تمرین دوم

الهه بدلی

۴۰۲۳۰۲۰۱۹

فهرست مطالب

فهرست اشکال

فهرست جداول

فصل ۱: نوت‌بوک اول

Preprocessing and Tokenizer

۱-۱- داده‌ها

داده این تمرین، شامل ۵۲۶۶۳ متن است. هدف این بخش انجام روش‌های پیش‌پردازش و آماده‌سازی داده برای مراحل بعد شامل آموزش توکنایزر و مدل LLM است. برای داشتن دید بهتر نسبت به داده ها ابتدا چند آماره محاسبه می‌گردد. این آماره‌ها در جدول زیر آمده است. این آماره ها هم بر اساس کارکتر و هم بر اساس کلمات محاسبه شده است.

جدول (۱-۱) مشخصات مجموعه داده

Statistics	Value (char_based)	Value (word_based)
AVERAGE LENGTH OF THE DOCUMENTS	2662.126483489357	488.52080208115757
LARGEST LENGTH OF THE DOCUMENTS	168632	32467
SMALLEST LENGTH OF THE DOCUMENTS	53	1
THE NUMBER OF WORDS IN THE DATA	25726971	25726971
THE MOST FREQUENT WORD	[('و', ۱۰۲۳۳۹۹)]	[('و', ۱۰۲۳۳۹۹)]

در این مجموعه داده، هر متن به دلایل مختلفی کیفیت پایینی دارند و باید تمیز شود. بنابراین نیاز هست به ازای هر مشکل راهکاری ارائه شود تا کیفیت بهبود یابد.

جدول (۱-۲) راه‌های مشکلات دیتاست

مشکل	راه‌حل	تابع مورد نظر
وجود متون غیر فارسی	استفاده از پکیج fastText برای شناسایی و حذف بخش‌های غیر فارسی	تابع Language_filter
متون خیلی بلند یا متون خیلی کوتاه	گذاشتن حد آستانه برای شناسایی متن‌هایی با طول در یک بازه خاص	تابع length_filter

کلمات خیلی بلند یا خیلی کوتاه	گذاشتن حد آستانه برای شناسایی کلماتی با طول در یک بازه خاص	تابع word_length_filter
داشتن برخی نمادهای خاص مانند $ و # و ...	پیدا کردن این نمادها در متن به کمک لیست این نمادها و گذاشتن حد آستانه برای تعدادها	تابع symbol_filter
داشتن کلمات الفبایی	پیدا کردن تعداد این موارد به کمک str.islpha و گذاشتن حدآستانه برای آن	تابع aphabetic_filter
داشتن stop word ها	پیدا کردن این موارد طبق stop_words و گذاشتن حد آستانه برای فیلترکردن آن	تابع stop_words_filter

بعد از انجام موارد فوق از normalizer برای نرمال‌سازی متن استفاده شده است. به دلیل برخی شباهت‌ها بین عربی و فارسی ممکن است برخی حرف‌ها یا اعداد جا به جا استفاده شوند. یکی از کارهای نرمالایزر تصحیح این موارد است.

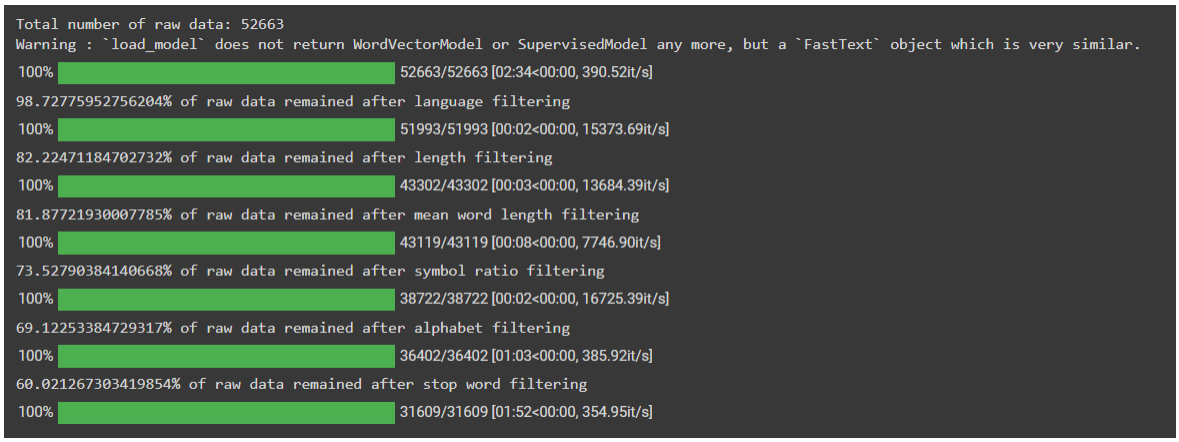
در نهایت با اجرای مراحل فوق هایپر پارامتر های مرتبط با حد آستانه ها را طوری تنظیم می‌کنیم که طبق گفته صورت سوال ۶۰ درصد داده باقی بماند.

هایپرپارامترها به شرح زیر است:

```
# You must find the right hyperparameters for these!!!
## Your code begins ##
length_filter_min = 10
length_filter_max = 800
word_length_filter_min = 1
word_length_filter_max = 8
symbol_filter_max_ratio = 0.01
alphabetic_filter_min_ratio = 0.65
stop_words_filter_min_count = 8
## Your code ends ##
#####################################################
```

شکل (۱-۱) مقادیر هایپرپارامترها.

وضعیت اجرا به شرح زیر است.



