INFS7410 Project - Part 1

version 1.0

Preamble

The due date for this assignment is **9 September 2022 16:00 Eastern Australia Standard Time**.

This part of project is worth 20% of the overall mark for INFS7410 (part 1 + part 2 = 40%). A detailed marking sheet for this assignment is provided alongside this notebook. The project is to be completed individually.

We recommend that you make an early start on this assignment, and proceed by steps. There are a number of activities you may have already tackled, including setting up the pipeline, manipulating the queries, implement some retrieval functions, and performing evaluation and analysis. Most of the assignment relies on knowledge and code you should have already have experienced in the computer practicals, however there are some hidden challenges here and there that you may require some time to solve.

Aim

Project aim: The aim of this project is to implement a number of classical information retrieval methods, evaluate them and compare them in the context of a real use-case.

Project Part 1 aim

The aim of Part 1 is to:

- Setup your infrastructure to index the collection and evaluate queries.
- Implement common information retrieval baselines.
- Tune your retrieval implementations to improve their effectiveness.
- Implement rank fusion methods.

The Information Retrieval Task: Web Passage Ranking

In this project we will consider the problem of open-domain passage ranking in answer to web queries. In this context, users pose queries to the search engine and expect answers in the form of a ranked list of passages (maximum 1000 passages to be retrieved).

The provided queries are real queries submitted to the Microsoft Bing search engine. In the collection, there are approximately 8.8 million passages and the goal is to rank them based on their relevance to the queries.

What we provide you with:

Files from practical

- A collection of 8.8 million text passages extracted from web pages (`collection.tsv` — provided in Week 1).

- A query file that contains 43 queries for you to perform retrieval experiments (`queries.tsv` — provided in Week 2).
- A qrel file that contains relevance judgements for you to tune your methods (`qrels.txt` — provided in Week 2).

Extra files for this project

- A leaderboard system for you to evaluate how well your system performs.
- A test query file that contains 54 queries for you to generate run files to submit to the leaderboard (`test_queries.tsv`).
- This jupyter notebook, which you will include inside it your implementation and report.

Put this notebook and provided files under the same directory.

What you need to produce

You need to produce:

- Correct implementations of the methods required by this project specifications.
- An explanation of the retrieval methods used, including the formulas that represent the models you implemented and code that implements that formula, an explanation of the evaluation settings followed, and a discussion of the findings.

You are required to produce both of these within this jupyter notebook.

Required methods to implement

In Part 1 of the project you are required to implement the following retrieval methods. All implementations should be based on your own code.

1. BM25: Create your own implementation, do not use the Pynserini API's implementation. See the videos in Week 3 for background information.
2. Pseudo-relevance feedback using BM25 for query expansion: create your own implementation using the Pyserini API to extract index statistics. See Week 5 practical for background information.
3. IDF-r query reduction: create your own implementation using the Pyserini API to extract index statistics. See Week 5 practical for background information.
4. The rank fusion method Borda; you need to create your own implementation of this. See Week 4 practical for background information.
5. The rank fusion method CombSUM; you need to create your own implementation of this. See Week 4 practical for background information.
6. The rank fusion method CombMNZ; you need to create your own implementation of this. See Week 4 practical for background information.

For your BM25, query expansion and query reduction implementations, you are also required to tune the parameters of these methods. You must perform a parameter search over at least 5 sensibly chosen parameter values depending on the method (10 when the method has two parameters).

For the rank fusion methods, consider fusing the highest performing tuned run from each of the BM25, query expansion and query reduction implementations.

You should have already attempted many of these implementations above as part of the computer pracs exercises.

Required evaluation to perform

In Part 1 of the project you are required to perform the following evaluation:

1. For all methods, tune using `queries.tsv` and `qrels.txt` (i.e., use this data to tune any parameter of a retrieval model, e.g. b and $k1$ for BM25, etc.) and submit your runs on the `test_queries.tsv` using the parameter values you selected from the `queries.tsv` to the learderboard system.
2. Report the results of every method on the `queries.tsv` (only for the run you selected the tuned paramters from) separately, into a table. Perform statistical significance analysis across the results of the methods and report them in the tables.
3. Produce a gain-loss plot that compares BM25 vs. Pseudo-relevance feedback query expansion using BM25; and plots that compare BM25 vs. each rank fusion method on the dev set.
4. Comment on trends and differences observed when comparing your findings. Is there a method that consistently outperform the others on the `queries.tsv` and the `test_queries.tsv`?
5. Provide insights of whether rank fusion works, or if it does not, e.g., with respect to runs to be considered in the fusion process, queries, etc.

In terms of evaluation measures, evaluate the retrieval methods with respect to nDCG at 10 (`ndcg_cut_10`). You should use this measure as the target measure for tuning. Also compute reciprocal rank at 1000 (`recip_rank`), MAP (`map`) and Recall at 1000 (`recall_1000`).

For all gain-loss plots, produce them with respect to nDCG at 10.

For all statistical significance analysis, use paired t-test; distinguish between p<0.05 and p<0.01.

How to submit

You will have to submit one file:

1. A zip file containing this notebook (.ipynb) and this notebook **as a PDF document**. The code should be able to be executed by us. Remeber to include all your discussion and analysis also in this notebook and not as a separate file.

It needs to be submitted via the relevant Turnitin link in the INFS7410 BlackBoard site, by **3 September 2021, 16:00 Eastern Australia Standard Time**, unless you have been given an extension (according to UQ policy), *before* the due date of the assignment.

Leaderboard Challenge

As part of this project, we present you the leaderboard challenge, where you can submit your best runs and compare them with others. You can submit multiple times during the whole process of your project.

For models, you can use any retrieval or rerank or fusion methods (including those you implement for the pracs and for the assessment), even combine multiple models together, not limited to the models we have learned in this class. You are allowed to be creative and come up with your own retrieval or rerank models.

Although multiple run files can be submitted, we recommend you to run your models with the queries we provided in pracs first, only to check if your model is valid. Then you can run on the test queries for your project, and submit it to leaderboard.

The submission link is: https://infs7410.uqcloud.net/leaderboard/

After opening the link, you will see the main page as below:

main

The submitted runs will appear here. You can see we have already submitted a sample run as `pyserini_BM25`. The name of the run is in the format of `student name(runtag)`, so make sure you name your run correctly in the run file's 6th column. If you use the same run tag for all the runs you submit, the latter ones will overwrite the previous runs: you will only have one run at the end. So to submit different runs, make sure to use different run tags.

If you want to make a submission, type the password in the textbox, and hit enter. (**The password is your student number**) It will take you to your private submission page to submit your run, as below:

submit

In this page, choose the run file you wish to upload by clicking the `Browse` button, then click `Submit` to submit the run.

We highly encourage you to submit your best runs to the leaderboard, so you will know how your model performs compared to your classmates.

We will also give prizes to students who have the top 3 best runs. More specifically:

Leaderboard winners: Certificate of award to the students with the best 3 runs (winner, + 2 runner-up) in the leaderboard (test queries) by the time the assessment closes (end of semester). +2% final mark provided to the student whose run has the highest effectiveness (ndcg_cut_10), +1% final mark to the student whose run has the second highest effectiveness (ndcg_cut_10) in the leaderboard by the end of the semester. A/Prof. Guido can provide recommendation letters to students with top 3 runs.

This leaderboard will be running throughout the semester.

Have fun!

Initialise packages and functions

You will need to decide which index, stemming algorithm and keeping stopwords or not in the following cell. You may want to try out different indexes and if you don't have one, follow week1 prac to create one (remember to add `-storeDocvectors`).

In [6]:

```
stemming = None     # None or 'poter' or anything else
stopwords = True   # False or True
index = 'indexes/lucene-index-msmarco-passage-vectors-noProcessing/'
```

Run the following cell to load and cache some useful packages and statistics that you will use later.

In [7]:

```
from pyserini.search import SimpleSearcher
from pyserini.analysis import Analyzer, get_lucene_analyzer
from pyserini.index import IndexReader
from tqdm import tqdm

lucene_analyzer = get_lucene_analyzer(stemming=stemming, stopwords=stopwords)
```

```
analyzer = Analyzer(lucene_analyzer)
searcher = SimpleSearcher(index)
searcher.set_analyzer(lucene_analyzer)
index_reader = IndexReader(index)


# Create document frequency dictionary to speed up scoring later, this will take around
df_dict = {}
for term in tqdm(index_reader.terms(), desc="loading idf dictionary:"):
        df_dict[term.term] = term.df

# cache document length and docids for the collection, this will take around 2 mins.
doc_len_dict = {}
doc_id_dict = {}
with open ('collection/collection.tsv', 'r') as f:
    lines = f.readlines()
    for line in tqdm(lines, desc="loading doc_length dictionary:"):
        docid, text = line.split('\t')
        doc_len_dict[docid] = len(text.split())
        internal_id = index_reader.convert_collection_docid_to_internal_docid(docid)
        doc_id_dict[internal_id] = docid
```

```
loading idf dictionary:: 2897192it [00:31, 91118.75it/s]
loading doc_length dictionary:: 100%|█| 8841823/8841823 [01:24<00:00, 105057.76i
```

Understand and run the following cell to define the search function.

NOTE: This search function is different from the search function in week3 prac. When you implement methods yourself, make sure to use this search function which we implemented with iterating posting lists, do not use the search function from Week 3 prac, as the Week 3 prac is only re-ranking BM25 results.

In [8]:

```
def search(query: str, k_1: float, b: int,k: int=1000,scorer=None):
    """
    Inputs:
        query (str): the query string to perform the search.
        k (int): the number of documents to be returned.
        scorer: your implemented scoring function, such as bm25.

    Output:
        results (list): the sorted result list, a list of tuples.
                        The first element in the tuples is the docid,
                        the second is the doc score.
    """

    assert scorer is not None
    print("------------------------------------------------------")
    print("Current query:", query)

    # get the analyzed term list
    q_terms = analyzer.analyze(query)
    doc_socres = {}
    for term in q_terms:
        # get the posting list for the current term
        postings_list = index_reader.get_postings_list(term, analyzer=None)
        if postings_list is not None:
            # get the document frequency of the current term
            df = df_dict[term]

            # iterate the posting list
            for posting in tqdm(postings_list, desc=f"Iterate posting for term '{term}'"
                internal_id = posting.docid
                # convert pyserini internal docid to the actual docid
                docid = doc_id_dict[internal_id]
                tf = posting.tf
                doc_len = doc_len_dict[docid]

                # Call the scoring function (you will implement these below).
                avg_dl = index_reader.stats()["total_terms"]/index_reader.stats()["docum
                score = scorer(tf, df, doc_len,avg_dl,k_1,b)
                if docid in doc_socres.keys():
                    doc_socres[docid] += score
                else:
                    doc_socres[docid] = score

    # Sort the results by the score.
    results = [(docid, doc_socre) for docid, doc_socre in doc_socres.items()]
    results = sorted(results, key=lambda x: x[1], reverse=True)[:k]
    return results
    print("-----------------------------------------------------")
```

After this line, feel free to edit this notebook however you like. You can use the following cells as a template to guide you in completing this project.

In [66]:

```
# Import all your python libraries and put setup code here.
import pytrec_eval
import numpy as np
from sklearn.preprocessing import minmax_scale
import scipy.stats
import matplotlib.pyplot as plt
import numpy as np


def normalise_run(run):
    for k, v in run.items():
        r = [(docid, score) for docid, score in v.items()]
        scores = minmax_scale([x[1] for x in r])
        run[k] = dict(zip([x[0] for x in r], scores))
    return run

def bm25(tf, df, doc_len,avg_dl,k_1,b):
    TF = tf/doc_len
    IDF = np.log(N/1+df)
    BM25 = IDF*((TF*(k_1+1))/(TF+k_1*(1-b+b*(doc_len/avg_dl))))
    return BM25

def prf_query_expansion_bm25(query: str, n: int, m: int):
    hits = searcher.search(query, k=50)
    top_n_doc = hits[:n]
    doc_bm25 = {}
    for doc in top_n_doc:
        tf = index_reader.get_document_vector(doc.docid)
        bm25_vector = {term: index_reader.compute_bm25_term_weight(doc.docid, term, anal
        doc_bm25[doc.docid] = bm25_vector

    bm25 = {}
    for docid, result in doc_bm25.items():
        for term, score in result.items():
            if term not in bm25:
                bm25[term] = score
            else:
                bm25[term] += score

    ranked_doc = sorted(bm25.items(), key = lambda x: x[1], reverse = True)
    ranked_doc_top_m = dict(ranked_doc[:m])

    expanded_query = query
    for term, value in ranked_doc_top_m.items():
        new_term = ' '+ term
        expanded_query += new_term

    return expanded_query

N = index_reader.stats()["documents"]

def idfr_query_reduction(query: str, n: int):
    terms = analyzer.analyze(query)
    term_idf = {}
    for term in terms:
        if term not in term_idf:
            df = index_reader.get_term_counts(term, analyzer=None)[0]
            term_idf[term] = np.log(N/(df+1))

    ranked_doc = sorted(term_idf.items(), key = lambda x: x[1], reverse = True)
    ranked_doc_top_m = dict(ranked_doc[:n])

    pruned_query = " "
    for term, idf in ranked_doc_top_m.items():
        new_term = term + " "
        pruned_query += new_term

    return pruned_query

def borda(runs):
    seen = {}
    for run in runs:
        for topic, results in run.items():
            if topic not in seen:
                seen[topic] = {}
            for i, docid in enumerate(results.keys()):
                n = len(results)
                rd = i
                score = (n-rd+1)/n
                if docid not in seen[topic]:
                    seen[topic][docid] = score
                else:
                    seen[topic][docid] += score
    return seen

def combsum(runs):
    seen = {}
    for run in runs:
        for topic, results in run.items():
            if topic not in seen:
                seen[topic] = {}
            for docid, score in results.items():
                if docid not in seen[topic]:
                    seen[topic][docid] = score
                else:
                    seen[topic][docid] += score

    return seen

def combmnz(runs):
    seen = {}
    for run in runs:
        for topic, results in run.items():
            if topic not in seen:
                seen[topic] = {}
            for docid, score in results.items():
                if docid not in seen[topic]:
                    seen[topic][docid] = score
                else:
                    seen[topic][docid] += score

    doc_count_in_run = {}
    for run in runs:
        for topic, results in run.items():
            if topic not in doc_count_in_run:
                doc_count_in_run[topic] = {}
            for docid, score in results.items():
                if docid not in doc_count_in_run[topic]:
                    doc_count_in_run[topic][docid] = 1
                else:
                    doc_count_in_run[topic][docid] += 1

    combmnz = {}
    for topic, doc in seen.items():
        if topic not in combmnz:
            combmnz[topic] = {}
        for docid, score in doc.items():
            combmnz[topic][docid] = score*doc_count_in_run[topic][docid]

    return combmnz
```

For the first method, I have implemented BM25 with saturation k1 = 1.2 and document length normalisation = 0.75.

