

```
"""
```

Task 3: Retrieval-Augmented Generation (RAG) Pipeline

This revised script:

- Prints each question and answer in the terminal/console for 5 example queries (BM25 & dense, top_k=5/10).
- Shows bar charts (token length, 'contains_answer' count, confidence).
- In interactive mode, it does the same for any user question, again printing question+answers and showing charts.

```
"""
```

```
import sys
from Bio import Entrez
import chromadb
from sentence_transformers import SentenceTransformer
from rank_bm25 import BM25Okapi
import nltk
from nltk.tokenize import word_tokenize
from transformers import AutoTokenizer, AutoModelForQuestionAnswering, pipeline
import pandas as pd
import matplotlib.pyplot as plt

# 1) ENV SETUP
nltk.download("punkt")
Entrez.email = "choksi.pr@northeastern.edu"

# If you want GPU usage, set USE_GPU = True, else it uses CPU
USE_GPU = False
DEVICE_ID = 0 if USE_GPU else -1

# 2) CHROMADB INIT
chroma_client = chromadb.PersistentClient(path="chroma_store")
collection_name = "scientific_papers_interactive"
collection = chroma_client.get_or_create_collection(collection_name)

# 3) SENTENCE TRANSFORMER
embedding_model = SentenceTransformer("all-MiniLM-L6-v2")

# -----
# PUBMED FETCH & INDEX
# -----
def fetch_pubmed_data(query, max_results=10):
    handle = Entrez.esearch(db="pubmed", term=query, retmax=max_results)
    record = Entrez.read(handle)
    handle.close()

    id_list = record.get("IdList", [])
    if not id_list:
        print("No PubMed articles found for query:", query)
        return []

    handle = Entrez.efetch(db="pubmed", id=id_list, rettype="xml")
    papers = Entrez.read(handle)
    handle.close()

    results = []
    for paper in papers.get('PubmedArticle', []):
        title = paper['MedlineCitation']['Article']['ArticleTitle']
        abstract = paper['MedlineCitation']['Article'].get('Abstract', {}).get('AbstractText', [""])[0]
        results.append({
            "id": paper['MedlineCitation']['PMID'],
            "title": title,
            "abstract": abstract,
            "text": abstract
        })
    return results

def index_papers(papers):
    for paper in papers:
        emb = embedding_model.encode(paper["text"]).tolist()
        collection.add(
            ids=[paper["id"]],
            embeddings=[emb],
            metadatas=[{"title": paper["title"], "abstract": paper["abstract"]}],
            documents=[paper["text"]]
        )

def maybe_fetch_and_index_papers(query, max_results=10):
    existing_count = collection.count()
    if existing_count > 0:
        print(f"\n>>> ChromaDB collection already has {existing_count} documents.")
        choice = input("Skip fetching from PubMed? (y/n) [default: y] >> ").strip().lower()
        if choice in ["n", "no"]:
```

```

        print("Deleting existing docs and fetching new ones...")
        collection.delete()
        new_papers = fetch_pubmed_data(query, max_results)
        if new_papers:
            index_papers(new_papers)
        else:
            print("No new papers fetched.")
    else:
        print("Skipping PubMed fetch, using existing data.")
else:
    print("ChromaDB collection is empty. Fetching new documents from PubMed...")
    new_papers = fetch_pubmed_data(query, max_results)
    if new_papers:
        index_papers(new_papers)
    else:
        print("No papers fetched from PubMed.")

maybe_fetch_and_index_papers("AI in healthcare", max_results=10)

all_docs = collection.get()
documents = all_docs.get("documents", [])
if not documents:
    print("No documents in ChromaDB. Exiting.")
    sys.exit(0)

tokenized_docs = [word_tokenize(doc.lower()) for doc in documents]
bm25 = BM25Okapi(tokenized_docs)

def retrieve_bm25(query, top_k=5):
    t_query = word_tokenize(query.lower())
    scores = bm25.get_scores(t_query)
    top_n_indices = sorted(range(len(scores)), key=lambda i: scores[i], reverse=True)[:top_k]
    return [documents[i] for i in top_n_indices]

def retrieve_dense(query, top_k=5):
    emb = embedding_model.encode(query).tolist()
    results = collection.query(query_embeddings=emb, n_results=top_k)
    print("\nDense Retrieval Results:", results)
    if not results.get("documents"):
        return []
    return results["documents"][0]

# QA MODEL
from transformers import AutoTokenizer, AutoModelForQuestionAnswering, pipeline

tokenizer = AutoTokenizer.from_pretrained("distilbert-base-uncased-distilled-squad")
model = AutoModelForQuestionAnswering.from_pretrained("distilbert-base-uncased-distilled-squad")
qa_pipeline = pipeline(
    "question-answering",
    model=model,
    tokenizer=tokenizer,
    device=DEVICE_ID # 0 => GPU, -1 => CPU
)

print(f"\nUsing device={DEVICE_ID} ('cuda:0' if 0, 'cpu' if -1)\n")

# EVALUATION
def check_retrieval_effectiveness(answer, retrieved_docs):
    ans_lower = answer.lower()
    for doc in retrieved_docs:
        if ans_lower in doc.lower():
            return True
    return False

def evaluate_response(query, retrieval_method="dense", top_k=5):
    if retrieval_method == "bm25":
        docs = retrieve_bm25(query, top_k)
    else:
        docs = retrieve_dense(query, top_k)

    if not docs:
        return {
            "question": query,
            "retrieval_method": retrieval_method,
            "top_k": top_k,
            "answer": "No documents retrieved.",
            "token_length": 0,
            "contains_answer": False,
            "confidence_score": 0.0
        }

    context = " ".join(docs)
    result = qa_pipeline({"question": query, "context": context})

