

## تمرین 2

8 آبان

مهدیه ساجدی پور

### 1. مقدمه

در این تمرین هدف آشنایی با مدل های زبانی و همچنین مفهوم **n-grams** می باشد. کار با یک PLM به نام BERT و بررسی embedding های تولید شده نیز از اهداف این تمرین است.

### 2. توضیحات

#### a. مدل های زبانی و n-grams

در پردازش زبان طبیعی حدس زدن کلمه بعدی کاربردهای بسیاری دارد؛ مثلاً دستیارهای صوتی احتیاج به چنین چیزی دارند تا بتوانند بین کلمات مشابه در صوت کلمه صحیح را برگزینند. به مدل هایی که برای دنباله کلمات احتمالاتی را تولید می کنند مدل های زبانی می گویند. ساده ترین مدل هایی که چنین کاری می کند مدل های **n-grams** هستند. برای حدس زدن یک کلمه به کلمه های قبلی احتیاج است. به طور مثال برای حدس زدن کلمه ی بعدی در جمله ی "تدریس پردازش زبان ... " به تمام کلمات موجود در لغت نامه احتمالی برای حضور به عنوان کلمه ی بعدی داده می شود. این احتمال برای کلمه "طبیعی" این گونه محاسبه می شود: تمام دفعاتی که "تدریس پردازش زبان طبیعی" در مجموعه داده مشاهده شده، تقسیم بر تمام دفعاتی که "تدریس پردازش زبان" در داده ها رویت شده است. **N** در **n-grams** نشان دهنده تعداد کلماتی است که به عنوان ترکیب برای محاسبه این احتمالات استفاده می شود. به طور مثال اگر  $n=2$  باشد، تعداد تکرار "زبان طبیعی" و "زبان" مد نظر است.

#### b. تحلیل احساسات با BERT فارسی

\*در این مرحله از نوت بوک [هوشواره](#) استفاده شده است.

این تسک بر روی دیتاست اسنپ فود انجام شده است.

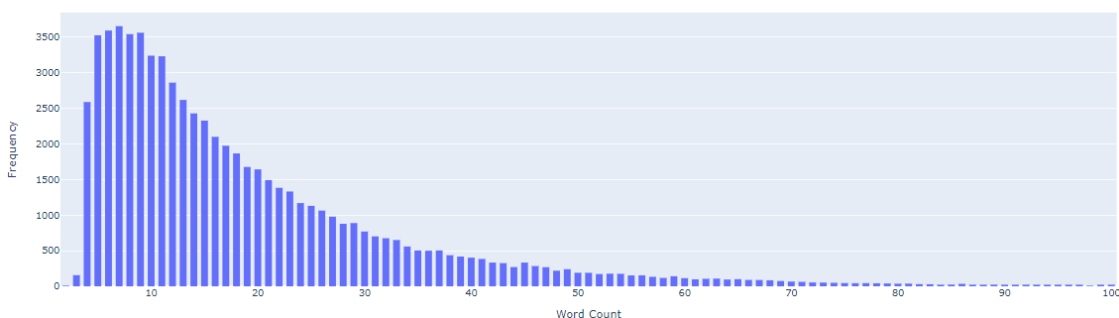
#### i. تجسم داده ها و پیش پردازش

Unnamed: 0		comment	label	label_id
0	NaN	واقعا حیف وقت که بنویسم سرویس دهیتون شده افتضاح	SAD	1.0
1	NaN	...قرار بود ۱ ساعته برسه ولی نیم ساعت زودتر از مو	HAPPY	0.0
2	NaN	...قیمت این مدل اصلا با کیفیتش سازگاری نداره. فقط	SAD	1.0
3	NaN	...عاللی بود همه چه درست و به اندازه و کیفیت خوب	HAPPY	0.0
4	NaN	...شیرینی وانیلی فقط یک مدل بود	HAPPY	0.0

دیتاست دارای ستون‌های نشان داده شده در تصویر بالا است. با بررسی دیتا، سطرهای دارای ایراد و یا بدون مقدار را حذف می‌کنیم. در نهایت 69480 سطر باقی می‌ماند.

