

TF-IDF Implementation Report

Tuan Hung Nguyen

Swinburne University of Technology

27/10/2023

Executive Summary

This Report aim to explain step-by-step the implementation of the TF-IDF computing in the Legal Textual Entailment Recognition task of VLSP. As all the instructions are informed in the previous report, the implementation will have a small change in the actual code.

Pre-Processing Code Implementation of TF-IDF

Tokenizing Text into Words

In Vietnamese, the construction of words is different because a definition of words in vietnamese could be single word or compound words. Therefore, I decided to use an external library of vietnamese word-tokenize which is “underthesea”.

```
In [4]: from underthesea import word_tokenize
import re
job_1 = {}
for i, row in enumerate(df.articles):
    for text_data in row:
        text = text_data["text"]

        tokens = word_tokenize(text)

        for token in tokens:
```

Figure 1. Underthesea library

Stopwords Removing

While Stopwords are as known as the words that does not majorly contribute to the meaning of the sentence evenly when they are being removed. After tokenzing, each token would be compared and removed in case of matching with the Vietnamese stopwords list.

```
stopwords_file = open("stopwords.txt", "r", encoding="utf-8")
stopwords_list = []

for line in stopwords_file:
    stopwords_list.append(line.strip())

stopwords_file.close()
```

Figure 2. Stopwords list importing

Before comparing with stopwords list, token is also needed to be cleaned from special character and removing space in head and tail.

```

for token in tokens:
    token = re.sub(r'[_!@#]', '', token).lower().strip()
    Check if the token is a number (digits only) or a word

    if not re.search(r'^\w\s\d]+', token):
        if not token in stopwords_list:

```

Figure 3. Filtering and cleaning token

TF-IDF Calculating

Job 1: Compute Term-Frequency (TF)

After tokenizing text, a key including term and document name that contain that term would be pass in a hash map with the value is the count of the term.

```

from underthesea import word_tokenize
import re
job_1 = {}
for i, row in enumerate(df.articles):
    for text_data in row:
        text = text_data["text"]

        tokens = word_tokenize(text)

        for token in tokens:
            token = re.sub(r'[_!@#]', '', token).lower().strip()
            # Check if the token is a number (digits only) or a word

            if not re.search(r'^\w\s\d]+', token):
                if not token in stopwords_list:
                    key = (token, doc_dict[i])

                    if job_1.get(key):
                        job_1[key] += 1
                    else:
                        job_1[key] = 1

print(job_1)

