ماژول صفرم: Normalizer

برای نرمالایزر از مثال آورده شده در خود ریپازیتوری مربوط به Dadmatools استفاده شده است. همانطور که مشخص است مدل علاوه بر اینکه " " را به جای آدرس سایت قرار نداده است و تنها یک فاصله گذاشته است، علامت > را نیز به عنوان یک بخش از URL در نظر گرفته است و آن را به همراه URL حذف کرده است.

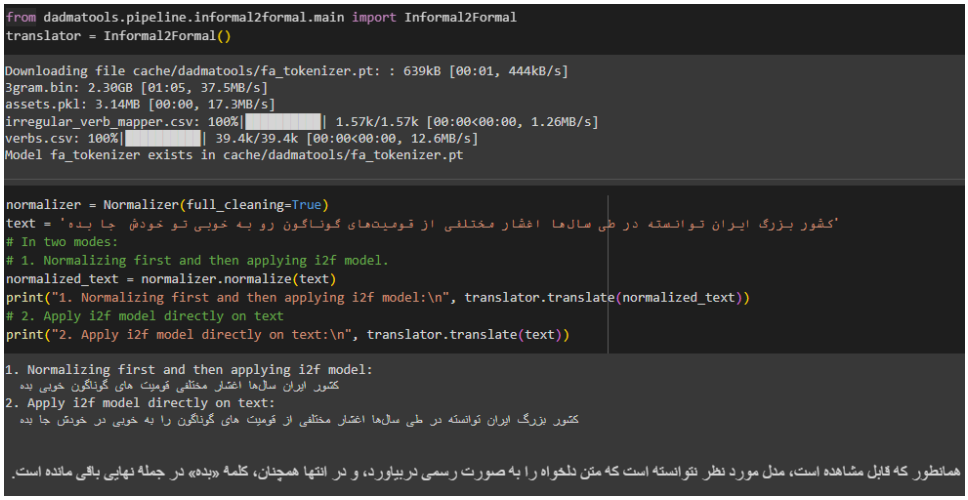
```
# Error is in replacing URL. While we needed "      " to be replaced with the url
# no space was added.
# Also the normalizer model has taken ">" character as part of the url. While
# according to "RFC 3986" the ">" character is not part of valid characters in
# URLs.

normalizer = Normalizer(
    full_cleaning=False,
    unify_chars=False,
    refine_punc_spacing=False,
    remove_extra_space=False,
    remove_puncs=False,
    remove_html=False,
    remove_stop_word=False,
    replace_email_with="<EMAIL>",
    replace_number_with=None,
    replace_url_with="      ",
    replace_mobile_number_with=None,
    replace_emoji_with=None,
    replace_home_number_with=None
)

text = """
<p> آدرس ما این است: <https://github.com/Dadmatech/DadmaTools> </p>
"""
print('input text: \n', text)
print('output text when replace emails and remove urls: \n', normalizer.normalize(text))
```

```
input text:

<p> آدرس ما این است: <https://github.com/Dadmatech/DadmaTools> </p>

output text when replace emails and remove urls:
 <p> آدرس ما این است: < </p>
```

ماژول اول: itf

برای این مدل از Informal2Formal موجود در بخش dadmatools.pipeline.informal2formal.main استفاده کرده‌ایم. برای این مورد نیز از مثال موجود در صفحهٔ گیتهاب مدل استفاده شده است. همانطور که مشخص است مدل به خوبی نتوانسته است که متن را به صورت رسمی در آورد و همچنان «بده» به «بدهد» تبدیل نشده است.

همچنین هر دو حالت نرمالیزه شدن و نشدن نیز تست شده‌اند که در هر دو این مشکل برقرار است.



```
from dadmatools.pipeline.informal2formal.main import Informal2Formal
translator = Informal2Formal()

Downloading file cache/dadmatools/fa_tokenizer.pt: : 639kB [00:01, 444kB/s]
3gram.bin: 2.30GB [01:05, 37.5MB/s]
assets.pkl: 3.14MB [00:00, 17.3MB/s]
irregular_verb_mapper.csv: 100%|| 1.57k/1.57k [00:00<00:00, 1.26MB/s]
verbs.csv: 100%|| 39.4k/39.4k [00:00<00:00, 12.6MB/s]
Model fa_tokenizer exists in cache/dadmatools/fa_tokenizer.pt

normalizer = Normalizer(full_cleaning=True)
text = 'کشور بزرگ ایران توانسته در طی سال‌ها اعشار مختلفی از قومیت‌های گوناگون رو به خوبی تو خودش جا بده'
# In two modes:
# 1. Normalizing first and then applying i2f model.
normalized_text = normalizer.normalize(text)
print("1. Normalizing first and then applying i2f model:\n", translator.translate(normalized_text))
# 2. Apply i2f model directly on text
print("2. Apply i2f model directly on text:\n", translator.translate(text))

1. Normalizing first and then applying i2f model:
  کشور ایران سال‌ها اعشار مختلفی قومیت های گوناگون خوبی بده
2. Apply i2f model directly on text:
  کشور بزرگ ایران توانسته در طی سال‌ها اعشار مختلفی از قومیت های گوناگون را به خوبی در خودش جا بده
```