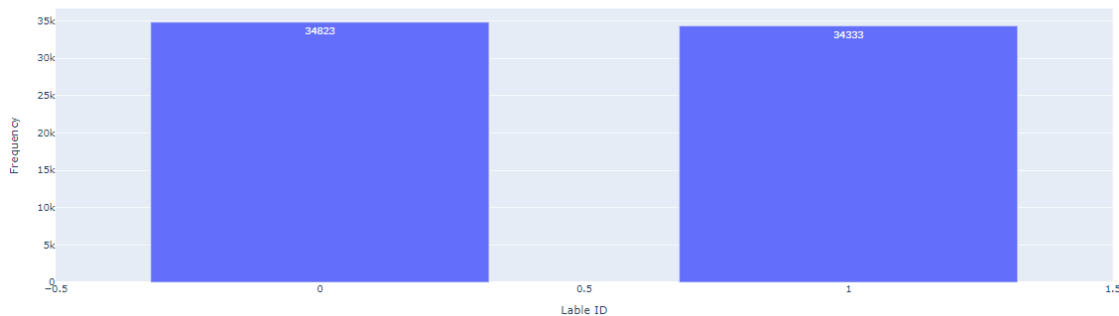
در مرحله بعدی با به دست آوردن ماکسیمم طول یک کامنت و محاسبه درصد کامنت‌های زیر ۱۰۰ کلمه (99.53%)، کامنت‌های با طول بیشتر را حذف می‌کنیم. توزیع باقی کامنت‌ها به صورت زیر است:

Distribution of word counts within comments



توزیع کامنت‌ها بر حسب رضایت نیز به این صورت است:

Distribution of label id within comments



سپس ستون‌های اضافه را حذف می‌کنیم و کامنت‌ها را نرمال‌سازی می‌کنیم:

comment	label_id
واقعاً حیف وقت که بنویسم سرویس دهیتون شده افتضاح	0
...قرار بود ۱ ساعته برسه ولی نیم ساعت زودتر از مو	1
...قیمت این مدل اصلاً با کیفیتش سازگاری نداره، فقط	2
...عالی بود همه چه درست و به اندازه و کیفیت خوب	3
...شیرینی وانیلی فقط بگ مدل بود	4

سپس دیتا را به نسبت‌های ۸، ۱، ۱ برای آموزش، اعتبارسنجی و تست، تقسیم بندی می‌کنیم.

## ii. لود کردن مدل

برای آموزش مدل و ذخیره اطلاعات GPU استفاده می‌کنیم. مدل پیش آموزش دیده را از ریپازیتوری طاقچه لود می‌کنیم و اطلاعات آن را پرینت می‌کنیم.

```
{
  "architectures": [
    "BertForMaskedLM"
  ],
  "attention_probs_dropout_prob": 0.1,
  "classifier_dropout": null,
  "hidden_act": "gelu",
  "hidden_dropout_prob": 0.1,
  "hidden_size": 768,
  "initializer_range": 0.02,
  "intermediate_size": 3072,
  "layer_norm_eps": 1e-12,
  "max_position_embeddings": 512,
  "model_type": "bert",
  "num_attention_heads": 12,
  "num_hidden_layers": 12,
  "pad_token_id": 0,
  "position_embedding_type": "absolute",
  "transformers_version": "4.33.0",
  "type_vocab_size": 2,
  "use_cache": true,
  "vocab_size": 100000
}
```

با استفاده از تابع `encode_plus` یک سمپل را انکد می‌کنیم و خروجی را چک می‌کنیم.