```

```

answer_text = result["answer"]
confidence = result.get("score", 0.0)

return {
    "question": query,
    "retrieval_method": retrieval_method,
    "top_k": top_k,
    "answer": answer_text,
    "token_length": len(answer_text.split()),
    "contains_answer": check_retrieval_effectiveness(answer_text, docs),
    "confidence_score": confidence
}

# 5 EXAMPLE TEST QUERIES
example_queries = [
    "How does AI help in drug discovery?",
    "How does AI help in diagnostics?",
    "What are the benefits of machine learning in healthcare?",
    "How is deep learning used in medical imaging?",
    "Does artificial intelligence improve patient outcomes?"
]
METHODS = ["bm25", "dense"]
TOP_K_VALUES = [5, 10]

results = []
for q_ in example_queries:
    for rm_ in METHODS:
        for tk_ in TOP_K_VALUES:
            results.append(evaluate_response(q_, rm_, tk_))

df_fixed = pd.DataFrame(results)
print("\n--- RESULTS FOR 5 EXAMPLE QUERIES (TABLE) ---\n")
print(df_fixed)

# Print Q&A in a more readable console format
print("\n--- DETAILED PRINT OF Q&A ---\n")
for idx, row in df_fixed.iterrows():
    print(f"Q: {row['question']}")
    print(f"Method={row['retrieval_method']} | top_k={row['top_k']}")
    print(f"Answer: {row['answer']}")
    print(f"Token Length: {row['token_length']}, Contains Answer: {row['contains_answer']}, Confidence: {row['confidence_score']:.3f}")
    print("-" * 60)

# GROUP & CHARTS
grouped_fixed = (
    df_fixed.groupby(["retrieval_method", "top_k"])
        .agg({
            "token_length": "mean",
            "contains_answer": "sum",
            "confidence_score": "mean"
        })
        .reset_index()
)

# (1) Average Token Length
plt.figure()
plt.bar(range(len(grouped_fixed)), grouped_fixed["token_length"])
plt.xticks(
    ticks=range(len(grouped_fixed)),
    labels=[f"{rm}-{tk}" for rm, tk in zip(grouped_fixed["retrieval_method"], grouped_fixed["top_k"])],
    rotation=45,
    ha="right"
)
plt.title("5 Example Queries: Average Token Length (by Method & top_k)")
plt.xlabel("Method-top_k")
plt.ylabel("Avg. Token Length")
plt.tight_layout()
plt.show()

# (2) Count of 'Contains Answer'
plt.figure()
plt.bar(range(len(grouped_fixed)), grouped_fixed["contains_answer"])
plt.xticks(
    ticks=range(len(grouped_fixed)),
    labels=[f"{rm}-{tk}" for rm, tk in zip(grouped_fixed["retrieval_method"], grouped_fixed["top_k"])],
    rotation=45,
    ha="right"
)
plt.title("5 Example Queries: Count of 'Contains Answer' (by Method & top_k)")
plt.xlabel("Method-top_k")
plt.ylabel("# Found In Docs")
plt.tight_layout()
plt.show()

```

```

plt.show()

# (3) Confidence
plt.figure()
plt.bar(range(len(grouped_fixed)), grouped_fixed["confidence_score"])
plt.xticks(
    ticks=range(len(grouped_fixed)),
    labels=[f"{rm}-{tk}" for rm, tk in zip(grouped_fixed["retrieval_method"], grouped_fixed["top_k"])],
    rotation=45,
    ha="right"
)
plt.title("5 Example Queries: Average Confidence Score (by Method & top_k)")
plt.xlabel("Method-top_k")
plt.ylabel("Avg. Confidence Score")
plt.tight_layout()
plt.show()

# INTERACTIVE
print("\n--- INTERACTIVE MODE (TEST ALL COMBOS) ---")
while True:
    user_input = input("\nEnter your question (or 'exit'/'quit'/'esc' to stop): ").strip()
    if user_input.lower() in ["exit", "quit", "esc"]:
        print("Exiting. Goodbye!")
        break

    combos_results = []
    for rm in METHODS:
        for tk_val in TOP_K_VALUES:
            combos_results.append(evaluate_response(user_input, rm, tk_val))

    df_interactive = pd.DataFrame(combos_results)
    print("\n--- RESULTS FOR YOUR QUESTION (TABLE) ---\n")
    print(df_interactive)

    # Print Q&A in detail
    print("\n--- DETAILED PRINT OF Q&A ---\n")
    for idx, row in df_interactive.iterrows():
        print(f"Q: {row['question']}")
        print(f"Method={row['retrieval_method']} | top_k={row['top_k']}")
        print(f"Answer: {row['answer']}")
        print(f"Token Length: {row['token_length']}, Contains Answer: {row['contains_answer']}, Confidence: {row['confidence_score']:.3f}")
        print("-" * 60)

    # Now group & plot the bar charts
    grouped_int = (
        df_interactive.groupby(["retrieval_method", "top_k"])
        .agg({
            "token_length": "mean",
            "contains_answer": "sum",
            "confidence_score": "mean"
        })
        .reset_index()
    )

    # (A) token_length
    plt.figure()
    plt.bar(range(len(grouped_int)), grouped_int["token_length"])
    plt.xticks(
        ticks=range(len(grouped_int)),
        labels=[f"{rm}-{tk}" for rm, tk in zip(grouped_int["retrieval_method"], grouped_int["top_k"])],
        rotation=45,
        ha="right"
    )
    plt.title(f"Interactive - Avg. Token Length for question: '{user_input}'")
    plt.xlabel("Method-top_k")
    plt.ylabel("Avg. Token Length")
    plt.tight_layout()
    plt.show()

    # (B) contains_answer
    plt.figure()
    plt.bar(range(len(grouped_int)), grouped_int["contains_answer"])
    plt.xticks(
        ticks=range(len(grouped_int)),
        labels=[f"{rm}-{tk}" for rm, tk in zip(grouped_int["retrieval_method"], grouped_int["top_k"])],
        rotation=45,
        ha="right"
    )
    plt.title(f"Interactive - Count of 'Contains Answer' for question: '{user_input}'")
    plt.xlabel("Method-top_k")
    plt.ylabel("# Found In Docs")
    plt.tight_layout()
    plt.show()

