

# Практическое задание №2. Применение линейных моделей для определения сентимента новости.

Петренко Дарья, 317 группа

24 ноября 2018 г.

## Содержание

<b>Введение</b>	<b>2</b>
<b>Теоретическая часть</b>	<b>2</b>
Вычисление градиента функции потерь для бинарной регрессии . . . . .	2
Вычисление градиента функции потерь для мультиномиальной регрессии . . . . .	2
Доказательство эквивалентности задачи мультиномиальной логистической регрессии с двумя классами задаче бинарной логистической регрессии . . . . .	3
<b>Эксперименты</b>	<b>4</b>
Эксперимент №1. Исследование поведения полного градиентного спуска . . . . .	4
Исследование зависимости от параметра <i>step_beta</i> . . . . .	4
Исследование зависимости от параметра <i>step_alpha</i> . . . . .	6
Исследование зависимости от начального приближения . . . . .	7
Эксперимент №2. Исследование поведения стохастического градиентного спуска. . . . .	8
Исследование зависимости от <i>batch_size</i> . . . . .	8
Исследование зависимости от <i>step_beta</i> . . . . .	10
Исследование зависимости от <i>step_alpha</i> . . . . .	11
Зависимость от начального приближения . . . . .	12
Выводы о поведении двух исследованных методов . . . . .	12
Эксперимент №3. Многоклассовая классификация . . . . .	12
Эксперимент №4. Предобработка данных . . . . .	13
Эксперимент №5. Исследование методов предобработки и представления данных . . . . .	13
Сравнение работы методов с использованием Bag of Words и Tf-Idf . . . . .	13
Исследование зависимости от параметра <i>min_df</i> . . . . .	13
Исследование зависимости от параметра <i>max_df</i> . . . . .	14
Эксперимент №6. Исследование ошибок метода с оптимальным набором параметров. . . . .	14
Пример 1 . . . . .	14
Пример 2 . . . . .	15

# Введение

В ходе выполнения задания были реализованы бинарный и многоклассовый классификаторы методом логистической регрессии. Минимизация функционала производилась методом градиентного спуска и стохастического градиентного спуска. Также были реализованы алгоритмы многоклассовой классификации "one vs all" и "all vs all" на основе бинарного классификатора.

Эксперименты производились на датасете русскоязычных новостей, разделенных на классы 'positive', 'negative' и 'neutral'. Для бинарной классификации использовались только объекты классов 'positive' и 'negative'.

## Теоретическая часть

### Вычисление градиента функции потерь для бинарной регрессии

$$Q(X, w) = \frac{1}{l} \cdot \sum_{i=1}^l \log(1 + \exp(-y_i \cdot \langle w, x_i \rangle)) + \frac{\lambda}{2} \cdot \|w\|_2^2$$

Градиент для одной пары  $(x_i, y_i)$  равен

$$\frac{dQ}{dw} = \frac{-y_i \cdot \exp(-y_i \cdot \langle w, x_i \rangle)}{1 + \exp(-y_i \cdot \langle w, x_i \rangle)} \cdot x_i$$

### Вычисление градиента функции потерь для мультиномиальной регрессии

$$Q(X, w) = -\frac{1}{l} \sum_{i=1}^l \log\left(\frac{\exp(\langle w_{y_i}, x_i \rangle)}{\sum_{k=1}^K \exp(\langle w_k, x_i \rangle)}\right) + \frac{\lambda}{2} \sum_{k=1}^K \|w_k\|_2^2$$

Таким образом, производная функции для конкретной пары  $(x_i, y_i)$  для разных компонент матрицы  $w$  различна в зависимости от того, совпадает значение  $y_i$  с индексом компоненты  $w$  или нет. Рассмотрим оба варианта.

Если  $y_i = w_j$ , то

$$\begin{aligned} \frac{dQ}{dw_j} &= -1 \cdot \frac{\sum_{k=1}^K \exp(\langle w_k, x_i \rangle) \exp(\langle w_j, x_i \rangle) \cdot x_i \cdot \sum_{k=1}^K \exp(\langle w_k, x_i \rangle) - \exp(\langle w_j, x_i \rangle) \cdot \exp(\langle w_j, x_i \rangle) \cdot x_i}{\left(\sum_{k=1}^K \exp(\langle w_k, x_i \rangle)\right)^2} + \\ &+ \lambda \cdot w_j = -x_i + \frac{\exp(\langle w_j, x_i \rangle)}{\sum_{k=1}^K \exp(\langle w_k, x_i \rangle)} \cdot x_i + \lambda \cdot w_j = x_i \left(-1 + \frac{\exp(\langle w_j, x_i \rangle)}{\sum_{k=1}^K \exp(\langle w_k, x_i \rangle)}\right) + \lambda \cdot w_j \end{aligned}$$

Если  $y_i \neq w_j$ , то

$$\begin{aligned} \frac{dQ}{dw_j} = & -1 \cdot \frac{\sum_{k=1}^K \exp(\langle w_k, x_i \rangle)}{\exp(\langle w_{y_i}, x_i \rangle)} \cdot -1 \cdot \frac{\exp(\langle w_{y_i}, x_i \rangle) \cdot \exp(\langle w_j, x_i \rangle) \cdot x_i}{\left( \sum_{k=1}^K \exp(\langle w_k, x_i \rangle) \right)^2} + \lambda \cdot w_j = \\ & x_i \cdot \frac{\exp(\langle w_j, x_i \rangle)}{\sum_{k=1}^K \exp(\langle w_k, x_i \rangle)} + \lambda \cdot w_j \end{aligned}$$

Таким образом, итоговое выражение можно записать как

$$\frac{dQ}{dw_j} = x_i \left( -1 \cdot [y_i = w_j] + \frac{\exp(\langle w_j, x_i \rangle)}{\sum_{k=1}^K \exp(\langle w_k, x_i \rangle)} \right) + \lambda \cdot w_j$$

## Доказательство эквивалентности задачи мультиномиальной логистической регрессии с двумя классами задаче бинарной логистической регрессии

Преобразуем функцию потерь мультиномиальной логистической регрессии (регуляризацию не будем учитывать, тк она не играет роли). За  $w_{-1}$  и  $w_1$  будем обозначать векторы весов для -1 и 1 классов соответственно.

$$\begin{aligned} -\frac{1}{l} \sum_{i=1}^l \log \left( \frac{\exp(\langle w_{y_i}, x_i \rangle)}{\exp(\langle w_{-1}, x_i \rangle) + \exp(\langle w_1, x_i \rangle)} \right) &= \frac{1}{l} \sum_{i=1}^l \log \left( \frac{\exp(\langle w_{-1}, x_i \rangle) + \exp(\langle w_1, x_i \rangle)}{\exp(\langle w_{y_i}, x_i \rangle)} \right) = \\ &= \frac{1}{l} \sum_{i=1}^l \log (1 + \exp(\langle w_{-y_i}, x_i \rangle - \langle w_{y_i}, x_i \rangle)) = \frac{1}{l} \sum_{i=1}^l \log (1 + \exp(\langle w_{-y_i} - w_{y_i}, x_i \rangle)) \rightarrow \min_{w_{-1}, w_1} \end{aligned}$$

Заметим, что при  $w = w_1 - w_{-1}$  полученное выражение в точности составляет функцию потерь для бинарной классификации: выражение под экспонентой равно  $-y_i \cdot \langle w_1 - w_{-1}, x_i \rangle$ . При минимизации исходной функции по  $w_1$ ,  $w_{-1}$  будет производиться минимизация полученной функции по  $w$ , что и соответствует задаче бинарной классификации.

