



گزارش پروژه پایانی – فاز اول

استخراج رابطه در زبان فارسی با بررسی وابستگی جهانی *Seraji* و *PerDT*

مبانی پردازش زبان و گفتار

استاد: دکتر بهروز مینایی بیدگلی

دانیال بازمانده – محمدحسین کریمیان

در این پروژه هدف اصلی یافتن روابط معنایی در زبان فارسی است به این صورت که بین دو *entity*، یک رابطه استخراج می-کنیم.

در این گام که گام اول پروژه است، ابتدا باید داده های زیادی به دست بیاوریم زیرا برای رسیدن به هدف نهاییمان، به مجموعه بزرگی از داده ها نیاز داریم. سپس باید این داده ها را *normalize* کنیم و با حذف *stop words* ها و پیدا کردن ریشه کلمات، تعداد تکرار شدن هر کلمه را بدست می آوریم و جمله ها را هم مشخص می کنیم.

برای به دست آوردن داده ها، باید *Crawl* کنیم که به معنای دنبال کردن لینک ها یا اصطلاحاً "خزیدن" در وب سایت ها است. برای استخراج داده ها از اینترنت، از کتابخانه *Beautiful Soup* استفاده می کنیم که برای استخراج داده از فایل های *html* کاربرد دارد.

در این تحقیق از منابع سایت *Wikipedia* فارسی برای جمع آوری *data* استفاده کردیم به این صورت که ابتدا به فهرست منابع ویکی پدیا یک ریکوئست می فرستیم تا بتوانیم فهرست الفبای آن را پیدا کنیم و مطابق قطعه کد زیر، که در فایل *collect_alphabets.py* می باشد، الفبای مورد نظر را که در ادامه قراره به لینک های مربوط به آن ها درخواست بزنیم، در یک فایل به نام *alphabets.txt* ذخیره می کنیم.

```
from bs4 import BeautifulSoup, Tag
import requests

resultsFile = open("../data/raw/alphabets.txt", "w", encoding="utf8")
url = "https://fa.wikipedia.org/wiki/فهرست_سریع"
web = requests.get(url)

soup = BeautifulSoup(web.content, "html5lib")
table = soup.find('table', attrs={'style': 'width: 80%; font-family: monospace; padding: 3px; background: #f7f8ff'})
for tr in table.contents[1].contents:
    if isinstance(tr, Tag):
        tds = tr.find_all("td")
        for td in tds:
            resultsFile.write(td.text)
```

مطابق شکل بالا وقتی به *url* مربوطه درخواست می دهیم، در صفحه *html*ی که وارد آن می شویم به دنبال جدولی می گردیم که دارای استایل مشخص شده باشد و همه موارد آن که حروف الفبای مورد نظر ما می باشند را، در فایل *alphabets.txt* می نویسیم. استایلی که برای جدول مشخص کردیم با بررسی کردن تگ های این صفحه مانند شکل زیر به دست می آید.