In [10]:

```
# Put your implementation of BM25 here, including parameter tuning.
queries = []
with open("queries.tsv", "r") as f:
    for line in f.readlines():
        parts = line.split("\t")
        # parts[0] ~> topic id
        # parts[1] ~> query
        queries.append((parts[0], parts[1].strip()))

# k_1 =1.2, b=0.75, k=10
bm_25 = []
for i in range(len(queries)):
    bm_25.append(search(queries[i][1], 1.2,0.75,1000, scorer=bm25))
```

```
-----------------------------------------------------
Current query: do goldfish grow
Iterate posting for term 'do': 100%|██| 479244/479244 [02:18<00:00, 3471.23it/s]
Iterate posting for term 'goldfish': 100%|█| 1393/1393 [00:00<00:00, 3201.36it/s
Iterate posting for term 'grow': 100%|██| 61594/61594 [00:17<00:00, 3483.41it/s]
-----------------------------------------------------
Current query: what is wifi vs bluetooth
Iterate posting for term 'what': 100%|█| 542299/542299 [02:34<00:00, 3504.46it/s
Iterate posting for term 'wifi': 100%|████| 4475/4475 [00:01<00:00, 3309.90it/s]
Iterate posting for term 'vs': 100%|████| 28413/28413 [00:08<00:00, 3486.22it/s]
Iterate posting for term 'bluetooth': 100%|█| 4585/4585 [00:01<00:00, 3555.84it/
-----------------------------------------------------
Current query: why did the us volunterilay enter ww1
Iterate posting for term 'why': 100%|█| 101667/101667 [00:28<00:00, 3565.57it/s]
Iterate posting for term 'did': 100%|███| 90156/90156 [00:27<00:00, 3330.16it/s]
Iterate posting for term 'us': 100%|██| 208281/208281 [01:01<00:00, 3376.86it/s]
Iterate posting for term 'enter': 100%|█| 65316/65316 [00:19<00:00, 3336.41it/s]
Iterate posting for term 'ww1': 100%|███████| 425/425 [00:00<00:00, 2938.09it/s]
-----------------------------------------------------
Current query: definition declaratory judgment
Iterate posting for term 'definition': 100%|█| 170760/170760 [00:49<00:00, 3446.
Iterate posting for term 'declaratory': 100%|█| 196/196 [00:00<00:00, 2709.64it/
Iterate posting for term 'judgment': 100%|█| 9868/9868 [00:02<00:00, 3404.39it/s
-----------------------------------------------------
```

```
Current query: right pelvic pain causes
Iterate posting for term 'right': 100%|█| 223259/223259 [01:04<00:00, 3462.16it/
Iterate posting for term 'pelvic': 100%|██| 7528/7528 [00:02<00:00, 3432.86it/s]
Iterate posting for term 'pain': 100%|█| 140027/140027 [00:40<00:00, 3460.26it/s
Iterate posting for term 'causes': 100%|█| 136109/136109 [00:40<00:00, 3373.87it
----------------------------------------------------
Current query: what are the social determinants of health
Iterate posting for term 'what': 100%|█| 542299/542299 [02:36<00:00, 3454.78it/s
Iterate posting for term 'social': 100%|█| 87525/87525 [00:25<00:00, 3450.49it/s
Iterate posting for term 'determinants': 100%|█| 798/798 [00:00<00:00, 3419.08it
Iterate posting for term 'health': 100%|█| 204625/204625 [01:00<00:00, 3360.07it
----------------------------------------------------
Current query: does legionella pneumophila cause pneumonia
Iterate posting for term 'does': 100%|█| 260662/260662 [01:17<00:00, 3369.40it/s
Iterate posting for term 'legionella': 100%|█| 350/350 [00:00<00:00, 3284.04it/s
Iterate posting for term 'pneumophila': 100%|█| 137/137 [00:00<00:00, 3464.15it/
Iterate posting for term 'cause': 100%|█| 227338/227338 [01:06<00:00, 3420.50it/
Iterate posting for term 'pneumonia': 100%|█| 8318/8318 [00:02<00:00, 3494.38it/
----------------------------------------------------
Current query: how is the weather in jamaica
Iterate posting for term 'how': 100%|█| 553931/553931 [02:40<00:00, 3454.88it/s]
Iterate posting for term 'weather': 100%|█| 87214/87214 [00:24<00:00, 3501.35it/
Iterate posting for term 'jamaica': 100%|█| 3779/3779 [00:01<00:00, 3548.39it/s]
----------------------------------------------------
Current query: types of dysarthria from cerebral palsy
Iterate posting for term 'types': 100%|█| 169667/169667 [00:48<00:00, 3492.45it/
Iterate posting for term 'dysarthria': 100%|█| 256/256 [00:00<00:00, 3608.74it/s
Iterate posting for term 'from': 100%|█| 1820726/1820726 [08:47<00:00, 3453.72it
Iterate posting for term 'cerebral': 100%|█| 6626/6626 [00:01<00:00, 3387.63it/s
Iterate posting for term 'palsy': 100%|███| 2534/2534 [00:00<00:00, 3489.41it/s]
----------------------------------------------------
Current query: who is robert gray
Iterate posting for term 'who': 100%|█| 479472/479472 [02:18<00:00, 3460.08it/s]
Iterate posting for term 'robert': 100%|█| 21332/21332 [00:06<00:00, 3394.47it/s
Iterate posting for term 'gray': 100%|██| 14829/14829 [00:04<00:00, 3395.65it/s]
----------------------------------------------------
Current query: what types of food can you cook sous vide
Iterate posting for term 'what': 100%|█| 542299/542299 [02:41<00:00, 3364.28it/s
Iterate posting for term 'types': 100%|█| 169667/169667 [00:48<00:00, 3500.47it/
Iterate posting for term 'food': 100%|█| 183952/183952 [00:54<00:00, 3351.99it/s
Iterate posting for term 'can': 100%|█| 1590108/1590108 [07:47<00:00, 3401.94it/
Iterate posting for term 'you': 100%|█| 1863876/1863876 [09:12<00:00, 3372.53it/
Iterate posting for term 'cook': 100%|██| 44859/44859 [00:12<00:00, 3450.85it/s]
Iterate posting for term 'sous': 100%|██████| 506/506 [00:00<00:00, 3383.24it/s]
Iterate posting for term 'vide': 100%|██████| 500/500 [00:00<00:00, 3560.12it/s]
----------------------------------------------------
Current query: how long is life cycle of flea
Iterate posting for term 'how': 100%|█| 553931/553931 [02:41<00:00, 3428.83it/s]
Iterate posting for term 'long': 100%|█| 286025/286025 [01:36<00:00, 2975.70it/s
Iterate posting for term 'life': 100%|█| 208672/208672 [16:24<00:00, 211.97it/s]
Iterate posting for term 'cycle': 100%|█| 41208/41208 [00:17<00:00, 2378.49it/s]
Iterate posting for term 'flea': 100%|████| 3425/3425 [00:01<00:00, 3134.83it/s]
----------------------------------------------------
Current query: what can contour plowing reduce
Iterate posting for term 'what': 100%|█| 542299/542299 [52:35<00:00, 171.83it/s]
Iterate posting for term 'can': 100%|█| 1590108/1590108 [1:48:48<00:00, 243.58it
Iterate posting for term 'contour': 100%|█| 1465/1465 [00:00<00:00, 3027.64it/s]
Iterate posting for term 'plowing': 100%|███| 338/338 [00:00<00:00, 3152.84it/s]
Iterate posting for term 'reduce': 100%|█| 61911/61911 [00:19<00:00, 3144.40it/s
----------------------------------------------------
Current query: when was the salvation army founded
Iterate posting for term 'when': 100%|█| 837432/837432 [04:21<00:00, 3196.54it/s
Iterate posting for term 'salvation': 100%|█| 1528/1528 [00:00<00:00, 3485.41it/
Iterate posting for term 'army': 100%|██| 29967/29967 [00:13<00:00, 2297.91it/s]
Iterate posting for term 'founded': 100%|█| 33188/33188 [00:10<00:00, 3131.42it/
-----------------------------------------------------
Current query: what is a active margin
Iterate posting for term 'what': 100%|█| 542299/542299 [02:48<00:00, 3227.29it/s
Iterate posting for term 'active': 100%|█| 57867/57867 [00:16<00:00, 3519.66it/s
Iterate posting for term 'margin': 100%|██| 7879/7879 [00:02<00:00, 3547.57it/s]
-----------------------------------------------------
Current query: difference between rn and bsn
Iterate posting for term 'difference': 100%|█| 75516/75516 [00:23<00:00, 3174.23
Iterate posting for term 'between': 100%|█| 416673/416673 [02:14<00:00, 3104.15i
Iterate posting for term 'rn': 100%|██████| 6217/6217 [00:01<00:00, 3209.90it/s]
Iterate posting for term 'bsn': 100%|█████| 2060/2060 [00:00<00:00, 3302.81it/s]
-----------------------------------------------------
Current query: medicare's definition of mechanical ventilation
Iterate posting for term 'medicare': 100%|█| 22024/22024 [03:49<00:00, 96.13it/s
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:03<00:00, 2704.
Iterate posting for term 'mechanical': 100%|█| 17693/17693 [00:05<00:00, 3019.92
Iterate posting for term 'ventilation': 100%|█| 4314/4314 [00:01<00:00, 3309.51i
-----------------------------------------------------
Current query: how to find the midsegment of a trapezoid
Iterate posting for term 'how': 100%|█| 553931/553931 [03:05<00:00, 2992.59it/s]
Iterate posting for term 'find': 100%|█| 272952/272952 [01:32<00:00, 2941.08it/s
Iterate posting for term 'midsegment': 100%|██| 38/38 [00:00<00:00, 2883.36it/s]
Iterate posting for term 'trapezoid': 100%|█| 611/611 [00:00<00:00, 3106.22it/s]
-----------------------------------------------------
Current query: what is an aml surveillance analyst
Iterate posting for term 'what': 100%|█| 542299/542299 [02:49<00:00, 3203.54it/s
Iterate posting for term 'aml': 100%|███████| 616/616 [00:00<00:00, 3354.13it/s]
Iterate posting for term 'surveillance': 100%|█| 2858/2858 [00:00<00:00, 3308.74
Iterate posting for term 'analyst': 100%|█| 6616/6616 [00:01<00:00, 3413.21it/s]
-----------------------------------------------------
Current query: what is the daily life of thai people
Iterate posting for term 'what': 100%|█| 542299/542299 [02:48<00:00, 3220.69it/s
Iterate posting for term 'daily': 100%|█| 81166/81166 [00:23<00:00, 3507.81it/s]
Iterate posting for term 'life': 100%|█| 208672/208672 [01:12<00:00, 2862.89it/s
Iterate posting for term 'thai': 100%|████| 2626/2626 [00:00<00:00, 3468.26it/s]
Iterate posting for term 'people': 100%|█| 446203/446203 [02:16<00:00, 3274.81it
-----------------------------------------------------
Current query: definition of a sigmet
Iterate posting for term 'definition': 100%|█| 170760/170760 [00:53<00:00, 3201.
Iterate posting for term 'sigmet': 100%|██████| 19/19 [00:00<00:00, 2176.95it/s]
-----------------------------------------------------
Current query: cost of interior concrete flooring
Iterate posting for term 'cost': 100%|█| 221704/221704 [01:06<00:00, 3341.43it/s
Iterate posting for term 'interior': 100%|█| 17121/17121 [00:04<00:00, 3461.11it
Iterate posting for term 'concrete': 100%|█| 21620/21620 [00:06<00:00, 3547.67it
Iterate posting for term 'flooring': 100%|█| 6960/6960 [00:01<00:00, 3601.11it/s
-----------------------------------------------------
Current query: what is the most popular food in switzerland
Iterate posting for term 'what': 100%|█| 542299/542299 [02:55<00:00, 3091.67it/s
Iterate posting for term 'most': 100%|█| 783486/783486 [04:00<00:00, 3263.45it/s
Iterate posting for term 'popular': 100%|█| 103940/103940 [00:32<00:00, 3211.54i
Iterate posting for term 'food': 100%|█| 183952/183952 [00:54<00:00, 3375.59it/s
Iterate posting for term 'switzerland': 100%|█| 6118/6118 [00:01<00:00, 3382.24i
-----------------------------------------------------
Current query: how are some sharks warm blooded
Iterate posting for term 'how': 100%|█| 553931/553931 [02:50<00:00, 3258.05it/s]
Iterate posting for term 'some': 100%|█| 586484/586484 [03:01<00:00, 3226.72it/s
Iterate posting for term 'sharks': 100%|██| 3549/3549 [00:01<00:00, 3467.59it/s]
Iterate posting for term 'warm': 100%|██| 48286/48286 [00:14<00:00, 3301.86it/s]
Iterate posting for term 'blooded': 100%|█| 1506/1506 [00:00<00:00, 3570.85it/s]
-----------------------------------------------------
Current query: what is durable medical equipment consist of
Iterate posting for term 'what': 100%|█| 542299/542299 [02:52<00:00, 3150.42it/s
Iterate posting for term 'durable': 100%|█| 8934/8934 [00:02<00:00, 3200.34it/s]
Iterate posting for term 'medical': 100%|█| 147835/147835 [00:41<00:00, 3556.19i
Iterate posting for term 'equipment': 100%|█| 40506/40506 [00:11<00:00, 3404.10i
Iterate posting for term 'consist': 100%|█| 12546/12546 [00:03<00:00, 3398.55it/
-----------------------------------------------------
Current query: exons definition biology
Iterate posting for term 'exons': 100%|█████| 361/361 [00:00<00:00, 3328.60it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:00<00:00, 2803.
Iterate posting for term 'biology': 100%|█| 14352/14352 [00:04<00:00, 3434.62it/
-----------------------------------------------------
Current query: define visceral?
Iterate posting for term 'define': 100%|█| 25586/25586 [00:07<00:00, 3529.46it/s
Iterate posting for term 'visceral': 100%|█| 2487/2487 [00:00<00:00, 3546.46it/s
-----------------------------------------------------
Current query: tracheids are part of _____.
Iterate posting for term 'tracheids': 100%|█| 125/125 [00:00<00:00, 3529.09it/s]
Iterate posting for term 'part': 100%|█| 289955/289955 [01:33<00:00, 3100.74it/s
-----------------------------------------------------
Current query: rsa definition key
Iterate posting for term 'rsa': 100%|███████| 597/597 [00:00<00:00, 3439.24it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [00:55<00:00, 3081.
Iterate posting for term 'key': 100%|███| 78308/78308 [00:26<00:00, 2915.37it/s]
-----------------------------------------------------
Current query: who formed the commonwealth of independent states
Iterate posting for term 'who': 100%|█| 479472/479472 [02:39<00:00, 3003.88it/s]
Iterate posting for term 'formed': 100%|█| 49002/49002 [00:14<00:00, 3377.21it/s
Iterate posting for term 'commonwealth': 100%|█| 5186/5186 [00:01<00:00, 3498.89
Iterate posting for term 'independent': 100%|█| 33631/33631 [00:09<00:00, 3525.9
Iterate posting for term 'states': 100%|█| 302247/302247 [01:35<00:00, 3158.09it
-----------------------------------------------------
Current query: causes of left ventricular hypertrophy
Iterate posting for term 'causes': 100%|█| 136109/136109 [00:38<00:00, 3531.86it
Iterate posting for term 'left': 100%|█| 115188/115188 [00:37<00:00, 3069.85it/s
Iterate posting for term 'ventricular': 100%|█| 3361/3361 [00:01<00:00, 3223.60i
Iterate posting for term 'hypertrophy': 100%|█| 1183/1183 [00:00<00:00, 3454.02i
-----------------------------------------------------
Current query: lps laws definition
Iterate posting for term 'lps': 100%|███████| 457/457 [00:00<00:00, 3503.71it/s]
Iterate posting for term 'laws': 100%|██| 37074/37074 [00:10<00:00, 3539.06it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [00:54<00:00, 3144.
-----------------------------------------------------
Current query: what are the three percenters?
Iterate posting for term 'what': 100%|█| 542299/542299 [03:01<00:00, 2993.26it/s
Iterate posting for term 'three': 100%|█| 260225/260225 [01:26<00:00, 3025.71it/
Iterate posting for term 'percenters': 100%|██| 37/37 [00:00<00:00, 3243.38it/s]
-----------------------------------------------------
Current query: causes of military suicide
Iterate posting for term 'causes': 100%|█| 136109/136109 [00:41<00:00, 3241.07it
Iterate posting for term 'military': 100%|█| 43218/43218 [00:12<00:00, 3391.07it
Iterate posting for term 'suicide': 100%|█| 6359/6359 [00:01<00:00, 3376.47it/s]
-----------------------------------------------------
Current query: what is theraderm used for
Iterate posting for term 'what': 100%|█| 542299/542299 [02:50<00:00, 3178.42it/s
Iterate posting for term 'theraderm': 100%|█████| 5/5 [00:00<00:00, 2577.30it/s]
Iterate posting for term 'used': 100%|█| 589843/589843 [1:06:40<00:00, 147.44it/
-----------------------------------------------------
Current query: what is famvir prescribed for
```

```
Iterate posting for term 'what': 100%|█| 542299/542299 [02:59<00:00, 3018.90it/s
Iterate posting for term 'famvir': 100%|██████| 59/59 [00:00<00:00, 2981.13it/s]
Iterate posting for term 'prescribed': 100%|█| 17263/17263 [00:09<00:00, 1841.79
-----------------------------------------------------
Current query: anthropological definition of environment
Iterate posting for term 'anthropological': 100%|█| 254/254 [00:00<00:00, 3403.6
Iterate posting for term 'definition': 100%|█| 170760/170760 [00:48<00:00, 3536.
Iterate posting for term 'environment': 100%|█| 50134/50134 [00:19<00:00, 2614.3
-----------------------------------------------------
Current query: axon terminals or synaptic knob definition
Iterate posting for term 'axon': 100%|████| 2677/2677 [00:00<00:00, 3500.95it/s]
Iterate posting for term 'terminals': 100%|█| 3680/3680 [00:01<00:00, 3539.34it/
Iterate posting for term 'synaptic': 100%|█| 1292/1292 [00:00<00:00, 3542.48it/s
Iterate posting for term 'knob': 100%|████| 2023/2023 [00:00<00:00, 3575.42it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [00:58<00:00, 2905.
-----------------------------------------------------
Current query: is cdg airport in main paris
Iterate posting for term 'cdg': 100%|███████| 190/190 [00:00<00:00, 3109.50it/s]
Iterate posting for term 'airport': 100%|█| 36250/36250 [00:10<00:00, 3533.71it/
Iterate posting for term 'main': 100%|█| 127477/127477 [00:47<00:00, 2708.42it/s
Iterate posting for term 'paris': 100%|█| 13242/13242 [00:03<00:00, 3334.17it/s]
-----------------------------------------------------
Current query: example of monotonic function
Iterate posting for term 'example': 100%|█| 234474/234474 [01:17<00:00, 3020.68i
Iterate posting for term 'monotonic': 100%|███| 68/68 [00:00<00:00, 2807.16it/s]
Iterate posting for term 'function': 100%|█| 89643/89643 [00:25<00:00, 3487.52it
-----------------------------------------------------
Current query: what is physical description of spruce
Iterate posting for term 'what': 100%|█| 542299/542299 [02:54<00:00, 3115.17it/s
Iterate posting for term 'physical': 100%|█| 77532/77532 [00:26<00:00, 2950.26it
Iterate posting for term 'description': 100%|█| 40874/40874 [00:12<00:00, 3400.4
Iterate posting for term 'spruce': 100%|██| 1771/1771 [00:00<00:00, 3388.44it/s]
-----------------------------------------------------
Current query: hydrogen is a liquid below what temperature
Iterate posting for term 'hydrogen': 100%|█| 22860/22860 [00:06<00:00, 3433.87it
Iterate posting for term 'liquid': 100%|█| 48415/48415 [00:19<00:00, 2511.88it/s
Iterate posting for term 'below': 100%|█| 206486/206486 [01:05<00:00, 3150.23it/
Iterate posting for term 'what': 100%|█| 542299/542299 [02:55<00:00, 3092.23it/s
Iterate posting for term 'temperature': 100%|█| 121196/121196 [00:34<00:00, 3520
-----------------------------------------------------
Current query: difference between a mcdouble and a double cheeseburger
Iterate posting for term 'difference': 100%|█| 75516/75516 [00:25<00:00, 2938.36
Iterate posting for term 'between': 100%|█| 416673/416673 [02:12<00:00, 3142.34i
Iterate posting for term 'mcdouble': 100%|████| 24/24 [00:00<00:00, 3165.41it/s]
Iterate posting for term 'double': 100%|█| 45518/45518 [00:17<00:00, 2572.07it/s
Iterate posting for term 'cheeseburger': 100%|█| 536/536 [00:00<00:00, 3380.96it
```

