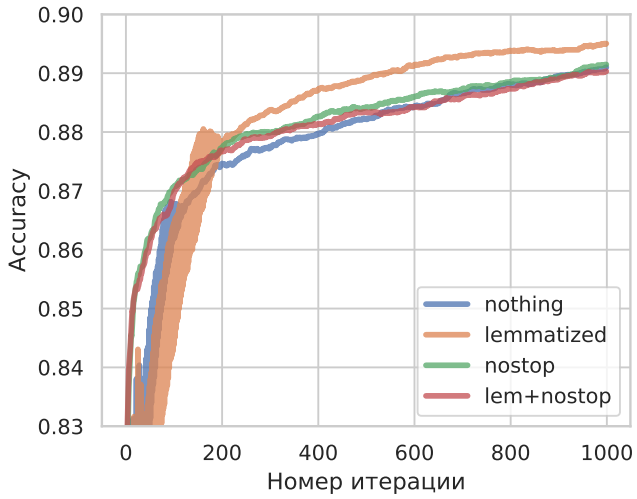
3.8 Предобработка корпуса.

С помощью `nltk.corpus.stopwords.words('english')` [2] были уделены стоп слова. Применена процедура лемматизации. Результаты приведены на рис. 6.

Наибольшую точность дала процедура лемматизации 6a. Причём удаление стоп слов из лемматизированного текста понизило качество. Значит, токсичность комментария сильно зависит от них.

Сокращение признакового пространства (табл. 2) отразилось на времени подсчёта каждой итерации (рис. 6b). Чем меньше признаков, тем быстрее.

(a) Accuracy



(b) Время одной итерации.

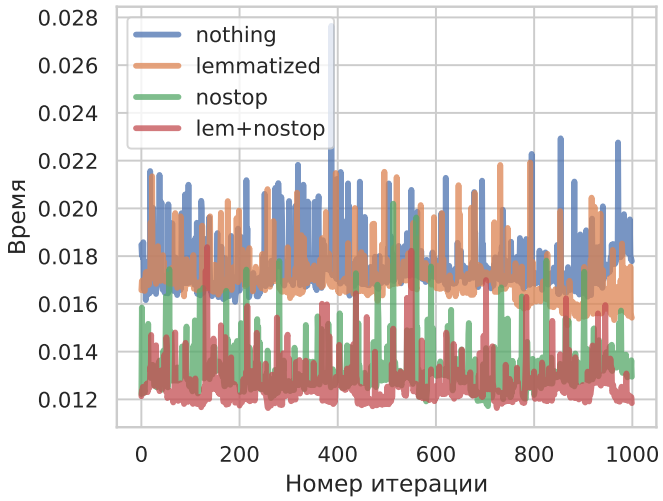


Рис. 6: Результаты предобработки корпуса.

Метод	Число признаков
`nothing`	15948
`lemmatized`	12992
`nostop`	15805
`lem+nostop`	12862

Таблица 2: Сокращение.

Число признаков	Время `BoW`	Время `Tf-Idf`
12992	20.6	13.1
3186	13.7	12.7
558	14.1	9.5
58	8.6	4.0

Таблица 3: Время счета в секундах.

3.9 Векторизация Tf-Idf.

Модель была обучена методом GD с параметрами $\alpha = 2.21,\ \beta = 0.1$ на двух версиях векторизации исходного текста: `Bag of Words` и `Tf-Idf`. С помощью параметра `min_df` [3] изменялось число используемых признаков. Результаты представлены на рис. 7.