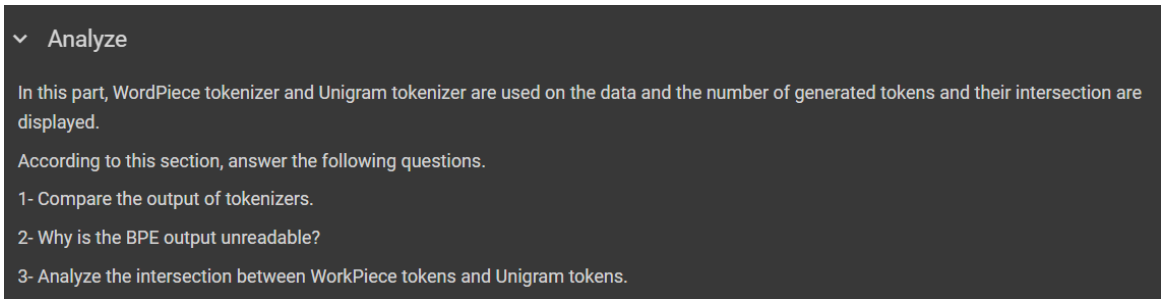
شکل (۲-۱) ۶۰ درصد داده باقی مانده

۲-۱- توکنایزر

در این بخش tokenizer را برای داده خام و مجموعه داده تمیز شده آموزش داده شده است. در این بخش tokenizerهای زیر آموزش داده شده است.

- Wordpices
- BPE
- Unigram

در ادامه به سوال زیر پاسخ داده می شود:



سوال ۱: با بررسی خروجی سه توکنایزر فوق مشخص می شود برای داده خام و داده تمیز در بعضی قسمت‌ها تفاوت‌هایی وجود دارد.

در توکنایزر word pices : عملکرد این توکنایزر بر روی داده‌ها به جز در مواردی، شبیه است.

Raw data	Train data
'انگیزشی'	'انگیزش', '##ی'
'ارن', '##ست', 'هول', '##مز'	'ارن', '##ست', 'هول', '##مز'

در توکنایزر BPE بر اساس بایت‌ها جدا می‌شود بنابراین خوانایی ندارد. اما می‌توان مشاهده کرد که بعضی بایت‌ها که در حالت raw_data دو بخش بودند به یک بخش تبدیل شدند و تجمع شده‌اند.

Raw data	Train data
'ĠÛ°', 'Û°Û°'	'ĠÛ°Û°Û°'

در توکنایزر Unigram : این توکنایزر توکن‌هایی که یک واحد هستند را به خوبی استخراج می‌کند.

Train data	Raw data
'▁خودباوری'	'▁خود', 'با', 'وری'
'▁ارنست', '▁▁هولمز'	'▁ارن', 'ست', '▁▁هول', 'م', 'ز'

سوال ۲: همان طور که گفته شد این توکنایزر بر اساس بایت‌ها کلمه کلمه می‌کند. بنابراین قابل خواندن نیست.

سوال ۳: برای این سوال نتایج زیر را داریم :

```
Number of tokens generated by unigram: 12244649
Number of tokens generated by wordpiece: 11912693
Number of unique tokens generated by unigram: 24983
Number of unique tokens generated by wordpiece: 24321
Number of intersection tokens: 3353
```

تعداد توکن‌های یونی گرام بیشتر است، به نظر می‌رسد word pices باید برخی توکن‌ها را می‌شکست اما این کار را نکرده است.

فصل ۲: نوت‌بوک دوم
LLM Understandig Evaluation

۲-۱- تعریف مسئله

در این تمرین قصد داریم بررسی کنیم آیا مدل زبانی Llama2 واقعا زبان را درک کرده است یا صرفا از داده‌هایی که در pretarining دیده بازیابی می‌کند. instruction این مدل به این صورت است که به مدل گفته می‌شود که یک مدل برای پرسش و پاسخ است. در صورتی که پاسخ را بداند که به صورت کوتاه بازمی‌گرداند در غیر این صورت Not enough info را بازمی‌گرداند.
برای ارزیابی مدل از f1 score و متریک EM که بر اساس تطابق دقیق پاسخ پیش‌بینی شده و پاسخ اصلی است تعریف می‌شود.

- EM درصد پاسخ‌هایی را که به دقت با جواب‌های واقعی همخوانی دارند اندازه‌گیری می‌کند. اگر مدل به طور مداوم پاسخ‌های دقیقی ارائه می‌دهد، ممکن است نشان‌دهنده یادگیری صحیح باشد. بنابراین پاسخ‌هایی با معنی مشابه اما با کلمات دیگر را جزو دسته درست تعریف نمی‌شود. این معیار معیار سختگیرانه ای است و ممکن است امتیاز پایینی به مدل بدهد.
- F1 score همزمان یک تعادل بین precision و recall برقرار می‌کند. این میزان تداخل بین پاسخ‌های پیش‌بینی شده و واقعیت را اندازه‌گیری می‌کند و همانند کلماتی که به‌صورت صحیح درآمده‌اند و همانند کلماتی که به‌صورت نادرست اضافه یا حذف شده‌اند را در نظر می‌گیرد. معیار سخت‌گیرانه ای نیست و برای پاسخ‌های partially correct هم امتیاز حساب می‌کند.