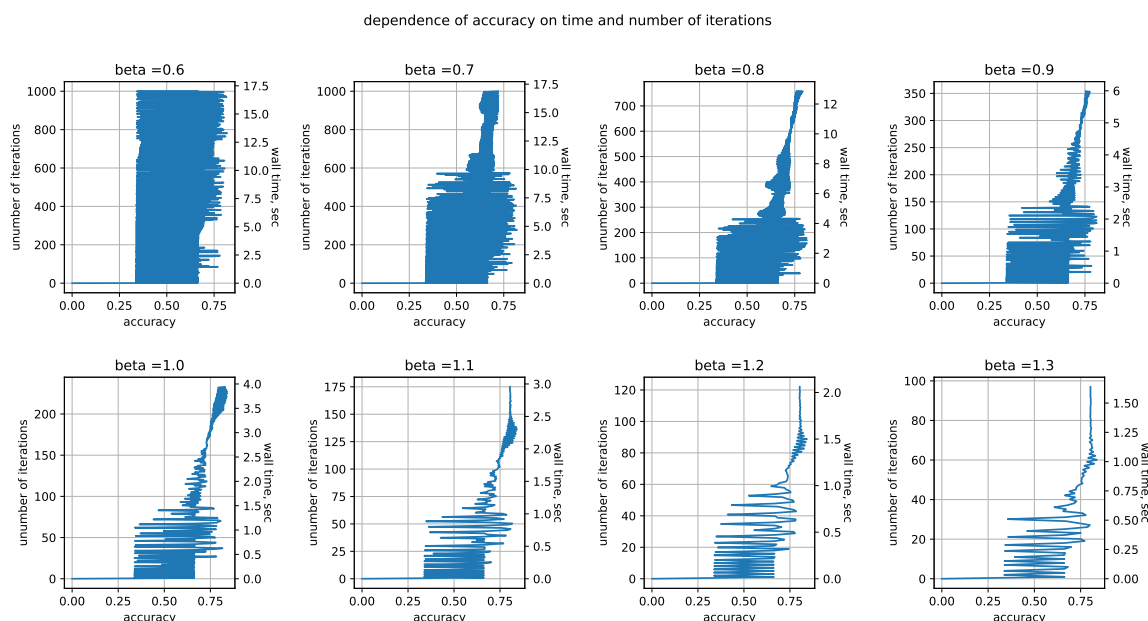
# Эксперименты

## Эксперимент №1. Исследование поведения полного градиентного спуска

Исследование проведено на предобработанном датасете русскоязычных новостей. Выбраны только новости из категорий 'negative' и 'positive', категория новости является значением целевой переменной. Из текстов исключены все символы, кроме букв и цифр. Далее каждый текст преобразован в разреженную матрицу при помощи функции *CountVectorizer* из библиотеки *sklearn.feature\_extraction.text*. Для ускорения проведения экспериментов размерность матриц была уменьшена путем выбора значения параметра  $min\_df = 0.1$  (число признаков составляет 284).

### Исследование зависимости от параметра $step\_beta$

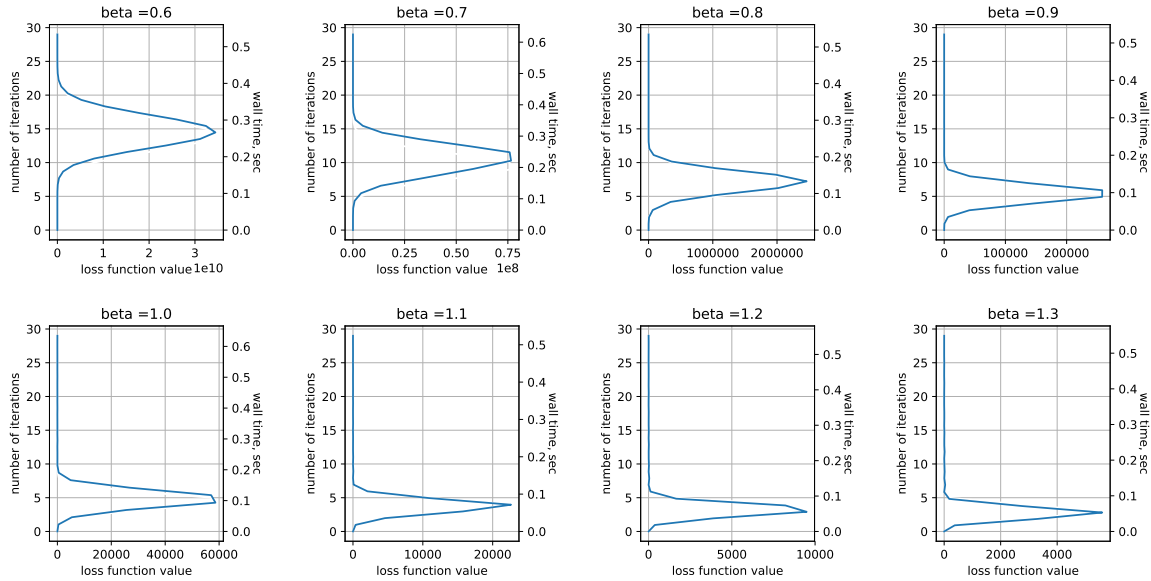
Рассмотрим поведение градиентного спуска в зависимости от изменения значения параметра  $step\_beta$ . Для этого зафиксируем значения остальных параметров и проведем ряд экспериментов, в ходе каждого из которых будем вычислять и сохранять в специальной структуре значение лосс-функции, полученную точность предсказания (проверяется на тестовой выборке; для этого тестовая выборка передается в метод *fit* в качестве параметра), а также текущее общее время работы программы на каждой итерации.



