NHÓM 12

Danh sách thành viên:

Họ và tên	MSSV
Phạm Anh Tuấn	20215662
Nguyễn Minh Sơn	20215637
Phạm Hoàng Thành	20210805
Vũ Thị Mai Linh	20215608
Bùi Anh Quốc	20215634
Phan Nguyên Vũ	20210952

Cài đặt Mapreduce cho thuật toán TF-IDF

1. Tính TF (Job 1):

Mapper: Đọc tài liệu và phát ra các cặp từ (t, d), trong đó t là từ và d là ID của tài liệu.

Reducer: Tính tần suất xuất hiện của mỗi từ trong từng tài liệu.

```
from mrjob.job import MRJob


class TFIDF1(MRJob):


    def mapper(self, _, line):
        words = line.split()
        doc_id = words[0]
        for word in words[1:]:
            yield (word, doc_id), 1


    def reducer(self, key, values):
        yield key, sum(values)


if __name__ == '__main__':
    TFIDF1.run()
```

2. Tính IDF (Job 2):

- **Mapper**: Lấy đầu ra từ job đầu tiên và phát ra (t, d) cùng với TF.
- **Reducer**: Tính số tài liệu chứa mỗi từ và IDF.

```
class TFIDF2(MRJob):

    def mapper(self, key, value):
        word, doc_id = key
        yield word, doc_id

    def reducer(self, key, values):
        doc_ids = list(set(values))
        df = len(doc_ids)
        for doc_id in doc_ids:
            yield (word, doc_id), df

if __name__ == '__main__':
    TFIDF2.run()
```

3. Tính TF-IDF (Job 3):

- **Mapper**: Lấy đầu ra từ hai job trước và tính TF-IDF cho mỗi từ trong mỗi tài liệu.
- **Reducer**: Kết hợp TF và IDF để tạo ra giá trị TF-IDF cuối cùng.

```
class TFIDF3(MRJob):

    def mapper(self, key, value):
        word, doc_id = key
        tf, df = value
        yield (word, doc_id), tf * math.log(1.0 / df)

    def reducer(self, key, values):
        yield key, sum(values)

if __name__ == '__main__':
    TFIDF3.run()
```