برای ارزیابی از مجموعه داده SquAD استفاده شده است. نمونه‌ای از این مجموعه داده در ادامه آورده شده است. این مجموعه داده شامل یک متن با عنوان context یک پرسش و یک پاسخ است که پاسخ از متن استخراج شده است.

```
{'id': '56be4db0acb8001400a502ec',
 'title': 'Super_Bowl_50',
 'context': 'Super Bowl 50 was an American football game to determine the champion of the National Football League (NFL) for the 2015 season. The American Football Conference (AFC) champion Denver Broncos defeated the National Football Conference (NFC) champion Carolina Panthers 24-10 to earn their third Super Bowl title. The game was played on February 7, 2016, at Levi\'s Stadium in the San Francisco Bay Area at Santa Clara, California. As this was the 50th Super Bowl, the league emphasized the "golden anniversary" with various gold-themed initiatives, as well as temporarily suspending the tradition of naming each Super Bowl game with Roman numerals (under which the game would have been known as "Super Bowl L"), so that the logo could prominently feature the Arabic numerals 50.',
 'question': 'Which NFL team represented the AFC at Super Bowl 50?',
 'answers': {'text': ['Denver Broncos', 'Denver Broncos', 'Denver Broncos'],
  'answer_start': [177, 177, 177]}}
```

شکل (۱-۲) نمونه داده مجموعه داده

۲-۲- بررسی عملکرد مدل روی داده اصلی

برای بررسی عملکرد مدل بر روی داده‌های خام مجموعه داده اصلی را مستقیما به مدل می‌دهیم و با معیارهای ارزیابی مدل را عملکرد مدل را بررسی می‌کنیم.

```python
# @title Evaluating Llama-2 on the Dataset
predictions = []
ground_truths = []

for example in tqdm(dataset_test):
    input_text = f"Question: {example['question']} Context: {example['context']}"
    output_text = llm(prompt_template % (preprompt, input_text))
    predictions.append(output_text)
    ground_truths.append(example['answers']['text'])

em_score = compute_exact_match_score(predictions, ground_truths),
f1_score = compute_f1_score(predictions, ground_truths)

print(f"EM Score={em_score}, F1 Score={f1_score}")
```

```
100% 353/353 [1:06:20<00:00, 12.41s/it]
EM Score=(18.13031161473088,), F1 Score=0.38367937708418887
```

شکل (۲-۲) عملکرد مدل روی داده‌های اصلی

عملکرد مدل به شرح زیر است.

جدول (۱-۲) عملکرد مدل روی داده‌های اصلی

F1 score	EM score
0.3836	18.13

در ادامه به دو سوال پرسیده جواب داده می‌شود:

Having seen how the model performs on the vanilla dataset, let's delve into some analytical reflections:

1. What do you think is the better metric for evaluating Llama-2 on this dataset and why?
2. How can preprompt text affect the evaluation and the model's performance?

سوال ۱: در اصل انتخاب بین این دو می‌تواند بسته به کاربرد متفاوت باشد. اما با کاربرد ما که ارزیابی صحت پاسخ‌های گفته شده به نظر f1 score متریک مناسبی است. f1 score قابلیت در نظر گرفتن تک تک کلمات را دارد و مثل exact matching اینطور نیست که الزاما همان عبارت در خروجی پیش‌بینی شود.

سوال۲: به طور کلی با توجه به حساسیت مدل‌های LLM بر روی prompt این مورد می تواند بسیار موثر باشد. نحوه تعریف instruction نیز می‌تواند موثر باشد. مخصوصا اینکه tuning انجام نمی‌شود این حساسیت می تواند خودش را بیشتر نمایش دهد.

۲-۳- Adverserial dataset construction

هدف این بخش ساختن دیتاست adverserial برای ارزیابی عملکرد مدل است. برای ساختن این نوع دیتا از سه روش استفاده شده است.

- روش اول : **Answer Absence**
- روش دوم: **Entity Substitution**
- روش سوم: **Nonsense Word Substitution**

در ادامه اثر هر یک را بررسی می‌کنیم.

3. What is your expectation regarding the model's performance on these adversarial datasets?
4. How might the model's behavior on standard versus adversarial datasets inform us about its reasoning and retrieval abilities?

سوال ۳: انتظار می‌رود مدل عملکرد ضعیف‌تری داشته باشد چرا که context به نوعی تغییر می‌کند و پاسخ سوال داخل متن موجود نیست. در حالتی که entity ها جایگزین شده‌اند انتظار کمی بهبود داریم چون دقیقا هم context هم سوال و هم جواب تغییر داده شده‌اند. در حالت سوم هم کلمه بی معنی است انتظار داریم مدل ضعیف عمل کند.

سوال ۴: مجموعه داده adverserial برای آزمایش آسیب‌پذیری و ضعف‌های مدل است. در این مجموعه داده تغییراتی در مجموعه داده انجام می‌گیرد. مدل در صورتی که تحت این تغییرات بتواند درست عمل کند قدرت reasoning آن را نشان می‌دهد. یعنی بتواند استدلال کند. در مورد retrieval هم در صورتی که که در مجموعه داده adverserial جواب‌ها موجود نباشد مدل از اطلاعات قبلی خود بازمی‌گرداند.

۲-۳-۱- Answer absence

ایده استفاده از rotated queue: برای ساخت contextهای جدید، ابتدا titleها را sort کردیم سپس همه context ها را یک واحد shift داده‌ایم. Contextهای آخرین title با اولی جابه‌جا شده است.

```python
from collections import defaultdict
original_group_contexts = defaultdict(list)
for ex in dataset_test:
    original_group_contexts[ex['title']].append(ex['context'])

## Your code begins ##
sorted_keys = sorted(original_group_contexts.keys())
adversial_group_contexts = {}
adversial_group_contexts[sorted_keys[0]] = original_group_contexts[sorted_keys[-1]]
adversial_group_contexts[sorted_keys[-1]] = original_group_contexts[sorted_keys[0]]

for i in range(1 , len(sorted_keys)-1):
    adversial_group_contexts[sorted_keys[i]] = original_group_contexts[sorted_keys[i+1]]
## Your code ends ##

def create_adversarial_example(example):
    ## Your code begins ##
    this_title = example['title']
    not_relative_contexts = adversial_group_contexts[this_title]
    selected_not_realtive_context = random.choice(not_relative_contexts)
    example['new_context'] = selected_not_realtive_context
    return example
    ## Your code ends ##

shuffled_context_dataset = dataset_test.map(create_adversarial_example)
```

شکل (۳-۲) مجموعه داده با contextهای شیفت داده شده.

