



Как простой бейзлайн занял 1 место* в соревновании **Авито**

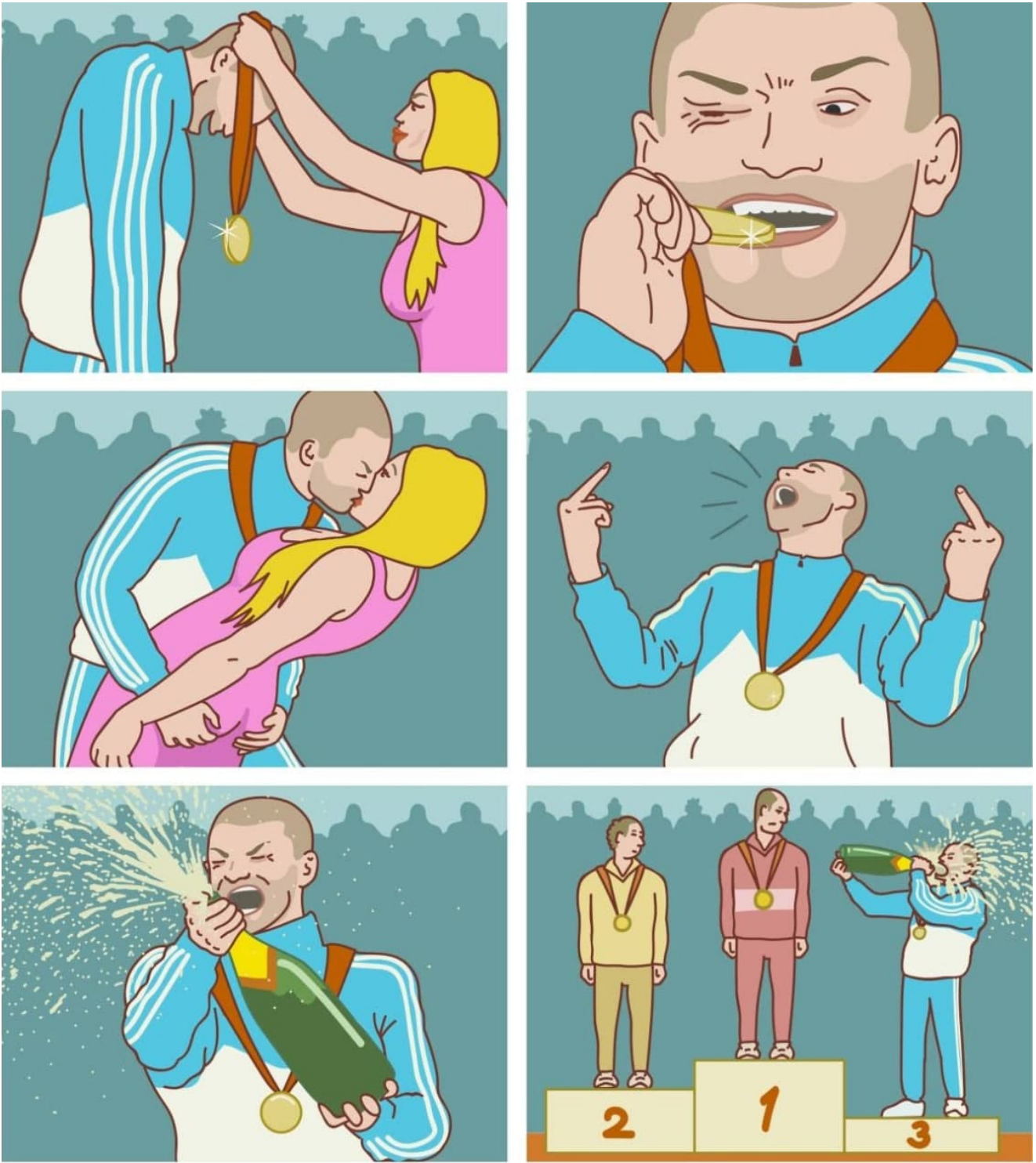
Александр Сенин

Студент 6 курса, ММП ВМК МГУ

Data Scientist, Альфа-Банк

Заголовок со звездочкой

#	△	Team	Members	Score	Entries	Last	Solution
1	▲ 1	Leon Zhebrik		0.85232	15	3mo	
2	▼ 1	Senin Alexander		0.85024	14	3mo	



Проблемы CV-бейзлайна

1

Шумный таргет

Флаг is_blocked
отвечает всему
объявлению



**Есть ли контакт
<- на этом фото?**

Спойлер: есть

2

Мало данных

Всего 100к
картинок и 11к
объявлений

3

Слабая архитектура

Модель 2015 года,
сжимающая картинку
до 224x224



Чиним CV-бейзлайн

1

Шумный таргет

Флаг `is_blocked`
отвечает всему
объявлению

Решение (?)

