

Thực hành truy hồi Boole

Nguyễn Mạnh Hiễn

hiennm@tlu.edu.vn

Cài đặt Whoosh

- Whoosh là thư viện tìm kiếm (truy hồi thông tin).
- Kiểm tra xem máy tính của bạn đã cài Whoosh chưa:
 - Gõ lệnh **import whoosh** ở dấu nhắc lệnh Python.
 - Không thấy báo lỗi gì nghĩa là đã cài Whoosh.
- Nếu máy tính của bạn chưa cài Whoosh, gõ lệnh **pip install whoosh** ở cửa sổ lệnh Windows (không gõ ở dấu nhắc lệnh Python).

Các đối tượng Index và Schema

- Đối tượng Index biểu diễn chỉ mục ngược.
- Đối tượng Schema biểu diễn lược đồ của chỉ mục ngược, chỉ rõ mỗi văn bản gồm những trường nào, như tiêu đề, thân, nội dung...

- Tạo lược đồ có hai trường title và content:

```
from whoosh.fields import Schema, TEXT
schema = Schema(title=TEXT, content=TEXT)
```

Ở đây, title và content đều có kiểu là TEXT, tức là văn bản thông thường.

- Tạo chỉ mục ngược trong thư mục index, có lược đồ schema:

```
from whoosh.index import create_in
ix = create_in("index", schema)
```

Một số kiểu trường trong lược đồ

- `whoosh.fields.ID`

Một dãy từ đơn nhất, không tách thành các từ đơn lẻ. Ví dụ: đường dẫn file, URL, ngày tháng...

- `whoosh.fields.STORED`

Chỉ lưu trữ lại, không lập chỉ mục trên kiểu trường này. Hữu ích cho những thông tin kèm theo kết quả tìm kiếm.

- `whoosh.fields.TEXT`

Văn bản thông thường, có hỗ trợ lưu thông tin vị trí từ để dùng trong tìm kiếm cụm từ.

- Những kiểu trường khác, tham khảo tài liệu Whoosh trên web (<https://whoosh.readthedocs.io/en/latest/index.html>).

Tạo/mở chỉ mục ngược

- Tạo chỉ mục mới:

```
import os.path
from whoosh.index import create_in
if not os.path.exists("index"):
    os.mkdir("index") # Tạo thư mục nếu cần
ix = create_in("index", schema)
```

- Mở chỉ mục đã có:

```
from whoosh.index import open_dir
ix = open_dir("index")
```

- Một số lệnh hữu ích:

- Xem thư mục hiện hành là thư mục nào: `os.getcwd()`
- Xem nội dung thư mục hiện hành: `os.listdir()`
- Thay đổi thư mục hiện hành: `os.chdir(<tên thư mục>)`

Đối tượng IndexWriter

Dùng để thêm các văn bản vào chỉ mục ngược (đang là ix trong các slide trước):

```
writer = ix.writer()
writer.add_document(title="My document",
                    content="This is my document!")
writer.add_document(title="Second try",
                    content="This is the second example.")
writer.add_document(title="Third time's the charm",
                    content="Examples are many.")
writer.commit() # Phải gọi commit để kết thúc
```

Đối tượng Searcher

- Dùng để tìm kiếm:

```
searcher = ix.searcher()
```

- Để đóng các file đang mở ngay sau khi kết thúc tìm kiếm, ta dùng từ khóa with:

```
with ix.searcher() as searcher:
```

```
...
```

- Gọi phương thức search để tiến hành tìm kiếm với một đối tượng Query (câu truy vấn):

```
results = searcher.search(myquery)
```


(Giả sử myquery đã được định nghĩa trước đó rồi)

Tạo đối tượng Query

- Cách 1: Tạo trực tiếp

```
from whoosh.query import *  
myquery = And([Term("content", "apple"),  
               Term("content", "bear")])
```

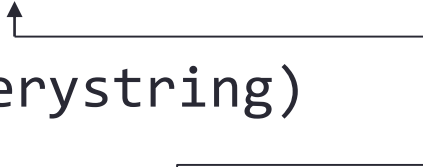
Tìm từ “apple” trong trường “content”



- Cách 2: Dùng đối tượng QueryParser để phân tích một chuỗi truy vấn

```
from whoosh.qparser import QueryParser  
parser = QueryParser("content", ix.schema)  
myquery = parser.parse(querystring)
```

“content” là trường tìm kiếm ngầm định



Ví dụ phân tích xâu truy vấn

```
>>> parser.parse("render shade animate")
And([Term("content", "render"), Term("content",
"shade"), Term("content", "animate")])

>>> parser.parse("render OR (title:shade
keyword:animate)")
Or([Term("content", "render"), And([Term("title",
"shade"), Term("keyword", "animate")])])

>>> parser.parse("rend*")
Prefix("content", "rend")
```

Tổng hợp

Đọc hiểu và chạy thử chương trình đầy đủ sau đây:

```
from whoosh.index import create_in
from whoosh.fields import *
schema = Schema(title=TEXT(stored=True), path=ID(stored=True),
                 content=TEXT)
ix = create_in("indexdir", schema)
writer = ix.writer()
writer.add_document(title="First document", path="/a",
                    content="This is the first document we've added!")
writer.add_document(title="Second document", path="/b",
                    content="The second one is even more interesting!")
writer.commit()
from whoosh.qparser import QueryParser
with ix.searcher() as searcher:
    query = QueryParser("content", ix.schema).parse("first")
    results = searcher.search(query)
    print(results[0])
```

Bài tập

Viết chương trình thực hiện tuần tự các yêu cầu sau đây:

1. Xây dựng chỉ mục ngược cho một tập văn bản theo cách làm như trong các slide phía trước.
2. Yêu cầu người dùng nhập vào câu truy vấn với giao diện như sau:

Tìm các văn bản thỏa mãn:

- chứa tất cả các từ này: <từ 1> <từ 2> ...
- chứa ít nhất một trong các từ này: <từ 3> <từ 4> ...
- không chứa các từ này: <từ 5> <từ 6> ...

3. Xử lý câu truy vấn và hiển thị kết quả lên màn hình.