For upcoming methods, I have implemented Pseudo-relevance feedback using BM25 for query expansion and IDF-r query reduction with different parameters.For query expansion, I have kept the top documents as 5 and variated the bewteen top terms as 10 and 15. For the query reduction, I have reduced the query to 3 words and 5 words respectively.

In [16]:

```python
expand_bm25_add_top10 = []
expand_bm25_add_top5 = []
reduce_bm25_5_leave = []
reduce_bm25_3_leave = []
for i in range(len(queries)):
    expand_bm25_add_top10.append(search(prf_query_expansion_bm25(queries[i][1],5,10), 1.
    expand_bm25_add_top5.append(search(prf_query_expansion_bm25(queries[i][1],5,15), 1.2
    reduce_bm25_5_leave.append(search(idfr_query_reduction(queries[i][1],5), 1.2,0.75,10
    reduce_bm25_3_leave.append(search(idfr_query_reduction(queries[i][1],3), 1.2,0.75,10
-----------------------------------------------------
Current query: do goldfish grow goldfish grow tank shubunkin stunted comet feeder unheal
thy fish fancy
```

```
Iterate posting for term 'do': 100%|██| 479244/479244 [02:36<00:00, 3072.07it/s]
Iterate posting for term 'goldfish': 100%|█| 1393/1393 [00:00<00:00, 3447.87it/s
Iterate posting for term 'grow': 100%|██| 61594/61594 [00:22<00:00, 2766.10it/s]
Iterate posting for term 'goldfish': 100%|█| 1393/1393 [00:00<00:00, 3410.58it/s
Iterate posting for term 'grow': 100%|██| 61594/61594 [00:17<00:00, 3537.15it/s]
Iterate posting for term 'tank': 100%|██| 17532/17532 [00:04<00:00, 3596.88it/s]
Iterate posting for term 'shubunkin': 100%|███| 25/25 [00:00<00:00, 2833.22it/s]
Iterate posting for term 'stunted': 100%|███| 500/500 [00:00<00:00, 3560.17it/s]
Iterate posting for term 'comet': 100%|███| 1190/1190 [00:00<00:00, 3507.55it/s]
Iterate posting for term 'feeder': 100%|██| 1752/1752 [00:00<00:00, 3494.45it/s]
Iterate posting for term 'unhealthy': 100%|█| 3515/3515 [00:00<00:00, 3530.11it/
Iterate posting for term 'fish': 100%|██| 50342/50342 [00:14<00:00, 3504.51it/s]
Iterate posting for term 'fancy': 100%|███| 4448/4448 [00:01<00:00, 3437.34it/s]
```

```
-----------------------------------------------------
Current query: do goldfish grow goldfish grow tank shubunkin stunted comet feeder unheal
thy fish fancy do growth die quot stunting
```

```
Iterate posting for term 'do': 100%|██| 479244/479244 [02:30<00:00, 3189.09it/s]
Iterate posting for term 'goldfish': 100%|█| 1393/1393 [00:00<00:00, 3450.88it/s
Iterate posting for term 'grow': 100%|██| 61594/61594 [00:17<00:00, 3538.33it/s]
Iterate posting for term 'goldfish': 100%|█| 1393/1393 [00:00<00:00, 3550.88it/s
Iterate posting for term 'grow': 100%|██| 61594/61594 [00:18<00:00, 3272.04it/s]
Iterate posting for term 'tank': 100%|██| 17532/17532 [00:05<00:00, 3447.27it/s]
Iterate posting for term 'shubunkin': 100%|███| 25/25 [00:00<00:00, 3253.92it/s]
Iterate posting for term 'stunted': 100%|███| 500/500 [00:00<00:00, 3455.69it/s]
Iterate posting for term 'comet': 100%|███| 1190/1190 [00:00<00:00, 3206.22it/s]
Iterate posting for term 'feeder': 100%|██| 1752/1752 [00:00<00:00, 3482.73it/s]
Iterate posting for term 'unhealthy': 100%|█| 3515/3515 [00:07<00:00, 497.97it/s
Iterate posting for term 'fish': 100%|██| 50342/50342 [00:15<00:00, 3329.81it/s]
Iterate posting for term 'fancy': 100%|████| 4448/4448 [00:07<00:00, 630.25it/s]
Iterate posting for term 'do': 100%|██| 479244/479244 [02:38<00:00, 3023.71it/s]
Iterate posting for term 'growth': 100%|█| 72399/72399 [00:20<00:00, 3463.43it/s
Iterate posting for term 'die': 100%|███| 21015/21015 [00:05<00:00, 3560.58it/s]
Iterate posting for term 'quot': 100%|████| 1190/1190 [00:00<00:00, 3456.57it/s]
Iterate posting for term 'stunting': 100%|██| 138/138 [00:00<00:00, 3431.41it/s]
```

```
-----------------------------------------------------
Current query:  goldfish grow do
```

```
Iterate posting for term 'goldfish': 100%|█| 1393/1393 [00:00<00:00, 3617.45it/s
Iterate posting for term 'grow': 100%|██| 61594/61594 [00:25<00:00, 2456.86it/s]
Iterate posting for term 'do': 100%|██| 479244/479244 [02:27<00:00, 3238.69it/s]
```

```
-----------------------------------------------------
Current query:  goldfish grow do
```

```
Iterate posting for term 'goldfish': 100%|█| 1393/1393 [00:00<00:00, 3499.56it/s
Iterate posting for term 'grow': 100%|██| 61594/61594 [00:22<00:00, 2770.12it/s]
Iterate posting for term 'do': 100%|██| 479244/479244 [02:31<00:00, 3154.94it/s]
```

```
-----------------------------------------------------
Current query: what is wifi vs bluetooth wifi bluetooth antenna lightinthebox router ext
ender fairlyâ tethering sonos wireless
```

```
Iterate posting for term 'what': 100%|█| 542299/542299 [03:02<00:00, 2977.53it/s
Iterate posting for term 'wifi': 100%|████| 4475/4475 [00:01<00:00, 3535.34it/s]
Iterate posting for term 'vs': 100%|████| 28413/28413 [00:08<00:00, 3532.81it/s]
Iterate posting for term 'bluetooth': 100%|█| 4585/4585 [00:01<00:00, 3577.54it/
Iterate posting for term 'wifi': 100%|████| 4475/4475 [00:01<00:00, 3572.83it/s]
Iterate posting for term 'bluetooth': 100%|█| 4585/4585 [00:01<00:00, 3605.22it/
Iterate posting for term 'antenna': 100%|█| 2826/2826 [00:00<00:00, 3617.58it/s]
Iterate posting for term 'lightinthebox': 100%|█| 8/8 [00:00<00:00, 3005.05it/s]
Iterate posting for term 'router': 100%|██| 8861/8861 [00:02<00:00, 3584.58it/s]
Iterate posting for term 'extender': 100%|██| 646/646 [00:00<00:00, 3597.12it/s]
Iterate posting for term 'fairlyâ': 100%|█████| 10/10 [00:00<00:00, 2974.26it/s]
Iterate posting for term 'tethering': 100%|█| 324/324 [00:00<00:00, 3652.77it/s]
Iterate posting for term 'sonos': 100%|█████| 204/204 [00:00<00:00, 3498.74it/s]
Iterate posting for term 'wireless': 100%|█| 18023/18023 [00:05<00:00, 3577.12it
```

```
-----------------------------------------------------
Current query: what is wifi vs bluetooth wifi bluetooth antenna lightinthebox router ext
ender fairlyâ tethering sonos wireless vs piconet connected range upto
Iterate posting for term 'what': 100%|█| 542299/542299 [03:05<00:00, 2917.16it/s
Iterate posting for term 'wifi': 100%|████| 4475/4475 [00:01<00:00, 3268.39it/s]
Iterate posting for term 'vs': 100%|████| 28413/28413 [00:08<00:00, 3435.85it/s]
Iterate posting for term 'bluetooth': 100%|█| 4585/4585 [00:01<00:00, 3568.86it/
Iterate posting for term 'wifi': 100%|████| 4475/4475 [00:01<00:00, 3633.74it/s]
Iterate posting for term 'bluetooth': 100%|█| 4585/4585 [00:01<00:00, 3604.64it/
Iterate posting for term 'antenna': 100%|█| 2826/2826 [00:00<00:00, 3586.63it/s]
Iterate posting for term 'lightinthebox': 100%|█| 8/8 [00:00<00:00, 2635.03it/s]
Iterate posting for term 'router': 100%|██| 8861/8861 [00:02<00:00, 3542.02it/s]
Iterate posting for term 'extender': 100%|██| 646/646 [00:00<00:00, 3674.86it/s]
Iterate posting for term 'fairlyâ': 100%|█████| 10/10 [00:00<00:00, 2710.55it/s]
Iterate posting for term 'tethering': 100%|█| 324/324 [00:00<00:00, 3612.28it/s]
Iterate posting for term 'sonos': 100%|█████| 204/204 [00:00<00:00, 3321.45it/s]
Iterate posting for term 'wireless': 100%|█| 18023/18023 [00:05<00:00, 3575.52it
Iterate posting for term 'vs': 100%|████| 28413/28413 [00:08<00:00, 3539.91it/s]
Iterate posting for term 'piconet': 100%|█████| 10/10 [00:00<00:00, 2775.66it/s]
Iterate posting for term 'connected': 100%|█| 30257/30257 [00:09<00:00, 3282.23i
Iterate posting for term 'range': 100%|█| 157499/157499 [00:54<00:00, 2884.33it/
Iterate posting for term 'upto': 100%|██████| 827/827 [00:00<00:00, 3455.76it/s]
-----------------------------------------------------
Current query:  wifi bluetooth vs what
Iterate posting for term 'wifi': 100%|████| 4475/4475 [00:01<00:00, 3504.53it/s]
Iterate posting for term 'bluetooth': 100%|█| 4585/4585 [00:01<00:00, 3480.84it/
Iterate posting for term 'vs': 100%|████| 28413/28413 [00:16<00:00, 1684.53it/s]
Iterate posting for term 'what': 100%|█| 542299/542299 [02:47<00:00, 3243.22it/s
-----------------------------------------------------
Current query:  wifi bluetooth vs
Iterate posting for term 'wifi': 100%|████| 4475/4475 [00:01<00:00, 3529.24it/s]
Iterate posting for term 'bluetooth': 100%|█| 4585/4585 [00:01<00:00, 3562.06it/
Iterate posting for term 'vs': 100%|████| 28413/28413 [00:15<00:00, 1813.94it/s]
-----------------------------------------------------
Current query: why did the us volunterilay enter ww1 ww1 why did germany britain ww2 war
outline.thanks them.ammianus °ð
Iterate posting for term 'why': 100%|█| 101667/101667 [00:30<00:00, 3327.93it/s]
Iterate posting for term 'did': 100%|███| 90156/90156 [00:33<00:00, 2669.66it/s]
Iterate posting for term 'us': 100%|██| 208281/208281 [00:59<00:00, 3511.40it/s]
Iterate posting for term 'enter': 100%|█| 65316/65316 [00:26<00:00, 2442.71it/s]
Iterate posting for term 'ww1': 100%|███████| 425/425 [00:00<00:00, 3260.68it/s]
Iterate posting for term 'ww1': 100%|███████| 425/425 [00:00<00:00, 3407.74it/s]
Iterate posting for term 'why': 100%|█| 101667/101667 [00:29<00:00, 3461.79it/s]
Iterate posting for term 'did': 100%|███| 90156/90156 [00:25<00:00, 3557.97it/s]
Iterate posting for term 'germany': 100%|█| 26318/26318 [00:07<00:00, 3436.18it/
Iterate posting for term 'britain': 100%|█| 17335/17335 [00:11<00:00, 1573.86it/
Iterate posting for term 'ww2': 100%|███████| 684/684 [00:00<00:00, 3250.27it/s]
Iterate posting for term 'war': 100%|███| 80223/80223 [00:22<00:00, 3522.80it/s]
Iterate posting for term 'outline.thanks': 100%|█| 1/1 [00:00<00:00, 1996.34it/s
Iterate posting for term 'them.ammianus': 100%|█| 2/2 [00:00<00:00, 2805.55it/s]
Iterate posting for term '°ð': 100%|██████████| 23/23 [00:00<00:00, 3049.54it/s]
-----------------------------------------------------
Current query: why did the us volunterilay enter ww1 ww1 why did germany britain ww2 war
outline.thanks them.ammianus °ð isolationists ammo belgium ago.e were
Iterate posting for term 'why': 100%|█| 101667/101667 [00:29<00:00, 3450.41it/s]
Iterate posting for term 'did': 100%|███| 90156/90156 [00:27<00:00, 3338.56it/s]
Iterate posting for term 'us': 100%|██| 208281/208281 [01:05<00:00, 3166.69it/s]
Iterate posting for term 'enter': 100%|█| 65316/65316 [00:34<00:00, 1868.10it/s]
Iterate posting for term 'ww1': 100%|███████| 425/425 [00:00<00:00, 3443.94it/s]
Iterate posting for term 'ww1': 100%|███████| 425/425 [00:00<00:00, 3410.31it/s]
Iterate posting for term 'why': 100%|█| 101667/101667 [00:29<00:00, 3405.16it/s]
Iterate posting for term 'did': 100%|███| 90156/90156 [00:36<00:00, 2497.08it/s]
Iterate posting for term 'germany': 100%|█| 26318/26318 [00:07<00:00, 3409.66it/
Iterate posting for term 'britain': 100%|█| 17335/17335 [00:05<00:00, 3036.33it/
Iterate posting for term 'ww2': 100%|███████| 684/684 [00:00<00:00, 2928.70it/s]
Iterate posting for term 'war': 100%|███| 80223/80223 [00:23<00:00, 3483.80it/s]
Iterate posting for term 'outline.thanks': 100%|█| 1/1 [00:00<00:00, 1749.08it/s
Iterate posting for term 'them.ammianus': 100%|█| 2/2 [00:00<00:00, 2840.71it/s]
Iterate posting for term '°ð': 100%|██████████| 23/23 [00:00<00:00, 3224.02it/s]
Iterate posting for term 'isolationists': 100%|█| 25/25 [00:00<00:00, 3236.25it/
Iterate posting for term 'ammo': 100%|██████| 561/561 [00:00<00:00, 3423.47it/s]
Iterate posting for term 'belgium': 100%|█| 5094/5094 [00:01<00:00, 3517.66it/s]
Iterate posting for term 'ago.e': 100%|███████| 35/35 [00:00<00:00, 2907.75it/s]
Iterate posting for term 'were': 100%|█| 313869/313869 [01:45<00:00, 2983.88it/s
-----------------------------------------------------
Current query:  volunterilay ww1 enter did why
Iterate posting for term 'ww1': 100%|███████| 425/425 [00:00<00:00, 3576.90it/s]
Iterate posting for term 'enter': 100%|█| 65316/65316 [00:28<00:00, 2303.89it/s]
Iterate posting for term 'did': 100%|███| 90156/90156 [00:26<00:00, 3432.08it/s]
Iterate posting for term 'why': 100%|█| 101667/101667 [00:29<00:00, 3471.78it/s]
-----------------------------------------------------
Current query:  volunterilay ww1 enter
Iterate posting for term 'ww1': 100%|███████| 425/425 [00:00<00:00, 3467.29it/s]
Iterate posting for term 'enter': 100%|█| 65316/65316 [00:18<00:00, 3453.62it/s]
-----------------------------------------------------
Current query: definition declaratory judgment declaratory judgment remedy relief court
action parties seek injunctive 2201
Iterate posting for term 'definition': 100%|█| 170760/170760 [00:56<00:00, 3015.
Iterate posting for term 'declaratory': 100%|█| 196/196 [00:00<00:00, 3512.73it/
Iterate posting for term 'judgment': 100%|█| 9868/9868 [00:02<00:00, 3600.22it/s
Iterate posting for term 'declaratory': 100%|█| 196/196 [00:00<00:00, 3428.51it/
Iterate posting for term 'judgment': 100%|█| 9868/9868 [00:02<00:00, 3594.79it/s
Iterate posting for term 'remedy': 100%|██| 6773/6773 [00:01<00:00, 3545.91it/s]
Iterate posting for term 'relief': 100%|█| 19047/19047 [00:05<00:00, 3588.14it/s
Iterate posting for term 'court': 100%|█| 65759/65759 [00:18<00:00, 3605.39it/s]
Iterate posting for term 'action': 100%|█| 61250/61250 [00:18<00:00, 3400.13it/s
