Яндекс Алгоритм 2018
ML track

...на протяжении 12 дней решение падает со второго места…

Предобработка

mой

вы і должны і

hет

Okей

ќет ! ћистер √убвиг !

"ќе говори мне "" силл "" , зерни !"

ќна не могла выгл€деть хуже .

"ќ , зоже , и штаны тоже ."

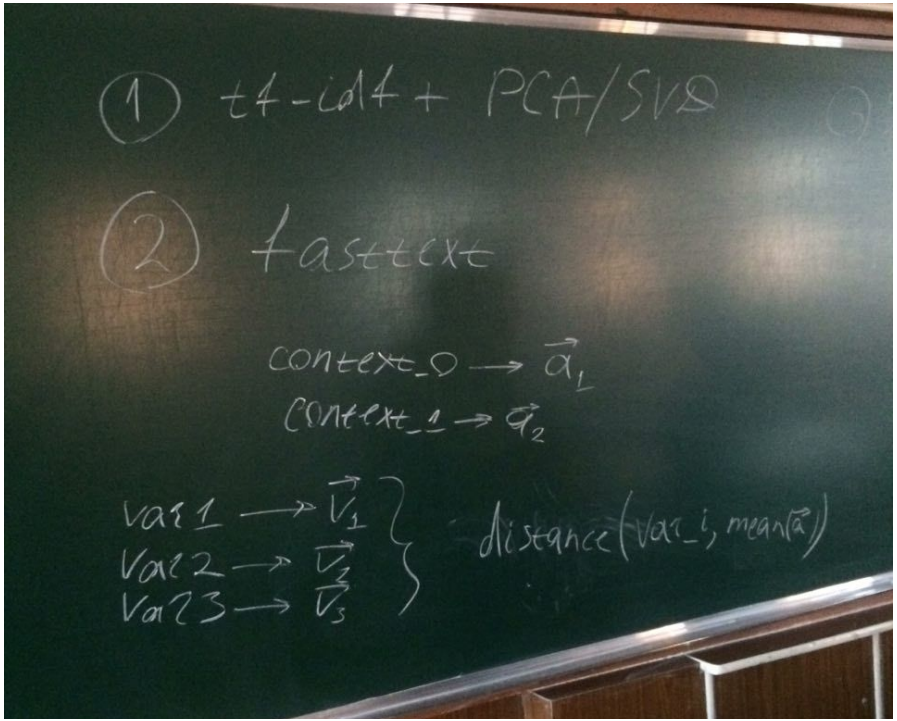
Вкрапления английских букв и странных символов (последствия OCR или перекодировок?)

Некоторые фразы диалогов целиком на английском (их немного, может, прогнать через API переводчика?)

Из чатике DMIA Соревнования

- Если заслать 0,1,2,3,4,5 в каждом id то скор будет 81302)

- Косинусное расстояние



Признаки морфологии

```
morph = pymorphy2.MorphAnalyzer()

def name_score(word):
    for p in morph.parse(word):
        if 'Name' in p.tag:
            return p.score
    return 0

def surn_score(word):
    for p in morph.parse(word):
        if 'Surn' in p.tag:
            return p.score
    return 0

all_data['pymorphy_word_is_known'] = all_data['Word'].apply(morph.word_is_known).astype('int8')
all_data['pymorphy_count_in_tag'] = all_data['Word'].apply(lambda x: len(morph.tag(x))).astype('int8')
all_data['pymorphy_score'] = all_data['Word'].apply(lambda x: morph.parse(x)[0].score)
all_data['pymorphy'] = all_data['Word'].apply(lambda x: morph.tag(x)[0])

all_data['pymorphy_animacy'] = all_data['pymorphy'].apply(lambda x: x.animacy)
all_data['pymorphy_POS'] = all_data['pymorphy'].apply(lambda x: x.POS)
all_data['pymorphy_case'] = all_data['pymorphy'].apply(lambda x: x.case)
all_data['pymorphy_number'] = all_data['pymorphy'].apply(lambda x: x.number)
all_data['pymorphy_gender'] = all_data['pymorphy'].apply(lambda x: x.gender)

all_data['pymorphy_name_score'] = all_data['Word'].apply(name_score)
all_data['pymorphy_surn_score'] = all_data['Word'].apply(surn_score)

columns_to_one_hot = ['pymorphy' , 'pymorphy_animacy', 'pymorphy_POS', 'pymorphy_case','pymorphy_number', 'pymorphy_gender']

for col in columns_to_one_hot:
    all_data[col] = LabelEncoder().fit_transform(list(all_data[col].fillna('nan')))

# https://github.com/applied-data-science/Data_Mining_in_Action_2018_Spring/blob/master/sport/hw0/solutions/001_Tushin_Kirill.ipynb
```

Фрагмент кода из решения другой задачи для демонстрации, что из морфологии можно вытянуть много признаков

Scorer

```
from sklearn.metrics import make_scorer

def DCG(label): return sum([float(label[i]/np.log2(i+2)) for i in range(len(label))])

def nDCG(label, best_label):
    label, best_label = DCG(label), DCG(best_label)
    if label != 0 and best_label != 0:
        return label/best_label
    else:
        return 0

scorer = make_scorer(nDCG)
```

https://github.com/applied-data-science/Data_Mining_in_Action_2018_Spring/blob/master/sport/hw2_yandex_algorithm_2018/benchmarks/Zuenko_Denis_82312_benchmark.ipynb

Из решений Kaggle Toxic Comments

- Bi-derectional GRU 2 layers
- Hierarchical attention NN
- Squizee and Excitation Networks (adapted for text)
- AC-BLSTM
- Перевод на язык и обратно
- SentencePiece
- Hyperopt

Запись трансляции тренировки ML 07.04.2018 | Kaggle Toxic, Whatever Hack, Boosters Raiffeisen
https://youtu.be/3mL9iP8g3fA