```

Figure 4. Job 1 full code

The output would be look like below:

```
{('phạm vi', 'Luật Viên chức 2010'): 5, ('điều chỉnh', 'Luật Viên chức 2010'): 1, ('luật', 'Luật Viên chức 2010'): 23, ('quy
định', 'Luật Viên chức 2010'): 108, ('viên chức', 'Luật Viên chức 2010'): 232, ('quyền', 'Luật Viên chức 2010'): 26, ('nghĩa
vụ', 'Luật Viên chức 2010'): 13, ('tuyển dụng', 'Luật Viên chức 2010'): 30, ('quản lý', 'Luật Viên chức 2010'): 96, ('sự nghi
ệp', 'Luật Viên chức 2010'): 117, ('công lập', 'Luật Viên chức 2010'): 113, ('công dân', 'Luật Viên chức 2010'): 1, ('việt na
m', 'Luật Viên chức 2010'): 3, ('việc làm', 'Luật Viên chức 2010'): 37, ('làm việc', 'Luật Viên chức 2010'): 93, ('chế độ',
'Luật Viên chức 2010'): 25, ('hợp đồng', 'Luật Viên chức 2010'): 73, ('huởng', 'Luật Viên chức 2010'): 22, ('lương', 'Luật Vi
ên chức 2010'): 16, ('quỹ', 'Luật Viên chức 2010'): 2, ('pháp luật', 'Luật Viên chức 2010'): 54, ('giải thích', 'Luật Viên ch
ức 2010'): 1, ('từ ngữ', 'Luật Viên chức 2010'): 2, ('1', 'Luật Viên chức 2010'): 62, ('bổ nhiệm', 'Luật Viên chức 2010'): 3
6, ('chức vụ', 'Luật Viên chức 2010'): 23, ('thời hạn', 'Luật Viên chức 2010'): 40, ('trách nhiệm', 'Luật Viên chức 2010'): 3
3, ('điều hành', 'Luật Viên chức 2010'): 4, ('tổ chức', 'Luật Viên chức 2010'): 31, ('công việc', 'Luật Viên chức 2010'): 20,
('công chức', 'Luật Viên chức 2010'): 13, ('phụ cấp', 'Luật Viên chức 2010'): 4, ('2', 'Luật Viên chức 2010'): 52, ('đạo đứ
c', 'Luật Viên chức 2010'): 8, ('nghề nghiệp', 'Luật Viên chức 2010'): 65, ('chuẩn mực', 'Luật Viên chức 2010'): 2, ('nhận th
ức', 'Luật Viên chức 2010'): 1, ('hành vi', 'Luật Viên chức 2010'): 10, ('đặc thù', 'Luật Viên chức 2010'): 5, ('lĩnh vực',
'Luật Viên chức 2010'): 18, ('hoạt động', 'Luật Viên chức 2010'): 40, ('cơ quan', 'Luật Viên chức 2010'): 30, ('thẩm quyền',
'Luật Viên chức 2010'): 34, ('3', 'Luật Viên chức 2010'): 39, ('quy tắc', 'Luật Viên chức 2010'): 5, ('ứng xử', 'Luật Viên ch
ức 2010'): 4, ('xử sự', 'Luật Viên chức 2010'): 1, ('thi hành', 'Luật Viên chức 2010'): 6, ('nhiệm vụ', 'Luật Viên chức 201
0'): 35, ('quan hệ', 'Luật Viên chức 2010'): 2, ('xã hội', 'Luật Viên chức 2010'): 13, ('nhà nước', 'Luật Viên chức 2010'): 1
2, ('ban hành', 'Luật Viên chức 2010'): 3, ('công khai', 'Luật Viên chức 2010'): 5, ('nhân dân', 'Luật Viên chức 2010'): 11,
('giám sát', 'Luật Viên chức 2010'): 2, ('chấp hành', 'Luật Viên chức 2010'): 6, ('4', 'Luật Viên chức 2010'): 26, ('lựa chọ
```

Job 2: Compute Document Inverted Frequency (IDF)

Figure 5. Job 1 output

In job 2, it would be divided into 2 smaller job including job 2 mapper and job 2 reducer.

Job 2 Mapper:

The input key which is belong to job 1 would be re-called in job 2 mapper to get tf. From here, the key of the hash map in job 2 mapper would be terms, and the output is doc_id, tf of the term in key, and 1 (Sums 1s to compute the number of documents containing term).

```
In [5]: job_2_mapper = {}
for term, doc_id in job_1:
    input_key = (term, doc_id)
    output_key = (doc_id, job_1[input_key], 1)
    if job_2_mapper.get(term):
        job_2_mapper[term].append(output_key)
    else:
        job_2_mapper[term] = [output_key]
```

Figure 6. Job 2 mapper full code

Job 2 Reducer:

After mapper, we sum 1s to get n (number of documents containing term) in job 2 reducer, then we will take N (the total number of document) divided for n (which is known above) and we get the final IDF.