**Доразмечаем
толокой**

2

Мало данных

Всего 100к
картинок и 11к
объявлений

Решение (?)

**Генерим контакты
самостоятельно**

3

Слабая архитектура

Модель 2015 года,
сжимающая картинку
до 224x224

Решение (?)

**resnet18 ->
50/101/152**

Плохо чиним CV-бейзлайн

1

Шумный таргет

Флаг is_blocked
отвечает всему
объявлению

Решение (?)

Доразмечаем
толокой

Проблема

Мало денег,
толокеры
ошибаются,
таргет слишком
сложный

2

Мало данных

Всего 100к
картинок и 11к
объявлений

Решение (?)

Генерим контакты
самостоятельно

Проблема

Нет чистой
разметки,
существуют FP
и FN картинки

3

Слабая архитектура

Модель 2015 года,
сжимающая картинку
до 224x224

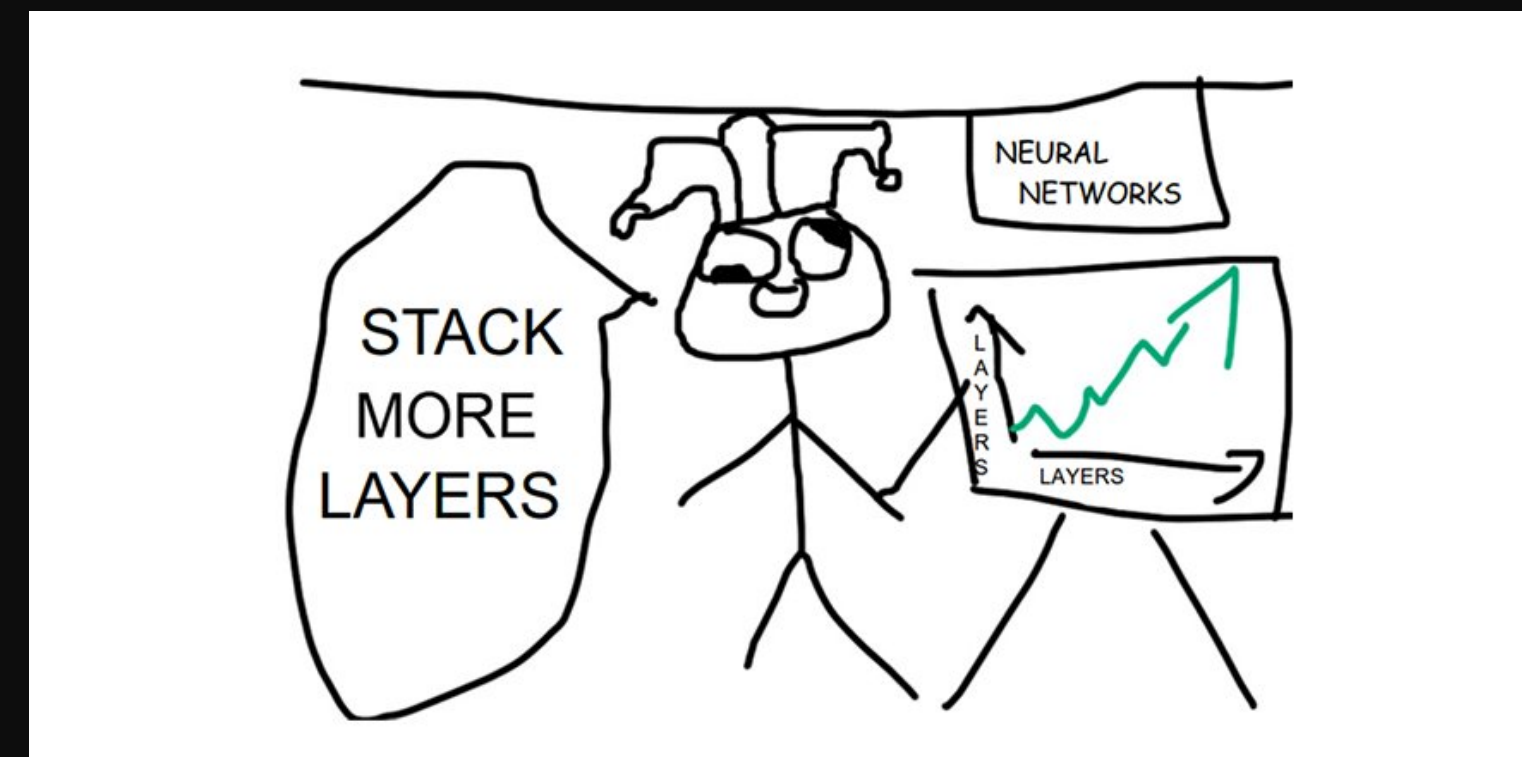
Решение (?)