Iterate posting for term 'parties': 100%|█| 18367/18367 [00:05<00:00, 3395.19it/
Iterate posting for term 'seek': 100%|██| 15104/15104 [00:04<00:00, 3399.73it/s]
Iterate posting for term 'injunctive': 100%|██| 61/61 [00:00<00:00, 3125.99it/s]
Iterate posting for term '2201': 100%|████████| 81/81 [00:00<00:00, 3261.51it/s]
-----------------------------------------------------
Current query: definition declaratory judgment declaratory judgment remedy relief court
action parties seek injunctive 2201 discretion govern preclude conclusive 9.3
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:01<00:00, 2761.
Iterate posting for term 'declaratory': 100%|█| 196/196 [00:00<00:00, 3515.69it/
Iterate posting for term 'judgment': 100%|█| 9868/9868 [00:02<00:00, 3510.49it/s
Iterate posting for term 'declaratory': 100%|█| 196/196 [00:00<00:00, 3515.78it/
Iterate posting for term 'judgment': 100%|█| 9868/9868 [00:02<00:00, 3592.77it/s
Iterate posting for term 'remedy': 100%|██| 6773/6773 [00:01<00:00, 3566.50it/s]
Iterate posting for term 'relief': 100%|█| 19047/19047 [00:05<00:00, 3558.24it/s
Iterate posting for term 'court': 100%|█| 65759/65759 [00:19<00:00, 3435.30it/s]
Iterate posting for term 'action': 100%|█| 61250/61250 [00:29<00:00, 2051.29it/s
Iterate posting for term 'parties': 100%|█| 18367/18367 [00:05<00:00, 3294.00it/
Iterate posting for term 'seek': 100%|██| 15104/15104 [00:04<00:00, 3387.98it/s]
Iterate posting for term 'injunctive': 100%|██| 61/61 [00:00<00:00, 2829.16it/s]
Iterate posting for term '2201': 100%|████████| 81/81 [00:00<00:00, 3102.15it/s]
Iterate posting for term 'discretion': 100%|█| 2855/2855 [00:00<00:00, 3398.92it
Iterate posting for term 'govern': 100%|██| 2924/2924 [00:00<00:00, 3416.04it/s]
Iterate posting for term 'preclude': 100%|██| 422/422 [00:00<00:00, 3466.72it/s]
Iterate posting for term 'conclusive': 100%|█| 732/732 [00:00<00:00, 3363.35it/s
Iterate posting for term '9.3': 100%|█████| 1007/1007 [00:00<00:00, 3369.40it/s]
-----------------------------------------------------
Current query:  declaratory judgment definition
Iterate posting for term 'declaratory': 100%|█| 196/196 [00:00<00:00, 3611.25it/
Iterate posting for term 'judgment': 100%|█| 9868/9868 [00:02<00:00, 3535.32it/s
Iterate posting for term 'definition': 100%|█| 170760/170760 [00:49<00:00, 3451.
-----------------------------------------------------
Current query:  declaratory judgment definition
```

```
Iterate posting for term 'declaratory': 100%|█| 196/196 [00:00<00:00, 3531.97it/
Iterate posting for term 'judgment': 100%|█| 9868/9868 [00:02<00:00, 3514.30it/s
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:03<00:00, 2693.
```

```
-----------------------------------------------------
Current query: right pelvic pain causes pelvic abdominal pain causes pathologies symptom
s continuation musculoskeletal lower discussed
```

```
Iterate posting for term 'right': 100%|█| 223259/223259 [01:20<00:00, 2776.31it/
Iterate posting for term 'pelvic': 100%|██| 7528/7528 [00:02<00:00, 3268.23it/s]
Iterate posting for term 'pain': 100%|█| 140027/140027 [00:48<00:00, 2877.53it/s
Iterate posting for term 'causes': 100%|█| 136109/136109 [00:45<00:00, 2979.69it
Iterate posting for term 'pelvic': 100%|███| 7528/7528 [00:07<00:00, 949.49it/s]
Iterate posting for term 'abdominal': 100%|█| 22220/22220 [00:06<00:00, 3339.79i
Iterate posting for term 'pain': 100%|█| 140027/140027 [00:47<00:00, 2976.07it/s
Iterate posting for term 'causes': 100%|█| 136109/136109 [00:46<00:00, 2942.93it
Iterate posting for term 'pathologies': 100%|█| 263/263 [00:00<00:00, 3448.37it/
Iterate posting for term 'symptoms': 100%|█| 163240/163240 [00:54<00:00, 2993.84
Iterate posting for term 'continuation': 100%|█| 2200/2200 [00:00<00:00, 3099.79
Iterate posting for term 'musculoskeletal': 100%|█| 2466/2466 [00:00<00:00, 3370
Iterate posting for term 'lower': 100%|█| 133174/133174 [00:44<00:00, 2988.24it/
Iterate posting for term 'discussed': 100%|█| 11070/11070 [00:03<00:00, 3232.75i
```

```
-----------------------------------------------------
Current query: right pelvic pain causes pelvic abdominal pain causes pathologies symptom
s continuation musculoskeletal lower discussed 43 cavity 39 digestive sexual
```

```
Iterate posting for term 'right': 100%|█| 223259/223259 [01:20<00:00, 2759.42it/
Iterate posting for term 'pelvic': 100%|██| 7528/7528 [00:02<00:00, 3337.31it/s]
Iterate posting for term 'pain': 100%|█| 140027/140027 [00:39<00:00, 3517.66it/s
Iterate posting for term 'causes': 100%|█| 136109/136109 [00:39<00:00, 3439.89it
Iterate posting for term 'pelvic': 100%|██| 7528/7528 [00:02<00:00, 3472.63it/s]
Iterate posting for term 'abdominal': 100%|█| 22220/22220 [00:06<00:00, 3551.33i
Iterate posting for term 'pain': 100%|█| 140027/140027 [00:52<00:00, 2685.78it/s
Iterate posting for term 'causes': 100%|█| 136109/136109 [00:45<00:00, 2967.43it
Iterate posting for term 'pathologies': 100%|█| 263/263 [00:00<00:00, 3463.04it/
Iterate posting for term 'symptoms': 100%|█| 163240/163240 [00:54<00:00, 2978.71
Iterate posting for term 'continuation': 100%|█| 2200/2200 [00:00<00:00, 3427.20
Iterate posting for term 'musculoskeletal': 100%|█| 2466/2466 [00:00<00:00, 3426
Iterate posting for term 'lower': 100%|█| 133174/133174 [00:44<00:00, 2979.28it/
Iterate posting for term 'discussed': 100%|█| 11070/11070 [00:03<00:00, 3408.37i
Iterate posting for term '43': 100%|████| 12241/12241 [00:03<00:00, 3467.84it/s]
Iterate posting for term 'cavity': 100%|█| 14528/14528 [00:04<00:00, 3260.95it/s
Iterate posting for term '39': 100%|████| 17185/17185 [00:05<00:00, 3433.06it/s]
Iterate posting for term 'digestive': 100%|█| 22792/22792 [00:06<00:00, 3539.30i
Iterate posting for term 'sexual': 100%|█| 23778/23778 [00:06<00:00, 3584.72it/s
```

```
-----------------------------------------------------
Current query:  pelvic causes pain right
```

```
Iterate posting for term 'pelvic': 100%|██| 7528/7528 [00:02<00:00, 3575.34it/s]
Iterate posting for term 'causes': 100%|█| 136109/136109 [00:47<00:00, 2855.07it
Iterate posting for term 'pain': 100%|█| 140027/140027 [00:47<00:00, 2931.59it/s
Iterate posting for term 'right': 100%|█| 223259/223259 [01:10<00:00, 3158.84it/
```

```
-----------------------------------------------------
Current query:  pelvic causes pain
```

```
Iterate posting for term 'pelvic': 100%|██| 7528/7528 [00:02<00:00, 3527.65it/s]
Iterate posting for term 'causes': 100%|█| 136109/136109 [00:38<00:00, 3549.19it
Iterate posting for term 'pain': 100%|█| 140027/140027 [00:40<00:00, 3500.05it/s
```

```
-----------------------------------------------------
Current query: what are the social determinants of health determinants social health pat
terning socioeconomic factors illness contribute behaviors interact
```

```
Iterate posting for term 'what': 100%|█| 542299/542299 [03:07<00:00, 2891.56it/s
Iterate posting for term 'social': 100%|█| 87525/87525 [00:25<00:00, 3430.18it/s
Iterate posting for term 'determinants': 100%|█| 798/798 [00:00<00:00, 3487.17it
Iterate posting for term 'health': 100%|█| 204625/204625 [01:19<00:00, 2566.76it
Iterate posting for term 'determinants': 100%|█| 798/798 [00:00<00:00, 3308.50it
Iterate posting for term 'social': 100%|█| 87525/87525 [00:33<00:00, 2640.05it/s
Iterate posting for term 'health': 100%|█| 204625/204625 [01:11<00:00, 2881.78it
Iterate posting for term 'patterning': 100%|█| 208/208 [00:00<00:00, 3154.76it/s
Iterate posting for term 'socioeconomic': 100%|█| 826/826 [00:00<00:00, 3315.91i
Iterate posting for term 'factors': 100%|█| 73135/73135 [00:20<00:00, 3483.01it/
Iterate posting for term 'illness': 100%|█| 26958/26958 [00:07<00:00, 3396.05it/
Iterate posting for term 'contribute': 100%|█| 18591/18591 [00:05<00:00, 3328.39
Iterate posting for term 'behaviors': 100%|█| 7201/7201 [00:02<00:00, 3446.03it/
Iterate posting for term 'interact': 100%|█| 7774/7774 [00:02<00:00, 3451.68it/s
------------------------------------------------------
Current query: what are the social determinants of health determinants social health pat
terning socioeconomic factors illness contribute behaviors interact recognize populatio
n.hese age.hese influence commonalities
Iterate posting for term 'what': 100%|█| 542299/542299 [03:08<00:00, 2884.49it/s
Iterate posting for term 'social': 100%|█| 87525/87525 [00:24<00:00, 3577.49it/s
Iterate posting for term 'determinants': 100%|█| 798/798 [00:00<00:00, 3413.16it
Iterate posting for term 'health': 100%|█| 204625/204625 [01:15<00:00, 2719.04it
Iterate posting for term 'determinants': 100%|█| 798/798 [00:00<00:00, 3269.45it
Iterate posting for term 'social': 100%|█| 87525/87525 [00:32<00:00, 2703.53it/s
Iterate posting for term 'health': 100%|█| 204625/204625 [01:08<00:00, 2994.93it
Iterate posting for term 'patterning': 100%|█| 208/208 [00:00<00:00, 3466.60it/s
Iterate posting for term 'socioeconomic': 100%|█| 826/826 [00:00<00:00, 3361.37i
Iterate posting for term 'factors': 100%|█| 73135/73135 [00:21<00:00, 3465.99it/
Iterate posting for term 'illness': 100%|█| 26958/26958 [00:07<00:00, 3460.59it/
Iterate posting for term 'contribute': 100%|█| 18591/18591 [00:05<00:00, 3501.45
Iterate posting for term 'behaviors': 100%|█| 7201/7201 [00:02<00:00, 3343.88it/
Iterate posting for term 'interact': 100%|█| 7774/7774 [00:02<00:00, 3345.41it/s
Iterate posting for term 'recognize': 100%|█| 12682/12682 [00:03<00:00, 3416.44i
Iterate posting for term 'population.hese': 100%|█| 6/6 [00:00<00:00, 2268.62it/
Iterate posting for term 'age.hese': 100%|██████| 8/8 [00:00<00:00, 2656.30it/s]
Iterate posting for term 'influence': 100%|█| 24683/24683 [00:14<00:00, 1707.33i
Iterate posting for term 'commonalities': 100%|█| 169/169 [00:00<00:00, 2658.41i
------------------------------------------------------
Current query:  determinants social health what
Iterate posting for term 'determinants': 100%|█| 798/798 [00:00<00:00, 3132.09it
Iterate posting for term 'social': 100%|█| 87525/87525 [00:25<00:00, 3405.78it/s
Iterate posting for term 'health': 100%|█| 204625/204625 [01:08<00:00, 2978.81it
Iterate posting for term 'what': 100%|█| 542299/542299 [03:00<00:00, 3006.44it/s
------------------------------------------------------
Current query:  determinants social health
Iterate posting for term 'determinants': 100%|█| 798/798 [00:00<00:00, 3408.90it
Iterate posting for term 'social': 100%|█| 87525/87525 [00:35<00:00, 2469.17it/s
Iterate posting for term 'health': 100%|█| 204625/204625 [01:01<00:00, 3331.47it
------------------------------------------------------
Current query: does legionella pneumophila cause pneumonia pneumophila chlamydophila leg
ionella pneumoniae pneumonia mycoplasma bacteria bacterial legionnaires cause
Iterate posting for term 'does': 100%|█| 260662/260662 [01:17<00:00, 3353.25it/s
Iterate posting for term 'legionella': 100%|█| 350/350 [00:00<00:00, 3516.21it/s
Iterate posting for term 'pneumophila': 100%|█| 137/137 [00:00<00:00, 3512.56it/
Iterate posting for term 'cause': 100%|█| 227338/227338 [01:14<00:00, 3061.19it/
Iterate posting for term 'pneumonia': 100%|█| 8318/8318 [00:08<00:00, 953.58it/s
Iterate posting for term 'pneumophila': 100%|█| 137/137 [00:00<00:00, 2965.30it/
Iterate posting for term 'chlamydophila': 100%|█| 64/64 [00:00<00:00, 3113.59it/
Iterate posting for term 'legionella': 100%|█| 350/350 [00:00<00:00, 3399.51it/s
Iterate posting for term 'pneumoniae': 100%|█| 1024/1024 [00:00<00:00, 3304.12it
Iterate posting for term 'pneumonia': 100%|█| 8318/8318 [00:02<00:00, 3321.26it/
Iterate posting for term 'mycoplasma': 100%|█| 658/658 [00:00<00:00, 3373.99it/s
Iterate posting for term 'bacteria': 100%|█| 59275/59275 [00:16<00:00, 3488.04it
Iterate posting for term 'bacterial': 100%|█| 22213/22213 [00:06<00:00, 3461.44i
Iterate posting for term 'legionnaires': 100%|█| 269/269 [00:00<00:00, 3604.44it
Iterate posting for term 'cause': 100%|█| 227338/227338 [01:21<00:00, 2791.30it/
------------------------------------------------------
Current query: does legionella pneumophila cause pneumonia pneumophila chlamydophila leg
ionella pneumoniae pneumonia mycoplasma bacteria bacterial legionnaires cause psittaci s
treptococcus atypical chlamydia disease
Iterate posting for term 'does': 100%|█| 260662/260662 [01:26<00:00, 3029.81it/s
Iterate posting for term 'legionella': 100%|█| 350/350 [00:00<00:00, 2800.69it/s
Iterate posting for term 'pneumophila': 100%|█| 137/137 [00:00<00:00, 3284.42it/
Iterate posting for term 'cause': 100%|█| 227338/227338 [01:26<00:00, 2636.40it/
Iterate posting for term 'pneumonia': 100%|█| 8318/8318 [00:02<00:00, 3206.28it/
Iterate posting for term 'pneumophila': 100%|█| 137/137 [00:00<00:00, 2998.67it/
Iterate posting for term 'chlamydophila': 100%|█| 64/64 [00:00<00:00, 3123.16it/
Iterate posting for term 'legionella': 100%|█| 350/350 [00:00<00:00, 3266.05it/s
Iterate posting for term 'pneumoniae': 100%|█| 1024/1024 [00:00<00:00, 3488.09it
Iterate posting for term 'pneumonia': 100%|█| 8318/8318 [00:02<00:00, 3529.15it/
Iterate posting for term 'mycoplasma': 100%|█| 658/658 [00:00<00:00, 3508.34it/s
