Определение эмоциональной окраски постов пользователей в социальных сетях.

Под постом подразумевается нечто, что состоит из картинки и текста, в объединении обладающими смысловой нагрузкой. То есть видео и аудио данные рассматриваться не будут.

Поскольку предполагается работать с текстовыми данными и изображениями, то предлагается использовать две модели: одна - отвечает за текст, вторая - за изображения.

Также, для простоты, ограничимся английским языком.

Данные.

Поскольку предполагается решать задачу классификации, причём двумя различными способами, то придётся использовать два датасета: в первом тексты и метки им соответствующие (эмоциональные окраски), аналогисно с картинками. Причем метки классов должны быть одинаковыми в обоих наборах.

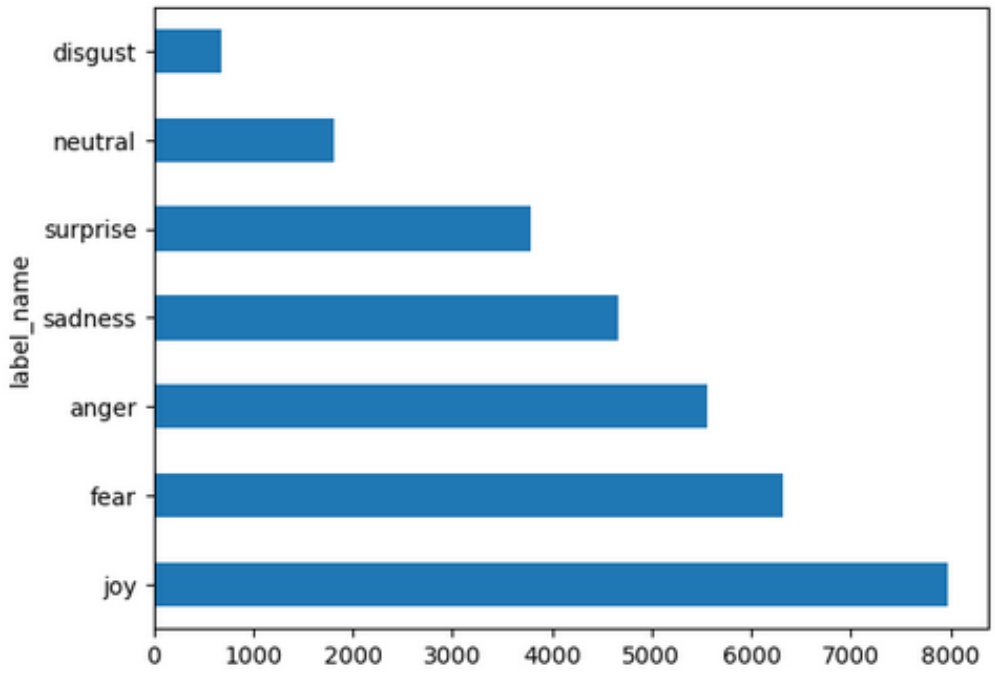

Figure 1: Количество примеров для каждого лэйбла

Топ 7 слов в каждом из классов:

Как этот топ 7 выглядит в терминах облаков слов.

В первую очередь подбирались данные с картинками, потому что такие наборы искать было труднее, чем текстовые. Поэтому в качестве датасета с картинками был выбран *https://huggingface.co/d*