```

```
# (C) confidence_score
plt.figure()
plt.bar(range(len(grouped_int)), grouped_int["confidence_score"])
plt.xticks(
    ticks=range(len(grouped_int)),
    labels=[f"{rm}-{tk}" for rm, tk in zip(grouped_int["retrieval_method"], grouped_int["top_k"])],
    rotation=45,
    ha="right"
)
plt.title(f"Interactive - Avg. Confidence Score for question: '{user_input}'")
plt.xlabel("Method-top_k")
plt.ylabel("Confidence Score")
plt.tight_layout()
plt.show()
```

```

[ nltk_data ] Downloading package punkt to
[ nltk_data ]      C:\Users\pchok\AppData\Roaming\nltk_data...
[ nltk_data ]      Package punkt is already up-to-date!

>>> ChromaDB collection already has 10 documents.
Skipping PubMed fetch, using existing data.
Device set to use cpu
c:\Users\pchok\anaconda3\Lib\site-packages\transformers\pipelines\question_answering.py:391: FutureWarning: Passing a list of SQuAD
  warnings.warn(

Using device=-1 ('cuda:0' if 0, 'cpu' if -1)

Dense Retrieval Results: {'ids': [['40010238', '40013176', '40012625', '40009619', '40014274']], 'embeddings': None, 'documents': [
Dense Retrieval Results: {'ids': [['40010238', '40013176', '40012625', '40009619', '40014274', '40010810', '40012602', '40010768',
Dense Retrieval Results: {'ids': [['40010238', '40013176', '40009619', '40012625', '40014274']], 'embeddings': None, 'documents': [
Dense Retrieval Results: {'ids': [['40010238', '40013176', '40009619', '40012625', '40014274', '40012602', '40010810', '40010768',
Dense Retrieval Results: {'ids': [['40010238', '40012602', '40012625', '40014274', '40013176']], 'embeddings': None, 'documents': [
Dense Retrieval Results: {'ids': [['40010238', '40012602', '40012625', '40014274', '40013176', '40009619', '40010810', '40013049',
Dense Retrieval Results: {'ids': [['40010238', '40012602', '40014274', '40009619', '40013176']], 'embeddings': None, 'documents': [
Dense Retrieval Results: {'ids': [['40010238', '40012602', '40014274', '40009619', '40013176', '40012625', '40010810', '40013049',
Dense Retrieval Results: {'ids': [['40013176', '40010238', '40014274', '40012602', '40009619']], 'embeddings': None, 'documents': [
Dense Retrieval Results: {'ids': [['40013176', '40010238', '40014274', '40012602', '40009619', '40012625', '40010810', '40013049',

--- RESULTS FOR 5 EXAMPLE QUERIES (TABLE) ---

      question retrieval_method top_k \
0      How does AI help in drug discovery?      bm25      5
1      How does AI help in drug discovery?      bm25     10
2      How does AI help in drug discovery?      dense      5
3      How does AI help in drug discovery?      dense     10
4      How does AI help in diagnostics?      bm25      5
5      How does AI help in diagnostics?      bm25     10
6      How does AI help in diagnostics?      dense      5
7      How does AI help in diagnostics?      dense     10
8  What are the benefits of machine learning in h...      bm25      5
9  What are the benefits of machine learning in h...      bm25     10
10 What are the benefits of machine learning in h...      dense      5
11 What are the benefits of machine learning in h...      dense     10
12 How is deep learning used in medical imaging?      bm25      5
13 How is deep learning used in medical imaging?      bm25     10
14 How is deep learning used in medical imaging?      dense      5
15 How is deep learning used in medical imaging?      dense     10
16 Does artificial intelligence improve patient o...      bm25      5
17 Does artificial intelligence improve patient o...      bm25     10
18 Does artificial intelligence improve patient o...      dense      5
19 Does artificial intelligence improve patient o...      dense     10

      answer token_length \
0  streamlining logistics and inventory management      5
1  streamlining logistics and inventory management      5
2      enhance vaccine supply chains      4
3      artificial intelligence      2
4  streamlining logistics and inventory management      5
5  streamlining logistics and inventory management      5
6  streamlining logistics and inventory management      5
7      artificial intelligence      2
8  negatively affects patient outcomes, wait time...      9
9  negatively affects patient outcomes, wait time...      9
10      transparency      1
11      greater oral healthcare inequities      4
12 magnetic resonance imaging and cerebrospinal f...      7
13 predictive analytics with interpretable rule-b...      6
14      Artificial Intelligence (AI) tools      4
15      Artificial Intelligence (AI) tools      4
16      improve the management of this population      6
17      improve the management of this population      6
18      improve the management of this population?      6
19      improve the management of this population?      6

      contains_answer confidence_score
0      True      0.422561
1      True      0.422561
2      True      0.316537
3      True      0.729459
4      True      0.489828
5      True      0.489828
6      True      0.393711
7      True      0.355121
8      True      0.283635
9      True      0.283635