resnet18 -> 50/101/152

Проблема

Все еще учимся
на шум

Выше 0.7 CV не потянуло



Доразмечал
собственными
глазами

Генерил
контакты и
вотермарки

Менял шедулер

Стакал больше
слоев

Ансамблировал

Фризил весь
backbone,
фризил часть
слоев

Учил много
эпох,
переобучал,
недообучал

Меняем парадигму на NLP

1

Вытаскиваем
текст с картинок

EasyOCR

2

Учим NLP-
модель

Бустинг над
ручными фичами

```
1 def generate_contact_features(df_):
2     df = df_.copy()
3     keywords = [
4         "http", "ht", "tp", "ww", "www", "ru",
5         "com", "co", "org", "ua", "py", "su",
6         "ins", "inst", "insta", "gram", "gra",
7         "te", "tele", "@", "dot", "ya", "yandex",
8         "dr", "drom", "om", "book", "ing", "booking",
9         "ci", "an", "cian", "др", "дром", "дом", "ом",
10        "авто", "ци", "ан", "циан", "ави", "то", "авито",
11        "mail", "google", "gle", "youla", "юла",
12        'qr', 'тел', "phone", "номер", "ном", "звон", "вацап",
13        'ватсап', 'вотсап', 'телеграм', 'теле', 'дрон', 'дрон',
14        'сайт', "site", "наш", "факс", "vk", "vk", "ok", "ok",
15        "auto", "mobile", "+7", "8", "8 9", "8 (", "8(", "89",
16        "8 7", "87", "8-9", "8-7", "8-8", "://", "tps", "https",
17        ".r", ".c", ".u", ".o", "id", "dro", "дро", "недв", "движ",
18        "car", "opt", "tics", "optic", "optics", "caroptic", "caroptics",
19        "инст", "cli", "ck", "click", "dom", "domclick", "realty", "rea", "lty",
20        "e-m", "em", "email", "e-mail"
21    ]
22    keyword_to_count = [
23        *str(i) for i in range(10)],
24        *["-", ".", "/", "w", "r", "u", "c", "o", "m", "@", "t",
25          "(", ")", "\\", "|", "_", "!", "~", "+", ":"],
26        *string.ascii_lowercase,
27        *list(set("АаБбВвГгДдЕеЁёЖжЗзИийЙккЛлМмНнОоПпСсТтУуФфХхЦцЧчШшЩщЪъЫыЬьЭэЮюЯя".lower()))
28    ]
29
30
31
32    df["text"] = df["text"].apply(lambda x: [el for el in x if not ("vito" in str(el))])
33    df["str_text"] = df["text"].apply(lambda el: " ".join([str(x) for x in el]))
34    df["str_text"] = df["str_text"].str.lower()
35
36    df["text_len"] = df["str_text"].str.len()
37
38    for keyword in keywords:
39        df[keyword + "_in_text"] = df["str_text"].apply(lambda x: x.count(keyword)).astype(int)
40    for keyword in keyword_to_count:
41        df[keyword + "_counter"] = df["str_text"].apply(lambda x: x.count(keyword)).astype(int)
42
43    return df
```