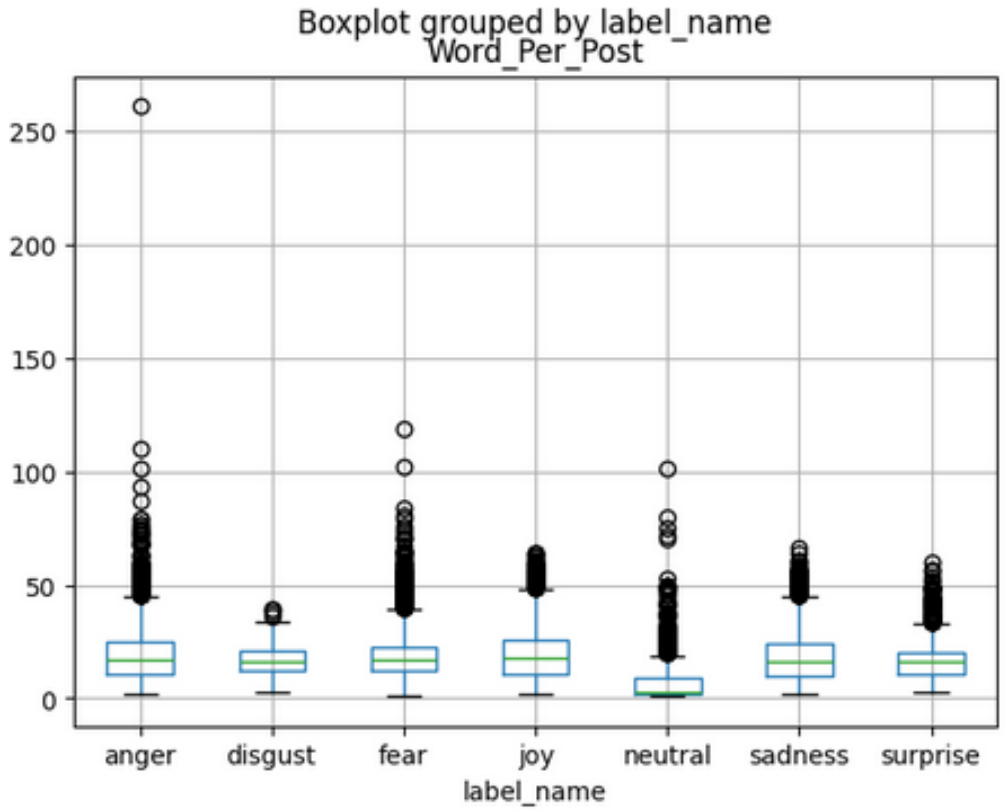

Figure 2: Длина текстов в каждом из лэйблов

содержащий в себе 9400 изображений лиц людей, разбитых по 8-ми категориям. В дальнейшем класс "content" пришлось убрать, т.к. соответствующих текстовых данных подобрать не удалось.

В качестве текстового датасета были выбраны два набора, метки которых объединении содержали все классы, присущие картинкам (лишние, разумеется пришлось устранить). Дянные распределены равномерно по лэйблам (судя по Hugginface)

По итогу, получилось два датасета с семью классами.

Модели.

Как говорилось выше - будет задействовано две модели. Для анализа текста была выбрана модель ($7 * 10^7$ параметров) $https://huggingface.co/distilbert/distilbert-base-uncased$

Эта модель в первую очередь предназначена для тонкой настройки на задачах, которые используют целое предложение (потенциально замаскированное) для принятия решений, таких как классификация последовательностей, классификация токенов или ответ на вопросы.