همانطور که قابل مشاهده است، مدل مورد نظر نتوانسته است که متن دلخواه را به صورت رسمی دربیاورد، و در انتها همچنان، کلمه «بده» در جمله نهایی باقی مانده است.

ماژول دوم: pos

در این بخش، متوجه شدم که مدل نسبت به کلماتی مانند مرید و دبیر و مانند این لغات، از خود حساسیت نشان داده و این لغات را به عنوان فعل تشخیص می‌دهد. همچنین در مثال زیر، متوجه می‌شویم که در حالت غیر نرمالایز شده، مدل دو کلمهٔ نخست را نیز به عنوان فعل در نظر گرفته است که اشتباه می‌باشد.

```
text = 'او دیروز با چهره ای خسته به دیدار دبیرش رفت.'

# In two modes:
# 1. Normalizing first and then applying pos model.
normalized_text = normalizer.normalize(text)
print("1. Normalizing first and then applying pos model:\n")
show_output(nlp(normalized_text), 'text', 'upos', 'xpos')
# 2. Apply pos model directly on text
print("2. Apply pos model directly on text:\n")
show_output(nlp(text), 'text', 'upos', 'xpos')

1. Normalizing first and then applying pos model:

1it [00:00, 15.92it/s]
چهره‌ای     ADJ     ADJ
دیدار       ADJ     ADJ
دبیرش       VERB    V_PA

2. Apply pos model directly on text:

1it [00:00, 18.69it/s]
او          VERB    V_PRS
دیروز       VERB    V_PA
با          ADP     P
چهره‌ای     ADJ     ADJ_INO
خسته        ADJ     ADJ
به          ADP     P
دیدار       VERB    V_PRS
دبیرش       VERB    V_PRS
رفت         VERB    V_PA
.           PUNCT   DELM
```

ماژول سوم: dep

برای بررسی این ماژول از مثال زیر استفاده شده است:

«ب علی به دیدار سحر رفت، اما سحر خانه نبود.»

در مثال زیر، در هر دو حالت نرمالایز شده و نشده، کلمهٔ «خانه» به عنوان root شناخته شده است. این در حالی است که برای آنکه یک لغت را به عنوان root شناسایی کنیم، نیاز است که آن لغت یک فعل باشد و نه یک اسم.

```
1. Normalizing first and then applying dep model:

1it [00:00, 21.13it/s]
علی      0       root
دیدار    1       fixed
سحر      1       advmod
سحر      5       advmod
خانه     3       nmod:poss

2. Apply dep model directly on text:

1it [00:00, 19.26it/s]
علی      9       nsubj
به       9       case
دیدار    9       nmod
سحر      9       conj
رفت      9       cc
،        9       punct
اما      9       cc
سحر      9       conj
خانه     0       root
نبود     9       cop
.        9       punct
```

ماژول چهارم: kasreh

برای بررسی این ماژول از مثال زیر استفاده شده است:

«بر در میکده دیدم که مقیم افتادست»

در این جا یک شعر از حافظ به عنوان مثال آورده شده است. همانطور که مشخص است، مدل نتوانسته است که کسره را در «در» تشخیص دهد و به اشتباه گمان کرده که «مقیم» واژه‌ای است که کسره به آن متصل است.

```
1. Normalizing first and then applying kasreh model:

1it [00:00, 45.32it/s]
میکده    O
دیدم     O
مقیم     S-kasreh
افتادست  O

2. Apply kasreh model directly on text:

1it [00:00, 43.23it/s]
بر       O
در       O
میکده    O
دیدم     O
که       O
مقیم     S-kasreh
افتادست  O
```

ماژول پنجم: lem

برای بررسی این ماژول از مثال زیر استفاده شده است:

«ک. دیروز به فروشگاه رفت، و از آنجا خرید کرد.»