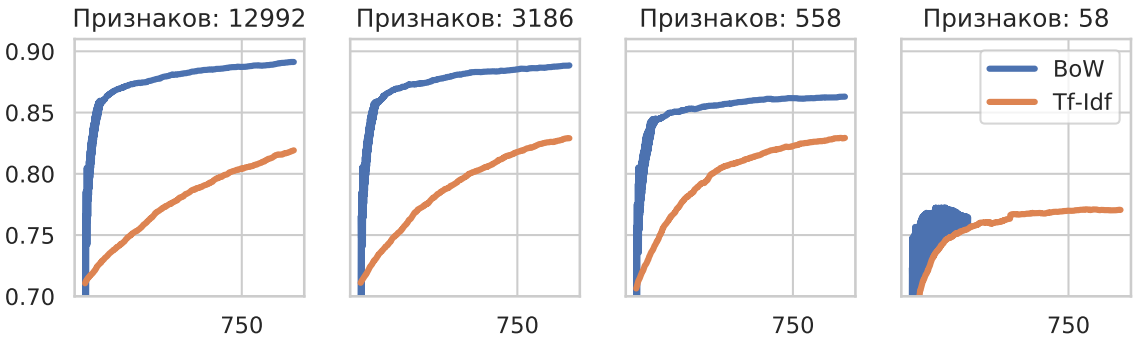


Рис. 7: Кривая обучения в зависимости от числа признаков

Векторизация `Tf-Idf` даёт меньшую точность. Это объяснимо тем, что для определения токсичности данного комментария необязательно учитывать все комментарии, как это делает `Tf-Idf`. Зависимость времени выполнения от числа признаков представлена в табл.

3.10 Оптимальный шаг GD.

Логистическая регрессия была обучена методом GD сетке параметров α и β. На рис. 8 представлены accuracy на валидационной выборке, которые в итоге удалось достичь. Ясно видна структура: к левому нижнему углу карты увеличивается accuracy, но в самом углу для некоторых моделей accuracy сильно низок. Это связано с ранее подмеченным наблюдением: при большом α и малом β метод осциллирует, поэтому финальная точность может быть какой угодно. Так что искать оптимальные параметры стоит в окрестности карты, где нет таких выбросов.

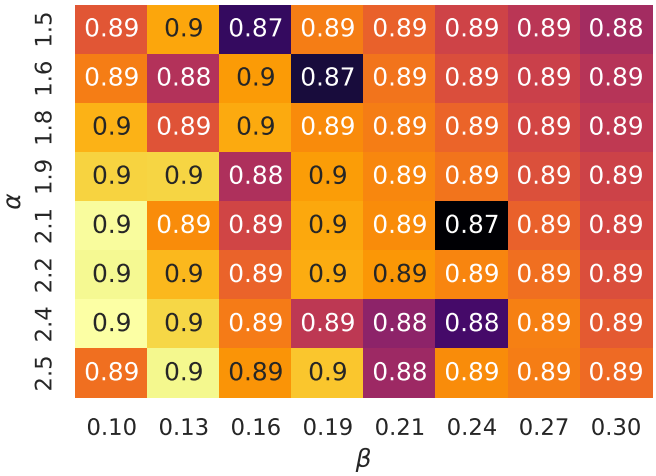


Рис. 8: Accuracy в зависимости от шага GD.

Рассмотрим окрестность карты $\alpha \in [1.5, 1.93]$, $\beta \in [0.21, 0.3]$. Она отделена от выбросов. Соответствующие ей модели не осциллириуют и не «застревают».

Поэтому оптимальными параметрами градиентного спуска будут $\alpha = 1.93$, $\beta = 0.21$, так как они обеспечивают компромисс между стабильностью сходимости и точностью.

3.11 Оптимальный шаг SGD.

Модель логистической регрессии была обучена методом SGD на сетке параметров α и β при batch size, равным 500. На рисунке 9 представлена зависимость accuracy от параметров шага. На данной карте повторяется тенденция, которая была в случае GD: чем больше α и меньше β, тем «нестабильнее» сходимость. Окрестностью, компромиссной относительно выбросов и точности, является $\alpha \in [3.6, 4.0]$, $\beta \in [0.04, 0.13]$.

В качестве оптимальных параметров SGD выберем $\alpha = 3.79$, $\beta = 0.13$.