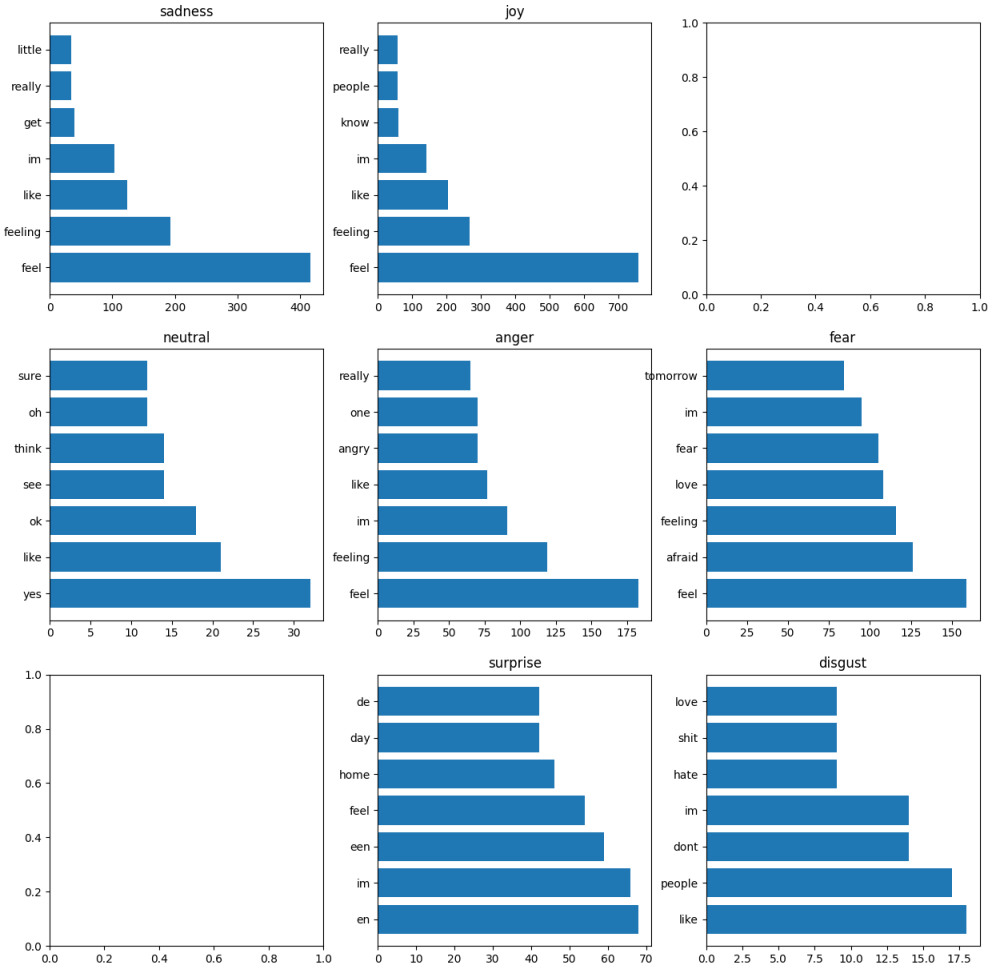

Figure 3: Топ 7 слов в каждом из классов

Для анализа изображений - задействована моель сопоставимых масштабов. $https://huggingface.co/google/vit-base-patch16-224-in21k$ Vision Transformer (ViT) — это модель кодировщик-трансформер (похожая на BERT), предварительно обученная на большой коллекции изображений контролируемым образом, а именно ImageNet-21k, с разрешением 224x224 пикселей.

Изображения представляются модели как последовательность фрагментов фиксированного размера (разрешение 16x16). Также добавляется токен [CLS] в начало последовательности, чтобы использовать ее для задач классификации. Также добавляются абсолютные позиционные эмбединги перед подачей последовательности на слои модели.

Обучение.