```

| | | |
|----|------|----------|
| 10 | True | 0.319089 |
| 11 | True | 0.375713 |
| 12 | True | 0.072802 |
| 13 | True | 0.201579 |
| 14 | True | 0.253473 |
| 15 | True | 0.253473 |
| 16 | True | 0.180981 |
| 17 | True | 0.180981 |
| 18 | True | 0.089209 |
| 19 | True | 0.089209 |

--- DETAILED PRINT OF Q&A ---

Q: How does AI help in drug discovery?

Method=bm25 | top_k=5

Answer: streamlining logistics and inventory management

Token Length: 5, Contains Answer: True, Confidence: 0.423

Q: How does AI help in drug discovery?

Method=bm25 | top_k=10

Answer: streamlining logistics and inventory management

Token Length: 5, Contains Answer: True, Confidence: 0.423

Q: How does AI help in drug discovery?

Method=dense | top_k=5

Answer: enhance vaccine supply chains

Token Length: 4, Contains Answer: True, Confidence: 0.317

Q: How does AI help in drug discovery?

Method=dense | top_k=10

Answer: artificial intelligence

Token Length: 2, Contains Answer: True, Confidence: 0.729

Q: How does AI help in diagnostics?

Method=bm25 | top_k=5

Answer: streamlining logistics and inventory management

Token Length: 5, Contains Answer: True, Confidence: 0.490

Q: How does AI help in diagnostics?

Method=bm25 | top_k=10

Answer: streamlining logistics and inventory management

Token Length: 5, Contains Answer: True, Confidence: 0.490

Q: How does AI help in diagnostics?

Method=dense | top_k=5

Answer: streamlining logistics and inventory management

Token Length: 5, Contains Answer: True, Confidence: 0.394

Q: How does AI help in diagnostics?

Method=dense | top_k=10

Answer: artificial intelligence

Token Length: 2, Contains Answer: True, Confidence: 0.355

Q: What are the benefits of machine learning in healthcare?

Method=bm25 | top_k=5

Answer: negatively affects patient outcomes, wait times, and resource efficiency

Token Length: 9, Contains Answer: True, Confidence: 0.284

Q: What are the benefits of machine learning in healthcare?

Method=bm25 | top_k=10

Answer: negatively affects patient outcomes, wait times, and resource efficiency

Token Length: 9, Contains Answer: True, Confidence: 0.284

Q: What are the benefits of machine learning in healthcare?

Method=dense | top_k=5

Answer: transparency

Token Length: 1, Contains Answer: True, Confidence: 0.319

Q: What are the benefits of machine learning in healthcare?

Method=dense | top_k=10

Answer: greater oral healthcare inequities

Token Length: 4, Contains Answer: True, Confidence: 0.376

Q: How is deep learning used in medical imaging?

Method=bm25 | top_k=5

Answer: magnetic resonance imaging and cerebrospinal fluid analysis

Token Length: 7, Contains Answer: True, Confidence: 0.073

Q: How is deep learning used in medical imaging?

Method=bm25 | top_k=10

Answer: predictive analytics with interpretable rule-based methods

Token Length: 6, Contains Answer: True, Confidence: 0.202

Q: How is deep learning used in medical imaging?

Method=dense | top_k=5

Answer: Artificial Intelligence (AI) tools

Token Length: 4, Contains Answer: True, Confidence: 0.253

Q: How is deep learning used in medical imaging?