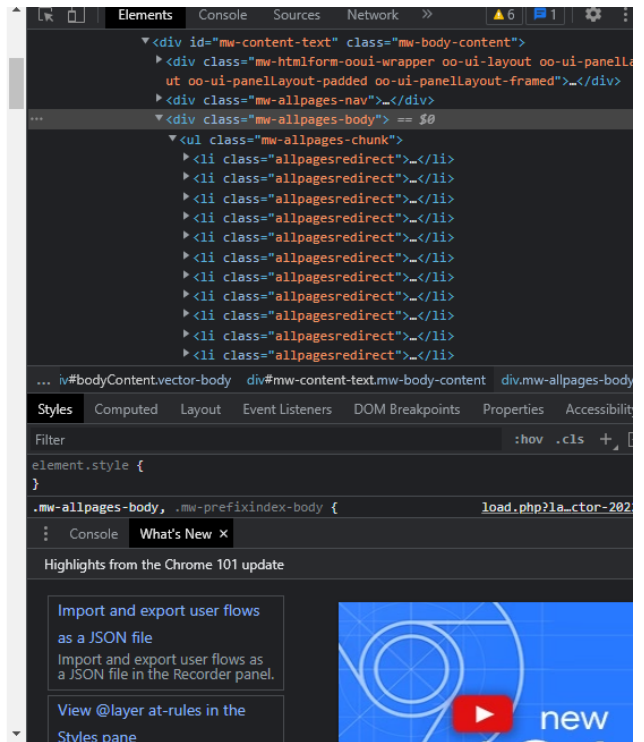
حال که لیست حروفی از الفبا داریم، باید به صفحه هر حرف رفته و لینک های موضوعاتی که با آن حرف شروع می شوند را به دست می آوریم. به این خاطر که داده های مورد نیاز ما حداکثر سی هزار جمله می باشد، فقط موضوعاتی که در صفحه اول هر حرف هستند را در نظر می گیریم و همچنین حروفی از فهرست که بیش از یک حرف دارند را هم حذف می کنیم. سپس مانند قطعه کد زیر که در فایل *collect_hrefs.py* می باشد، به این صفحات یک ریکوئست می فرستیم و آدرس صفحات مرتبط با آن ها را به دست می آوریم و در فایل *hrefs.txt* می نویسیم.

```
data_cleaner.py x collect_alphabets.py x script.py x collect_hrefs.py x _init_.py x
Visual layout of bidirectional text can depend on the base direction (View | Bidi Text Base Direction) Choose direction Hide notification Don't show again

7 url = "https://fa.wikipedia.org/wiki/ویژه:تمام_صفحه_ها"
8 alphabets = open('../data/raw/alphabets.txt', 'r', encoding="utf8")
9 hrefs = open('../data/raw/hrefs.txt', 'w', encoding="utf8")
10 count = 1
11
12 for alphabet in alphabets.read().splitlines():
13     if len(alphabet) == 1:
14         try:
15             # send a request to website
16             web = requests.get(url + 'from=' + alphabet + '&to=&namespace=0')
17         except:
18             # handle a problem that too requests to a website that gives us max_retries error
19             # This block code will get the url and retry 3 times if it gets error
20             session = requests.Session()
21             retry = Retry(connect=3, backoff_factor=0.5)
22             adapter = HTTPAdapter(max_retries=retry)
23             session.mount('http://', adapter)
24             session.mount('https://', adapter)
25
26             web = requests.get(url + 'from=' + alphabet + '&to=&namespace=0')
27
28             soup = BeautifulSoup(web.content, "html5lib")
29             table = soup.find('ul', attrs={'class': 'mw-allpages-chunk'})
30             for to in table.contents:
```

مطابق کد بالا ابتدا فهرست الفبایی که از فایل آن می خوانیم را با *UTF8* اینکود می کنیم تا فرمت آن به دلیل فارسی بودن به هم نریزد. سپس به صفحه اول هر کدام از حروف الفبا در ویکیپدیا درخواست داده و با یافتن جدولی که کلاس مورد نظر ما را دارد و لیست موضوعات صفحه های ویکیپدیا در آن است، هر کدام از این موضوعات که در اسم آن حروفی مانند خط فاصله یا پرانتز است، یا درباره موضوع ابهام زدایی می باشد را نادیده گرفته و بقیه را *decode* می کنیم و در فایل *hrefs.txt* می نویسیم. کلاسی که برای جدول مشخص کرده ایم با بررسی آن صفحه به صورت زیر به دست آمده است.

4-1. $\frac{1}{2} \log \frac{1}{2}$



حال برای اطمینان از نتایج، قسمتی از فایل *hrefs.txt* را در زیر می بینیم:



حال که لینک هایی که می خواهیم از آن ها اطلاعات بگیریم را ذخیره کردیم، باید به هر کدام جداگانه یک ریکوئست بزنیم و متنی که داخل آن نوشته شده است را استخراج کنیم و در پوشه *raw* در فایل *raw_dataset.json* ذخیره کنیم. برای این کار با *encode* کردن لینک های ذخیره شده در فایل *hrefs.txt* در مرحله قبل، به آن ها در خواست می دهیم و با کاوش کردن در فایل *html* سایت ها همه تگ های *p* آن ها که مربوط به متن است را به دست می آوریم. مطابق شکل زیر بعضی جمله ها که در همه مقاله ها می آید را نادیده می گیریم:

```
final_text = ''
for p_tag in div.findChildren("p", recursive=False):
    p_text = ''
    for element in p_tag.contents:
        if isinstance(element, Tag):
            if element.name != 'sup' and element.name != 'br':
                if 'می‌توانید با گسترش آن به ویکی‌پدیا کمک کنید' not in element.text \
                    and element.text.replace('\n', '') != '':
                    p_text += element.text
            else:
                if 'می‌توانید با گسترش آن به ویکی‌پدیا کمک کنید' not in element \
                    and element.replace('\n', '') != '':
                    p_text += element

    text = p_text.replace('\n', '')
    if p_text != '':
        final_text += p_text + ' '

dataset.append({'id': count, 'title': title, 'text': final_text})
count += 1
```

نتیجه آن را در فایل *raw_dataset.json* مشاهده می کنیم:

ی فارسی و الفبای عربی و الفبای عبری و سایر ابجد های استفاده شده توسط زبان های سامی است. این حرف در زبان فارسی شروع کننده خیلی از اسم هاست. "الف", "title": "الف", "id": 1, "text": "این نشان «ا» جزو نشانه های سمیوسه گانه ای که الفبای فارسی را تشکیل می دهند آورده نشده است. تمام مطالب این بخش از لغت نامه دهخدا، ماده «ا»، نقل شده است؛ مگر آن که از منبع دیگری نام برده شده باشد»
 لان با کسب ۷ عنوان قهرمانی در لیگ قهرمانان اروپا پر افتخارترین تیم ایتالیایی در اروپا و بعد از رئال مادرید دومین تیم پرافتخار این رقابت ها شناخته می شود (ssotfat'tsjo:ne 'kaltfo 'mi:lan). از پرافتخارترین تیم های اروپاست و نسبت به رقبای داخلی اینتر میلان و یوونتوس، در رتبه بالاتری قرار دارد شده است. آت میلان اولین عنوان قهرمانی خود در جام قهرمانی ایتالیا را در سال ۱۹۰۱ و دو قهرمانی ۱۹۰۶ و ۱۹۰۷ میلان تا فصل (۵-۱۹۵۰) موفق به کسب قهرمانی در ایتالیا نشد. در دهه ۵۰، میلان به اوج فوتبال ایتالیا با هدایت ملثت سوندی گره-نولی که متشکل از گونار گرن، گونار نوردال و نیلس لیدهولم، بازگشت قرار داد با پائولو مالدینی بود. در سال ۱۹۸۶، سیلیو برلوسکونی، باشگاه را خریداری کرد و یک سال بعد با به خدمت گیری آریگو ساکی به عنوان سرمربی، و بازیکنانی همچون مارکو فان باستن، رود گولیت و ف ۰ ن را بدون شکست قهرمان ایتالیا کرد. میلان دو فصل بعد را هم با قهرمانی ایتالیا پشت سر گذاشت. کاپو در اروپا هم اقتدار ساکی را ادامه داد و همراه میلان، در ۳ فینال پیاپی لیگ قهرمانان/جام باشگاه ها حاضر شد و، آن هم در ورزشگاه خانگی، بدترین نتیجه دوران برلوسکونی است. میلان در اولین دهه قرن، یکی از قدرتمندترین باشگاه های اروپا و جهان بود. در فاصله سال های ۲۰۰۲ تا ۲۰۰۷ سه بار به فینال اروپا راه یافت شروع فصل ۰۷-۲۰۰۶ و عدم حضور این تیم در لیگ قهرمانان اروپا را صادر کرد که این حکم بعد از بازنگری، به کسر هشت امتیاز و حضور در مرحله پلی آف لیگ قهرمانان اروپا، کاهش پیدا کرد. سال ۲۰۰۷ یکی اضر شد و در فینال بوکا جونیورز را با نتیجه ۳-۲ شکست داد و برای نخستین بار این جام معتبر بین المللی را تحت نام جدیدش فتح کرد. میلان که فصل ۰۸-۲۰۰۷ را با پینتر از سطح انتظار به اتمام رسانده بود، در نبال ایتالیا و اروپا بود. کاکا نیز در پایان فصل با قراردادی به ارزش ۶۸٫۵ میلیون یورو به رئال مادرید پیوست. همچنین کارلو آنچلوتی سرمربی میلان پس از ۸ فصل هدایت باشگاه و کسب ۸ جام، راهی چلسی شد. د برینه پیوونتوس ترک گفت. کوچ پیرلو سرآغاز تغییر نسل طلایی میلان بود که در طول ده سال، جام های معتبر داخلی و بین المللی را فتح کرده و میلان را به جایگاه پرافتخارترین باشگاه در سطح بین المللی رسانده بودند به مه آتزا تغییر نام داد. در سال ۱۹۸۹ این ورزشگاه برای نوسازی مدت کوتاهی تعطیل شد و پس از بازگشایی ظرفیت آن به حدود ۸۳ هزار نفر رسید. بسیاری از بازی های ایتالیا در همین ورزشگاه برگزار می شود ، بین تنها یکبار موفق به فتح جام شده است. ن گروه های هواداری اولترای ایتالیا، در شهر میلان مستقر است و از میلان حمایت می کند. در حال حاضر بریکت روسونره اصلی ترین گروه اولترای حامی میلان است ساب می آید. میلان در طول تاریخ خود از سال ۱۸۹۹ تاکنون، رؤسای گوناگونی داشته است. برخی از این روسا هم زمان مالک باشگاه هم بوده اند، درحالی که تعدادی از آن ها ریاست افتخاری باشگاه را برعهده داشته اند مین توپ طلای میلان را