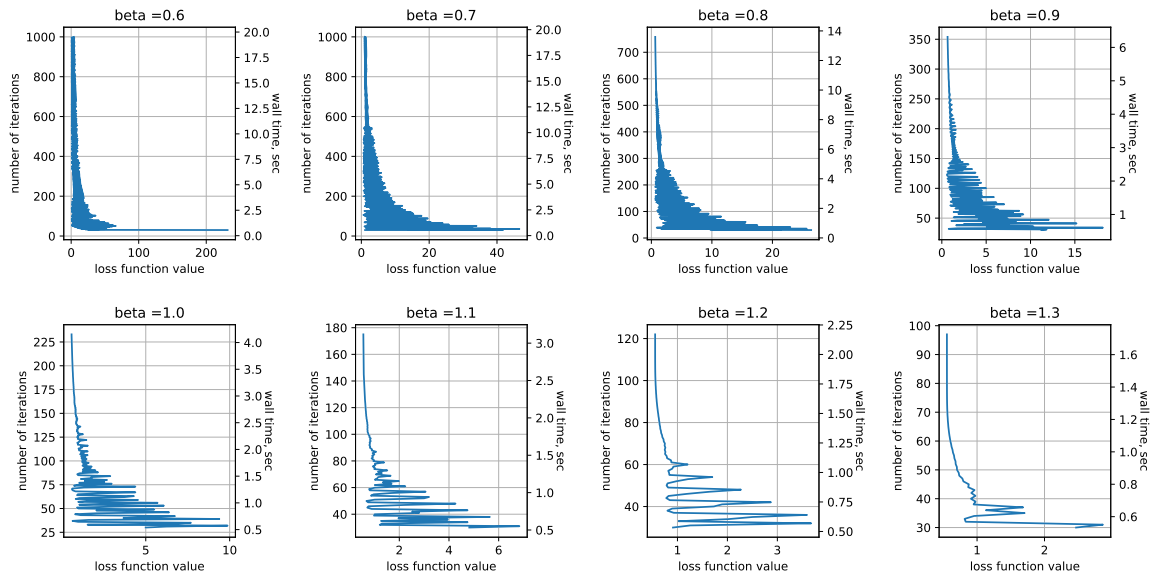
При слишком маленьких значениях параметра по мере приближения к искомой точке экстремума функции потерь шаг продолжает оставаться достаточно большим, поэтому метод не может попасть в искомую точку, а постоянно "перепрыгивает" через нее и возвращается обратно. С увеличением значения параметра метод сходится быстрее (это можно увидеть на графиках: количество итераций и полное время работы уменьшаются), а точность попа-

дания в экстремум увеличивается. При  $step\_beta=0.2$  и  $step\_beta=0.3$  конечная точность предсказаний почти одинакова и составляет 0.803. При дальнейшем увеличении значения параметра точность сначала остается такой же, а затем начинает падать. Это может быть связано с тем, что шаг уменьшается так быстро, что метод просто не успевает дойти до точки экстремума до того, как модуль изменения значения лосс-функции на соседних итерациях станет меньше заданного в методе `fit` числа и произойдет остановка.

dependence of loss function value on time and number of iterations



dependence of loss function value on time and number of iterations



При исследовании зависимости функции потерь от времени работы и числа итераций диапазон принимаемых значений на первых нескольких десятках итераций и на оставшихся сильно различается, поэтому на графиках в силу удобства и большей наглядности изображены отдельно первые 30 и последующие итерации.

При любом значении параметра сначала происходит резкое увеличение значений функции потерь, а затем резкий спад, значения функции потерь становятся близкими к нулю. Это видно на графиках, изображающих первые 30 итераций. Чем меньше значение параметра, тем больше диапазон значений во время скачка и тем медленнее он происходит. Вероятно, это связано с тем, что чем меньше  $step\_beta$ , тем больше шаг.

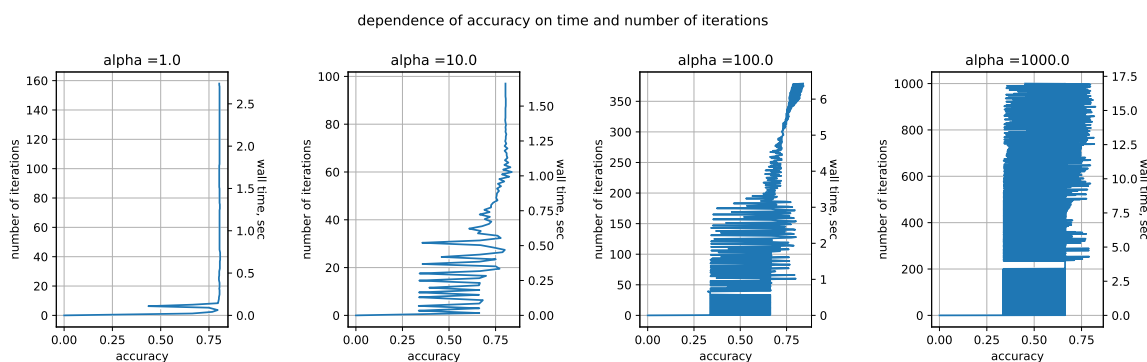
На последующих итерациях лосс-функция постепенно сходится к нулю. Аналогично исследованной зависимости точности, чем меньше значение параметра, тем больше колеблется функция по мере сходимости и тем больше итоговое значение функции потерь. Итоговые значения для различных значений  $step\_beta$  приведены в таблице.

Зависимость итоговой точности от значения параметра

значение параметра	точность предсказания
0.6	2.68
0.7	0.96
0.8	0.63
0.9	0.64
1.0	0.57
1.1	0.55
1.2	0.55