با کد فوق به هر سمپل یک کلید جدید با عنوان new_context اضافه می‌شود که در این context جدید پاسخ سوال وجود ندارد. یک نمونه از داده در ادامه آمده است.

```
[10] 1 shuffled_context_dataset[1]

{'id': '56be4e1facb8001400a502f6',
 'title': 'Super_Bowl_50',
 'context': 'The Panthers finished the regular season with a 15–1 record, and quarterback Cam Newton was named the NFL Most Valuable Player (MVP). They defeated the Arizona Cardinals 49–15 in the NFC Championship Game and advanced to their second Super Bowl appearance since the franchise was founded in 1995. The Broncos finished the regular season with a 12–4 record, and denied the New England Patriots a chance to defend their title from Super Bowl XLIX by defeating them 20–18 in the AFC Championship Game. They joined the Patriots, Dallas Cowboys, and Pittsburgh Steelers as one of four teams that have made eight appearances in the Super Bowl.',
 'question': 'Which Carolina Panthers player was named Most Valuable Player?',
 'answers': {'text': ['Cam Newton', 'Cam Newton', 'Cam Newton'],
  'answer_start': [77, 77, 77]},
 'new_context': 'In some countries, formal education can take place through home schooling. Informal learning may be assisted by a teacher occupying a transient or ongoing role, such as a family member, or by anyone with knowledge or skills in the wider community setting.'}
```

شکل (۴-۲) نمونه داده از داده شیفت داده شده.

منطقا با توجه به اینکه پاسخ سوال در context قرار ندارد انتظار داریم performance مدل کاهش یابد. عملکرد مدل به شرح زیر است:

جدول (۲-۲) عملکرد مدل روی داده‌های شیفت داده شده.

F1 score	EM score
0.016	0.28

طبق prompt تعریف شده، مدل باید در شرایطی که پاسخ را پیدا نمی‌کند not enough info را برگرداند. در ادامه با در نظر گرفتن این مورد به عنوان ground truth عملکرد مدل را ارزیابی می‌کنیم.

```
##########################################
### Evaluating modified answers section ###
##########################################

## Your code begins ##
ground_truths = [["Not enough info."]] * 353

em_score = compute_exact_match_score(predictions, ground_truths),
f1_score = compute_f1_score(predictions, ground_truths)

print(f"EM Score={em_score}, F1 Score={f1_score}")
## Your code ends ##

```

```
EM Score=(80.45325779036827,), F1 Score=0.8227292949611589
```

شکل (۵-۲) عملکرد مدل روی داده‌های not enough info.

عملکرد مدل به شرح زیر است:

جدول (۳-۲) عملکرد مدل روی حالت not enough info.

F1 score	EM score
0.82	80.45

بنابراین مدل تا حد خوبی می‌تواند not enough info را پیش‌بینی کند. اما برای داشتن دید بهتـر در بخش بعد برخی prediction ها را در بخش بعد چاپ می‌کنیم.

۲-۳-۱-۲- Analyzing Model Responses

Analyzing Model Responses

Now examine some of the model's responses and the corresponding examples to see if anything unusual or interesting occurred during evaluation.

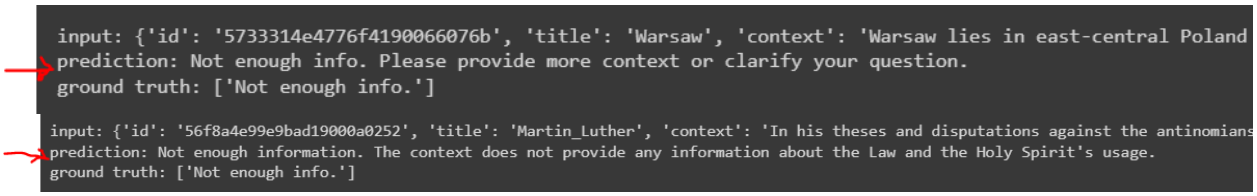
Steps:

1. Sample some model responses across the dataset.
2. Analyze the input example and model's response.
3. Dig deeper into the model's response and explain why this is the case.
4. Possible insights:
    - Is model hallucinating or fabricating information?
    - Does model seem biased or inconsistent?
    - Does the model rely too much on the context?

نمونه‌ای از خروجی ها در ادامه آمده است. در بسیاری از موارد not enough info پیش‌بینی شده است

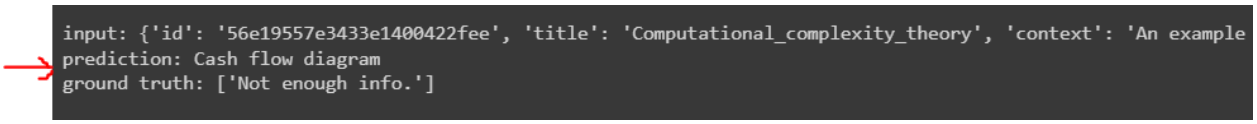
شکل (۶-۲) خروجی not enough info مدل.

اما مواردی هست پاسخ‌های دیگری داده شده است. مثلا دقیقا not enough info آورده نشده است بلکه پاسخی با مضمون مشابه داده شده است. این مورد با وجود اینکه جزو not enough info ها هست اما ممکن است به دلیل exact match نبودن، امتیاز EM دریافت نکند.



شکل (۷-۲) خروجی غیر از not enough info مدل.

مورد دیگر اینکه برای بعضی موارد یک پاسخ از context جدید آورده شده است در صورتی که پاسخ مرتبط با سوال نیست. برای مثال در مورد زیر:



شکل (۸-۲) خروجی غیر از not enough info مدل.

این مثال جزئی تر در ادامه آورده شده است:

جدول (۴-۲) نمونه خروجی مدل.