Method=dense | top_k=10

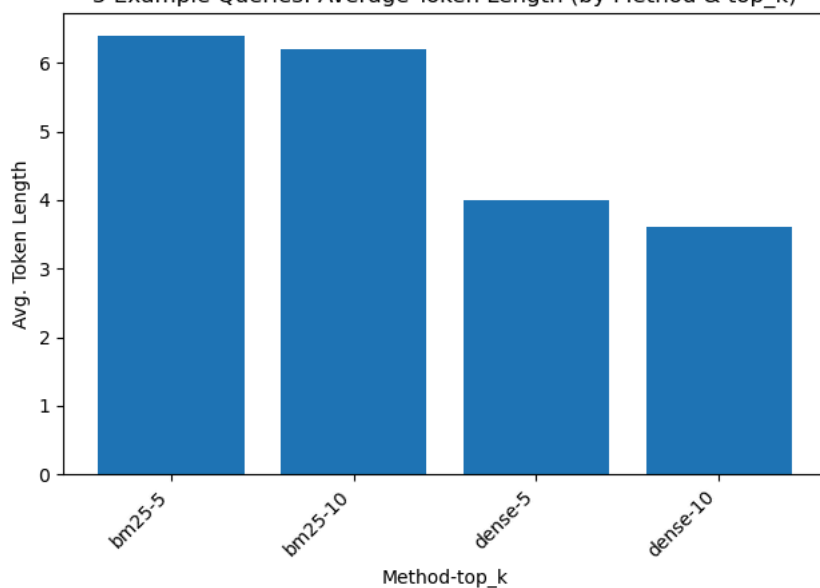
Answer: Artificial Intelligence (AI) tools

```

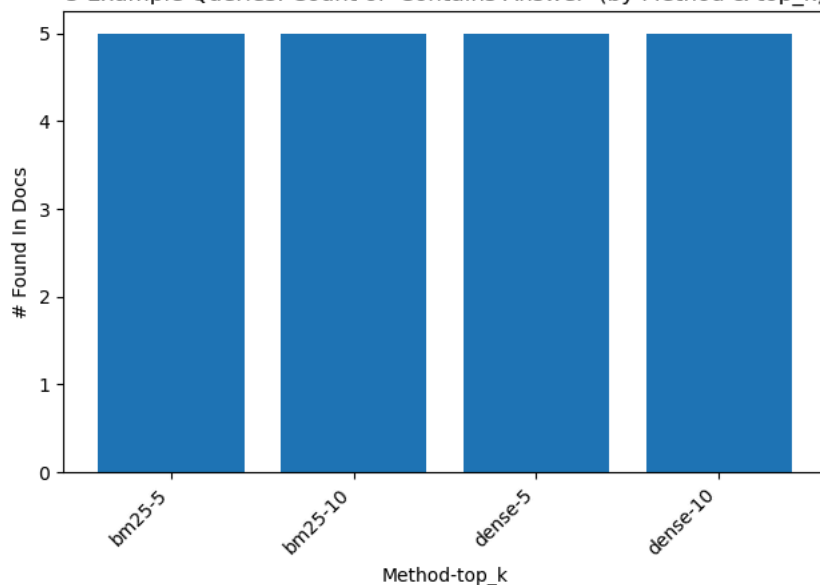
Answer: Artificial intelligence (AI) tools
Token Length: 4, Contains Answer: True, Confidence: 0.253
-----
Q: Does artificial intelligence improve patient outcomes?
Method=bm25 | top_k=5
Answer: improve the management of this population
Token Length: 6, Contains Answer: True, Confidence: 0.181
-----
Q: Does artificial intelligence improve patient outcomes?
Method=bm25 | top_k=10
Answer: improve the management of this population
Token Length: 6, Contains Answer: True, Confidence: 0.181
-----
Q: Does artificial intelligence improve patient outcomes?
Method=dense | top_k=5
Answer: improve the management of this population?
Token Length: 6, Contains Answer: True, Confidence: 0.089
-----
Q: Does artificial intelligence improve patient outcomes?
Method=dense | top_k=10
Answer: improve the management of this population?
Token Length: 6, Contains Answer: True, Confidence: 0.089
-----

```

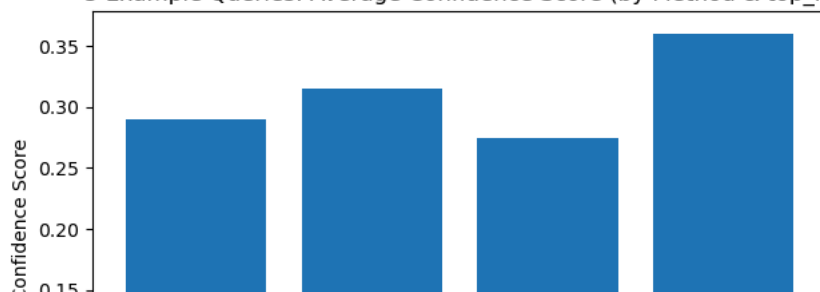
5 Example Queries: Average Token Length (by Method & top_k)