گوست چیزبرگر خام بود و خوب پخته نشده بود

```
input_ids:
tensor([[ 2, 5835, 4370, 20215, 5014, 2834, 1379, 4124, 11208, 4338,
        2834,  4,  0,  0,  0,  0,  0,  0,  0,  0,
         0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
         0,  0]])
token_type_ids:
tensor([[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
         0, 0, 0, 0, 0, 0, 0]])
attention_mask:
tensor([[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
         0, 0, 0, 0, 0, 0, 0]])
```

دیتای آموزش اعتبارسنجی و تست را به دیتا لودر تبدیل می‌کنیم. سپس مدل تحلیل احساس را با استفاده از برت پیش آموزشی و دو لایه `dropout` و `classifier` تعریف می‌کنیم. مدل را با `epoch=3` و `optimizer=AdamW` و `loss_function=CrossEntropyLoss` و استفاده از کتابخانه `tqdm` (برای نشان دادن روند آموزش) آموزش می‌دهیم. و در هر مرحله که در اعتبارسنجی مدل بهتر عمل کرد، پارامتر هارا ذخیره می‌کنیم. همانطور که در مرحله نرمال‌سازی پارسیوار از هضم بهتر عمل کرده‌است، در این مرحله نیز بهتر عمل می‌کند.

### iii. تست مدل

با استفاده از `classification_report` عملکرد مدل را بررسی می‌کنیم.

	precision	recall	f1-score	support
HAPPY	0.89	0.84	0.87	3560
SAD	0.84	0.89	0.87	3356
accuracy			0.87	6916
macro avg	0.87	0.87	0.87	6916
weighted avg	0.87	0.87	0.87	6916

## c. BERT encodings + Random Forest

در این بخش با استفاده از دیتای تولید شده توسط برت توکنایزر یک مدل `random forest` را آموزش می‌دهیم.

### i. نرمال سازی دیتا

در این مرحله مانند تمام تمرین‌های قبل دیتا را بارگذاری و آن را نرمال سازی می‌کنیم.

comment	label id
واقعا حیف وقت که بنویسم سرویس دهیتون شده افتضاح	1
...قرار بود ا ساعته برسه ولی نیم ساعت زودتر از مو	0
...قیمت این مدل اصلا با کیفیتش سازگاری نداره، فقط	2
...،عالی بود همه چه درست و به اندازه و کیفیت خوب	3
...شیرینی وانیلی فقط یک مدل بود	0

### ii. ساختن انکودینگ ها

با استفاده از تابع `encode_plus` از `BertTokenizer`، انکود شده ی تمام کامنت ها را می‌سازیم. با ضرب `attention_mask` و `input_ids` ورودی مورد نظر را تولید می‌کنیم. و با `train_test_split` دیتا را به دو بخش آموزش و تست تقسیم می‌کنیم.

### iii. آموزش و تست

بعد از آموزش مدل `Random Forest`، آن را تست می‌کنیم. نتایج به صورت زیر است:

	precision	recall	f1-score	support
0	0.75	0.61	0.68	5245
1	0.67	0.79	0.72	5129
accuracy			0.70	10374
macro avg	0.71	0.70	0.70	10374
weighted avg	0.71	0.70	0.70	10374

### 3. نتیجه‌گیری

پارس برت یک مدل آموزش داده شده ی فارسی است که میتوان از آن برای اهداف مختلف، من جمله تحلیل احساسات استفاده کرد. همچنین میتوان از دیتای کد شده آن نیز در مدل‌های دیگر استفاده کرد.

### 4. منابع

[https://colab.research.google.com/github/hooshvare/parsbert/blob/master/notebooks/Taaghche\\_Sentiment\\_Analysis.ipynb#scrollTo=aQB4nI4VP2-V](https://colab.research.google.com/github/hooshvare/parsbert/blob/master/notebooks/Taaghche_Sentiment_Analysis.ipynb#scrollTo=aQB4nI4VP2-V)

<https://web.stanford.edu/~jurafsky/slp3/ed3book.pdf>