Результат

0.8 ROC AUC

С минимальным
тюнингом прямо
на `is_blocked`

0.85 ROC AUC

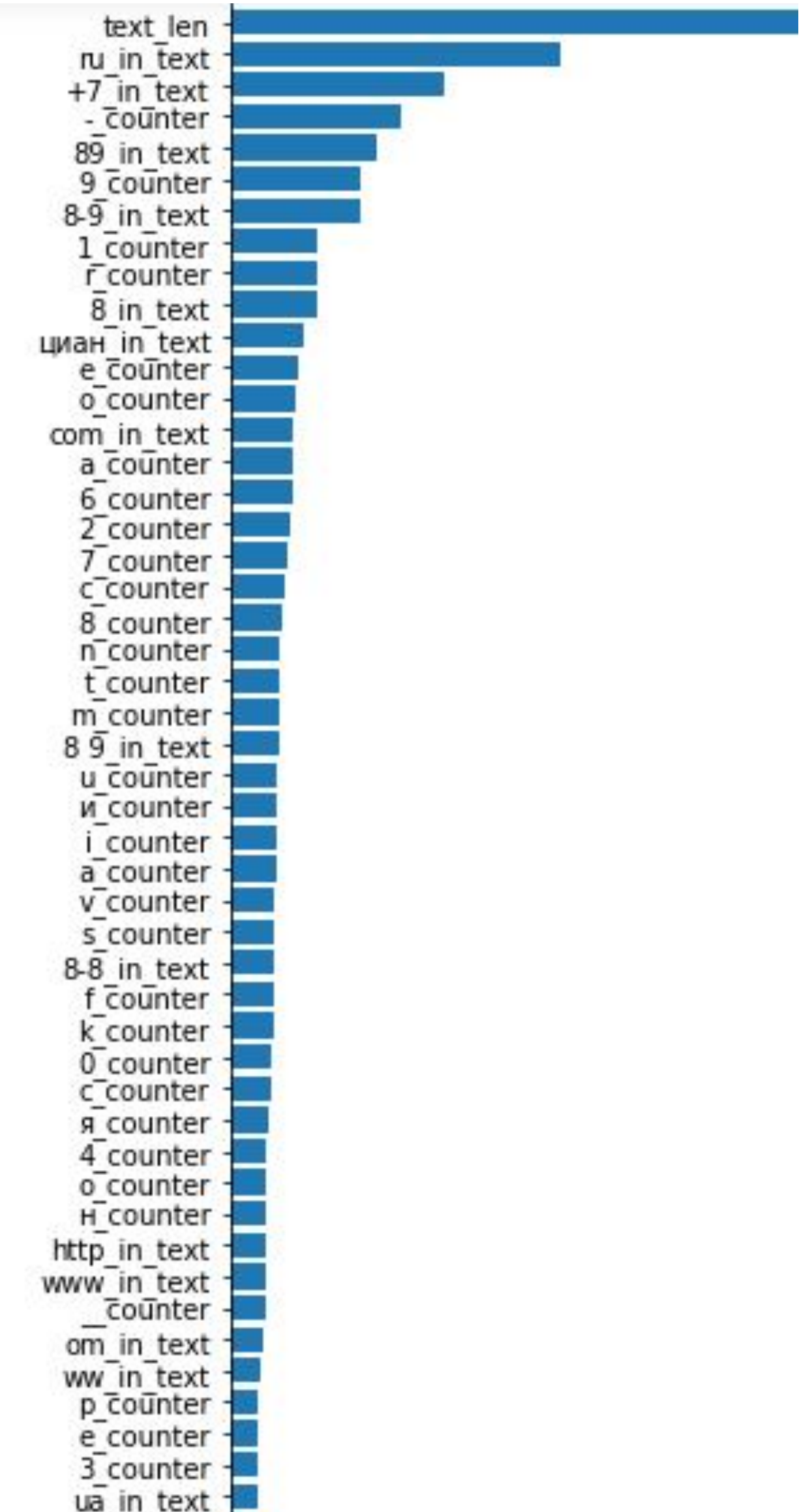
С доразметкой и
ансамблем из коробки
LightAutoML



NLP



CV



Что можно улучшить

1

Заансамблировать
ResNet и NLP-
модель

2

Взять токены не
из головы, а по
частоте: BPE и тп

3

Вместо бустинга
на счетчиках
обучить RNN