در سال ۲۰۰۴ کسب کرد. جدی ترین رقبای او دکو و رونالدینیو بودند (balz daz): از گرفت. جانی ریورا، اولین بازیکن میلان و دومین ایتالیایی برنده توپ طلای اروپا (تلفظ فرانسوی رد. طولانی ترین دوران مربیگری میلان با ۴۵۰ بازی، در اختیار اوست. سیلیو برلوسکونی، طولانی ترین دوران ریاست باشگاه را با ۳۱ سال ریاست در اختیار دارد. میلان پس از رئال مادرید، بیشترین جام بین الملل اشگاه در برابر مودنا بدست آمده است. میلان در یکی از دیدارهای فصل ۱۵-۱۹۱۴، مودنا را با نتیجه ۱۳-۰ شکست داده است. بولونیا در فصل ۲۳-۱۹۲۲، سنگین ترین شکست را با ۸ گل به میلان تحویل کرده است ود در سال ۱۹۹۶ با این شرکت کاملاً یکبارجه شد و تحت نام دایملر-بنز آ.گ. در آمدند، بخش های باقی مانده آ.گ. به دایملر-کرایسلر-هوافضا که از شرکت های وابسته دایملر-بنز بود و اندرزن الحاق شدند (henau) ین زمان تولید شد، شروع سابقه ای طولانی در امر تأمین تجهیزات الکتریکی حمل و نقل ریلی آلمان شد .\n لیه هویت سازمانی این شرکت با محصولات و به اشتراک گذاری تبلیغات ویزگهای طراحی مشترک گشت رسمی یکی از پرافتخارترین تیم های اروپاست و نسبت به رقبای داخلی اینتر میلان و یوونتوس، در رتبه بالاتری قرار دارد .\n ایتالیایی در اروپا و بعد از رئال مادرید دومین تیم پرافتخار این رقابت ها شناخته می شود ن اتفاقات میلان تا فصل (۵۱-۱۹۵۰) موفق به کسب قهرمانی در ایتالیا نشد. در دهه ۵۰، میلان به اوج فوتبال ایتالیا با هدایت ملثت سوندی گره-نولی که متشکل از گونار گرن، گونار نوردال و نیلس لیدهولم، بازگش . ت ازنشستکی جیانی ریورا، میلان دوران افت را تجربه کرد. دورانی که ماجرای توننرو یا همان شرط بندی و پرداخت رشوه از سوی باشگاه برای تغییر نتیجه مسابقات در سال ۱۹۸۰ و مجازات حضور در س .\n تانی سوپر جام اروپا و ۳ قهرمانی جام بین فاره ای). وی در این دوره یک نایب قهرمانی در سری آ و یک نایب قهرمانی در کوپا ایتالیا را نیز در کارنامه خود دارد. پس از او فابیو کاپو، هدایت میلان را برعهده گرفت و ن را بدون شکست قهرمان ایتالیا کرد. میلان دو فصل بعد را هم با قهرمانی ایتالیا پشت سر گذاشت. کاپو در اروپا هم اقتدار ساکی را ادامه داد و همراه میلان، در ۳ فینال پیاپی لیگ قهرمانان/جام باشگاه ها حاضر شد به سود میلان به پایان برسد. آقا، در نیمه دوم ورق برگشت و لیورپول موفق شد در مدت زمان ۶ دقیقه، نتیجه را به تساوی بکشاند. استیون جرارد، ولادیمیر اسمیچر و ژابی آونسو سه گل پیاپی به ثمر رساندند تا نهایتاً لان را با نتیجه ۳ بر ۲ شکست دهد. اما این پایان راه نبود و در شب درخشش کاکا، میلان بازی برگشت را با نتیجه ۳ بر ۰ به سود خود به پایان برد و ضمن جلوگیری از فینال تمام انگلیسی، زمینه انتقام گیری از لیور حمت گرفتند. همچنین جیانلوکا زامبروتا از بارسلونا، متیو فلامینینی از آرسنال و مارکو بوریلو از جنوا دیگر بازیکنانی بودند که در بازار تابستانی جذب باشگاه شدند. در بازار نقل و انتقالات زمستانی دیوید بکام ستاره ا ن این فصل آندره آ پیرلو پس از ده سال میلان را به قصد پیوستن به رقیب دیرینه پیوونتوس ترک گفت. کوچ پیرلو سرآغاز تغییر نسل طلایی میلان بود که در طول ده سال، جام های معتبر داخلی و بین المللی را فتح کرده گی، جیانلوکا زامبروتا، جئارو گنوسو، مارکو فن بومل، کلارنس سیدورف، تیاگو سیلوا، زلاتان ابراهیموویچ و تغییر کامل نسل طلایی بود .\n و میلان را به جایگاه پرافتخارترین باشگاه در سطح بین المللی رسانده بودند