Iterate posting for term 'bacteria': 100%|█| 59275/59275 [00:17<00:00, 3375.32it
Iterate posting for term 'bacterial': 100%|█| 22213/22213 [00:06<00:00, 3519.52i
Iterate posting for term 'legionnaires': 100%|█| 269/269 [00:00<00:00, 3400.47it
Iterate posting for term 'cause': 100%|█| 227338/227338 [01:15<00:00, 3030.00it/
Iterate posting for term 'psittaci': 100%|███| 74/74 [00:00<00:00, 3367.31it/s]
Iterate posting for term 'streptococcus': 100%|█| 2804/2804 [00:00<00:00, 3495.8
Iterate posting for term 'atypical': 100%|█| 1948/1948 [00:00<00:00, 3461.31it/s
Iterate posting for term 'chlamydia': 100%|█| 2697/2697 [00:00<00:00, 3496.59it/
Iterate posting for term 'disease': 100%|█| 157505/157505 [00:52<00:00, 3015.02i
```

```
-----------------------------------------------------
Current query:  pneumophila legionella pneumonia cause does
```

```
Iterate posting for term 'pneumophila': 100%|█| 137/137 [00:00<00:00, 3122.22it/
Iterate posting for term 'legionella': 100%|█| 350/350 [00:00<00:00, 3286.97it/s
Iterate posting for term 'pneumonia': 100%|█| 8318/8318 [00:02<00:00, 3564.43it/
Iterate posting for term 'cause': 100%|█| 227338/227338 [01:13<00:00, 3090.90it/
Iterate posting for term 'does': 100%|█| 260662/260662 [01:28<00:00, 2951.57it/s
```

```
-----------------------------------------------------
Current query:  pneumophila legionella pneumonia
```

```
Iterate posting for term 'pneumophila': 100%|█| 137/137 [00:00<00:00, 2826.66it/
Iterate posting for term 'legionella': 100%|█| 350/350 [00:00<00:00, 3501.16it/s
Iterate posting for term 'pneumonia': 100%|█| 8318/8318 [00:02<00:00, 3480.24it/
```

```
-----------------------------------------------------
Current query: how is the weather in jamaica jamaica weather montego forecast bay destin
ations month perfect averages january
```

```
Iterate posting for term 'how': 100%|█| 553931/553931 [03:21<00:00, 2747.22it/s]
Iterate posting for term 'weather': 100%|█| 87214/87214 [00:24<00:00, 3560.69it/
Iterate posting for term 'jamaica': 100%|█| 3779/3779 [00:01<00:00, 3500.16it/s]
Iterate posting for term 'jamaica': 100%|█| 3779/3779 [00:01<00:00, 3623.27it/s]
Iterate posting for term 'weather': 100%|█| 87214/87214 [00:35<00:00, 2428.80it/
Iterate posting for term 'montego': 100%|███| 474/474 [00:00<00:00, 3358.63it/s]
Iterate posting for term 'forecast': 100%|█| 21427/21427 [00:06<00:00, 3266.30it
Iterate posting for term 'bay': 100%|███| 26869/26869 [00:07<00:00, 3400.31it/s]
Iterate posting for term 'destinations': 100%|█| 7058/7058 [00:02<00:00, 3440.22
Iterate posting for term 'month': 100%|█| 111884/111884 [00:32<00:00, 3471.09it/
Iterate posting for term 'perfect': 100%|█| 44872/44872 [00:26<00:00, 1695.93it/
Iterate posting for term 'averages': 100%|█| 17230/17230 [00:05<00:00, 3251.40it
Iterate posting for term 'january': 100%|█| 70591/70591 [00:20<00:00, 3496.94it/
```

```
-----------------------------------------------------
Current query: how is the weather in jamaica jamaica weather montego forecast bay destin
ations month perfect averages january deciding holidays monthly sunshine season
```

```
Iterate posting for term 'how': 100%|█| 553931/553931 [03:06<00:00, 2976.38it/s]
Iterate posting for term 'weather': 100%|█| 87214/87214 [00:24<00:00, 3612.06it/
Iterate posting for term 'jamaica': 100%|█| 3779/3779 [00:01<00:00, 3625.64it/s]
Iterate posting for term 'jamaica': 100%|█| 3779/3779 [00:01<00:00, 3641.93it/s]
Iterate posting for term 'weather': 100%|█| 87214/87214 [00:36<00:00, 2360.89it/
Iterate posting for term 'montego': 100%|███| 474/474 [00:00<00:00, 3111.28it/s]
Iterate posting for term 'forecast': 100%|█| 21427/21427 [00:06<00:00, 3401.35it
Iterate posting for term 'bay': 100%|███| 26869/26869 [00:07<00:00, 3434.57it/s]
Iterate posting for term 'destinations': 100%|█| 7058/7058 [00:02<00:00, 3488.08
Iterate posting for term 'month': 100%|█| 111884/111884 [00:32<00:00, 3456.03it/
Iterate posting for term 'perfect': 100%|█| 44872/44872 [00:12<00:00, 3487.19it/
Iterate posting for term 'averages': 100%|█| 17230/17230 [00:04<00:00, 3474.76it
Iterate posting for term 'january': 100%|█| 70591/70591 [00:31<00:00, 2217.18it/
Iterate posting for term 'deciding': 100%|█| 5642/5642 [00:01<00:00, 3417.58it/s
Iterate posting for term 'holidays': 100%|█| 9269/9269 [00:02<00:00, 3526.27it/s
Iterate posting for term 'monthly': 100%|█| 39920/39920 [00:11<00:00, 3520.55it/
Iterate posting for term 'sunshine': 100%|█| 10308/10308 [00:02<00:00, 3571.20it
Iterate posting for term 'season': 100%|█| 88323/88323 [00:24<00:00, 3535.07it/s
-----------------------------------------------------
Current query:  jamaica weather how
```

```
Iterate posting for term 'jamaica': 100%|█| 3779/3779 [00:01<00:00, 3577.97it/s]
Iterate posting for term 'weather': 100%|█| 87214/87214 [00:24<00:00, 3491.25it/
Iterate posting for term 'how': 100%|█| 553931/553931 [03:13<00:00, 2856.99it/s]
-----------------------------------------------------
Current query:  jamaica weather how
```

```
Iterate posting for term 'jamaica': 100%|█| 3779/3779 [00:01<00:00, 3549.20it/s]
Iterate posting for term 'weather': 100%|█| 87214/87214 [00:25<00:00, 3367.48it/
Iterate posting for term 'how': 100%|█| 553931/553931 [03:17<00:00, 2801.52it/s]
-----------------------------------------------------
Current query: types of dysarthria from cerebral palsy palsy cerebral dysarthria spastic
ataxic impairment speech types speaking movement
```

```
Iterate posting for term 'types': 100%|█| 169667/169667 [00:54<00:00, 3087.25it/
Iterate posting for term 'dysarthria': 100%|█| 256/256 [00:00<00:00, 3045.49it/s
Iterate posting for term 'from': 100%|█| 1820726/1820726 [10:44<00:00, 2825.42it
Iterate posting for term 'cerebral': 100%|█| 6626/6626 [00:01<00:00, 3350.85it/s
Iterate posting for term 'palsy': 100%|███| 2534/2534 [00:00<00:00, 3626.38it/s]
Iterate posting for term 'palsy': 100%|███| 2534/2534 [00:00<00:00, 3526.71it/s]
Iterate posting for term 'cerebral': 100%|█| 6626/6626 [00:01<00:00, 3629.09it/s
Iterate posting for term 'dysarthria': 100%|█| 256/256 [00:00<00:00, 3450.74it/s
Iterate posting for term 'spastic': 100%|███| 506/506 [00:00<00:00, 3440.75it/s]
Iterate posting for term 'ataxic': 100%|██████| 77/77 [00:00<00:00, 2669.72it/s]
Iterate posting for term 'impairment': 100%|█| 4731/4731 [00:01<00:00, 3461.10it
Iterate posting for term 'speech': 100%|█| 23255/23255 [00:06<00:00, 3554.87it/s
Iterate posting for term 'types': 100%|█| 169667/169667 [00:47<00:00, 3541.82it/
Iterate posting for term 'speaking': 100%|█| 20792/20792 [00:05<00:00, 3532.83it
Iterate posting for term 'movement': 100%|█| 52317/52317 [00:14<00:00, 3565.07it
-----------------------------------------------------
Current query: types of dysarthria from cerebral palsy palsy cerebral dysarthria spastic
ataxic impairment speech types speaking movement sclerosis four discussed brain detail
```

```
Iterate posting for term 'types': 100%|█| 169667/169667 [00:54<00:00, 3103.15it/
Iterate posting for term 'dysarthria': 100%|█| 256/256 [00:00<00:00, 3161.38it/s
Iterate posting for term 'from': 100%|█| 1820726/1820726 [10:41<00:00, 2839.22it
Iterate posting for term 'cerebral': 100%|█| 6626/6626 [00:04<00:00, 1653.38it/s
Iterate posting for term 'palsy': 100%|███| 2534/2534 [00:01<00:00, 2061.35it/s]
Iterate posting for term 'palsy': 100%|███| 2534/2534 [00:00<00:00, 3510.36it/s]
Iterate posting for term 'cerebral': 100%|█| 6626/6626 [00:01<00:00, 3558.28it/s
Iterate posting for term 'dysarthria': 100%|█| 256/256 [00:00<00:00, 2869.92it/s
Iterate posting for term 'spastic': 100%|███| 506/506 [00:00<00:00, 3067.95it/s]
Iterate posting for term 'ataxic': 100%|██████| 77/77 [00:00<00:00, 2644.32it/s]
Iterate posting for term 'impairment': 100%|█| 4731/4731 [00:01<00:00, 3052.32it
Iterate posting for term 'speech': 100%|█| 23255/23255 [00:07<00:00, 3095.16it/s
Iterate posting for term 'types': 100%|█| 169667/169667 [01:13<00:00, 2293.47it/
Iterate posting for term 'speaking': 100%|█| 20792/20792 [00:06<00:00, 3111.46it
Iterate posting for term 'movement': 100%|█| 52317/52317 [00:17<00:00, 2956.37it
Iterate posting for term 'sclerosis': 100%|█| 4288/4288 [00:01<00:00, 3385.64it/
Iterate posting for term 'four': 100%|█| 158057/158057 [01:10<00:00, 2252.61it/s
Iterate posting for term 'discussed': 100%|█| 11070/11070 [00:03<00:00, 3230.89i
Iterate posting for term 'brain': 100%|█| 76679/76679 [00:50<00:00, 1522.15it/s]
Iterate posting for term 'detail': 100%|█| 16237/16237 [00:05<00:00, 2861.86it/s
-----------------------------------------------------
Current query:  dysarthria palsy cerebral types from
```

```
Iterate posting for term 'dysarthria': 100%|█| 256/256 [00:00<00:00, 2957.25it/s
Iterate posting for term 'palsy': 100%|███| 2534/2534 [00:00<00:00, 3275.01it/s]
Iterate posting for term 'cerebral': 100%|█| 6626/6626 [00:01<00:00, 3429.58it/s
Iterate posting for term 'types': 100%|█| 169667/169667 [01:08<00:00, 2474.89it/
Iterate posting for term 'from': 100%|█| 1820726/1820726 [10:27<00:00, 2902.52it
-----------------------------------------------------
Current query:  dysarthria palsy cerebral
```

```
Iterate posting for term 'dysarthria': 100%|█| 256/256 [00:00<00:00, 2834.50it/s
Iterate posting for term 'palsy': 100%|███| 2534/2534 [00:00<00:00, 3226.72it/s]
Iterate posting for term 'cerebral': 100%|█| 6626/6626 [00:01<00:00, 3382.34it/s
```

```
-----------------------------------------------------
Current query: who is robert gray gray robert governor mississippi offical democratic he
election rediviva 1792
```

```
Iterate posting for term 'who': 100%|█| 479472/479472 [02:54<00:00, 2740.15it/s]
Iterate posting for term 'robert': 100%|█| 21332/21332 [00:06<00:00, 3405.05it/s
Iterate posting for term 'gray': 100%|██| 14829/14829 [00:04<00:00, 3528.44it/s]
Iterate posting for term 'gray': 100%|██| 14829/14829 [00:04<00:00, 3505.64it/s]
Iterate posting for term 'robert': 100%|█| 21332/21332 [00:16<00:00, 1276.37it/s
Iterate posting for term 'governor': 100%|█| 11841/11841 [00:03<00:00, 3153.42it
Iterate posting for term 'mississippi': 100%|█| 12559/12559 [00:05<00:00, 2456.3
Iterate posting for term 'offical': 100%|███| 170/170 [00:00<00:00, 2650.74it/s]
Iterate posting for term 'democratic': 100%|█| 11284/11284 [00:03<00:00, 3279.81
Iterate posting for term 'he': 100%|██| 364179/364179 [02:16<00:00, 2665.77it/s]
Iterate posting for term 'election': 100%|█| 16226/16226 [00:04<00:00, 3331.28it
Iterate posting for term 'rediviva': 100%|██████| 7/7 [00:00<00:00, 2836.18it/s]
Iterate posting for term '1792': 100%|████| 1190/1190 [00:00<00:00, 3485.28it/s]
```

```
-----------------------------------------------------
Current query: who is robert gray gray robert governor mississippi offical democratic he
election rediviva 1792 truck who bless shies didn't
```

```
Iterate posting for term 'who': 100%|█| 479472/479472 [03:14<00:00, 2463.61it/s]
Iterate posting for term 'robert': 100%|█| 21332/21332 [00:08<00:00, 2613.56it/s
Iterate posting for term 'gray': 100%|██| 14829/14829 [00:04<00:00, 3278.35it/s]
Iterate posting for term 'gray': 100%|██| 14829/14829 [00:04<00:00, 3149.55it/s]
Iterate posting for term 'robert': 100%|█| 21332/21332 [00:09<00:00, 2188.48it/s
Iterate posting for term 'governor': 100%|█| 11841/11841 [00:03<00:00, 2984.82it
Iterate posting for term 'mississippi': 100%|█| 12559/12559 [00:03<00:00, 3428.1
Iterate posting for term 'offical': 100%|███| 170/170 [00:00<00:00, 3478.61it/s]
Iterate posting for term 'democratic': 100%|█| 11284/11284 [00:03<00:00, 3451.71
Iterate posting for term 'he': 100%|██| 364179/364179 [02:30<00:00, 2419.39it/s]
Iterate posting for term 'election': 100%|█| 16226/16226 [00:05<00:00, 3231.36it
Iterate posting for term 'rediviva': 100%|██████| 7/7 [00:00<00:00, 2540.90it/s]
Iterate posting for term '1792': 100%|████| 1190/1190 [00:00<00:00, 3272.11it/s]
Iterate posting for term 'truck': 100%|█| 16076/16076 [00:04<00:00, 3281.35it/s]
Iterate posting for term 'who': 100%|█| 479472/479472 [02:55<00:00, 2733.66it/s]
Iterate posting for term 'bless': 100%|█████| 788/788 [00:00<00:00, 2698.83it/s]
Iterate posting for term 'shies': 100%|███████| 15/15 [00:00<00:00, 2616.10it/s]
Iterate posting for term 'didn't': 100%|█| 22061/22061 [00:14<00:00, 1528.49it/s
```

```
-----------------------------------------------------
Current query:  gray robert who
```

```
Iterate posting for term 'gray': 100%|██| 14829/14829 [00:04<00:00, 3384.59it/s]
Iterate posting for term 'robert': 100%|█| 21332/21332 [00:06<00:00, 3399.44it/s
Iterate posting for term 'who': 100%|█| 479472/479472 [02:48<00:00, 2844.73it/s]
```

```
-----------------------------------------------------
Current query:  gray robert who
```

```
Iterate posting for term 'gray': 100%|██| 14829/14829 [00:04<00:00, 3342.55it/s]
Iterate posting for term 'robert': 100%|█| 21332/21332 [00:06<00:00, 3540.55it/s
Iterate posting for term 'who': 100%|█| 479472/479472 [02:57<00:00, 2696.08it/s]
```

```
-----------------------------------------------------
Current query: what types of food can you cook sous vide vide sous cook food cooked chic