در این مثال ک. اسم یک فرد است که به صورت خلاصه آورده شده است. همانطور که مشخص است، دو واژهٔ فروشگاه و نیز خرید به اشتباه ریشه‌یابی شده‌اند و هر دو معادل خودشان در خروجی آورده شده است. این در حالی است که برای واژهٔ «خرید»، واژهٔ «خر» و نیز برای «فروشگاه» واژهٔ «فرو» ریشه‌های صحیح هستند.

```
1. Normalizing first and then applying lemma model:

1it [00:00, 22.02it/s]
فروشگاه   فروشگاه
خرید      خرید

2. Apply lemma model directly on text:

1it [00:00, 19.55it/s]
ک         ک
.          .
دیروز     دیروز
به        به
فروشگاه   فروشگاه
رفت       رفت#رو
،          ،
و         و
از        از
آنجا      آنجا
خرید      خرید
کرد       کرد#کن
.          .
```

ماژول ششم: tok

برای بررسی این ماژول از مثال زیر استفاده شده است:

« خانهٔ کوچک ما در مرکز شهر می سی سی پی مستقر شده بود.»

در مثال داده شده، می سی سی پی به صورت جدا از هم آورده شده است. در این حالت در نرمالایر با حذف می و پی به طور کامل عبارت ناقص می‌شود و همچنین در حالت بدون استفاده از نرمالایز نیز، واژه به ۴ بخش شکسته شده و هر بخش به صورت یک توکن مجزا در خروجی آورده شده‌اند.

```
1. Normalizing first and then applying tok model:

1it [00:00, 42.39it/s]
خانه
کوچک
مرکز
شهر
سی
سی
مستقر

2. Apply tok model directly on text:

1it [00:00, 31.68it/s]
خانهٔ
کوچک
ما
در
مرکز
شهر
می
سی
سی
لی
مستقر
شده
بود
.
```

ماژول هفتم: ner

برای بررسی این ماژول از مثال زیر استفاده شده است:

«ک. سخت به خود تکانی داد تا بتواند با چشمانی نیم‌بسته کنت سباستین را نگاه کند.»

در خروجی اما، مدل ner در هر دو حالت نرمالایز شده و نشده، «کنت سباستین» را به عنوان یک موجودیت در نظر نگرفته است. همچنین در حالت نرمالایز نشده نیز، «ک.» را به عنوان یک ماژول در نظر نگرفته است.

```
1. Normalizing first and then applying ner model:

1it [00:00, 38.14it/s]
تکانی      O
چشمانی     O
نیم‌بسته   O
کنت        O
سباستین    O

2. Apply ner model directly on text:

1it [00:00, 29.65it/s]
ک          O
.          O
سخت        O
به         O
خود        O
تکانی      O
داد        O
تا         O
بتواند     O
با         O
چشمانی     O
نیم‌بسته   O
کنت        O
سباستین    O
را         O
نگاه       O
کند        O
.          O
```

ماژول هشتم: spellchecker

برای بررسی این ماژول از مثال زیر استفاده شده است:

« در چرخهٔ تکراری فلاکط، زیر فشار چرخ‌دهندهٔ بدبختی به لابح و زاری افتاده بود.»

در خروجی اما، مدل spellchecker نتوانسته است شکل صحیح لغت «لابح» را که «لابه» می‌باشد، تشخیص دهد.

```
1. Normalizing first and then applying spellchecker model:

1it [00:00, 42.10it/s]
{'checked_words': [('فلاکت' ,'فلاکط'), ('لباس' ,'لابح')],
 'corrected': 'دهنده بدبختی لباس زاری افتاده\u200cچرخه تکراری فلاکت فشار چرخ',
 'orginal': 'دهنده بدبختی لابح زاری افتاده\u200cچرخه تکراری فلاکط فشار چرخ'}

2. Apply spellchecker model directly on text:

1it [00:00, 28.72it/s]
{'checked_words': [('فلاکت' ,'فلاکط'), ('سلاح' ,'لابح')],
 'corrected': ' دهندهٔ بدبختی به سلاح و\u200cدر چرخهٔ تکراری فلاکت،زیر فشار چرخ '
              'زاری افتاده بود.',
 'orginal': ' دهندهٔ بدبختی به لابح و\u200cدر چرخهٔ تکراری فلاکط زیر فشار چرخ '
            'زاری افتاده بود.'}
```

ماژول نهم: sent

برای بررسی این ماژول از مثال زیر استفاده شده است:

« در ۲۴ ساعت گذشته، تنها ۴ ساعت خوابیدم. واقعاً که زندگی هنوز قشنگی‌هاش رو داره.»

اما در نتیجه مدل sent نتوانسته است که کنایه بودن این جمله را تشخیص دهد و با درصد اطمینان خوبی آن را به عنوان یک جملهٔ مثبت تلقی کرده است.

```
1. Normalizing first and then applying sentiment model:

1it [00:00, 23.18it/s]
[{'label': 'positive', 'score': 0.9117720127105713}]

2. Apply sentiment model directly on text:

1it [00:00, 23.99it/s]
[{'label': 'positive', 'score': 0.8430431485176086}]
```