Yıldız Teknik Üniversitesi
Bilgisayar Mühendisliği Bölümü

Doğal Dil İşleme Dersi Ödev-1

Mert TÜRKMENOĞLU

17011005

KULLANILAN REGEX:

/([Ss][Oo][Kk]([\.\s])*)|([Ss][Oo][Kk][Aa][Kk][\.\s])|([Ss][Kk][\.\s])|([Bb][Uu][Ll][Vv][Aa][Rr][Iı])|([Bb][Uu][Ll][Vv]([\.\s])*)|((\s)+[Bb][Ll][Vv]([\.\s])+)|([Mm][Aa][Hh]([\.\s])+)|([Mm][Hh]([\.\s])+)|(\s)+([Mm][\.\s])|([Cc][Aa][Dd][Dd][Ee][Ss][İiIı]([\.\s])*)|([Cc][Aa][Dd]([\.\s])*)|([Cc][Dd][\.\s])|([Cc][\.\s])|([Aa][Pp][Aa][Rr][Tt][Mm][Aa][Nn][Iı][\.\s])|([Aa][Pp][Tt][\.\s])|[Nn][Oo][:\.\s]| (\/\s)/gm

Yukarıda verilen RegEx'i kullanarak input dosyasındaki her bir satırı ayırıp JSON formatında kaydeden Python kodu:

```
import re
import json
import itertools


street_regex = r'([Ss][Oo][Kk]([\.\s])*)|([Ss][Oo][Kk][Aa][Kk][\.\s])|([Ss][Kk][\.\s])'
boulevard_regex = r'([Bb][Uu][Ll][Vv][Aa][Rr][Iı])|([Bb][Uu][Ll][Vv]([\.\s])*)|((\s)+[Bb][Ll][Vv]([\.
\s])+)'
district_regex = r'([Mm][Aa][Hh]([\.\s])+)|([Mm][Hh]([\.\s])+)|(\s)+([Mm][\.\s])'
avenue_regex = r'([Cc][Aa][Dd][Dd][Ee][Ss][İiIı]([\.\s])*)|([Cc][Aa][Dd]([\.\s])*)|([Cc][Dd][\.\s])|
([Cc][\.\s])'
apt_regex = r'([Aa][Pp][Aa][Rr][Tt][Mm][Aa][Nn][Iı][\.\s])|([Aa][Pp][Tt][\.\s])'
no_regex = r'[Nn][Oo][:\.\s]'
my_regex = '|'.join([street_regex, boulevard_regex, district_regex, avenue_regex, no_regex, apt_regex,
r'(\/\s)'])

# CHANGE TO YOUR FILE NAME
INPUT_FILE_NAME = 'adres.txt'

label_regex_dict = {
    'street': street_regex,
    'boulevard': boulevard_regex,
    'district': district_regex,
    'avenue': avenue_regex,
    'apartment': apt_regex
}

invalid = [None, '.', '/', '/ ', ' ']
```

```python
def grouper(L, n):
    args = [iter(L)] * n
    return ([e for e in t if e != None] for t in itertools.zip_longest(*args))


def get_lines(f_name):
    res = []

    with open(f_name, 'r') as f:
        for line in f:
            res.append([e.strip() for e in re.split(my_regex, line[:-1]) if not e in invalid])

    return res


def make_dict_from_line(r):
    d = {}
    splitted = re.split(r' ', r[-2])

    for e in grouper(r[:-2], 2):
        if e[0] != '' and e[0] != ' ':
            label = ([ l for l, regex in label_regex_dict.items() if re.search(regex, " ".join(e)) != None] + ['other'])[0]
            d[label] = e[0]

    no_or_desc = " ".join(splitted[:-1]).strip()
    if no_or_desc != '':
        d['no' if re.search(r'[0-9]', no_or_desc) != None else 'desc'] = no_or_desc

    d['county'] = splitted[-1].strip()
    d['province'] = r[-1]

    return d


def write_to_file(f_name, data):
    with open(f_name, 'w') as f:
        f.write(json.dumps({'data': data}, indent=2, ensure_ascii=False))


lines = get_lines(INPUT_FILE_NAME)
result = [make_dict_from_line(line) for line in lines]
write_to_file('output.json', result)
```

output.json dosyasının bir kısmı:

```
{
  "data": [
    {
      "desc": "YENİBOSNA  METRO İSTASYONU",
      "county": "BAKIRKÖY",
      "province": "İSTANBUL"
    },
    {
      "avenue": "KENNEDY",
      "desc": "SİRKECİ ARABALI VAPUR İSKELESİ",
      "county": "FATİH",
      "province": "İSTANBUL"
    },
    {
      "district": "YAVUZTÜRK",
      "avenue": "KARADENİZ",
      "no": "2",
      "county": "ÜSKÜDAR",
      "province": "İSTANBUL"
    },
    {
      "district": "HAMİDİYE",
      "street": "ALPEREN",
      "no": "15/2",
      "county": "ÇEKMEKÖY",
      "province": "İSTANBUL"
    },
    {
      "district": "UĞUR MUMCU",
      "avenue": "YUNUS EMRE",
      "no": "25",
      "county": "KARTAL",
      "province": "İSTANBUL"
    },
    {
      "other": "BAĞLARBAŞI",
      "avenue": "İNÖNÜ",
      "no": "3",
      "county": "MALTEPE",
      "province": "İSTANBUL"
    },
  ]
}
```

Hata oluşan durumlara örnekler:

- MIGROS ALIŞVERİŞ MERKEZİ E5 KARAYOLU ÜZERİ GIRIS KAT BEYLİKDÜZÜ/ İSTANBUL

- CUMHURİYET MAH. ŞEHİTLER CAD. BEYLİKDÜZÜ BULVAR EVLERİ ÇARŞISI NO:134 ESENYURT/ İSTANBUL

- MEHTERÇEŞME MAH. CUMH.CAD 1810 SOK. NO:1 ESENYURT/ İSTANBUL

Bu örneklerin JSON dosyasından alınıp birleştirilmiş hali:


```
{
  "data": [
    {
      "no": "MIGROS ALIŞVERİŞ MERKEZİ E5 KARAYOLU ÜZERİ GIRIS
KAT", "county": "BEYLİKDÜZÜ",
      "province": "İSTANBUL"
    },
    {
      "district": "CUMHURİYET",
      "avenue": "ŞEHİTLER",
      "boulevard": "BEYLİKDÜZÜ",
      "other": "AR EVLERİ ÇARŞISI",
      "no": "134",
      "county": "ESENYURT",
      "province": "İSTANBUL"
    },
    {
      "district": "CU",
      "street": "1810",
      "no": "1",
      "county": "ESENYURT",
      "province": "İSTANBUL"
    }
  ]
}
```

Sonuç:

- RegEx, düzgün formatta verilen adreslerin tamamını tanımıştır.
- Yazımında farklılık / yanlışlık bulunan adreslerde bunları “other” kategorisinde değerlendirmiştir.
- Uç noktalara ilişkin hatalı örnekler yukarıda belirtilmiştir.
- Tamamiyle başarısız olduğu (adresin her parçasının yanlış bulunduğu) hiçbir veri bulunmamaktadır.

Toplam veri sayısı: 6831

Bulunan adres sayısı: 6831

Bulunamayan adres sayısı: 0

Hatalı bölümlenen adres sayısı: 590

Başarı oranı: %91.363