Inverted files(2)

from collections import defaultdict

# Step 1: Build the inverted index

def build\_inverted\_index(documents):

inverted\_index = defaultdict(list)

for doc\_id, doc in enumerate(documents):

for term in set(doc.lower().split()): # Tokenize and avoid duplicates

inverted\_index[term].append(doc\_id)

return inverted\_index

# Step 2: Search for documents containing terms

def search(query, inverted\_index):

return [inverted\_index.get(term, []) for term in query.lower().split()]

# Sample documents

documents = [

"The quick brown fox jumps over the lazy dog",

"Never jump over the lazy dog quickly",

"A brown fox is quick and jumps high",

]

# Build and query the inverted index

inverted\_index = build\_inverted\_index(documents)

query = "brown dog"

result = search(query, inverted\_index)

# Display results

for idx, term in enumerate(query.split()):

print(f"Documents containing {term}:", result[idx])