Distilbert обучалась в 6 эпох на датасете из 30803 примеров,

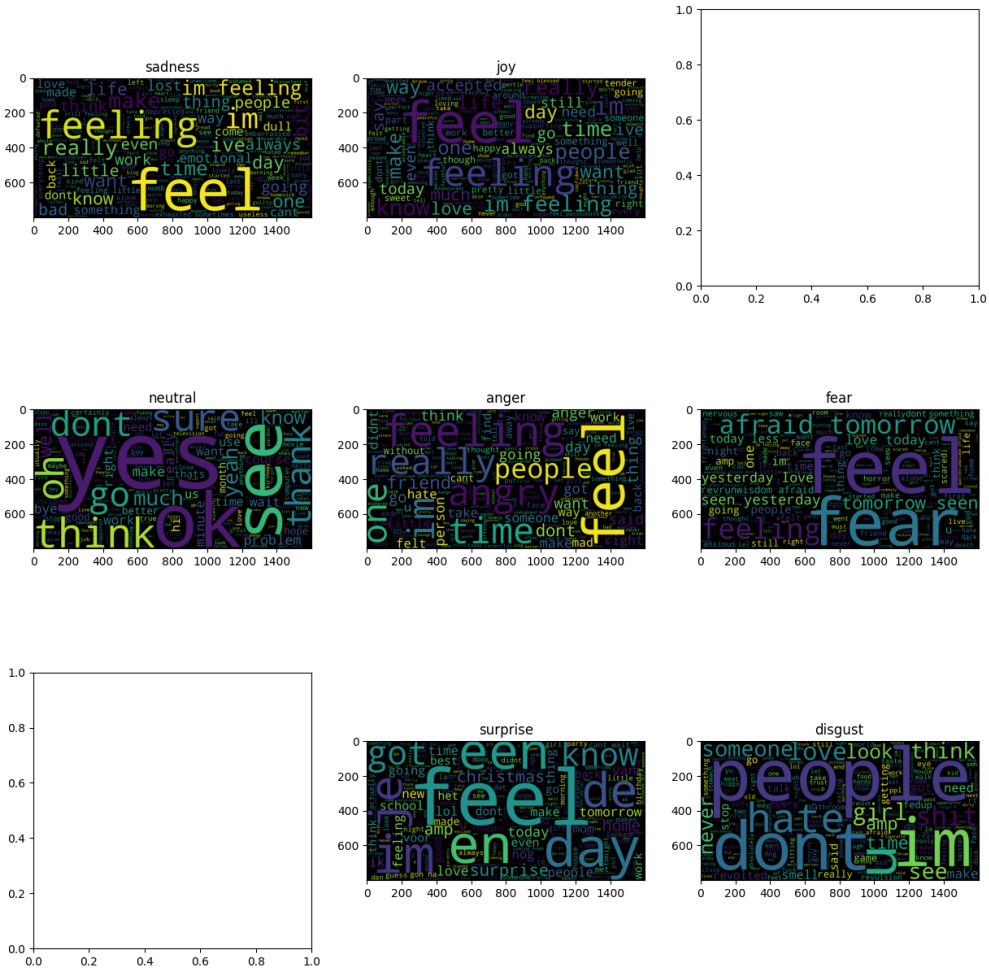

Figure 4: топ 7 выглядит в терминах облаков слов

при помощи стандартных методов Trainer.

ViT обучалась в 6 эпох на датасете из 7254 примеров, при помощи стандартных методов Trainer.

Эксперименты.

Для текстовой модели посмотрим, с какой точностью ей удаётся определять каждый из классов. Поскольку датасет разбалансир (класс "disgust" в диффиците), то следует ожидать. что самый малочисленный класс будет определяться хуже остальных. То есть на инференсе представитель самого бедного класса может часто падать в другой, но в малочисленный класс редко будут попадать по ошибке. См. рис.

Посмотрим, какие слова чаще всего попадаются в предложениях, отнесённых к определённому классу.

То же самое, но на облаках слов.

Epoch	Training Loss	Validation Loss	Accuracy	F1
1	No log	0.541501	0.805689	0.788816
2	0.711000	0.391791	0.864689	0.861598
3	0.322300	0.264887	0.916619	0.916233
4	0.236000	0.185536	0.946445	0.946267
5	0.174200	0.141388	0.960585	0.960343
6	0.128900	0.125051	0.965461	0.965403

Figure 5: Обучение Distilbert

Step	Training Loss	Validation Loss	Accuracy
100	1.186600	1.256971	0.533713
200	0.751900	1.039499	0.624881
300	0.714200	0.946472	0.654321
400	0.373200	0.920718	0.685660
500	0.254000	0.942241	0.685660

Figure 6: Обучение ViT

Видно, что почти везде встречаются слова "feel", "feeling", "like", "de", "en". Попробуем выкинуть их, чтобы большую роль играли уникальные слова, что, возможно, приведет к лучшему качеству классификации. Проверим гипотезу, рассмотрев аналогичные картинки. Из таблиц видно, что столь агрессивный подход привёл к заметной потере качества. Что говорит о важности не конкретных слов, а их комбинаций.

Рассмотрим качество классификации в зависимости от длинны исходного предложения рис 11.

Для ViT посмотреть какие элементы картинки оказывают наибольшее влияние на классификацию - более сложная процедура, поэтому рассмотрим с какой точностью ей удаётся определять каждый из классов. Исходя из более скромных показателей ViT