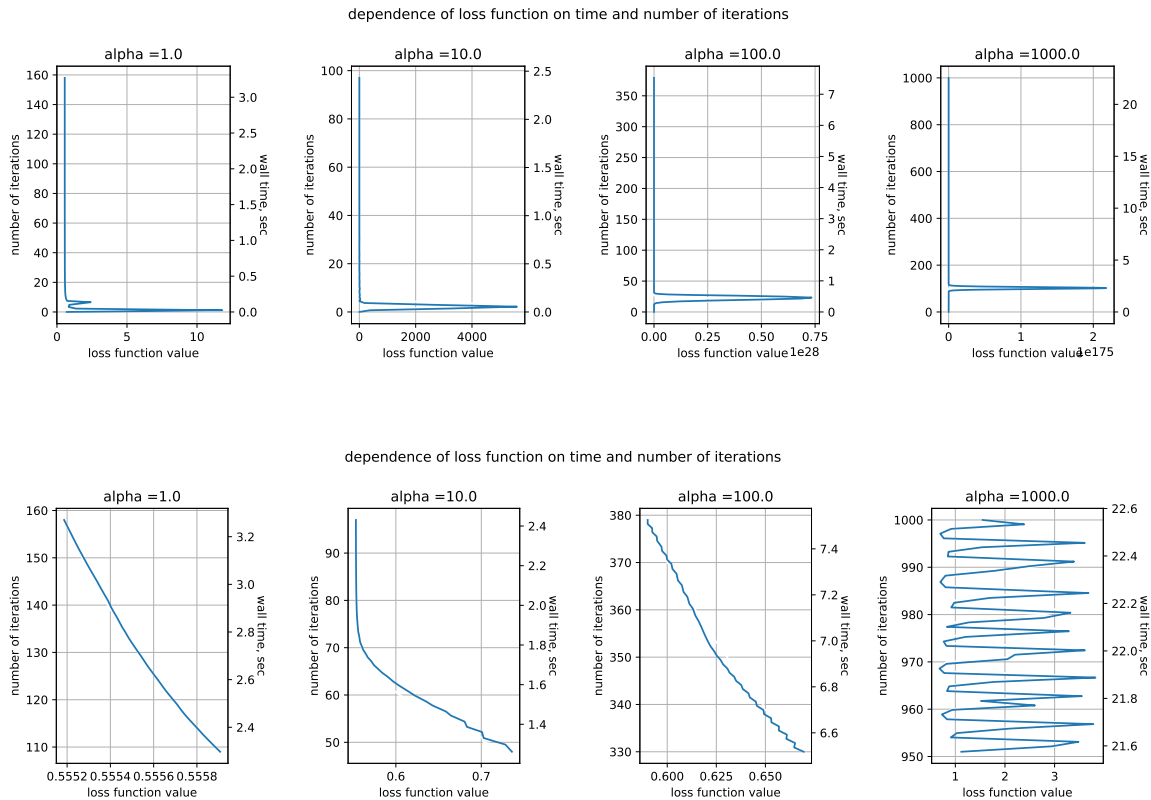
## Исследование зависимости от параметра $step\_alpha$

Зафиксируем значения всех параметров, кроме  $step\_alpha$  (значение  $step\_beta$  возьмем равным оптимальному по результатам предыдущего эксперимента) и исследуем поведение метода при изменении этого параметра.



Точность, достигаемая при значениях параметра, равных 1 и 10, совпадает и равна 0.803. Различается только количество итераций, которое понадобилось для сходимости метода (чем больше шаг, тем быстрее сходимость). Поэтому, выбирая из этих методов оптималь-

ный, можно принять решение в пользу  $step\_alpha = 10$ , тк он позволяет ускорить обучение без ущерба качеству. Дальнейшее увеличение параметра негативно сказывается на сходимости метода к точке экстремума. На графиках видно, что метод начинает колебаться около искомой точки, на каждой итерации "перепрыгивая" ее и возвращаясь, тк размер шага слишком большой для точного попадания. При исследовании зависимости значения

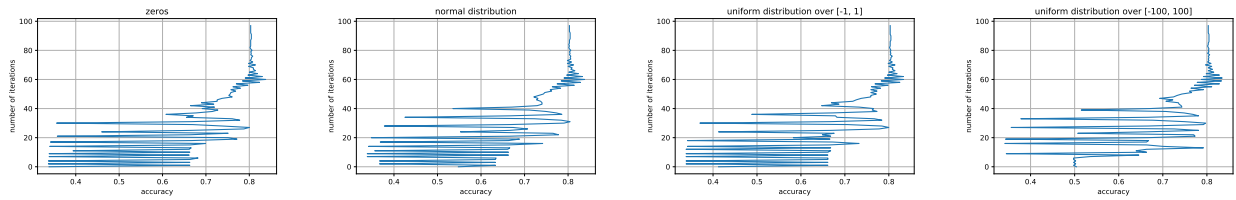


лосс-функции от времени и от числа итераций аналогично предыдущему эксперименту были отдельно построены графики для всего числа итераций и для последних 50-ти. На первых графиках виден резкий скачок (чем больше значение параметра, тем большие значения достигаются на пике; это ожидаемо, тк чем больше  $step\_alpha$ , тем больше шаг), а затем постепенное стремление к нулю.

Более детально сходимость на последних итерациях видна на втором наборе графиков. Чем больше значение параметра, тем больше колеблется метод при приближении к нулю.

## Исследование зависимости от начального приближения

Для исследования зависимости были зафиксированы значения всех параметров, кроме  $w_0$ , и в качестве значения исследуемого аргумента поданы векторы значений, взятых из какого-то распределения (равномерного на различных интервалах, равномерного), нулевой вектор. Графики зависимости точности от количества итераций приведены ниже.



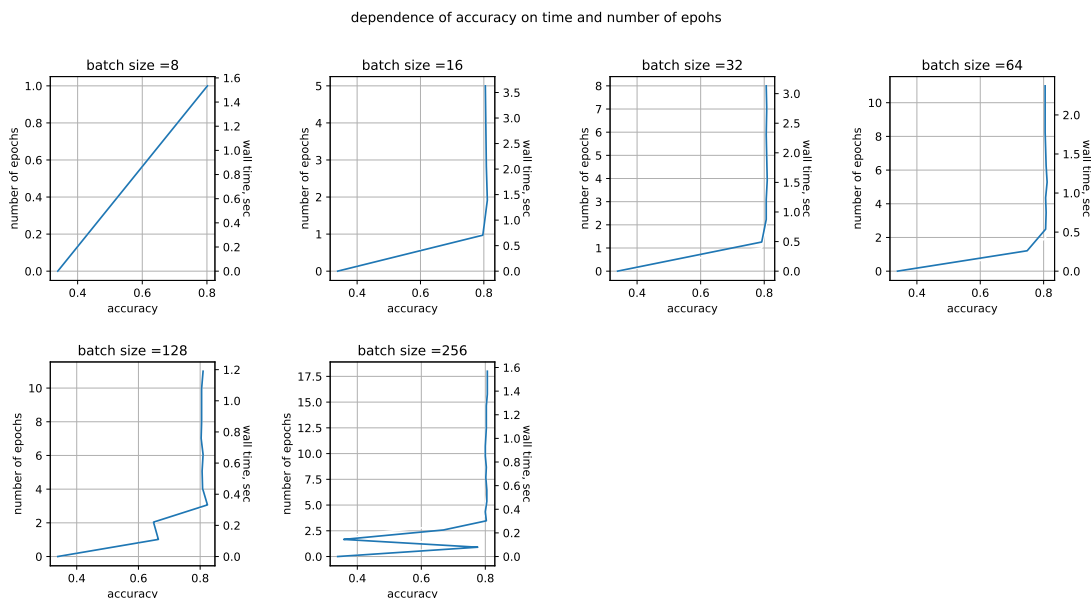
Можно сделать вывод, что зависимость от начального приближения незначительна и все различия исчезают после первых нескольких десятков итераций. Общее число итераций для всех проведенных экспериментов одинаково и равно 98, конечная точность также совпадает и составляет 0.803.