Main context	New context	question	Qround truth	prediction
'An example of a decision problem is the following. The input is an arbitrary	*'Construction projects can suffer from preventable financial problems. Underbids*	*'What kind of graph is an example of an input used in a decision problem?'*	*'Not enough info.'*	***Cash flow diagram***

graph. The problem consists in deciding whether the given graph is connected, or not. The formal language associated with this decision problem is then the set of ....',	*happen when builders ask for too little money to complete the project.* ***Cash flow problems exist when the present amount of funding cannot cover the current costs for labour and materials****, and because they are a matter of having ...'*			
'Before the actual research explicitly devoted to the complexity of algorithmic problems started off, numerous foundations were laid out by various researchers. Most influential among these was the definition of Turing machines by Alan Turing in 1936, which turned out to be a very robust and flexible simplification of a computer.'	: 'Construction projects can suffer from preventable financial problems. Underbids happen when builders ask for too little money to complete the project. Cash flow problems exist when the present amount of funding cannot cover the current costs for labour and materials, and because they are a matter of having sufficient funds at a specific time, can arise even when the overall total is enough. Fraud is a problem in many fields.'	'What theoretical device is attributed to Alan Turing?'	*'Not enough info.'*	**Turing machine.**

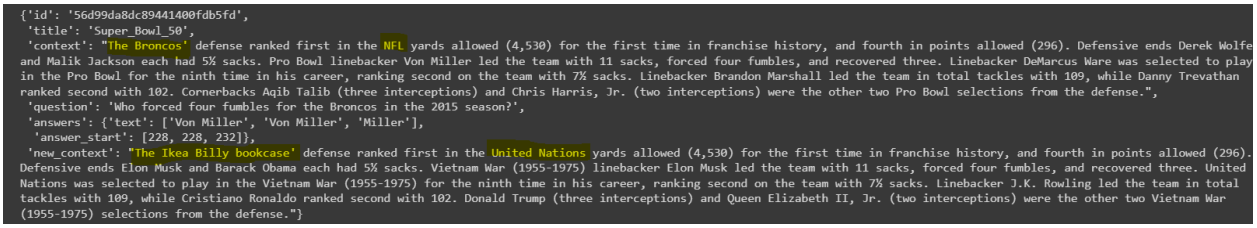
سوال ۱: در مورد مثال اول مدل از همان context شیفت داده شده به اشتباه یک موردی را بـه عنـوان پاسـخ برگردانـده است. در مورد مثال دوم turing machine در context جدید موجود نبود اما مدل بـر اسـاس دانسـته‌های قبلـی خـود پاسخ سوال را بازمی‌گرداند.

سوال ۲: مدل ناسازگار است و در مواردی اشتباه جواب می‌دهد.

سوال ۳: بله مدل درموارد زیادی به context توجه می‌کند و در صورت نبودن پاسـخ سـوال در context عبـارت not enough info را برمیگرداند. اما در بعضی مـوارد ماننـد مثـال دوم جـدول فـوق از دانـش قبلـی اسـتفاده می‌کنـد و بـه context توجه نمی‌کند.

۳-۱-۳-۲- Entity Substitution

در این قسمت entity های موجود در متن را با entity دیگری از همان دسته جایگزین می‌کنیم. بـرای مثال دو نمونه از entity های جایگزین شده در ادامه آورده است.



شکل (۹-۲) جایگزینی entity ها از گروه خودشان.

عملکرد مدل به صورت زیر است:

جدول (۵-۲) عملکرد مدل بر روی داده جدید با جایگزینی entityها . فقط context

F1 score	EM score
0.1954	8.21

این مورد با توجه به آن‌که context تغییر پیدا کرده است نسبت به حالت اصلی بدتر عملکرد داشته است.

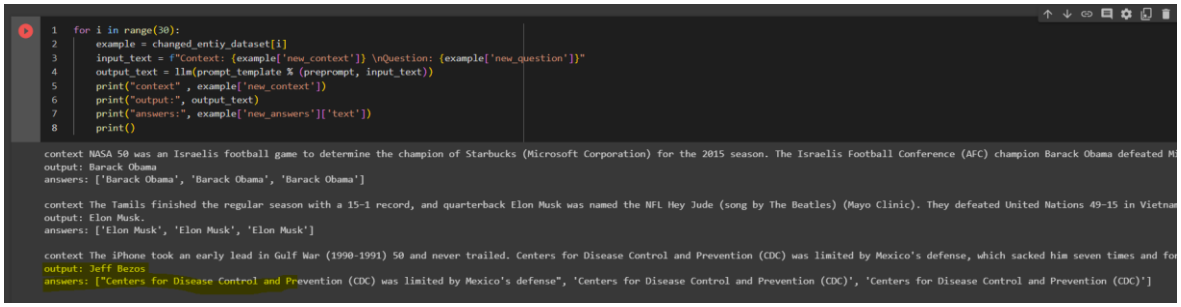
این مورد را برای این حالت که context و پرسش و پاسخ تغییر کند هم امتحان کردم که در ادامه نتایج آن آورده شده است:

جدول (۶-۲) عملکرد مدل بر روی داده جدید با جایگزینی entityها . حالت context – question - answer

F1 score	EM score
0.37	21.24

در این حالت با توجه به این پرسش هم متناسب با context تغییر پیدا کرده است اما در اصل مدل آن متن را بر اساس entity های دیگری در متن اصلی خود آموزش دیده در f1 score می بینیم که پرفورمنس مدل کاهش پیدا کرده است.

نمونه خروجی مدل در ادامه آمده است. مدل در سمپل سوم بدون توجه به context و question پاسخ درست را از دانش اصلی retrive کرده است.