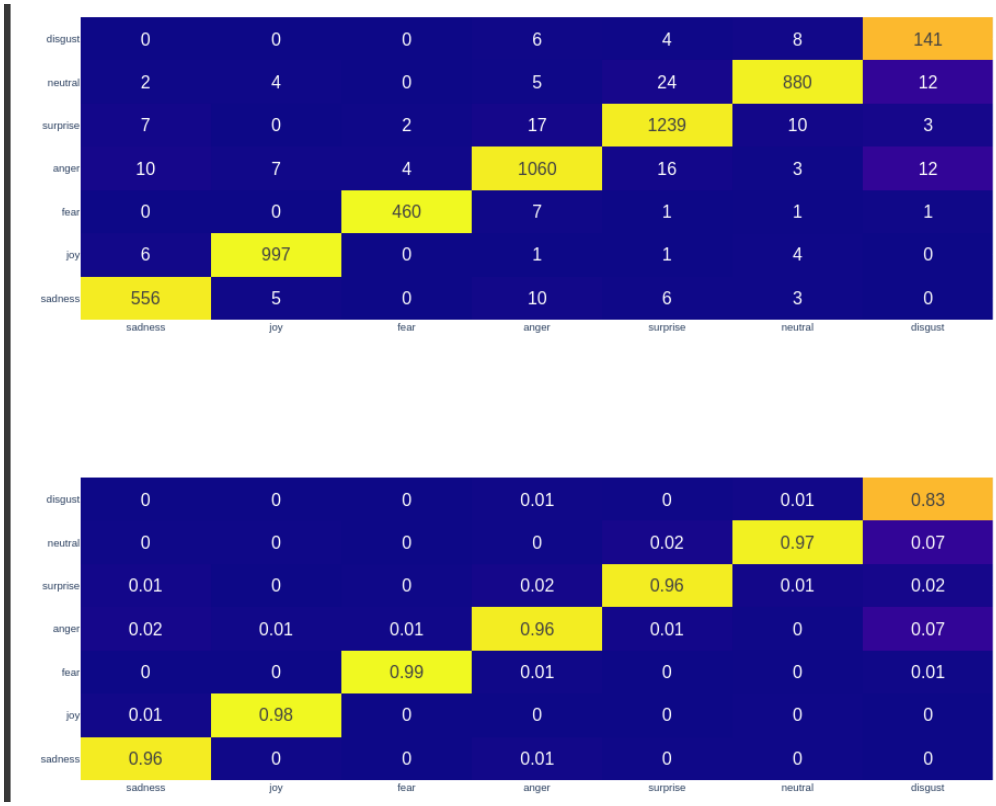

Figure 7: зависимость предсказаний модели от истиного класса. Сверху - колличественные оценки на тестовом наборе, снизу - частотные

по сравнению с Distilbert, не трудно догадаться, что большее колличество тестовых объектов попадёт вне побочной диагонали. Здесь скорее больший интерес представляет вопрос, а какие классы труднее всего отличать друг от друга. рассмотрим рисунок. 12

Предсказание в комплексе.

Изначально планировалось проводить классификацию с опорой на две модели. Однако, анализ текстов оказался более простой задачей. Поэтому можно ожидать, что во время инференса на объекте, состоящем и из картинки, и из текста, стоит серьёзнее относиться к предсказаниям Distilbert.

Web-интерфейс. Наверное, один из самых разумных способов моделировать пост - отправлять телеграм боту картинку с подписью. Однако реализовать его я не успел.