## Эксперимент №2. Исследование поведения стохастического градиентного спуска.

### Исследование зависимости от $batch\_size$

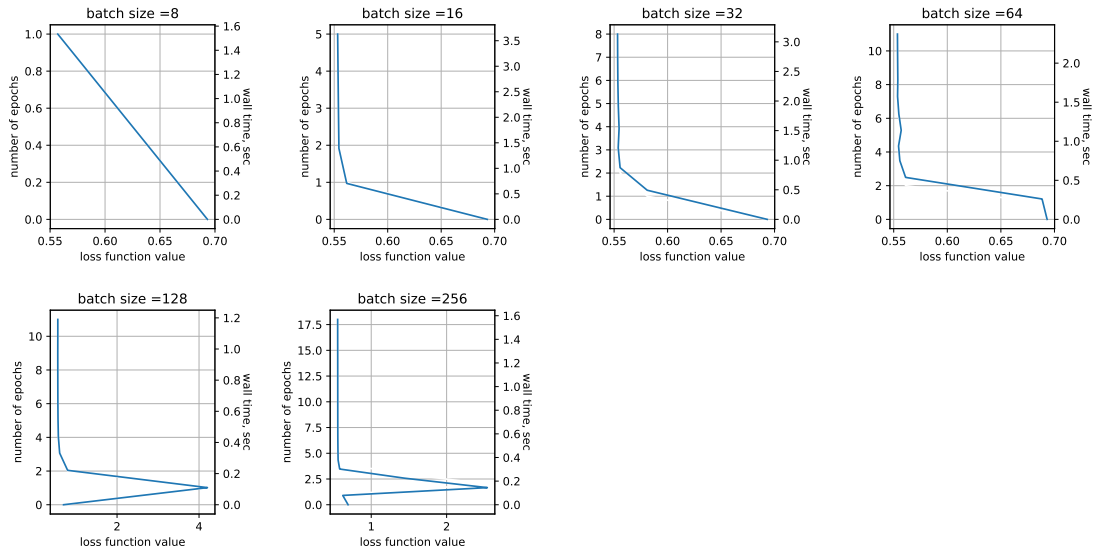
Зафиксируем значения всех параметров, кроме исследуемого, и посмотрим на поведение метода при его изменении. Тк одна эпоха включает в себя несколько итераций, то для достижения сходимости метода количество эпох нужно взять больше, чем в экспериментах по исследованию полного градиентного спуска. Более того, на каждой итерации функция потерь меняется не так сильно, как при полном спуске, тк градиент берется только по нескольким ее компонентам. Поэтому во избежание преждевременного выхода из цикла в методе `fit` значение параметра *tolerance* необходимо взять меньше, чем в предыдущих экспериментах.

Исследования проводились на том же датасете, что и эксперимент №1, при значениях параметров  $batch\_size = 100000$ ,  $tolerance = 1e - 6$ ,  $step\_alpha = 10$ ,  $step\_beta = 1.3$ .

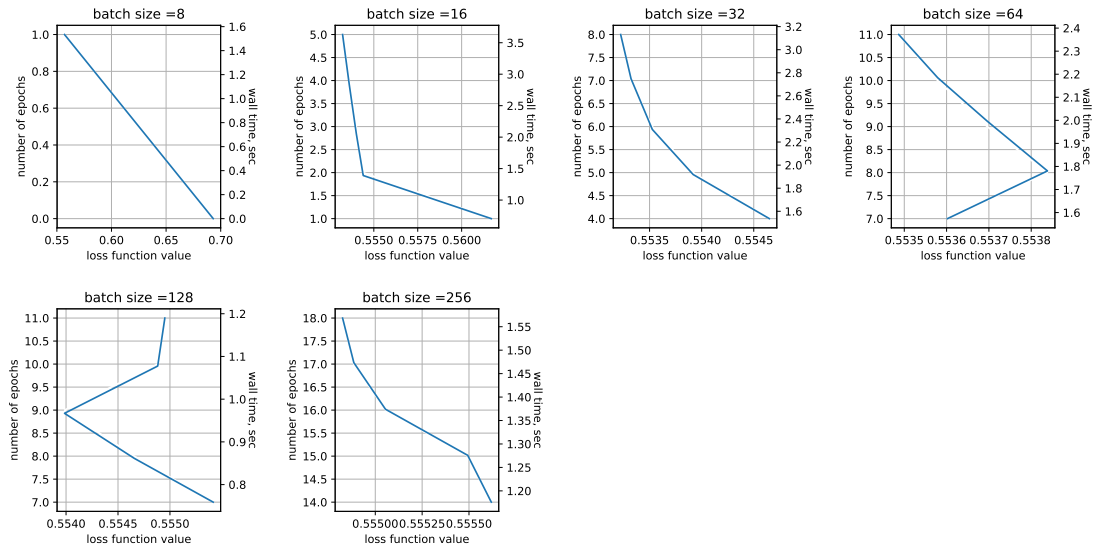




dependence of loss function value on time and number of epochs



dependence of loss function value on time and number of epochs

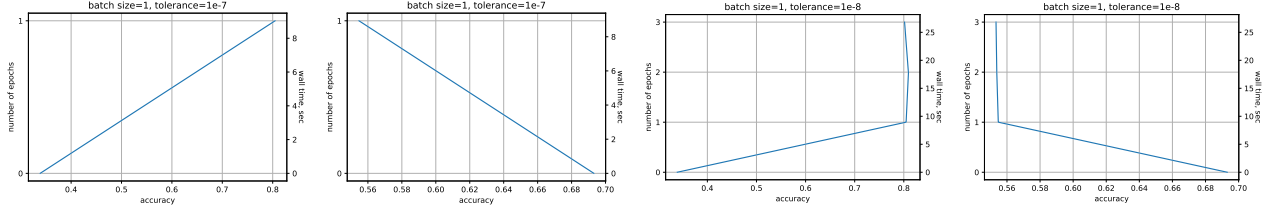


Итоговая точность, достигнутая на каждом эксперименте, приведена в таблице.