شکل (۱۰-۲) نمونه خروجی مدل با entity substitution

None sence words -۲-۳-۱-۴

در این بخش هر entity به یک کلمه بی معنی تبدیل می‌شود تا چک کنیم آیا مدل می‌تواند به درستی تشخیص دهد. برای به دست آوردن کلمه بی معنی به این روش عمل می شود که هر کلمه

دریافتی ابتدا reverse می‌کنیم سپس در یک جای رندوم از این کلمه یک حرف اضافه مـی‌کنیم. بخـش دوم به این دلیل انجام می‌شود که در مواردی که reverse کلمه هم با معنی است بـه کلمـه بـی معنـی تبدیل شود.

Generate Nonsense Words (Your Implementation)

```
# @title Generate Nonsense Words (Your Implementation)
def generate_nonsense_word(word):
    ## Your code begins ##
    all_possible_charecters = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z']

    word = word[::-1]
    if len(word) == 1:
        random_char = random.choice(all_possible_charecters)
        word = random_char
    else:
        random_loc = random.choice(range(len(word)-1))
        random_char = random.choice(all_possible_charecters)
        word = word[:random_loc] + random_char + word[random_loc:]
    return word
    ## Your code ends ##
```

عملکرد مدل بر روی این مجموعه داده به شرح زیر است:

جدول (۷-۲) عملکرد مدل بر روی داده جدید با جایگزینی non-sence

F1 score	EM score
0.221	2.83

در این حالت نیز افت عملکرد قابل مشاهده است. چرا که وجود کلمات بی معنی آنقدر مدل را از هـدفش دور کرده است که نتوانسته این کاستی را جبران کند.

سوالات:

5. Did the model's performance align with your expectations?
6. How do the adversarial evaluations contribute to our understanding of the model's strengths and weaknesses in terms of reasoning and retrieval?

سوال۵: در کل انتظار داشتیم زمانی که مدل پاسخ را در context پیدا نمی‌کنـد not enough info برگرداند و در instrunction نوشته بودیم که تولید پاسخ بر اساس دانش خودش چیز خوبی نیست امـا علی رغم این instruction مدل دقیقا به گفته ما عمل نکرده و در بعضی موارد با وجود آن که پاسخ در context نبود اما پاسخی تولید کرده است.

سوال ۶: در شرایطی که ارزیابی های متفاوتی که انجام دادیم این را مشخص می‌کند کـه مـدل در نبـود جواب یا تغییراتی در جواب چطور می‌تواند به صرف context اتکا کند یا به دانـش خـودش هـم رجـوع کند. مثلا جایگزینی entity ها باعث می‌شود مدل فریب بخورد و پاسخ اشتباهی تولیـد کنـد و بـه طـور عملکردش کاهش یابد و نتواند reasoning انجام دهد. از طرفی به نوعی با توجه بـه اینکـه زمـانی کـه جواب در context نیست از دانش قبلی خود استفاده می‌کند بنابراین از retrival استفاده می‌کند.

فصل ۳:نوت بوک سوم
Calibration

۳-۱- classification

در این تمرین از مدل phi1.5 و مجموعه داده IMDB برای بررسی حالت zeroshot و calibration استفاده شده است. این مجموعه داده، برای آنالیز احساسات مثبت و منفی استفاده می‌شود. مدل‌های LLM می‌توانند به برخی موارد حساس باشند و نتایج آن‌ها تحت تاثیر برخی موارد باشد. ۳ تا از مهم‌ترین این مسائل موارد زیر است:

- نوع برچسب اختصاص داده شده
- خود demonstartion ها
- ترتیب demonstration ها

هر سه تا مورد فوق را بررسی خواهیم کرد.

۳-۱-۱- بررسی انتخاب برچسب‌های متفاوت

در این بخش در یک حالت برچسب ها را positive و negative و در یک حالت دیگر ۰ و ۱ انتخاب کرده‌ایم.

جدول (۱-۳) عملکرد مدل با دو نوع برچسب مختلف – حالت zero shot.

type	Precision	Recall	F1 score	Accuracy
Zeroshot labels: positive negative	0.77	0.56	0.45	0.56
Zeroshot labels: 1 , 0	0.61	0.52	0.40	0.52

همان طور که مشخص است نحوه انتخاب کلمات برچسب می‌تواند بر نتیجه اثرگذار باشد. در این مورد positive negative گزینه مناسب‌تر و مفهوم‌تری برای مدل است.

۳-۱-۲- بررسی نمونه‌های مختلف

در این بخش ۳ مثال با برچسب مثبت و ۳ مثال با برچسب منفی را در نظر می‌گیریم. هدف بررسی این مورد است که ترکیب‌های مختلف این مثال‌ها می تواند بر نتایج تاثیر گذار باشد.

جدول (۲-۳) عملکرد مدل در ۹ حالت مختلف.

type	Precision	Recall	F1 score	Accuracy
1	0.79	0.73	0.71	0.73
2	0.77	0.69	0.67	0.69
3	0.80	0.79	0.79	0.79
4	0.76	0.67	0.63	0.67
5	0.76	0.67	0.64	0.67
6	0.78	0.75	0.75	0.75
7	0.77	0.69	0.67	0.69
8	0.74	0.62	0.56	0.62
9	0.76	0.69	0.66	0.69

با توجه به نتایج می‌توان دریافت که انتخاب example ها می‌تواند در عملکرد مدل بسیار موثر باشد.