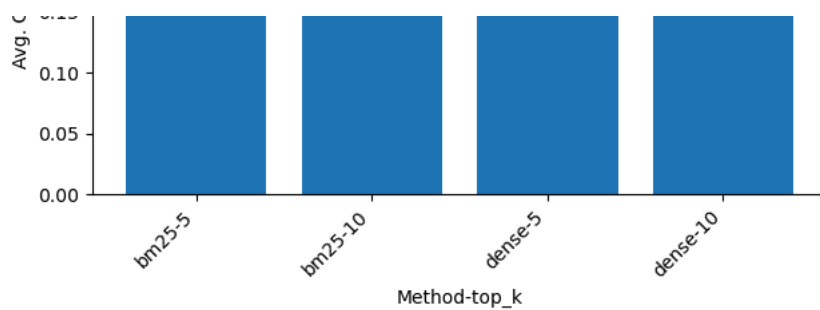


5 Example Queries: Count of 'Contains Answer' (by Method & top_k)



5 Example Queries: Average Confidence Score (by Method & top_k)





```
--- INTERACTIVE MODE (TEST ALL COMBOS) ---
c:\Users\pchok\anaconda3\Lib\site-packages\transformers\pipelines\question_answering.py:391: FutureWarning: Passing a list of SQuAD
warnings.warn(
```

```
Dense Retrieval Results: {'ids': [['40010810', '40014274', '40012602', '40011748', '40009619']], 'embeddings': None, 'documents': [
```

```
Dense Retrieval Results: {'ids': [['40010810', '40014274', '40012602', '40011748', '40009619', '40010238', '40013176', '40012625'],
```

```
--- RESULTS FOR YOUR QUESTION (TABLE) ---
```

| | question | retrieval_method | top_k | \ |
|---|------------------------------------------|------------------|-------|---|
| 0 | What model is used for metabolics study? | bm25 | 5 | |
| 1 | What model is used for metabolics study? | bm25 | 10 | |
| 2 | What model is used for metabolics study? | dense | 5 | |
| 3 | What model is used for metabolics study? | dense | 10 | |

| | answer | token_length | contains_answer | confidence_score |
|---|-------------------|--------------|-----------------|------------------|
| 0 | AI | 1 | True | 0.090576 |
| 1 | predictive models | 2 | True | 0.646343 |
| 2 | predictive models | 2 | True | 0.007803 |
| 3 | Biomarker | 1 | True | 0.427688 |

```
--- DETAILED PRINT OF Q&A ---
```

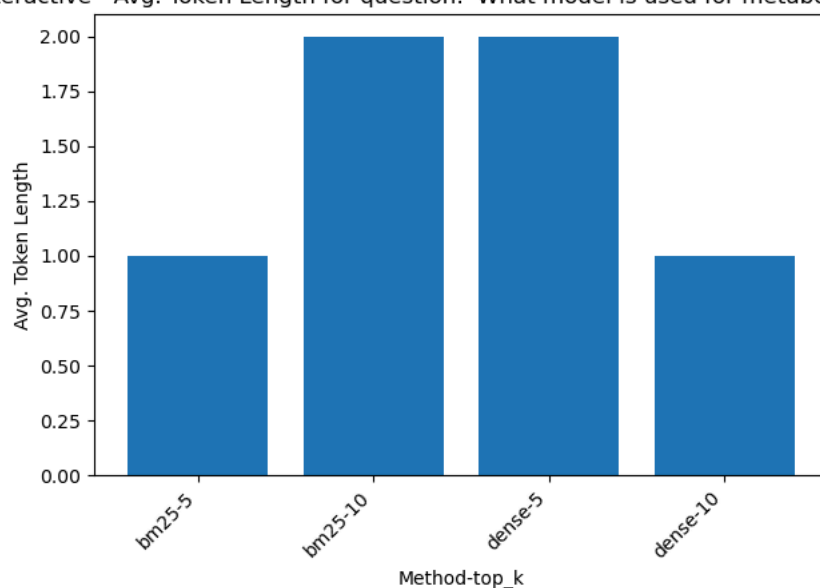
```
Q: What model is used for metabolics study?
Method=bm25 | top_k=5
Answer: AI
Token Length: 1, Contains Answer: True, Confidence: 0.091
```

```
-----
Q: What model is used for metabolics study?
Method=bm25 | top_k=10
Answer: predictive models
Token Length: 2, Contains Answer: True, Confidence: 0.646
```

```
-----
Q: What model is used for metabolics study?
Method=dense | top_k=5
Answer: predictive models
Token Length: 2, Contains Answer: True, Confidence: 0.008
```

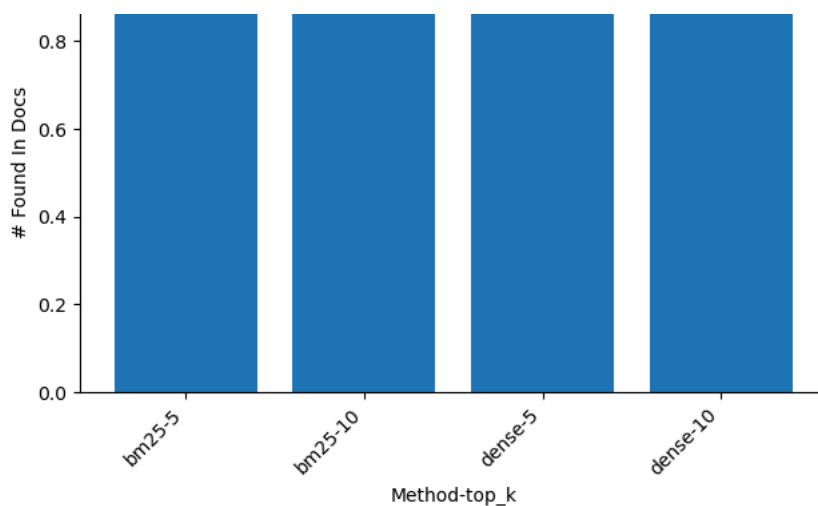
```
-----
Q: What model is used for metabolics study?
Method=dense | top_k=10
Answer: Biomarker
Token Length: 1, Contains Answer: True, Confidence: 0.428
```

Interactive - Avg. Token Length for question: 'What model is used for metabolics study?'