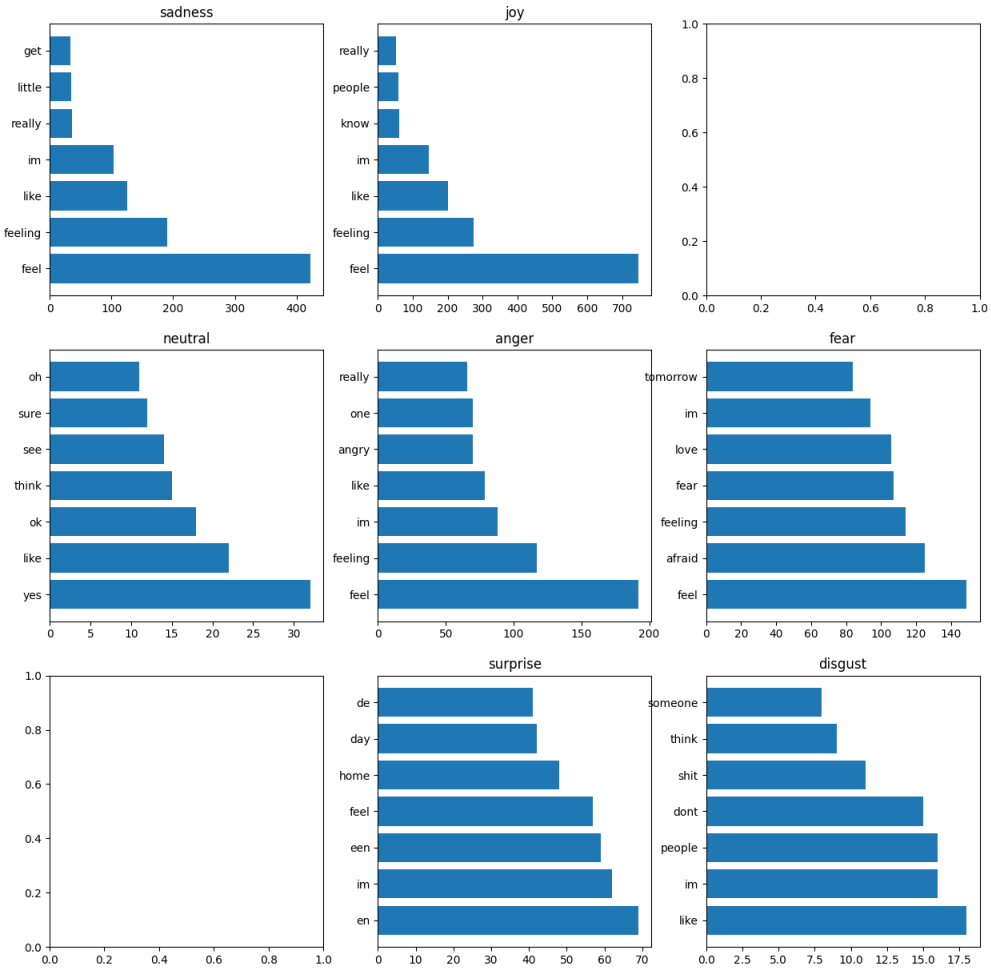

Figure 8: Топ 7 слов в каждом из классов с точки зрения модели

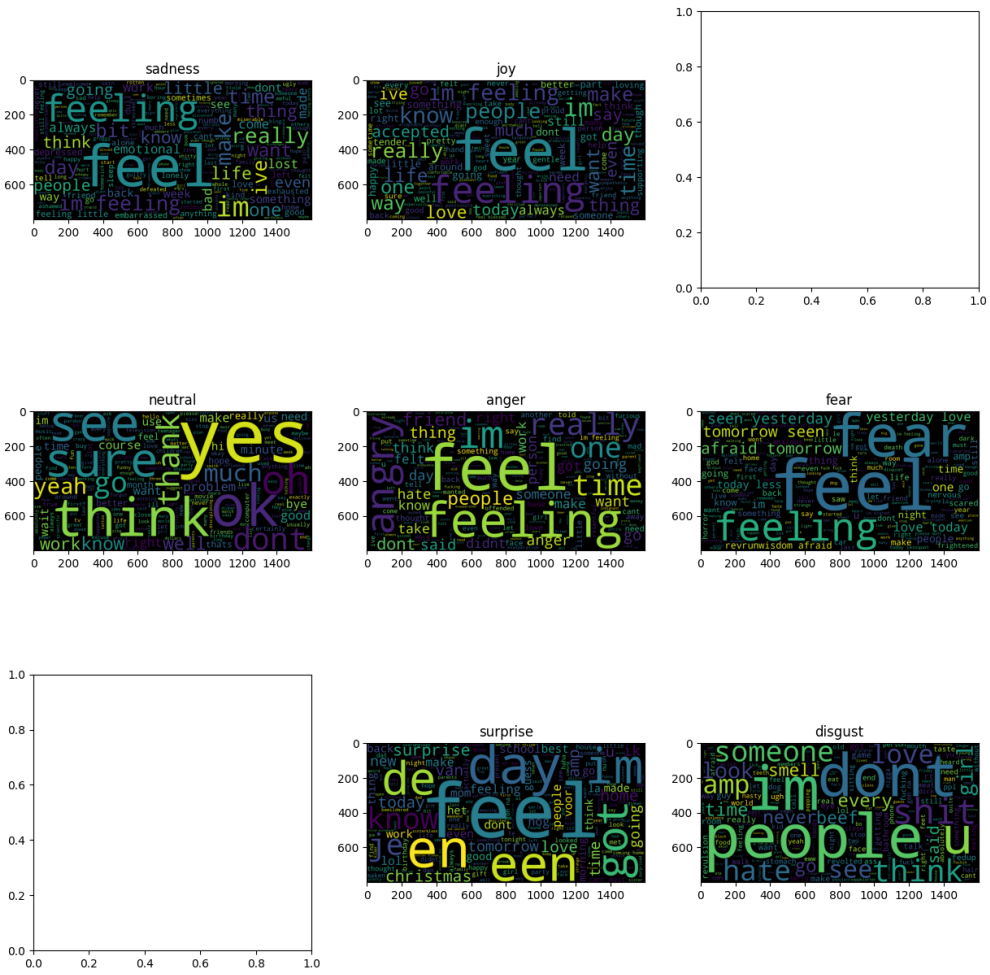

Figure 9: Облака слов для оп 7 слов в каждом из классов с точки зрения модели