ken cooking wrappersâ searzall shellsâ
```

```
Iterate posting for term 'what': 100%|█| 542299/542299 [03:20<00:00, 2711.22it/s
Iterate posting for term 'types': 100%|█| 169667/169667 [00:49<00:00, 3405.35it/
Iterate posting for term 'food': 100%|█| 183952/183952 [01:07<00:00, 2709.85it/s
Iterate posting for term 'can': 100%|█| 1590108/1590108 [09:09<00:00, 2893.14it/
Iterate posting for term 'you': 100%|█| 1863876/1863876 [11:05<00:00, 2800.38it/
Iterate posting for term 'cook': 100%|██| 44859/44859 [00:13<00:00, 3394.31it/s]
Iterate posting for term 'sous': 100%|██████| 506/506 [00:00<00:00, 3399.33it/s]
Iterate posting for term 'vide': 100%|██████| 500/500 [00:00<00:00, 3495.92it/s]
Iterate posting for term 'vide': 100%|██████| 500/500 [00:00<00:00, 3538.34it/s]
Iterate posting for term 'sous': 100%|██████| 506/506 [00:00<00:00, 3568.38it/s]
Iterate posting for term 'cook': 100%|██| 44859/44859 [00:12<00:00, 3559.33it/s]
Iterate posting for term 'food': 100%|█| 183952/183952 [00:52<00:00, 3476.16it/s
Iterate posting for term 'cooked': 100%|█| 27890/27890 [00:08<00:00, 3248.91it/s
Iterate posting for term 'chicken': 100%|█| 35606/35606 [00:10<00:00, 3284.55it/
Iterate posting for term 'cooking': 100%|█| 38474/38474 [00:11<00:00, 3321.86it/
Iterate posting for term 'wrappersâ': 100%|█████| 4/4 [00:00<00:00, 3081.78it/s]
Iterate posting for term 'searzall': 100%|██████| 2/2 [00:00<00:00, 2735.12it/s]
Iterate posting for term 'shellsâ': 100%|█████| 13/13 [00:00<00:00, 2918.17it/s]
```

```
-----------------------------------------------------
Current query: what types of food can you cook sous vide vide sous cook food cooked chic
ken cooking wrappersâ searzall shellsâ water plastic youâ what foods
```

```
Iterate posting for term 'what': 100%|█| 542299/542299 [04:10<00:00, 2165.97it/s
Iterate posting for term 'types': 100%|█| 169667/169667 [01:16<00:00, 2228.63it/
Iterate posting for term 'food': 100%|█| 183952/183952 [01:13<00:00, 2502.44it/s
Iterate posting for term 'can': 100%|█| 1590108/1590108 [10:02<00:00, 2641.05it/
Iterate posting for term 'you': 100%|█| 1863876/1863876 [11:47<00:00, 2636.26it/
Iterate posting for term 'cook': 100%|██| 44859/44859 [00:13<00:00, 3436.44it/s]
Iterate posting for term 'sous': 100%|██████| 506/506 [00:00<00:00, 3598.63it/s]
Iterate posting for term 'vide': 100%|██████| 500/500 [00:00<00:00, 3529.78it/s]
Iterate posting for term 'vide': 100%|██████| 500/500 [00:00<00:00, 3448.44it/s]
Iterate posting for term 'sous': 100%|██████| 506/506 [00:00<00:00, 3522.86it/s]
Iterate posting for term 'cook': 100%|██| 44859/44859 [00:24<00:00, 1859.09it/s]
Iterate posting for term 'food': 100%|█| 183952/183952 [00:55<00:00, 3340.03it/s
Iterate posting for term 'cooked': 100%|█| 27890/27890 [00:08<00:00, 3123.98it/s
Iterate posting for term 'chicken': 100%|█| 35606/35606 [00:32<00:00, 1109.46it/
Iterate posting for term 'cooking': 100%|█| 38474/38474 [00:12<00:00, 2975.30it/
Iterate posting for term 'wrappersâ': 100%|█████| 4/4 [00:00<00:00, 2461.45it/s]
Iterate posting for term 'searzall': 100%|██████| 2/2 [00:00<00:00, 3164.32it/s]
Iterate posting for term 'shellsâ': 100%|█████| 13/13 [00:00<00:00, 2654.62it/s]
Iterate posting for term 'water': 100%|█| 302936/302936 [02:03<00:00, 2450.73it/
Iterate posting for term 'plastic': 100%|█| 31702/31702 [00:09<00:00, 3437.58it/
Iterate posting for term 'youâ': 100%|█| 101918/101918 [00:47<00:00, 2160.45it/s
Iterate posting for term 'what': 100%|█| 542299/542299 [03:45<00:00, 2403.94it/s
Iterate posting for term 'foods': 100%|█| 84392/84392 [00:24<00:00, 3432.42it/s]
```

```
-----------------------------------------------------
Current query:  vide sous cook types food
```

```
Iterate posting for term 'vide': 100%|██████| 500/500 [00:00<00:00, 3460.04it/s]
Iterate posting for term 'sous': 100%|██████| 506/506 [00:00<00:00, 3461.41it/s]
Iterate posting for term 'cook': 100%|██| 44859/44859 [00:27<00:00, 1604.10it/s]
Iterate posting for term 'types': 100%|█| 169667/169667 [00:53<00:00, 3192.61it/
Iterate posting for term 'food': 100%|█| 183952/183952 [01:41<00:00, 1812.60it/s
```

```
-----------------------------------------------------
Current query:  vide sous cook
```

```
Iterate posting for term 'vide': 100%|██████| 500/500 [00:00<00:00, 2388.35it/s]
Iterate posting for term 'sous': 100%|██████| 506/506 [00:00<00:00, 3175.17it/s]
Iterate posting for term 'cook': 100%|██| 44859/44859 [00:12<00:00, 3474.04it/s]
```

```
-----------------------------------------------------
Current query: how long is life cycle of flea flea cycle larvae pupa life eggs fleas las
t anywhere dormant
```

```
Iterate posting for term 'how': 100%|█| 553931/553931 [03:50<00:00, 2404.04it/s]
Iterate posting for term 'long': 100%|█| 286025/286025 [01:54<00:00, 2492.61it/s
Iterate posting for term 'life': 100%|█| 208672/208672 [01:55<00:00, 1801.33it/s
Iterate posting for term 'cycle': 100%|█| 41208/41208 [00:11<00:00, 3458.35it/s]
Iterate posting for term 'flea': 100%|████| 3425/3425 [00:00<00:00, 3488.00it/s]
Iterate posting for term 'flea': 100%|████| 3425/3425 [00:00<00:00, 3602.22it/s]
Iterate posting for term 'cycle': 100%|█| 41208/41208 [00:11<00:00, 3594.63it/s]
Iterate posting for term 'larvae': 100%|██| 3754/3754 [00:01<00:00, 3529.97it/s]
Iterate posting for term 'pupa': 100%|██████| 444/444 [00:00<00:00, 3603.13it/s]
Iterate posting for term 'life': 100%|█| 208672/208672 [01:40<00:00, 2070.24it/s
Iterate posting for term 'eggs': 100%|██| 39377/39377 [00:11<00:00, 3527.69it/s]
Iterate posting for term 'fleas': 100%|███| 3418/3418 [00:00<00:00, 3513.41it/s]
Iterate posting for term 'last': 100%|█| 192994/192994 [01:18<00:00, 2461.49it/s
Iterate posting for term 'anywhere': 100%|█| 42217/42217 [00:11<00:00, 3577.90it
Iterate posting for term 'dormant': 100%|█| 3026/3026 [00:00<00:00, 3568.98it/s]
```

```
-----------------------------------------------------
Current query: how long is life cycle of flea flea cycle larvae pupa life eggs fleas las
```

```
t anywhere dormant depends long influences control.ancel control.part
```

```
Iterate posting for term 'how': 100%|█| 553931/553931 [04:09<00:00, 2224.49it/s]
Iterate posting for term 'long': 100%|█| 286025/286025 [01:55<00:00, 2472.63it/s
Iterate posting for term 'life': 100%|█| 208672/208672 [01:39<00:00, 2090.74it/s
Iterate posting for term 'cycle': 100%|█| 41208/41208 [00:11<00:00, 3460.59it/s]
Iterate posting for term 'flea': 100%|████| 3425/3425 [00:00<00:00, 3526.67it/s]
Iterate posting for term 'flea': 100%|████| 3425/3425 [00:00<00:00, 3535.80it/s]
Iterate posting for term 'cycle': 100%|█| 41208/41208 [00:25<00:00, 1616.40it/s]
Iterate posting for term 'larvae': 100%|██| 3754/3754 [00:01<00:00, 3138.20it/s]
Iterate posting for term 'pupa': 100%|██████| 444/444 [00:00<00:00, 3290.28it/s]
Iterate posting for term 'life': 100%|█| 208672/208672 [01:30<00:00, 2306.52it/s
Iterate posting for term 'eggs': 100%|██| 39377/39377 [00:26<00:00, 1483.95it/s]
Iterate posting for term 'fleas': 100%|███| 3418/3418 [00:01<00:00, 3254.88it/s]
Iterate posting for term 'last': 100%|█| 192994/192994 [01:21<00:00, 2363.08it/s
Iterate posting for term 'anywhere': 100%|█| 42217/42217 [00:28<00:00, 1469.71it
Iterate posting for term 'dormant': 100%|█| 3026/3026 [00:01<00:00, 2702.47it/s]
Iterate posting for term 'depends': 100%|█| 67001/67001 [00:19<00:00, 3374.76it/
Iterate posting for term 'long': 100%|█| 286025/286025 [01:57<00:00, 2425.61it/s
Iterate posting for term 'influences': 100%|█| 6257/6257 [00:01<00:00, 3469.40it
Iterate posting for term 'control.ancel': 100%|█| 1/1 [00:00<00:00, 2357.68it/s]
Iterate posting for term 'control.part': 100%|██| 1/1 [00:00<00:00, 2474.52it/s]
```

```
-----------------------------------------------------
Current query:  flea cycle life long how
```

```
Iterate posting for term 'flea': 100%|████| 3425/3425 [00:00<00:00, 3584.61it/s]
Iterate posting for term 'cycle': 100%|█| 41208/41208 [00:25<00:00, 1635.88it/s]
Iterate posting for term 'life': 100%|█| 208672/208672 [01:24<00:00, 2483.92it/s
Iterate posting for term 'long': 100%|█| 286025/286025 [01:58<00:00, 2420.46it/s
Iterate posting for term 'how': 100%|█| 553931/553931 [03:47<00:00, 2439.00it/s]
```

```
-----------------------------------------------------
Current query:  flea cycle life
```

```
Iterate posting for term 'flea': 100%|████| 3425/3425 [00:01<00:00, 3144.95it/s]
Iterate posting for term 'cycle': 100%|█| 41208/41208 [00:30<00:00, 1342.30it/s]
Iterate posting for term 'life': 100%|█| 208672/208672 [01:22<00:00, 2524.89it/s
```

```
-----------------------------------------------------
Current query: what can contour plowing reduce plowing contour erosion soil 1851268 by__
_ land reduce translations plowed
```

```
Iterate posting for term 'what': 100%|█| 542299/542299 [03:50<00:00, 2352.03it/s
Iterate posting for term 'can': 100%|█| 1590108/1590108 [10:27<00:00, 2532.49it/
Iterate posting for term 'contour': 100%|█| 1465/1465 [00:00<00:00, 2471.56it/s]
Iterate posting for term 'plowing': 100%|███| 338/338 [00:00<00:00, 2859.45it/s]
Iterate posting for term 'reduce': 100%|█| 61911/61911 [00:20<00:00, 2965.85it/s
Iterate posting for term 'plowing': 100%|███| 338/338 [00:00<00:00, 3243.49it/s]
Iterate posting for term 'contour': 100%|█| 1465/1465 [00:00<00:00, 3405.52it/s]
Iterate posting for term 'erosion': 100%|█| 5132/5132 [00:01<00:00, 3423.91it/s]
Iterate posting for term 'soil': 100%|██| 39943/39943 [00:11<00:00, 3501.97it/s]
Iterate posting for term '1851268': 100%|███████| 1/1 [00:00<00:00, 2603.54it/s]
Iterate posting for term 'by___': 100%|█████████| 1/1 [00:00<00:00, 2939.25it/s]
Iterate posting for term 'land': 100%|██| 76384/76384 [00:46<00:00, 1652.14it/s]
Iterate posting for term 'reduce': 100%|█| 61911/61911 [00:19<00:00, 3222.93it/s
Iterate posting for term 'translations': 100%|█| 11575/11575 [00:05<00:00, 2073.
Iterate posting for term 'plowed': 100%|████| 231/231 [00:00<00:00, 1854.14it/s]
```

```
-----------------------------------------------------
Current query: what can contour plowing reduce plowing contour erosion soil 1851268 by__
_ land reduce translations plowed contoured definitions runoff mean terraces
```

```
Iterate posting for term 'what': 100%|█| 542299/542299 [04:05<00:00, 2209.00it/s
Iterate posting for term 'can': 100%|█| 1590108/1590108 [10:04<00:00, 2630.90it/
Iterate posting for term 'contour': 100%|█| 1465/1465 [00:01<00:00, 1411.23it/s]
Iterate posting for term 'plowing': 100%|███| 338/338 [00:00<00:00, 1639.26it/s]
Iterate posting for term 'reduce': 100%|█| 61911/61911 [00:18<00:00, 3259.30it/s
Iterate posting for term 'plowing': 100%|███| 338/338 [00:00<00:00, 3498.53it/s]
Iterate posting for term 'contour': 100%|█| 1465/1465 [00:00<00:00, 3500.24it/s]
Iterate posting for term 'erosion': 100%|█| 5132/5132 [00:01<00:00, 3508.32it/s]
Iterate posting for term 'soil': 100%|██| 39943/39943 [00:11<00:00, 3530.59it/s]
Iterate posting for term '1851268': 100%|███████| 1/1 [00:00<00:00, 1782.53it/s]
Iterate posting for term 'by___': 100%|█████████| 1/1 [00:00<00:00, 2341.88it/s]
Iterate posting for term 'land': 100%|██| 76384/76384 [00:22<00:00, 3468.93it/s]
Iterate posting for term 'reduce': 100%|█| 61911/61911 [00:43<00:00, 1425.31it/s
Iterate posting for term 'translations': 100%|█| 11575/11575 [00:03<00:00, 3156.
Iterate posting for term 'plowed': 100%|████| 231/231 [00:00<00:00, 2949.63it/s]
Iterate posting for term 'contoured': 100%|█| 298/298 [00:00<00:00, 3407.55it/s]
Iterate posting for term 'definitions': 100%|█| 23262/23262 [00:06<00:00, 3384.7
Iterate posting for term 'runoff': 100%|██| 1573/1573 [00:00<00:00, 3338.14it/s]
Iterate posting for term 'mean': 100%|██| 96428/96428 [00:30<00:00, 3185.91it/s]
Iterate posting for term 'terraces': 100%|██| 414/414 [00:00<00:00, 3434.37it/s]
```

```
-----------------------------------------------------
Current query:  plowing contour reduce what can
```

```
Iterate posting for term 'plowing': 100%|███| 338/338 [00:00<00:00, 3498.45it/s]
Iterate posting for term 'contour': 100%|█| 1465/1465 [00:00<00:00, 3297.28it/s]
Iterate posting for term 'reduce': 100%|█| 61911/61911 [00:18<00:00, 3344.13it/s
Iterate posting for term 'what': 100%|█| 542299/542299 [03:33<00:00, 2534.95it/s
Iterate posting for term 'can': 100%|█| 1590108/1590108 [10:40<00:00, 2481.85it/
```

```
-----------------------------------------------------
Current query:  plowing contour reduce
```

```
Iterate posting for term 'plowing': 100%|███| 338/338 [00:00<00:00, 2195.45it/s]
Iterate posting for term 'contour': 100%|█| 1465/1465 [00:00<00:00, 2886.38it/s]
Iterate posting for term 'reduce': 100%|█| 61911/61911 [00:17<00:00, 3446.68it/s
```

```
-----------------------------------------------------
Current query: when was the salvation army founded salvation army booth methodist locati
ons daly william ascot railton colma
```

```
Iterate posting for term 'when': 100%|█| 837432/837432 [05:50<00:00, 2391.05it/s
Iterate posting for term 'salvation': 100%|█| 1528/1528 [00:00<00:00, 2879.21it/