Зависимость итоговой точности от размера батча

размер батча	точность предсказания
8	0.802
16	0.804
32	0.806
64	0.804
128	0.809
256	0.806

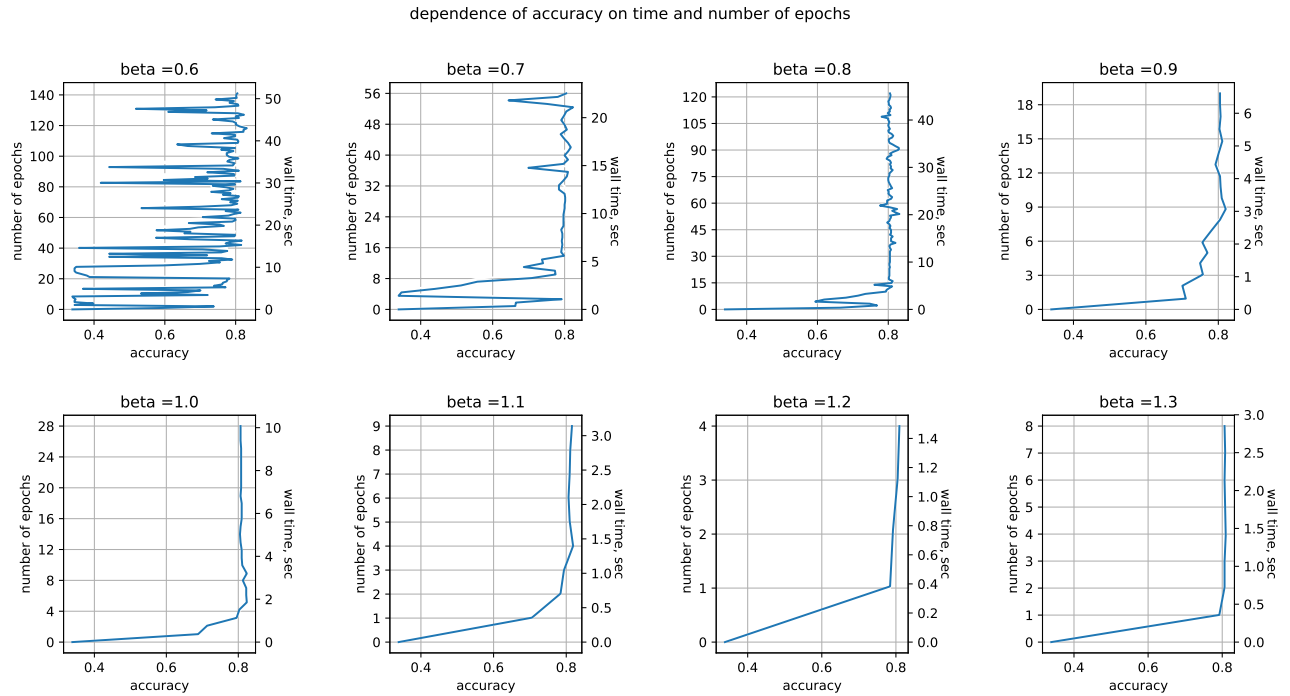
Исследования для  $batch\_size = 1$  проведем при других значениях параметра  $tolerance$ , тк при  $tolerance = 1e - 6$  работа метода завершается до окончания первой эпохи. При  $tolerance = 1e - 7$  итоговая точность составила 0.804, при  $tolerance = 1e - 8$  - 0.801.



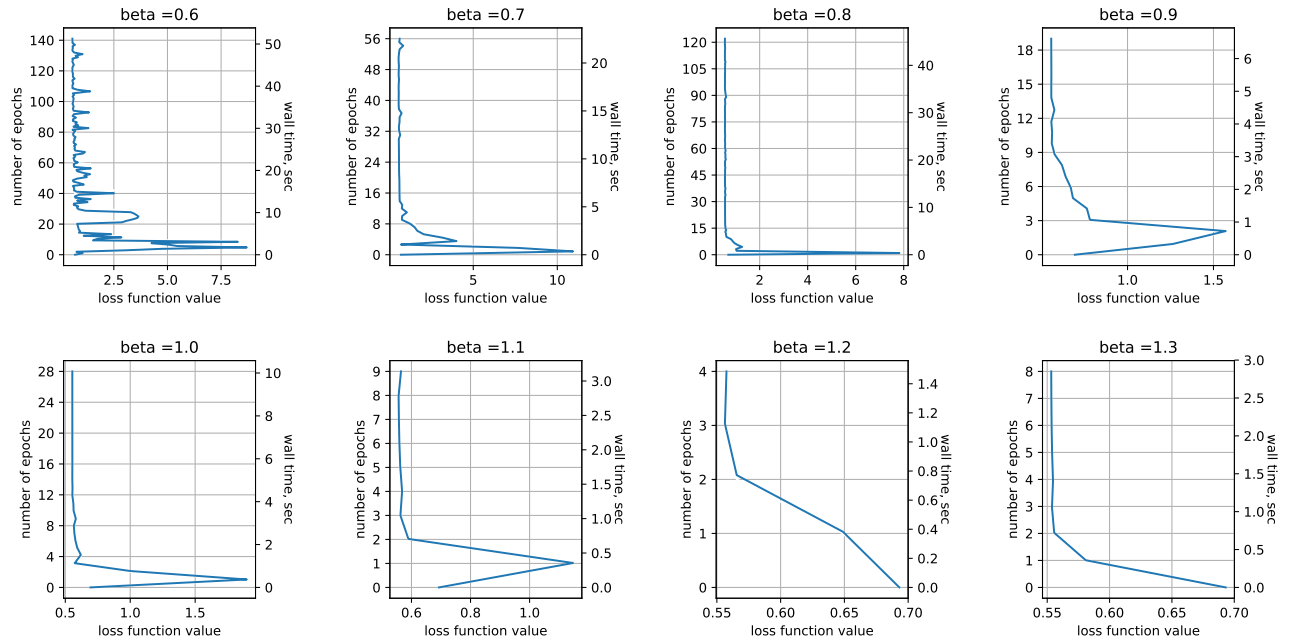
Таким образом, стохастический спуск с размером батча = 1 работает значительно медленнее, но не дает прироста точности. Увеличение размера батча позволяет снизить время работы и затраты вычислительной мощности без потери качества.