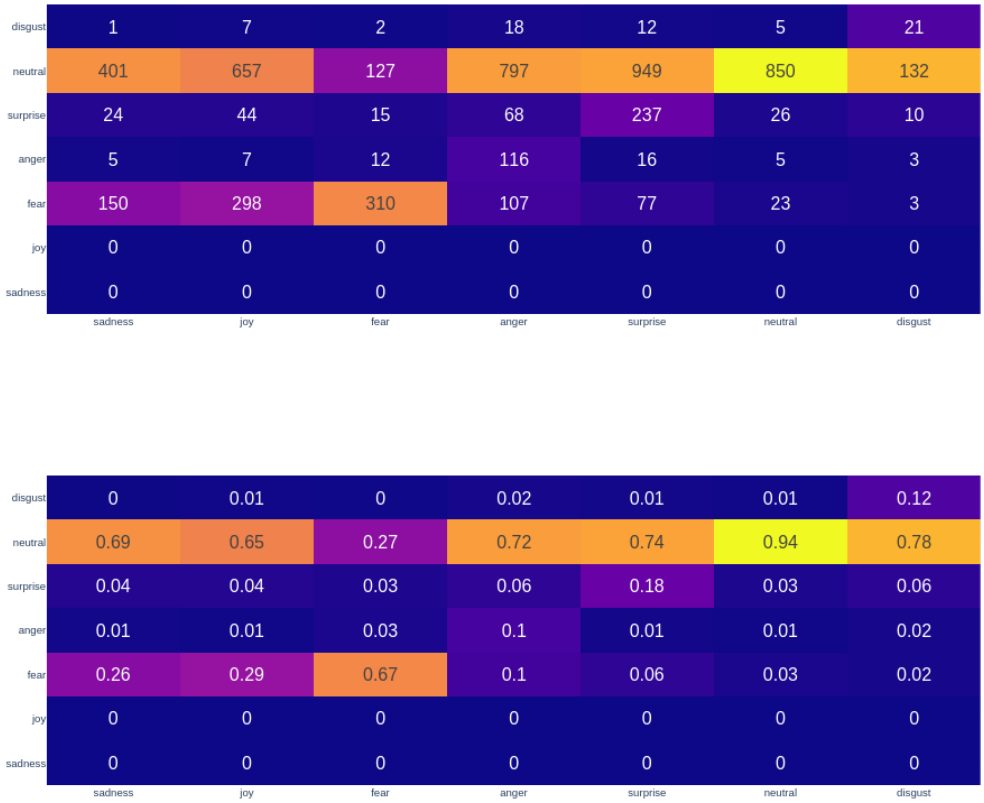

Figure 10: зависимость предсказаний модели после выкидывания некоторых слов от истиного класса. Сверху - колличественные оценки на тестовом наборе, снизу - частотные