Iterate posting for term 'army': 100%|██| 29967/29967 [00:08<00:00, 3427.35it/s]
Iterate posting for term 'founded': 100%|█| 33188/33188 [00:09<00:00, 3512.71it/
Iterate posting for term 'salvation': 100%|█| 1528/1528 [00:00<00:00, 3567.89it/
Iterate posting for term 'army': 100%|██| 29967/29967 [00:08<00:00, 3582.44it/s]
Iterate posting for term 'booth': 100%|███| 2460/2460 [00:00<00:00, 3502.40it/s]
Iterate posting for term 'methodist': 100%|█| 1353/1353 [00:00<00:00, 3623.72it/
Iterate posting for term 'locations': 100%|█| 37028/37028 [00:12<00:00, 2967.29i
Iterate posting for term 'daly': 100%|██████| 628/628 [00:00<00:00, 3065.03it/s]
Iterate posting for term 'william': 100%|█| 24887/24887 [00:28<00:00, 859.60it/s
Iterate posting for term 'ascot': 100%|█████| 143/143 [00:00<00:00, 2863.14it/s]
Iterate posting for term 'railton': 100%|███████| 7/7 [00:00<00:00, 3080.81it/s]
Iterate posting for term 'colma': 100%|███████| 25/25 [00:00<00:00, 2958.90it/s]
```

```
-----------------------------------------------------
Current query: when was the salvation army founded salvation army booth methodist locati
ons daly william ascot railton colma hanger racehorse murrells bramwell nigel
```

```
Iterate posting for term 'when': 100%|█| 837432/837432 [05:50<00:00, 2390.43it/s
Iterate posting for term 'salvation': 100%|█| 1528/1528 [00:00<00:00, 3334.46it/
Iterate posting for term 'army': 100%|██| 29967/29967 [00:08<00:00, 3536.71it/s]
Iterate posting for term 'founded': 100%|█| 33188/33188 [00:09<00:00, 3577.94it/
Iterate posting for term 'salvation': 100%|█| 1528/1528 [00:00<00:00, 3551.97it/
Iterate posting for term 'army': 100%|██| 29967/29967 [00:08<00:00, 3511.90it/s]
Iterate posting for term 'booth': 100%|███| 2460/2460 [00:00<00:00, 3503.04it/s]
Iterate posting for term 'methodist': 100%|█| 1353/1353 [00:00<00:00, 3533.17it/
Iterate posting for term 'locations': 100%|█| 37028/37028 [00:34<00:00, 1076.86i
Iterate posting for term 'daly': 100%|██████| 628/628 [00:00<00:00, 3501.48it/s]
Iterate posting for term 'william': 100%|█| 24887/24887 [00:08<00:00, 2991.46it/
Iterate posting for term 'ascot': 100%|█████| 143/143 [00:00<00:00, 3330.31it/s]
Iterate posting for term 'railton': 100%|███████| 7/7 [00:00<00:00, 3003.29it/s]
Iterate posting for term 'colma': 100%|███████| 25/25 [00:00<00:00, 2915.79it/s]
Iterate posting for term 'hanger': 100%|████| 796/796 [00:00<00:00, 3396.09it/s]
Iterate posting for term 'racehorse': 100%|█| 227/227 [00:00<00:00, 3427.40it/s]
Iterate posting for term 'murrells': 100%|████| 45/45 [00:00<00:00, 2830.76it/s]
Iterate posting for term 'bramwell': 100%|████| 19/19 [00:00<00:00, 2904.22it/s]
Iterate posting for term 'nigel': 100%|█████| 413/413 [00:00<00:00, 3361.06it/s]
```

```
-----------------------------------------------------
Current query:  salvation army founded when
```

```
Iterate posting for term 'salvation': 100%|█| 1528/1528 [00:00<00:00, 3419.56it/
Iterate posting for term 'army': 100%|██| 29967/29967 [00:08<00:00, 3556.29it/s]
Iterate posting for term 'founded': 100%|█| 33188/33188 [00:10<00:00, 3194.02it/
Iterate posting for term 'when': 100%|█| 837432/837432 [05:34<00:00, 2500.04it/s
```

```
-----------------------------------------------------
Current query:  salvation army founded
```

```
Iterate posting for term 'salvation': 100%|█| 1528/1528 [00:00<00:00, 3428.83it/
Iterate posting for term 'army': 100%|██| 29967/29967 [00:08<00:00, 3352.03it/s]
Iterate posting for term 'founded': 100%|█| 33188/33188 [00:10<00:00, 3124.06it/
```

```
-----------------------------------------------------
Current query: what is a active margin margin continental passive oceanic margins active
volcanic plate ocean lithosphere
```

```
Iterate posting for term 'what': 100%|█| 542299/542299 [03:59<00:00, 2262.04it/s
Iterate posting for term 'active': 100%|█| 57867/57867 [00:41<00:00, 1388.58it/s
Iterate posting for term 'margin': 100%|██| 7879/7879 [00:02<00:00, 3425.57it/s]
Iterate posting for term 'margin': 100%|██| 7879/7879 [00:02<00:00, 3576.70it/s]
Iterate posting for term 'continental': 100%|█| 12563/12563 [00:03<00:00, 3471.4
Iterate posting for term 'passive': 100%|█| 5794/5794 [00:01<00:00, 3480.47it/s]
Iterate posting for term 'oceanic': 100%|█| 4748/4748 [00:01<00:00, 3534.01it/s]
Iterate posting for term 'margins': 100%|█| 3353/3353 [00:00<00:00, 3500.61it/s]
Iterate posting for term 'active': 100%|█| 57867/57867 [00:16<00:00, 3517.31it/s
Iterate posting for term 'volcanic': 100%|█| 7406/7406 [00:02<00:00, 3503.11it/s
Iterate posting for term 'plate': 100%|█| 24190/24190 [00:07<00:00, 3453.81it/s]
Iterate posting for term 'ocean': 100%|█| 34710/34710 [00:25<00:00, 1356.76it/s]
Iterate posting for term 'lithosphere': 100%|█| 1784/1784 [00:00<00:00, 3404.56i
```

```
-----------------------------------------------------
Current query: what is a active margin margin continental passive oceanic margins active
volcanic plate ocean lithosphere earthquake area.arthquakes crust.together coast mountai
ns
```

```
Iterate posting for term 'what': 100%|█| 542299/542299 [03:20<00:00, 2702.55it/s
Iterate posting for term 'active': 100%|█| 57867/57867 [00:40<00:00, 1434.58it/s
Iterate posting for term 'margin': 100%|██| 7879/7879 [00:04<00:00, 1802.64it/s]
Iterate posting for term 'margin': 100%|██| 7879/7879 [00:02<00:00, 3516.89it/s]
Iterate posting for term 'continental': 100%|█| 12563/12563 [00:04<00:00, 3017.8
Iterate posting for term 'passive': 100%|█| 5794/5794 [00:01<00:00, 2948.66it/s]
Iterate posting for term 'oceanic': 100%|█| 4748/4748 [00:01<00:00, 3200.87it/s]
Iterate posting for term 'margins': 100%|█| 3353/3353 [00:01<00:00, 3137.26it/s]
Iterate posting for term 'active': 100%|█| 57867/57867 [00:18<00:00, 3171.73it/s
Iterate posting for term 'volcanic': 100%|█| 7406/7406 [00:02<00:00, 3339.30it/s
Iterate posting for term 'plate': 100%|█| 24190/24190 [00:09<00:00, 2634.83it/s]
Iterate posting for term 'ocean': 100%|██| 34710/34710 [00:42<00:00, 825.71it/s]
Iterate posting for term 'lithosphere': 100%|█| 1784/1784 [00:00<00:00, 3030.09i
Iterate posting for term 'earthquake': 100%|█| 6670/6670 [00:02<00:00, 3042.97it
Iterate posting for term 'area.arthquakes': 100%|█| 1/1 [00:00<00:00, 2216.86it/
Iterate posting for term 'crust.together': 100%|█| 3/3 [00:00<00:00, 1928.12it/s
Iterate posting for term 'coast': 100%|█| 33856/33856 [00:11<00:00, 3016.09it/s]
Iterate posting for term 'mountains': 100%|█| 21451/21451 [00:06<00:00, 3175.51i
```

```
-----------------------------------------------------
Current query:  margin active what
```

```
Iterate posting for term 'margin': 100%|██| 7879/7879 [00:02<00:00, 3256.85it/s]
Iterate posting for term 'active': 100%|█| 57867/57867 [00:17<00:00, 3375.71it/s
Iterate posting for term 'what': 100%|█| 542299/542299 [03:50<00:00, 2354.10it/s
```

```
-----------------------------------------------------
Current query:  margin active what
```

```
Iterate posting for term 'margin': 100%|██| 7879/7879 [00:02<00:00, 3362.16it/s]
Iterate posting for term 'active': 100%|█| 57867/57867 [00:16<00:00, 3531.39it/s
Iterate posting for term 'what': 100%|█| 542299/542299 [04:38<00:00, 1948.61it/s
```

```
-----------------------------------------------------
Current query: difference between rn and bsn bsn rn difference adn pls between attained
noticed nurse degree
```

```
Iterate posting for term 'difference': 100%|█| 75516/75516 [00:26<00:00, 2817.56
Iterate posting for term 'between': 100%|█| 416673/416673 [03:28<00:00, 1994.72i
Iterate posting for term 'rn': 100%|███████| 6217/6217 [00:20<00:00, 309.22it/s]
Iterate posting for term 'bsn': 100%|█████| 2060/2060 [00:00<00:00, 3135.08it/s]
Iterate posting for term 'bsn': 100%|█████| 2060/2060 [00:00<00:00, 3560.96it/s]
Iterate posting for term 'rn': 100%|██████| 6217/6217 [00:01<00:00, 3320.64it/s]
Iterate posting for term 'difference': 100%|█| 75516/75516 [00:22<00:00, 3317.58
Iterate posting for term 'adn': 100%|███████| 682/682 [00:00<00:00, 3408.79it/s]
Iterate posting for term 'pls': 100%|███████| 496/496 [00:00<00:00, 3473.46it/s]
Iterate posting for term 'between': 100%|█| 416673/416673 [03:25<00:00, 2030.11i
Iterate posting for term 'attained': 100%|█| 1850/1850 [00:00<00:00, 3397.29it/s
Iterate posting for term 'noticed': 100%|█| 9942/9942 [00:02<00:00, 3462.08it/s]
Iterate posting for term 'nurse': 100%|█| 21917/21917 [00:06<00:00, 3494.66it/s]
Iterate posting for term 'degree': 100%|█| 77062/77062 [00:40<00:00, 1900.63it/s
```

```
-----------------------------------------------------
Current query: difference between rn and bsn bsn rn difference adn pls between attained
noticed nurse degree truly confused 67,449 bls asn
```

```
Iterate posting for term 'difference': 100%|█| 75516/75516 [00:23<00:00, 3240.70
Iterate posting for term 'between': 100%|█| 416673/416673 [03:23<00:00, 2049.56i
Iterate posting for term 'rn': 100%|██████| 6217/6217 [00:01<00:00, 3390.92it/s]
Iterate posting for term 'bsn': 100%|█████| 2060/2060 [00:00<00:00, 3507.10it/s]
Iterate posting for term 'bsn': 100%|█████| 2060/2060 [00:00<00:00, 3542.48it/s]
Iterate posting for term 'rn': 100%|██████| 6217/6217 [00:01<00:00, 3522.69it/s]
Iterate posting for term 'difference': 100%|█| 75516/75516 [00:40<00:00, 1852.10
Iterate posting for term 'adn': 100%|███████| 682/682 [00:00<00:00, 3336.89it/s]
Iterate posting for term 'pls': 100%|███████| 496/496 [00:00<00:00, 3303.83it/s]
Iterate posting for term 'between': 100%|█| 416673/416673 [03:34<00:00, 1946.69i
Iterate posting for term 'attained': 100%|█| 1850/1850 [00:00<00:00, 2968.13it/s
Iterate posting for term 'noticed': 100%|█| 9942/9942 [00:03<00:00, 3309.14it/s]
Iterate posting for term 'nurse': 100%|█| 21917/21917 [00:06<00:00, 3548.75it/s]
Iterate posting for term 'degree': 100%|█| 77062/77062 [00:40<00:00, 1911.46it/s
Iterate posting for term 'truly': 100%|█| 14011/14011 [00:04<00:00, 3409.90it/s]
Iterate posting for term 'confused': 100%|█| 14513/14513 [00:04<00:00, 3369.76it
Iterate posting for term '67,449': 100%|████████| 2/2 [00:00<00:00, 1879.59it/s]
Iterate posting for term 'bls': 100%|█████| 8111/8111 [00:02<00:00, 3365.54it/s]
Iterate posting for term 'asn': 100%|███████| 232/232 [00:00<00:00, 3227.38it/s]
```

```
-----------------------------------------------------
Current query:  bsn rn difference between
```

```
Iterate posting for term 'bsn': 100%|█████| 2060/2060 [00:00<00:00, 3609.87it/s]
Iterate posting for term 'rn': 100%|██████| 6217/6217 [00:01<00:00, 3560.14it/s]
Iterate posting for term 'difference': 100%|█| 75516/75516 [00:53<00:00, 1411.59
Iterate posting for term 'between': 100%|█| 416673/416673 [03:13<00:00, 2147.98i
```

```
-----------------------------------------------------
Current query:  bsn rn difference
```

```
Iterate posting for term 'bsn': 100%|█████| 2060/2060 [00:00<00:00, 3228.50it/s]
Iterate posting for term 'rn': 100%|██████| 6217/6217 [00:01<00:00, 3337.27it/s]
Iterate posting for term 'difference': 100%|█| 75516/75516 [00:44<00:00, 1699.87
```

```
-----------------------------------------------------
Current query: medicare's definition of mechanical ventilation ventilation mechanical ve
ntilator beneficiary medicare claims 73 documentation showed hours
```

```
Iterate posting for term 'medicare': 100%|█| 22024/22024 [00:06<00:00, 3430.55it
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:23<00:00, 2055.
Iterate posting for term 'mechanical': 100%|█| 17693/17693 [00:05<00:00, 3356.80
Iterate posting for term 'ventilation': 100%|█| 4314/4314 [00:01<00:00, 3496.25i
Iterate posting for term 'ventilation': 100%|█| 4314/4314 [00:01<00:00, 2890.86i
Iterate posting for term 'mechanical': 100%|█| 17693/17693 [00:05<00:00, 3287.63
Iterate posting for term 'ventilator': 100%|█| 726/726 [00:00<00:00, 3325.47it/s
Iterate posting for term 'beneficiary': 100%|█| 4589/4589 [00:28<00:00, 159.92it
Iterate posting for term 'medicare': 100%|█| 22024/22024 [00:06<00:00, 3188.12it
Iterate posting for term 'claims': 100%|█| 27103/27103 [00:08<00:00, 3337.96it/s
Iterate posting for term '73': 100%|██████| 5907/5907 [00:01<00:00, 3424.43it/s]
Iterate posting for term 'documentation': 100%|█| 9344/9344 [00:02<00:00, 3450.1
Iterate posting for term 'showed': 100%|█| 12563/12563 [00:03<00:00, 3187.79it/s
Iterate posting for term 'hours': 100%|█| 187884/187884 [02:19<00:00, 1346.49it/
```

```
-----------------------------------------------------
Current query: medicare's definition of mechanical ventilation ventilation mechanical ve
ntilator beneficiary medicare claims 73 documentation showed hours hospitals ventilatio
n.â log interface.n selected
```

```
Iterate posting for term 'medicare': 100%|█| 22024/22024 [00:07<00:00, 2983.37it
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:28<00:00, 1920.
Iterate posting for term 'mechanical': 100%|█| 17693/17693 [00:05<00:00, 3464.01
Iterate posting for term 'ventilation': 100%|█| 4314/4314 [00:01<00:00, 3569.26i
Iterate posting for term 'ventilation': 100%|█| 4314/4314 [00:01<00:00, 3657.40i
Iterate posting for term 'mechanical': 100%|█| 17693/17693 [00:24<00:00, 719.48i