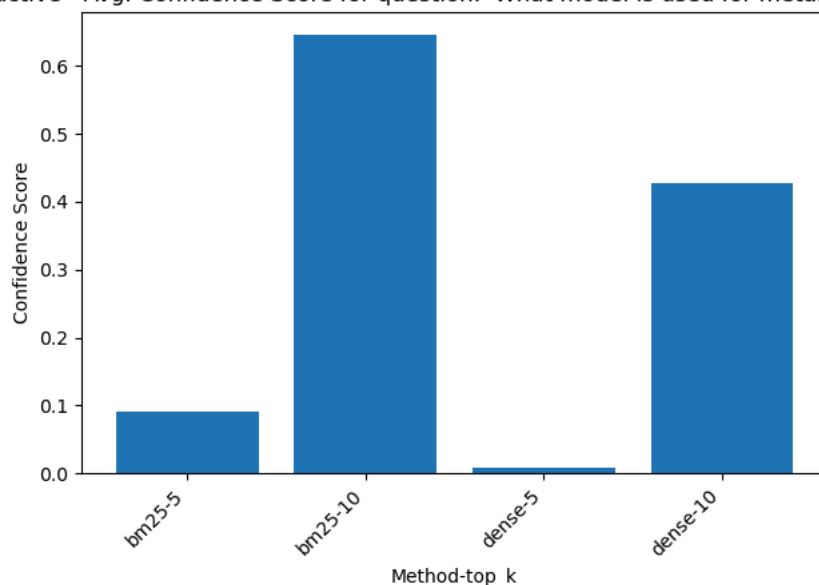


Interactive - Count of 'Contains Answer' for question: 'What model is used for metabolics study?'





Interactive - Avg. Confidence Score for question: 'What model is used for metabolics study?'



c:\Users\pchok\anaconda3\Lib\site-packages\transformers\pipelines\question_answering.py:391: FutureWarning: Passing a list of SQuAD warnings.warn()

Dense Retrieval Results: {'ids': [['40010238', '40012602', '40013176', '40014274', '40012625']], 'embeddings': None, 'documents': [

Dense Retrieval Results: {'ids': [['40010238', '40012602', '40013176', '40014274', '40012625', '40009619', '40010810', '40010768'],

--- RESULTS FOR YOUR QUESTION (TABLE) ---

| | question | retrieval_method | top_k \ |
|---|---------------------------------------------------|------------------|---------|
| 0 | How can Generative AI help for image classific... | bm25 | 5 |
| 1 | How can Generative AI help for image classific... | bm25 | 10 |
| 2 | How can Generative AI help for image classific... | dense | 5 |
| 3 | How can Generative AI help for image classific... | dense | 10 |

| | answer | token_length \ |
|---|---------------------------------------------------|----------------|
| 0 | innovative tools and technologies, including a... | 7 |
| 1 | innovative tools and technologies, including a... | 7 |
| 2 | Artificial Intelligence (AI) tools, which anal... | 8 |
| 3 | patch-clamp recordings and molecular dynamic s... | 6 |

| | contains_answer | confidence_score |
|---|-----------------|------------------|
| 0 | True | 0.386894 |
| 1 | True | 0.386894 |
| 2 | True | 0.180171 |
| 3 | True | 0.224399 |

--- DETAILED PRINT OF Q&A ---

Q: How can Generative AI help for image classifications?
Method=bm25 | top_k=5
Answer: innovative tools and technologies, including artificial intelligence
Token Length: 7, Contains Answer: True, Confidence: 0.387

Q: How can Generative AI help for image classifications?
Method=bm25 | top_k=10
Answer: innovative tools and technologies, including artificial intelligence
Token Length: 7, Contains Answer: True, Confidence: 0.387

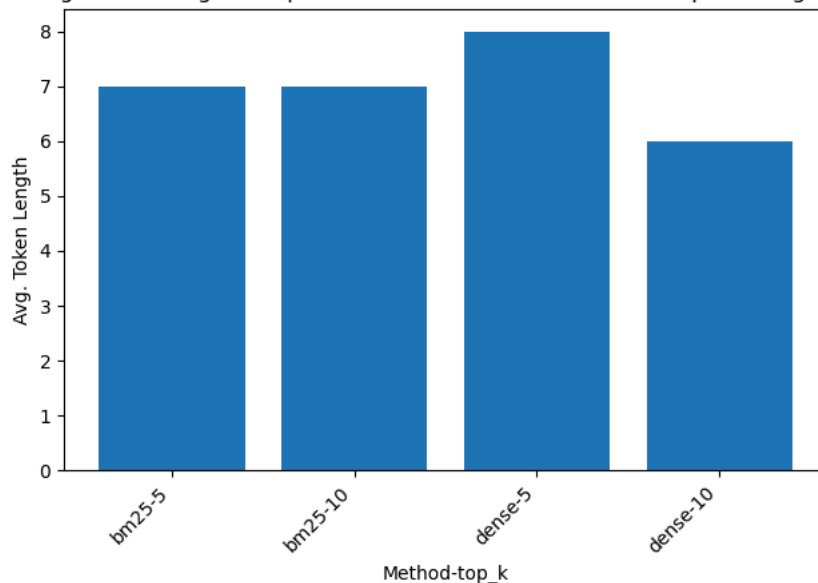
Q: How can Generative AI help for image classifications?
Method=dense | top_k=5
Answer: Artificial Intelligence (AI) tools, which analyze large datasets