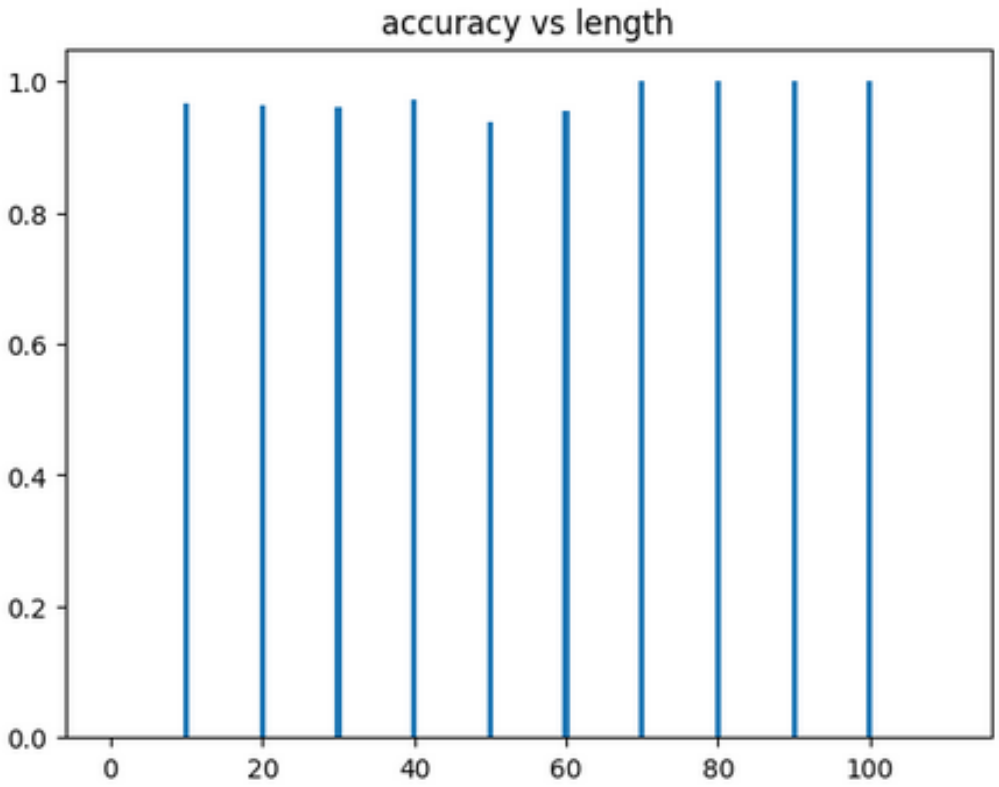

Figure 11: качество от длины промпта

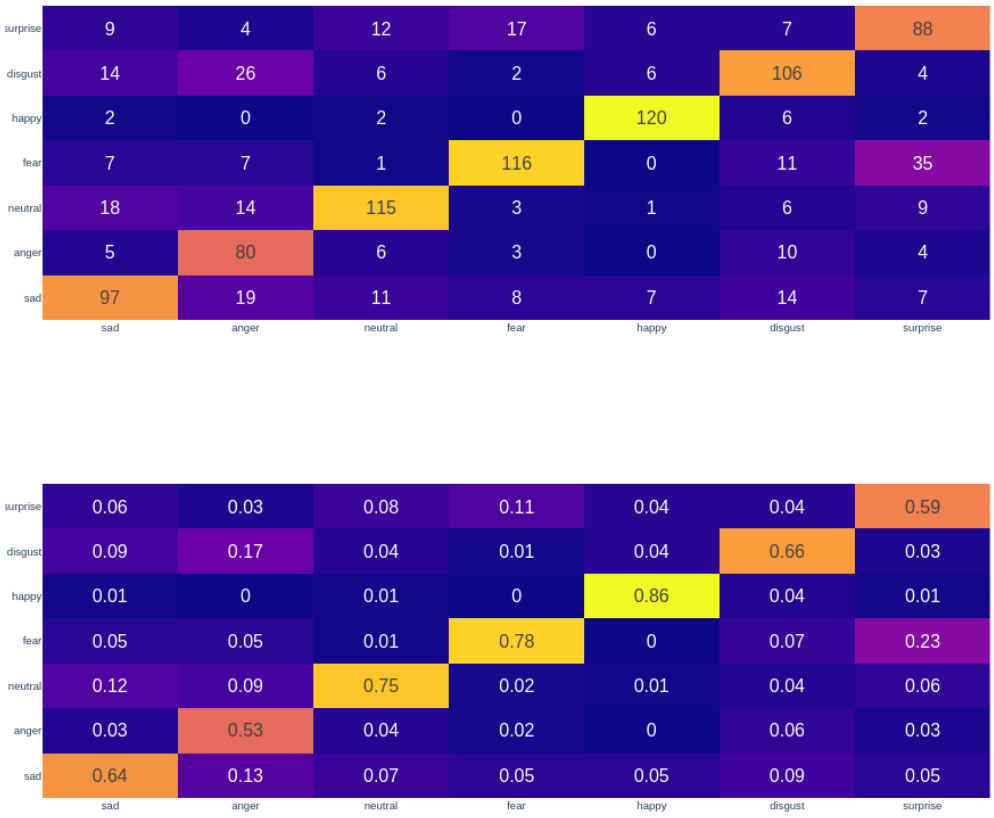

Figure 12: зависимость предсказаний модели по картинкам от истинного класса