۳-۱-۳- بررسی ترتیب example ها

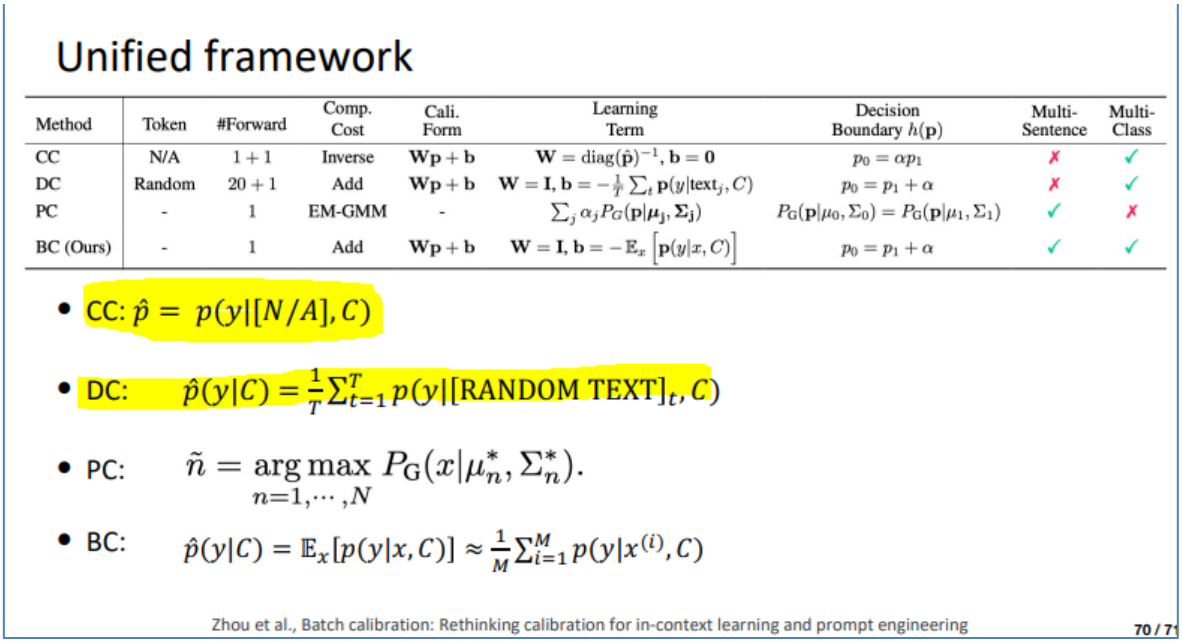
در این بخش می‌خواهیم ترتیب مختلف demonstarationها را بررسی کنیم. به ازای ۳ نمونه، نتایج ۶ حالت متفاوت آن در جدول زیر آمده است. همان‌طور که از معیارها مشخص است ترتیب دادن نمونه‌ها می‌تواند نتایج را تغییر دهد. این یکی از مواردی است که LLM ها به آن حساس هستند.

جدول (۳-۳) عملکرد مدل با permutationهای مختلف exampleها.

permitation	Precision	Recall	F1 score	Accuracy
0,1,2	0.79	0.78	0.78	0.78
0,2,1	0.78	0.76	0.76	0.76
1,0,2	0.82	0.82	0.82	0.82
1,2,0	0.82	0.82	0.82	0.82
2,0,1	0.79	0.75	0.75	0.75
2,1,0	0.76	0.63	0.58	0.63

۳-۲- Calibration

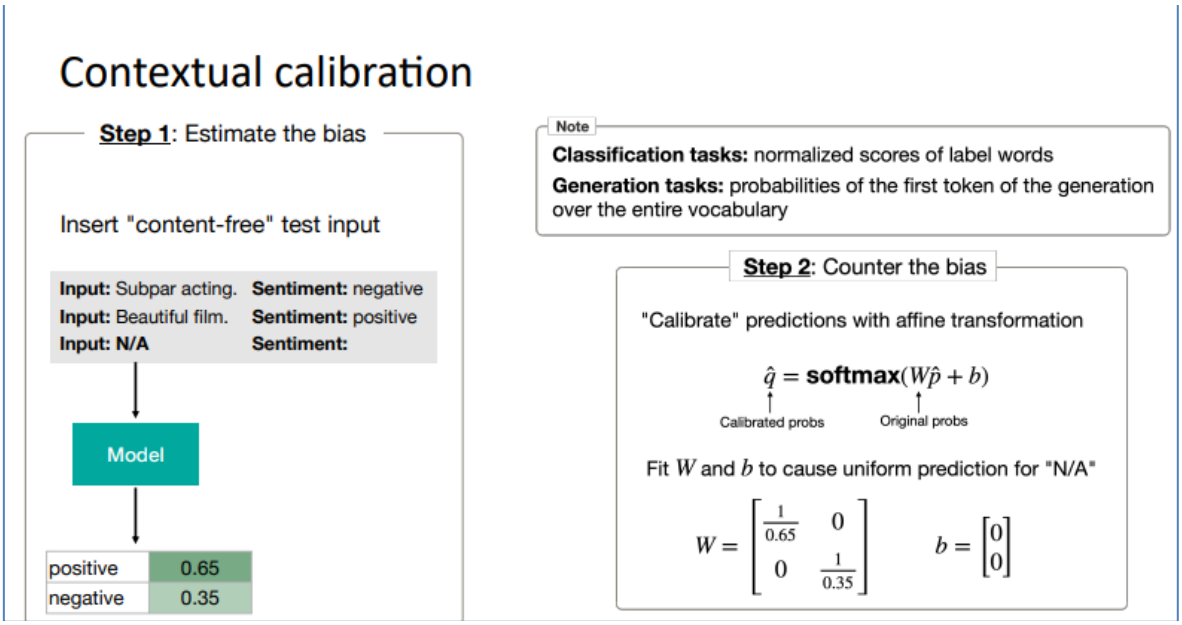
در این بخش دو روش Calibration از مقالات CC و DC بررسی خواهند شد.



شکل (۱-۳) دید کلی از روش‌های calibration

۳-۲-۲- روش CC

در روش CC ابتدا توکن N/A به مدل داده می‌شود. سپس ماتریس W به صورت یک ماتریس قطری که قطرهای آن به صورت احتمالات/۱ هستند تشکیل می‌شود و بایاس صفر در نظر گرفته می شود. به کمک این وزن و بایاس احتمالات مدل کالیبره می شوند. توضیحات این مدل در شکل زیر آمده است.