بعد از این کار زمان *normalize* کردن این دیتا می رسد. در فایل *data_cleaner.py*، از کتابخانه *Hazm* استفاده می کنیم که کتابخانه ای است تقریباً مشابه *nlTK* که برای پردازش زبان فارسی کاربرد دارد و *documentation* آن از این لینک <https://www.roshan-ai.ir/hazm/docs> قابل مشاهده است. ابتدا با دستور *pip install hazm*، این کتابخانه را نصب می کنیم، سپس با استفاده از توابع *normalizer* و *lemmatizer*، داده هایی که در *raw_dataset.json* ذخیره کرده بودیم را نرمالایز می کنیم.

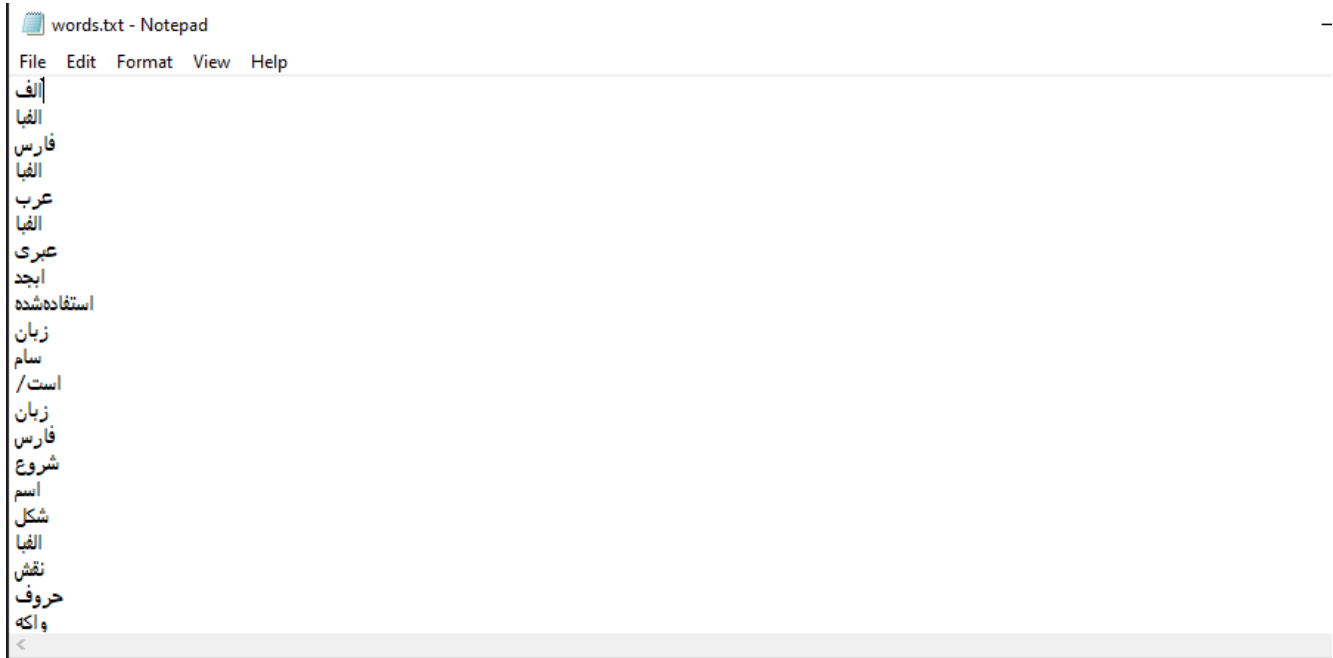
```

for row in raw_dataset:
    cleaned_text = normalizer.normalize(row.get('text'))
    cleaned_text_data.append({
        'id': row.get('id'),
        'text': cleaned_text,
        'subject': row.get('title')
    })

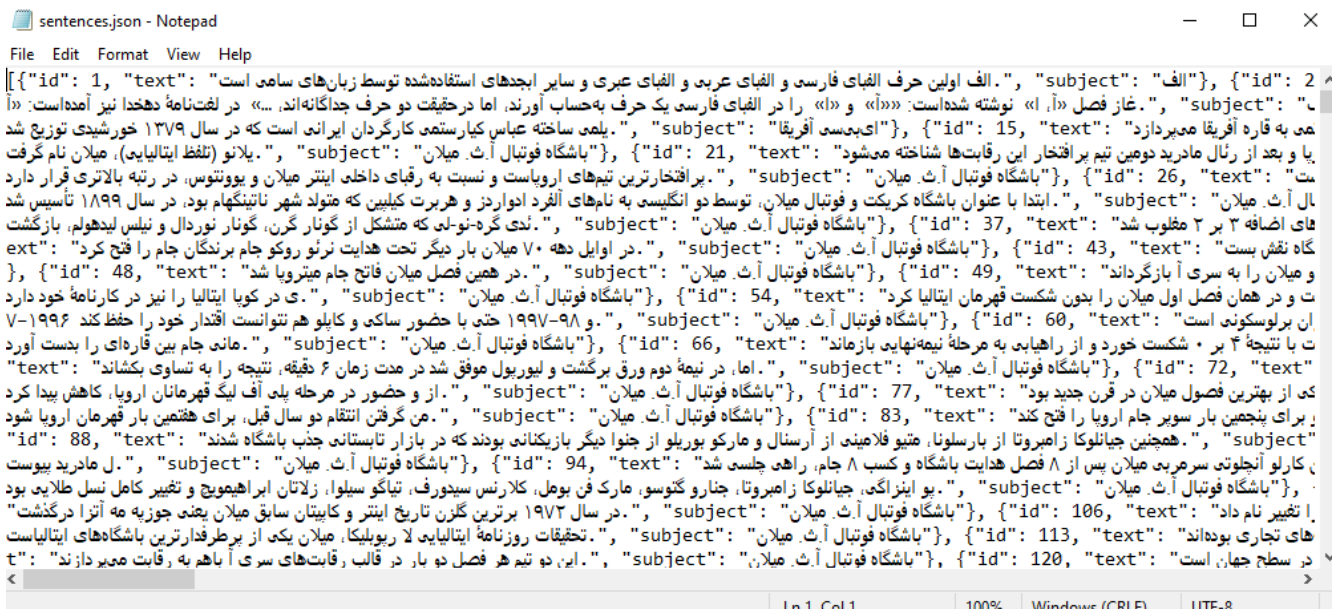
for sentence in sent_tokenize(cleaned_text):
    sentences_data.append({
        'id': sentence_counter,
        'text': sentence,
        'subject': row.get('title')
    })
    sentence_counter += 1
for word in word_tokenize(cleaned_text):
    new_word = lemmatizer.lemmatize(word)
    if new_word not in stop_words:
        if '#' in new_word:
            new_word = new_word.replace('#', '/')
        if new_word in counts:
            counts[new_word] += 1
        else:
            counts[new_word] = 1