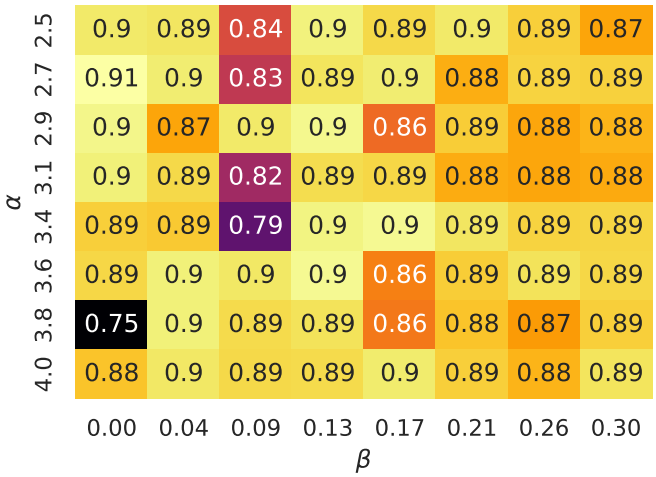


Рис. 9: Accuracy в зависимости от шага SGD.

3.12 Контроль качества.

С помощью модели логистической регрессии, обученной методом GD с оптимальными параметрами, были даны предсказания для тестовой выборки. Accuracy составила 0.8549.

Если взять элементы с максимальным отступом (т.е. те, в ответах к которым модель уверена больше всего), то нетоксичные комментарии содержат благодарности и нейтральные слова о комментируемом материале, а токсичные комментарии по большей части состоят из ругательств.

Что характерно для ошибок false positive:

- Слова kill, hate, которые употреблены не в отношении человека. Они необходимы в данном контесте: «black mamba it is ponious snake of the word and but it not kills many people but king cobra kills many people in india».

- Есть комментарий на тему истории одного ругательного слова. Это слово повторяется много раз.
- Очевидно, слово `slavery` окрашено негативно. Пример: «islam and slavery is wikipedia articles for deletion islam and slavery would you care to vote thx».
- Некоторые, кажется, произошли из-за ошибок при составлении датасета: «hello everyone i m just here to tell you that you re all freaks». Либо алгоритм смог распознать в этом шутку, а не оскорбление.
- Слово `nazi` с точки зрения модели имеет яркий негативный окрас.
- Некоторые слова имеют двойной смысл, один из них окрашен негативно. Например, «you guys are sick» помечено негативно, хотя чаще всего такую фразу употребляют, чтобы похвалить и выразить крайний восторг.

Ошибки false negative:

- Пользователи писали ругательства с парой пропущенных букв, искажая слово каким-то уникальным способом.
- Алгоритм не может распознать нераспространённые ругательства.
- Некоторые комментарии не кажутся токсичными, а просто выражают недовольство тем, как плохо сделана работа, которую они комментируют.
- Много нейтральных слов плюс несколько нелицеприятных или имеющих двойной смысл – верный способ получить ошибку false negative.

4 Выводы.

Градиентные методы GD и SGD чувствительны к выбору шага (learning rate) и начальному значению w_0. Если шаг будет слишком большим, то метод будет осциллировать вокруг оптимума. Если шаг будет слишком быстро угасать или начальное значение будет далеко от оптимума, то метод может «не добраться» до него. Оптимальный шаг должен достичь компромисс между осцилляциями и точностью.

Метод SGD дополнительно чувствителен к размеру батча. Чем он меньше, тем грубее используемая на каждой итерации оценка градиента и тем медленнее и с большими осцилляциями сходится метод. В целом, он менее устойчив в сравнении с GD и требует более точной настройки параметров.

Время работы обоих методов зависит от размера признакового пространства и размеров обрабатываемых датасетов.

Список литературы

1. Toxic Comment Classification Challenge. — URL: https://www.kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge. — (Дата обращения: 13.11.2022).
2. NLTK Documentation, Sample usage for corpus. — URL: https://www.nltk.org/howto/corpus.html?highlight=stopwords. — (Дата обращения: 13.11.2022).
3. scikit-learn documentation, TfIdfVectorizer. — URL: https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html. — (Дата обращения: 15.11.2022).