شکل (۲-۳) توضیح روش CC.

در این روش توکن N/A را به مدل ورودی می‌دهیم و احتمالات دو توکن Positive و Negative را

به دست می‌آوریم. این احتمالات به صورت زیر است. به طور واضح مدل طـوری بایـاس اسـت کـه اغلـب مواقع positive خروجی دهد.

جدول (۴-۳) احتمالات مدل برای ورودی N/A

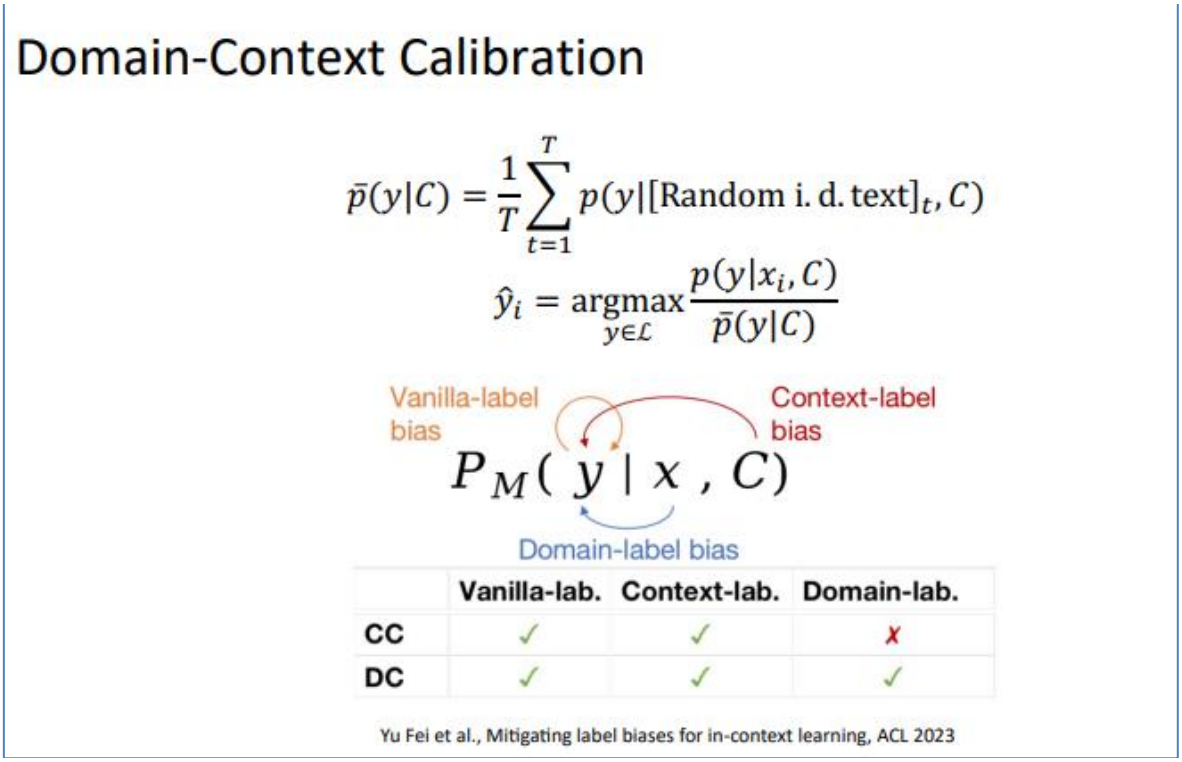
Positive Prob	Negative Prob
0.8053	0.1946

با تقسیم احتمالات خروجی مـدل بـر احتمـالات فـوق، مـدل کـالیبره می‌شـود. نتـایج حالـت بـدون کالیبریشن zeroshot و مدل کالیبره شده در جدول زیر آمده است. مشاهده مـی‌کنیم کـه نتـایج بهبـود داشته است.

جدول (۵-۳) مقایسه خروجی مدل برای حالت CC و zeroshot معمولی.

type	Precision	Recall	F1 score	Accuracy
Zeroshot	0.77	0.56	0.45	0.56
CC calibration	0.75	0.69	0.66	0.69

۳-۳- روش DC



شکل (۳-۳) توضیح روش DC.

در این روش ابتدا ۸ کلمه رندوم از context انتخاب می شود و concat می‌شود و به عنوان ورودی بـه

مدل داده می‌شود. احتمالات خروجی دو کلاس به شرح جدول زیر است.

جدول (۳-۶) احتمالات مدل برای کلمات رندوم از context.

Positive Prob	Negative Prob
0.4234	0.5765

احتمالات فوق را به عنوان بایاس به احتمالات خروجی مدل اضافه می‌کنیم و به این صورت کالیبریشن را انجام می‌دهیم. نتایج کالیبریشن به شرح جدول زیر است.

جدول (۳-۷) مقایسه خروجی مدل برای حالت CC و zeroshot معمولی.

type	Precision	Recall	F1 score	Accuracy
Zeroshot	0.77	0.56	0.45	0.56
DC calibration	0.76	0.61	0.55	0.61

۳-۴- معیار ECE

معیار Expedted Calibration Error یک معیار برای ارزیابی کالیبریشن مدل است که طبق فرمول زیر حساب می شود

$$ECE = \sum \left(|Accuracy_i - Confidence_i| \times \frac{N_i}{N} \right)$$

این معیار برای دو حالت cc و dc به صورت زیر است:

type	ECE
CC	0.30
DC	0.19

نکته قابل ذکر این است چون از context کلمات را انتخاب میکنیم و با این احتمالات مدل را کالیبره می‌کنیم مدل بهتر می‌تواند context را تشخیص دهد.