Answer: Artificial Intelligence (AI) tools, which analyze large datasets
Token Length: 8, Contains Answer: True, Confidence: 0.180

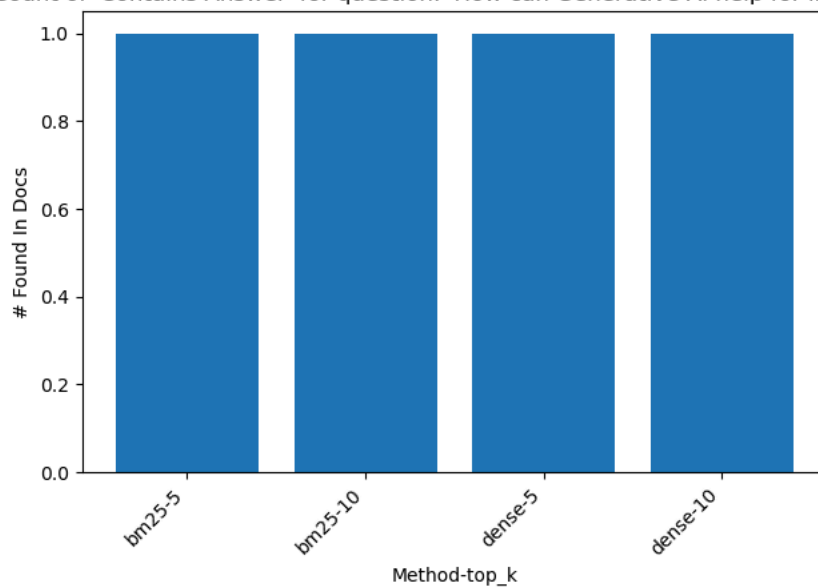
Q: How can Generative AI help for image classifications?
Method=dense | top_k=10

Answer: patch-clamp recordings and molecular dynamic simulations
Token Length: 6, Contains Answer: True, Confidence: 0.224

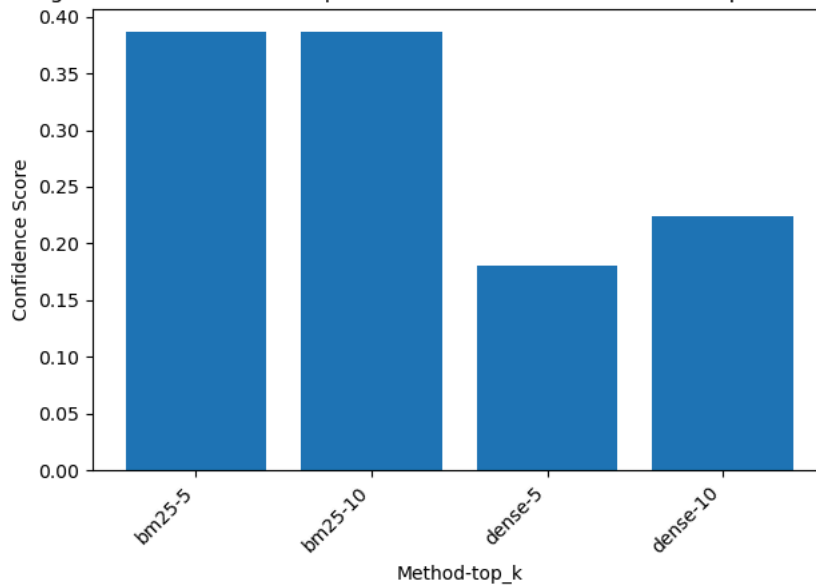
Interactive - Avg. Token Length for question: 'How can Generative AI help for image classifications?'



Interactive - Count of 'Contains Answer' for question: 'How can Generative AI help for image classifications?'



Interactive - Avg. Confidence Score for question: 'How can Generative AI help for image classifications?'



Exiting. Goodbye!

Observations and Conclusion

Observations:

1. Retrieval Method Comparison (BM25 vs Dense)

- BM25 tends to retrieve **longer and more interpretable answers**, while Dense retrieval occasionally provides **shorter or vague responses**.
- In some cases, BM25 answers are **consistent across different top_k values**, whereas Dense retrieval results **vary significantly** when increasing top_k.

2. Answer Quality and Relevance

- Some retrieved answers, such as *"artificial intelligence"* or *"Biomarker"*, *are overly generic and lack specific insights*.
- The **confidence scores** for many responses are **low (<0.5)**, indicating uncertainty in retrieval accuracy.
- Responses related to **Generative AI for image classification** provide reasonable detail, but some retrieved answers (e.g., *"patch-clamp recordings and molecular dynamic simulations"*) *seem out of context*.

3. Token Length and Answer Specificity

- Shorter answers (**1-2 tokens**) often lack specificity and **may not provide useful information** (e.g. *"AI"* *"Biomarker"*).