```

سپس با *stemming* ریشه اصلی آن ها را به دست می آوریم و در نهایت چک می کنیم که این کلمات بین *stop words* های زبان فارسی هستند یا خیر و اگر در فایل *stop words* نبود، آن را در لیست کلمات در فایل *words.txt* می نویسیم که در شکل زیر این فایل را می بینیم.



لیست *stop words* ها در پوشه *raw* در فایل *stopwords-fa.txt* قرار دارد. سپس هر جمله که توسط این توابع جدا شده اند را در *sentences.json*، همراه با آیدی و موضوع آن می نویسیم. فرمت ذخیره شده را در شکل زیر می بینیم.



همچنین برای این که هر کلمه را با تعداد تکرارش داشته باشیم، یک دیکشنری تعریف می کنیم و هر بار که کلمه ای تکرار شد یکی به تعداد تکرار آن اضافه می کنیم و در نهایت آن را پس از *sort* کردن بر اساس تکرار بیشتر، در فایل *count.txt* ذخیره می کنیم.

count.txt - Notepad

File Edit Format View Help

اشد/شو : 8162
است : 8139/
کرد/کن : 7173
بود/باش : 5650
سال : 4763
داشت/دار : 3722
ایران : 2665
داد/ده : 1914
شده/است : 1632
گرفت/گیر : 1587
شهر : 1194
هست : 1177/
کار : 1149
کشور : 1121
انگلیسی : 1086
توانست/توان : 1052
تاریخ : 1042
گفت/گو : 1017
کتاب : 932
بن : 916
زمین : 882
دست : 807
یافت/یاب : 805
رفت/رو : 766
آمد/آ : 766
جهان : 741
گروه : 734
شرکت : 722
نظر : 714
آلمان : 711
آب : 710
تولید : 690
می‌توان : 670
دوران : 659
گشت/گرد : 657
علی : 657

<

برای این که براساس تعداد مرتب کنیم، از قطعه کد زیر استفاده می کنیم:

```

words_count = open('../data/word_count/count.txt', 'w', encoding='utf8')

for i in sorted(counts, key=counts.get, reverse=True):
    words_count.write(i + " : "+str(counts[i])+"\n")

cleaned_dataset_file.write(json.dumps(cleaned_text_data, ensure_ascii=False))
sentence_broken_file.write(json.dumps(sentences_data, ensure_ascii=False))