### Исследование зависимости от $step\_beta$

Эксперимент проводился при значении параметра  $batch\_size = 32$



dependence of loss function value on time and number of epochs

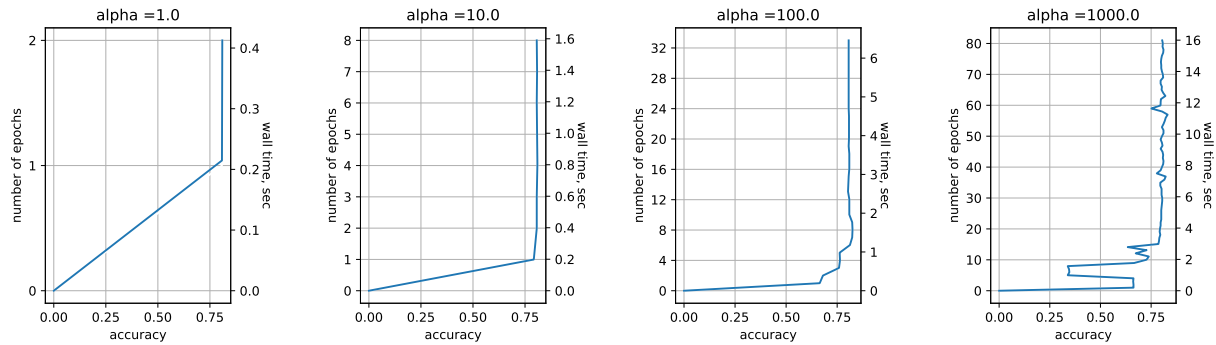


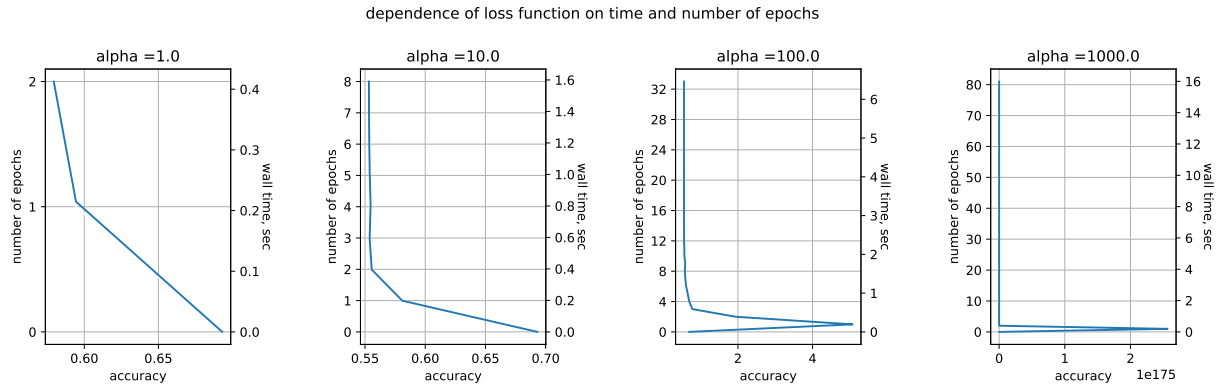
Общие тенденции зависимостей аналогичны полученным для полного градиентного спуска.

### Исследование зависимости от $step\_alpha$

Были зафиксированы значения всех параметров, кроме исследуемого. Значение параметра  $batch\_size = 32$ .

dependence of accuracy on time and number of epochs





Зависимость поведения метода от параметра также аналогична поведению полного градиентного спуска.

### Зависимость от начального приближения

Аналогично исследованию полного градиентного спуска, было проведено несколько экспериментов с различным начальным приближением при фиксированных значениях остальных параметров. Разница в поведении метода присутствует только на нескольких начальных итерациях, на итоговое качество и время работы начальное приближение не влияет.

### Выводы о поведении двух исследованных методов

В результате экспериментов было выявлено, что метод стохастического градиентного спуска позволяет значительно сократить время работы без потери качества предсказания.

### Эксперимент №3. Многоклассовая классификация

Для проведения эксперимента были зафиксированы все параметры ( $step\_alpha = 10$ ,  $step\_beta = 1.3$ ). Датасет предобработан аналогично предыдущим экспериментам: из текстов исключены все символы, кроме букв и цифр, далее каждый текст преобразован в разреженную матрицу при помощи функции *CountVectorizer* из библиотеки *sklearn.feature\_extraction.text* (параметр *min\_df* взят по умолчанию, так что размер пространства признаков составляет 131954). Для тестирования методов многоклассовой классификации были взяты все объекты (в том числе и те, для которых значение целевой переменной равно 'neutral').

Зависимость итоговой точности от использованного метода

метод	точность предсказания	время работы, с
all vs all	0.639	99.85
one vs all	0.642	122.80
multinomial	0.660	588.98

Несмотря на большое время работы, итоговая точность на методе *multinomial* значительно превышает остальные, поэтому будем использовать этот метод в дальнейших экспериментах.

## Эксперимент №4. Предобработка данных

В данном эксперименте была произведена следующая стадия предобработки данных: помимо уже произведенных ранее преобразований была произведена лемматизация (с использованием функций библиотеки *py morphology2*), также из текста были удалены стоп-слова (слова из списка *nlTK*).

Размерность пространства сократилась с 131954 до 56532. Достигнутая на методе *multinomial* точность составляет 0.671, время работы равно 309.85 секунд. Таким образом, при увеличении качества время работы снизилось почти в 2 раза.

## Эксперимент №5. Исследование методов предобработки и представления данных

### Сравнение работы методов с использованием Bag of Words и Tf-Idf

Тк параметры, подобранные для модели BagOfWords, не подходят для моделTf-Idf, то возникла необходимость заново подбирать параметры. На лучшем из полученных наборов была достигнута точность 0.627 при времени работы 530.50 секунд на предобработанной выборке. Этот результат хуже, чем на модели BagOfWords, поэтому в дальнейших экспериментах будем использовать последнюю.

### Исследование зависимости от параметра *min\_df*

Сначала исследуем зависимость от значения параметра *max\_df*.

Зависимость итоговой точности и времени работы от значения параметра

<i>min_df</i>	точность предсказания	время работы, с
5	0.673	95.83
10	0.674	74.77
15	0.673	59.82

При значении параметра, равном 10, полученная точность выше точности с параметром по умолчанию, а время работы значительно меньше. Возможно, из текстов удаляются слова, которые являются типичными для подобного жанра и не могут встречаться вне зависимости от тональности. После их удаления более значимые слова начинают играть более существенную роль.

## Исследование зависимости от параметра $max\_df$

Результаты экспериментов представлены в таблице.

Зависимость итоговой точности и времени работы от значения параметра

$max\_df$	точность предсказания	время работы, с
10	0.558	7.76
50	0.648	10.21
100	0.632	14.83

При дальнейшем увеличении значения параметра точность продолжает падать. Возможно, редко встречающиеся слова являются специфичными для каких-то определенных тоналностей текста, поэтому их удаление приводит к снижению качества распознавания.

## Эксперимент №6. Исследование ошибок метода с оптимальным набором параметров.

По результатам экспериментов, лучшей моделью является *GDClassifier* с параметрами  $step\_alpha = 10$ ,  $step\_beta = 1.3$ .