Figure 7. Job 2 reducer full code

'[phạm vi': 1.0, 'điều chỉnh': 1.0, 'luật': 1.0, 'quy định': 1.0, 'viên chức': 1.2857142857142858, 'quyền': 1.0, 'nghĩa vụ': 1.125, 'tuyên dụng': 3.0, 'quản lý': 1.0, 'sự nghiệp': 2.0, 'cống lập': 2.5714285714285716, 'cống dân': 1.125, 'viết mẫu': 1.0, 'việc làm': 2.5714285714285716, 'lâm việc': 1.125, 'chế độ': 1.2857142857142858, 'hợp đồng': 2.0, 'huống': 1.3846153846153846, 'lương': 3.6, 'quy': 2.0, 'pháp luật': 2.0, 'giải thích': 1.0588235294117647, 'từ ngữ': 1.2857142857142858, '1': 1.0, 'b': 0, 'nhiệm': 2.0, 'chức vụ': 3.0, 'thời hạn': 1.0, 'trách nhiệm': 1.0, 'điều hành': 1.6363636363636365, 'tổ chức': 1.0, 'cống vi': 1.6363636363636365, 'cống chức': 1.3846153846153846, 'phụ cấp': 4.5, '2': 1.0, 'đạo đức': 1.2857142857142858, 'nghề nghiệp': 1.5, 'chuẩn mực': 3.0, 'nhân thức': 1.3846153846153846, 'hành vi': 1.0, 'đặc thù': 3.0, 'lĩnh vực': 1.3846153846153846, 'hoạt động': 1.0, 'cơ quan': 1.0, 'thẩm quyền': 1.0, '3': 1.0, 'quy tắc': 2.0, 'ứng xử': 2.0, 'xử sự': 6.0, 'thành': 1.0, 'nhiệm vụ': 1.0588235294117647, 'quan hệ': 1.5, 'xã hội': 1.0, 'nhà nước': 1.0, 'ban hành': 1.125, 'cống khai': 1.3846153846153846, 'nhân dân': 1.125, 'giám sát': 1.2, 'chấp hành': 1.3846153846153846, '4': 1.0, 'lựa chọn': 1.2857142857142858, 'phẩm chất': 3.6, 'trình độ': 1.6363636363636365, 'năng lực': 1.125, '5': 1.0, 'thỏa thuận': 1.125, 'văn bản': 1.0, 'đứng': 1.2857142857142858, 'đầu': 1.125, 'tiền lương': 4.5, 'đãi ngộ': 4.5, 'kỹ năng': 2.0, 'chuyên môn': 1.3846153846153846, 'nghệ thuật': 1.3846153846153846, 'nguyên tắc': 1.0, 'tuần thủ': 1.2857142857142858, 'tận tụy': 6.0, 'phục vụ': 1.2, 'quy trình': 1.6363636363636365, 'thành tra': 1.125, 'kiểm tra': 1.0588235294117647, 'bảo đảm': 1.0, 'lãnh đạo': 2.25, 'đang công sản vật nhân': 4.5, 'thống nhất': 1.0, 'chủ động': 1.8, 'đề cao': 6.0, 'cơ sở': 1.0588235294117647, 'tiêu chuẩn': 1.5, 'chức danh': 3.6, 'cán bộ': 1.2857142857142858, 'bình đẳng': 1.3846153846153846, 'giới': 2.0, 'chính sách': 1.0588235294117647, 'vụ đài': 1.6363636363636365, 'tài năng': 4.5, 'dân tộc thiểu số': 2.0, 'cống': 2.5714285714285716, 'cách mạng': 3.0, 'miền': 1.3846153846153846, 'núi': 1.8, 'biên giới': 2.25, 'hải đảo': 1.5, 'vùng sâu': 4.5, 'vùng xa': 4.5, 'kinh tế': 1.0588235294117647, 'gần': 3.0, 'tư tưởng': 2.25, 'xác định': 1.2, 'số lượng': 1.5, 'cơ cấu': 2.25, 'chính phủ': 1.0, 'phương pháp': 2.5714285714285716, 'trình

Job 3: Compute TF-IDF

```
In [7]: job_3 = {}
        for term, doc_id in job_1:
            input_key = (term, doc_id)
            job_3[input_key] = job_2_reducer[term] * job_1[input_key]
        print(job_3)
```