```

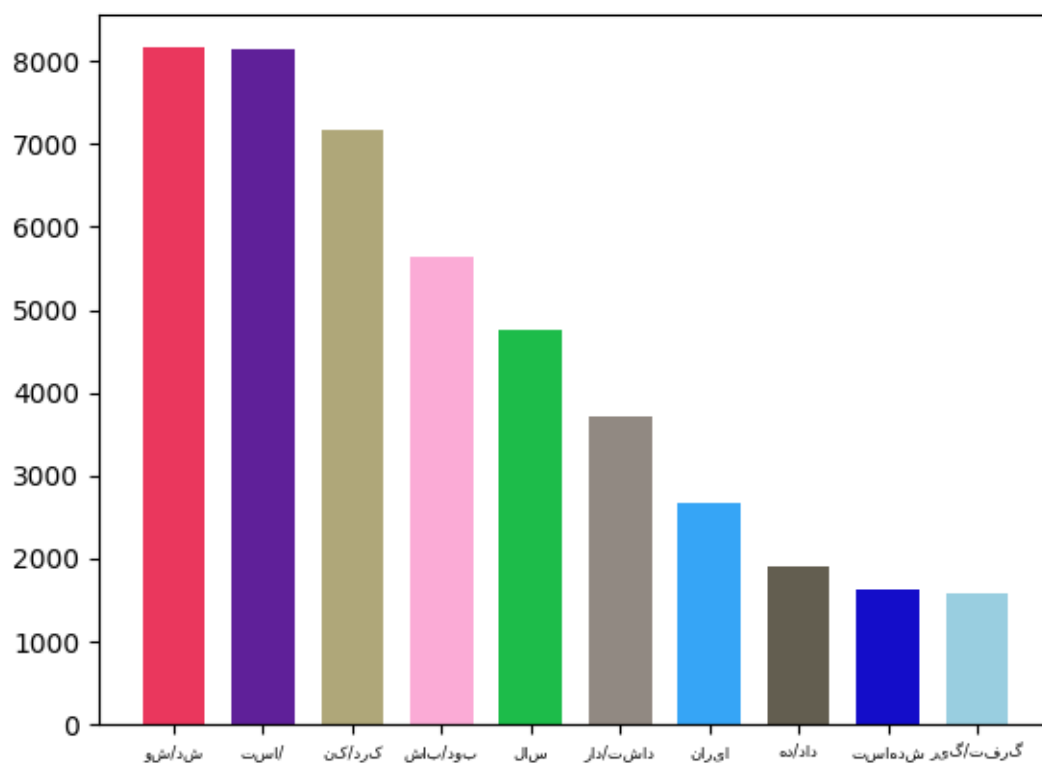
حال که کلماتی که بیشترین تکرار را دارند پیدا کرده ایم، ده تا از آن ها که بیشترین تکرار را داشته اند در یک نمودار نمایش می دهیم. در فایل *plot.py* ، ما با خواندن فایل *count.txt* و انتخاب ده مورد اول آن نمودار کلمات و تعداد تکرار آن ها را با رنگ های *random* رسم می کنیم و عکس نمودار را در پوشه *plot_img* ذخیره می کنیم.

```

1 from __future__ import unicode_literals
2
3 import matplotlib.pyplot as plt
4 import numpy as np
5 import random
6 from bidi.algorithm import get_display
7 wordcount = open('../data/word_count/count.txt', 'r', encoding='utf8').read().splitlines()
8 x = []
9 y = []
10 for i in range(10):
11     row = wordcount[i].split(' : ')
12     y.append(int(row[1]))
13     x.append(get_display(row[0]))
14
15 rnd = []
16 for i in range(10):
17     r = random.random()
18     b = random.random()
19     g = random.random()
20     rnd.append((r, g, b))
21 plt.bar(x, y, width=0.7, bottom=None, align='center', data=None, color=_rnd_)
22 plt.xticks(fontsize=6)
23 |
24 plt.savefig('../data/plot_img/plot.png')
25

```

در نهایت در شکل زیر عکس نمودار را مشاهده می کنیم.



در انتها فایل *script.py*، همه توابع موجود برای گرفتن داده و نرمالایز کردن را انجام می دهد که کد آن در زیر نشان داده شده است.

```

1  import data_cleaner
2  import collect_alphabets
3  import collect_hrefs
4  import collect_articles
5
6  if __name__ == "__main__":
7      collect_alphabets.func()
8      collect_hrefs.func()
9      collect_articles.func()
10     data_cleaner.func()
11

```