Данные приведены к нижнему регистру, из них исключены все символы кроме букв и цифр. Затем к ним применены процедуры лемматизации и удаления стоп-слов, данные преобразованы к представлению BagOfWords с параметром  $min\_df = 10$ . Полученная точность на тестовой выборке составляет 0.674, время работы алгоритма равно 89.55 секунд.

Построим матрицу ошибок для трех классов.

$$\begin{bmatrix} 85 & 106 & 24 \\ 22 & 475 & 108 \\ 1 & 143 & 276 \end{bmatrix}$$

Как и ожидалось, больше всего объектов было отнесено к нейтральному классу, как верно, так и ошибочно.

Приведем примеры текстов, категория которых была предсказана неправильно.

### Пример 1

*Ситуация на железнодорожном вокзале Астаны нормализовалась, все поезда идут согласно своему графику, сообщает Казинформ. В преддверии празднования Нового года в помещении ж/д вокзала не осталось и следа от пассажирского коллапса, который образовался пару дней назад. Пассажиры приобретают билеты, ожидают очереди у касс и прибытия своих поездов. Напомним, 27-28 декабря опаздывали 12 пассажирских поездов в Акмолинской области. Все они застряли из-за схода 27 декабря на станции Бабатай Карагандинского отделения дороги электровоза и 12 вагонов грузового поезда. По данным КТЖ, в тот вечер на вокзале Астаны скопление людей достигло 800 человек. Рано утром 28 декабря все желающие уехать пассажиры были отправлены, еще 250-ти были возвращены деньги за билеты. «Это был один из самых тяжелых дней. И справочным бюро досталось, и кассирам. Пассажиры не понимают, что вокзал дает готовую инфор-*

мацию в эфир, мы не перевозчики, не отвечаем за путь, сход вагонов. Даем информацию, которую нам передает диспетчер. Даже когда нет билетов, а касса - это совершенно другая организация, приходят ко мне и ругаются со мной. Очень устали, выслушали всякое в свой адрес, записывали и бейджик мой, и фамилию, куда меня только не посылали пассажиры. Слезы на глаза наворачивались (...) Конечно, пассажиров можно было понять, они настроились уехать, а поездов нет, но и они нас даже не хотели слушать», - рассказала агентства дежурившая в ту ночь работница вокзала.

Этот текст имеет категорию "positive но был отнесен к категории "negative". Это можно объяснить высоким содержанием в нем слов, которые типичны для текстов с негативным содержанием, например "коллапс "застряли "тяжелых "слезы".

## Пример 2

По данным союза предприятий автомобильной отрасли Казахстана «КазАвтоПром» за 9 месяцев 2016 года казахстанцы приобрели 30,8 тыс. новых легковых автомобилей на сумму \$569 млн, что на 59% ниже аналогичного показателя минувшего года. По результатам января-сентября официальными дилерами было реализовано 30 759 легковых и легких коммерческих автомобилей, что на 59% ниже аналогичного показателя минувшего года (74 911 ед.). Продажи в стоимостном выражении в отчетный период составили \$569 млн., уступив 58,7% значению соответствующего периода 2015 года. В сентябре объем продаж сократился до отметки 3 207 автомобилей против 8129 ед. в том же месяце 2015 г. (-60%). Отрицательная динамика сбыта на автомобильном рынке республики сохраняется начиная со II квартала 2014 года. Наряду с другими сегментами казахстанского ретейла авторынок продолжает испытывать негативные последствия девальвации национальной валюты и снижения доступности кредитных ресурсов. Казахстанский автопарк «помолодел»... По информации аналитической службы *Energyprom.kz* на 1 сентября доля машин возрастом менее 3 лет составляла 16,2%. Для сравнения: к осени 2015 нов... 23 568 автомобилей (76%), реализованных по итогам 9 месяцев, пришлось на импорт, 7154 покупателя (24%) остановили выбор на продукции казахстанских автозаводов. Рейтинг наиболее востребованных казахстанцами брендов по итогам отчетного периода возглавляет Lada: за 9 месяцев 2016 г. на территории республики продано 6 086 автомобилей российской марки. Сбыт дилеров Toyota составил 5 728 единиц. Тройку лидеров замыкает Renault с показателем 2 925 реализованных автомобилей. В десятку наиболее успешных брендов вошли также Hyundai (2 659 ед.), KIA (2 325 ед.), Volkswagen (1583 ед.), GAZ (1 467 ед.), UAZ (1 196 ед.), Nissan (1 103 ед.) и Chevrolet (644 ед.). Лидерство в рейтинге моделей сохраняет Lada Priora. Российский седан предпочли 2 237 покупателей. Пятерку бестселлеров составили также Toyota Camry (2 232 ед.), Lada 44 (1 534 ед.), Renault Duster (1394 ед.) и Hyundai Accent (1261 ед.). Среди моделей, вошедших в TOP-10 потребительских предпочтений, также Volkswagen Polo (1236 ед.), Lada Granta (1164 ед.), KIA Rio (1015 ед.), Renault Sandero (1012 ед.) и Toyota Land Cruiser Prado (916 ед.). Статус крупнейшей дилерской группы на рынке Казахстана удерживает холдинг «БПЭК АВТО – АЗИЯ АВТО». По итогам отчетного

*период 33,3% всех покупателей новых легковых автомобилей (включая LCV) остановили выбор на предложениях этой компании. 20,6% рынка пришлось на сбыт «Toyota Motor Kazakhstan». Доля КМК «Astana Motors» составила 9,7%. В десятку наиболее заметных участников казахстанского авторемейла вошли также компании «Вираж» (доля – 7,4%), «Mercur Auto» (5,3%), «Aster Auto» (3,8%), «Урал-Кров Авто» (2,7%), «Allur Auto» (2,6%), «Автомир» (2%), «Astek-Auto» (1,5%). Наибольшую активность в приобретении легковых автомобилей за этот период проявили жители Алматы (9281 реализованных авто), Астаны (5333 ед.), Атырау (2234 ед.), Усть-Каменогорска (1656 ед.), Караганды (1623 ед.), Шымкента (1573 ед.), Актау (1370 ед.), Костаная (1368 ед.), Уральска (1320 ед.) и Актобе (1196 ед.). Источник: Курсивъ Поделится в WhatsApp*

Этот текст имеет категорию "negative но был отнесен к положительной категории. Возможно, дело в том, что несмотря на отрицательный общий смысл, в нем присутствует много слов, которые могут выражать положительные эмоции: "лидерство "успешных "заметных".

Таким образом, во многих неправильно распознанных текстах несмотря на их общую эмоциональную окраску присутствуют слова, характерные для другой категории текстов.