The output would look like this:

```
{('phạm vi', 'Luật Viên chức 2010'): 5.0, ('điều chỉnh', 'Luật Viên chức 2010'): 1.0, ('luật', 'Luật Viên chức 2010'): 23.0, ('quy định', 'Luật Viên chức 2010'): 108.0, ('viên chức', 'Luật Viên chức 2010'): 298.28571428571433, ('quyền', 'Luật Viên chức 2010'): 26.0, ('nghĩa vụ', 'Luật Viên chức 2010'): 14.625, ('tuyển dụng', 'Luật Viên chức 2010'): 90.0, ('quản lý', 'Luật Viên chức 2010'): 96.0, ('sự nghiệp', 'Luật Viên chức 2010'): 234.0, ('công lập', 'Luật Viên chức 2010'): 290.5714285714286, ('công dân', 'Luật Viên chức 2010'): 1.125, ('việt nam', 'Luật Viên chức 2010'): 3.0, ('việc làm', 'Luật Viên chức 2010'): 95.14285714285715, ('làm việc', 'Luật Viên chức 2010'): 104.625, ('chế độ', 'Luật Viên chức 2010'): 32.142857142857146, ('hợp đồng', 'Luật Viên chức 2010'): 146.0, ('hưởng', 'Luật Viên chức 2010'): 30.46153846153846, ('lương', 'Luật Viên chức 2010'): 57.6, ('quy', 'Luật Viên chức 2010'): 4.0, ('pháp luật', 'Luật Viên chức 2010'): 54.0, ('giải thích', 'Luật Viên chức 2010'): 1.0588235294117647, ('từ ngữ', 'Luật Viên chức 2010'): 2.5714285714285716, ('1', 'Luật Viên chức 2010'): 62.0, ('bổ nhiệm', 'Luật Viên chức 2010'): 72.0, ('chức vụ', 'Luật Viên chức 2010'): 69.0, ('thời hạn', 'Luật Viên chức 2010'): 40.0, ('trách nhiệm', 'Luật Viên chức 2010'): 33.0, ('điều hành', 'Luật Viên chức 2010'): 6.545454545454546, ('tổ chức', 'Luật Viên chức 2010'): 31.0, ('công việc', 'Luật Viên chức 2010'): 32.72727272727273, ('công chức', 'Luật Viên chức 2010'): 18.0, ('phụ cấp', 'Luật Viên chức 2010'): 18.0, ('2', 'Luật Viên chức 2010'): 52.0, ('đạo đức', 'Luật Viên chức 2010'): 10.285714285714286, ('nghề nghiệp', 'Luật Viên chức 2010'): 97.5, ('chuẩn mực', 'Luật Viên chức 2010'): 6.0, ('nhận thức', 'Luật Viên chức 2010'): 1.3846153846153846, ('hành vi', 'Luật Viên chức 2010'): 10.0, ('đặc thù', 'Luật Viên chức 2010'): 15.0, ('lĩnh vực', 'Luật Viên chức 2010'): 24.923076923076923, ('hoạt động', 'Luật Viên chức 2010'): 40.0, ('cơ quan', 'Luật Viên chức 2010'): 30.0, ('tầm quyền', 'Luật Viên chức 2010'): 34.0, ('3', 'Luật Viên chức 2010'): 39.0, ('quy tắc', 'Luật Viên chức 2010'): 10.0, ('ứng xử', 'Luật Viên chức 2010'): 8.0, ('xử sự', 'Luật Viên chức 2010'): 6.0, ('thi hành', 'Luật Viên chức 2010'): 6.0, ('nhiệm vụ', 'Luật Viên chức 2010'): 37.05882352941177, ('quan hệ', 'Luật Viên chức 2010'): 3.0, ('xã hội', 'Luật Viên chức 2010'): 1.0}
```

Figure 10. Job 3 output

Discussion

Above is my final implementation to calculating the TF-IDF for the task of checking the legalness of a passage which deployed by the hadoop abstract code. In term of enhancing the algorithm to calculating TF-IDF, I think it could be enhanced by integrate the job 2 mapper and reducer together to get a cleaner code.