Iterate posting for term 'ventilator': 100%|█| 726/726 [00:00<00:00, 2176.51it/s
Iterate posting for term 'beneficiary': 100%|█| 4589/4589 [00:01<00:00, 3019.18i
Iterate posting for term 'medicare': 100%|█| 22024/22024 [00:06<00:00, 3363.46it
Iterate posting for term 'claims': 100%|█| 27103/27103 [00:08<00:00, 3191.56it/s
Iterate posting for term '73': 100%|██████| 5907/5907 [00:02<00:00, 2733.05it/s]
Iterate posting for term 'documentation': 100%|█| 9344/9344 [00:02<00:00, 3145.6
Iterate posting for term 'showed': 100%|█| 12563/12563 [00:04<00:00, 3126.79it/s
Iterate posting for term 'hours': 100%|█| 187884/187884 [01:55<00:00, 1622.99it/
Iterate posting for term 'hospitals': 100%|█| 13696/13696 [00:04<00:00, 2945.85i
Iterate posting for term 'ventilation.â': 100%|█| 2/2 [00:00<00:00, 2750.36it/s]
Iterate posting for term 'log': 100%|███| 18657/18657 [00:05<00:00, 3300.91it/s]
Iterate posting for term 'interface.n': 100%|███| 8/8 [00:00<00:00, 1465.52it/s]
Iterate posting for term 'selected': 100%|█| 24937/24937 [00:07<00:00, 3386.07it
```

```
-----------------------------------------------------
Current query:  ventilation mechanical medicare definition
```

```
Iterate posting for term 'ventilation': 100%|█| 4314/4314 [00:01<00:00, 3451.05i
Iterate posting for term 'mechanical': 100%|█| 17693/17693 [00:05<00:00, 3369.76
Iterate posting for term 'medicare': 100%|█| 22024/22024 [00:32<00:00, 673.34it/
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:09<00:00, 2451.
```

```
-----------------------------------------------------
Current query:  ventilation mechanical medicare
```

```
Iterate posting for term 'ventilation': 100%|█| 4314/4314 [00:01<00:00, 3515.80i
Iterate posting for term 'mechanical': 100%|█| 17693/17693 [00:04<00:00, 3571.43
Iterate posting for term 'medicare': 100%|█| 22024/22024 [00:06<00:00, 3497.39it
```

```
-----------------------------------------------------
Current query: how to find the midsegment of a trapezoid midsegment trapezoid midpoints
bases parallel segment lengths length sides sum
```

```
Iterate posting for term 'how': 100%|█| 553931/553931 [04:24<00:00, 2095.60it/s]
Iterate posting for term 'find': 100%|█| 272952/272952 [01:51<00:00, 2437.91it/s
Iterate posting for term 'midsegment': 100%|██| 38/38 [00:00<00:00, 2612.07it/s]
Iterate posting for term 'trapezoid': 100%|█| 611/611 [00:00<00:00, 3061.95it/s]
Iterate posting for term 'midsegment': 100%|██| 38/38 [00:00<00:00, 3205.36it/s]
Iterate posting for term 'trapezoid': 100%|█| 611/611 [00:00<00:00, 3323.83it/s]
Iterate posting for term 'midpoints': 100%|█| 149/149 [00:00<00:00, 2864.66it/s]
Iterate posting for term 'bases': 100%|████| 9849/9849 [00:22<00:00, 437.76it/s]
Iterate posting for term 'parallel': 100%|█| 12239/12239 [00:04<00:00, 2974.88it
Iterate posting for term 'segment': 100%|█| 9662/9662 [00:03<00:00, 3037.17it/s]
Iterate posting for term 'lengths': 100%|█| 6253/6253 [00:01<00:00, 3317.42it/s]
Iterate posting for term 'length': 100%|█| 75031/75031 [00:21<00:00, 3436.46it/s
Iterate posting for term 'sides': 100%|█| 32126/32126 [00:28<00:00, 1143.78it/s]
Iterate posting for term 'sum': 100%|███| 15735/15735 [00:04<00:00, 3232.12it/s]
```

```
-----------------------------------------------------
Current query: how to find the midsegment of a trapezoid midsegment trapezoid midpoints
bases parallel segment lengths length sides sum base two transversals conjecture connect
s
```

```
Iterate posting for term 'how': 100%|█| 553931/553931 [04:36<00:00, 2006.69it/s]
Iterate posting for term 'find': 100%|█| 272952/272952 [01:49<00:00, 2498.63it/s
Iterate posting for term 'midsegment': 100%|██| 38/38 [00:00<00:00, 3270.75it/s]
Iterate posting for term 'trapezoid': 100%|█| 611/611 [00:00<00:00, 3268.67it/s]
Iterate posting for term 'midsegment': 100%|██| 38/38 [00:00<00:00, 2803.19it/s]
Iterate posting for term 'trapezoid': 100%|█| 611/611 [00:00<00:00, 3470.01it/s]
Iterate posting for term 'midpoints': 100%|█| 149/149 [00:00<00:00, 3084.11it/s]
Iterate posting for term 'bases': 100%|███| 9849/9849 [00:02<00:00, 3420.55it/s]
Iterate posting for term 'parallel': 100%|█| 12239/12239 [00:03<00:00, 3484.34it
Iterate posting for term 'segment': 100%|█| 9662/9662 [00:02<00:00, 3480.73it/s]
Iterate posting for term 'lengths': 100%|█| 6253/6253 [00:01<00:00, 3414.86it/s]
Iterate posting for term 'length': 100%|█| 75031/75031 [00:38<00:00, 1932.19it/s
Iterate posting for term 'sides': 100%|█| 32126/32126 [00:10<00:00, 3055.48it/s]
Iterate posting for term 'sum': 100%|███| 15735/15735 [00:04<00:00, 3393.72it/s]
Iterate posting for term 'base': 100%|██| 72522/72522 [00:45<00:00, 1598.57it/s]
Iterate posting for term 'two': 100%|█| 524625/524625 [03:59<00:00, 2191.48it/s]
Iterate posting for term 'transversals': 100%|█| 12/12 [00:00<00:00, 2625.95it/s
Iterate posting for term 'conjecture': 100%|█| 363/363 [00:00<00:00, 2958.00it/s
Iterate posting for term 'connects': 100%|█| 11349/11349 [00:03<00:00, 3419.28it
-----------------------------------------------------
Current query:  midsegment trapezoid find how
Iterate posting for term 'midsegment': 100%|██| 38/38 [00:00<00:00, 3244.25it/s]
Iterate posting for term 'trapezoid': 100%|█| 611/611 [00:00<00:00, 3544.35it/s]
Iterate posting for term 'find': 100%|█| 272952/272952 [02:04<00:00, 2191.92it/s
Iterate posting for term 'how': 100%|█| 553931/553931 [05:05<00:00, 1811.01it/s]
-----------------------------------------------------
Current query:  midsegment trapezoid find
Iterate posting for term 'midsegment': 100%|██| 38/38 [00:00<00:00, 2760.89it/s]
Iterate posting for term 'trapezoid': 100%|█| 611/611 [00:00<00:00, 3062.46it/s]
Iterate posting for term 'find': 100%|█| 272952/272952 [02:39<00:00, 1709.04it/s
-----------------------------------------------------
Current query: what is an aml surveillance analyst analyst aml surveillance jobs salarie
s 54,259 hired 9449 769 postings
Iterate posting for term 'what': 100%|█| 542299/542299 [04:39<00:00, 1943.40it/s
Iterate posting for term 'aml': 100%|███████| 616/616 [00:00<00:00, 3285.17it/s]
Iterate posting for term 'surveillance': 100%|█| 2858/2858 [00:00<00:00, 3441.60
Iterate posting for term 'analyst': 100%|█| 6616/6616 [00:01<00:00, 3529.79it/s]
Iterate posting for term 'analyst': 100%|█| 6616/6616 [00:01<00:00, 3584.21it/s]
Iterate posting for term 'aml': 100%|███████| 616/616 [00:00<00:00, 3546.43it/s]
Iterate posting for term 'surveillance': 100%|█| 2858/2858 [00:00<00:00, 3549.90
Iterate posting for term 'jobs': 100%|██| 38953/38953 [00:31<00:00, 1225.97it/s]
Iterate posting for term 'salaries': 100%|█| 41432/41432 [00:11<00:00, 3558.99it
Iterate posting for term '54,259': 100%|████████| 1/1 [00:00<00:00, 2070.24it/s]
Iterate posting for term 'hired': 100%|███| 8301/8301 [00:02<00:00, 3471.26it/s]
Iterate posting for term '9449': 100%|████████| 13/13 [00:00<00:00, 2543.19it/s]
Iterate posting for term '769': 100%|███████| 237/237 [00:00<00:00, 3405.56it/s]
Iterate posting for term 'postings': 100%|█| 3158/3158 [00:00<00:00, 3534.12it/s
-----------------------------------------------------
Current query: what is an aml surveillance analyst analyst aml surveillance jobs salarie
s 54,259 hired 9449 769 postings nationwideverage casino reno job nv
Iterate posting for term 'what': 100%|█| 542299/542299 [04:56<00:00, 1830.95it/s
Iterate posting for term 'aml': 100%|███████| 616/616 [00:00<00:00, 3125.71it/s]
Iterate posting for term 'surveillance': 100%|█| 2858/2858 [00:00<00:00, 3298.59
Iterate posting for term 'analyst': 100%|█| 6616/6616 [00:01<00:00, 3480.59it/s]
Iterate posting for term 'analyst': 100%|█| 6616/6616 [00:01<00:00, 3678.62it/s]
Iterate posting for term 'aml': 100%|███████| 616/616 [00:00<00:00, 3154.04it/s]
Iterate posting for term 'surveillance': 100%|█| 2858/2858 [00:00<00:00, 2946.25
Iterate posting for term 'jobs': 100%|██| 38953/38953 [00:29<00:00, 1303.76it/s]
Iterate posting for term 'salaries': 100%|█| 41432/41432 [00:11<00:00, 3515.32it
Iterate posting for term '54,259': 100%|████████| 1/1 [00:00<00:00, 1476.87it/s]
Iterate posting for term 'hired': 100%|███| 8301/8301 [00:02<00:00, 2824.96it/s]
Iterate posting for term '9449': 100%|████████| 13/13 [00:00<00:00, 2607.28it/s]
Iterate posting for term '769': 100%|███████| 237/237 [00:00<00:00, 3191.49it/s]
Iterate posting for term 'postings': 100%|█| 3158/3158 [00:00<00:00, 3506.37it/s
Iterate posting for term 'nationwideverage': 100%|█| 90/90 [00:00<00:00, 3375.34
Iterate posting for term 'casino': 100%|██| 5212/5212 [00:01<00:00, 2966.45it/s]
Iterate posting for term 'reno': 100%|████| 1610/1610 [00:00<00:00, 3244.24it/s]
Iterate posting for term 'job': 100%|█| 109676/109676 [01:12<00:00, 1508.22it/s]
Iterate posting for term 'nv': 100%|██████| 2580/2580 [00:00<00:00, 2742.10it/s]
-----------------------------------------------------
Current query:  aml surveillance analyst what
Iterate posting for term 'aml': 100%|███████| 616/616 [00:00<00:00, 3212.40it/s]
Iterate posting for term 'surveillance': 100%|█| 2858/2858 [00:00<00:00, 3333.20
Iterate posting for term 'analyst': 100%|█| 6616/6616 [00:01<00:00, 3406.48it/s]
Iterate posting for term 'what': 100%|█| 542299/542299 [05:16<00:00, 1711.14it/s
-----------------------------------------------------
Current query:  aml surveillance analyst
Iterate posting for term 'aml': 100%|███████| 616/616 [00:00<00:00, 2716.25it/s]
Iterate posting for term 'surveillance': 100%|█| 2858/2858 [00:01<00:00, 2383.09
Iterate posting for term 'analyst': 100%|█| 6616/6616 [00:02<00:00, 3177.25it/s]
-----------------------------------------------------
Current query: what is the daily life of thai people thai sanuk everyday calorie livestr
ong.com chat friends values à curry
Iterate posting for term 'what': 100%|█| 542299/542299 [18:39<00:00, 484.37it/s]
Iterate posting for term 'daily': 100%|█| 81166/81166 [00:27<00:00, 2923.31it/s]
Iterate posting for term 'life': 100%|█| 208672/208672 [01:51<00:00, 1878.11it/s
Iterate posting for term 'thai': 100%|████| 2626/2626 [00:00<00:00, 3423.72it/s]
Iterate posting for term 'people': 100%|█| 446203/446203 [04:20<00:00, 1712.90it
Iterate posting for term 'thai': 100%|████| 2626/2626 [00:00<00:00, 3269.00it/s]
Iterate posting for term 'sanuk': 100%|███████| 14/14 [00:00<00:00, 2682.03it/s]
Iterate posting for term 'everyday': 100%|█| 14336/14336 [00:07<00:00, 2021.97it
Iterate posting for term 'calorie': 100%|█| 20772/20772 [00:06<00:00, 3001.88it/
Iterate posting for term 'livestrong.com': 100%|█| 64/64 [00:00<00:00, 3364.32it
Iterate posting for term 'chat': 100%|████| 6597/6597 [00:02<00:00, 2748.98it/s]
Iterate posting for term 'friends': 100%|█| 37603/37603 [00:11<00:00, 3404.12it/
Iterate posting for term 'values': 100%|█| 44228/44228 [00:51<00:00, 865.44it/s]
Iterate posting for term 'à': 100%|███████| 2590/2590 [00:00<00:00, 2932.31it/s]
Iterate posting for term 'curry': 100%|███| 3934/3934 [00:01<00:00, 3146.45it/s]
-----------------------------------------------------
Current query: what is the daily life of thai people thai sanuk everyday calorie livestr
ong.com chat friends values à curry fun jai persons life daily
Iterate posting for term 'what': 100%|█| 542299/542299 [1:05:44<00:00, 137.50it/
Iterate posting for term 'daily': 100%|█| 81166/81166 [00:59<00:00, 1372.19it/s]
Iterate posting for term 'life': 100%|█| 208672/208672 [01:45<00:00, 1973.23it/s
Iterate posting for term 'thai': 100%|████| 2626/2626 [00:00<00:00, 3481.27it/s]
Iterate posting for term 'people': 100%|█| 446203/446203 [04:09<00:00, 1791.44it
Iterate posting for term 'thai': 100%|████| 2626/2626 [00:00<00:00, 3270.08it/s]
Iterate posting for term 'sanuk': 100%|███████| 14/14 [00:00<00:00, 3197.05it/s]
Iterate posting for term 'everyday': 100%|█| 14336/14336 [00:04<00:00, 3453.03it
Iterate posting for term 'calorie': 100%|█| 20772/20772 [00:05<00:00, 3566.91it/
Iterate posting for term 'livestrong.com': 100%|█| 64/64 [00:00<00:00, 2643.43it
Iterate posting for term 'chat': 100%|████| 6597/6597 [00:01<00:00, 3539.02it/s]
Iterate posting for term 'friends': 100%|█| 37603/37603 [00:28<00:00, 1302.98it/
Iterate posting for term 'values': 100%|█| 44228/44228 [00:13<00:00, 3342.09it/s
Iterate posting for term 'à': 100%|███████| 2590/2590 [00:00<00:00, 3457.00it/s]
Iterate posting for term 'curry': 100%|███| 3934/3934 [00:01<00:00, 2879.52it/s]
Iterate posting for term 'fun': 100%|███| 34349/34349 [00:10<00:00, 3236.23it/s]
Iterate posting for term 'jai': 100%|███████| 316/316 [00:00<00:00, 3533.42it/s]
Iterate posting for term 'persons': 100%|█| 20293/20293 [00:06<00:00, 3205.24it/
Iterate posting for term 'life': 100%|█| 208672/208672 [02:08<00:00, 1618.69it/s
Iterate posting for term 'daily': 100%|█| 81166/81166 [00:23<00:00, 3463.25it/s]
-----------------------------------------------------
Current query:  thai daily life people what
Iterate posting for term 'thai': 100%|████| 2626/2626 [00:00<00:00, 2775.42it/s]
Iterate posting for term 'daily': 100%|█| 81166/81166 [00:56<00:00, 1437.77it/s]
Iterate posting for term 'life': 100%|█| 208672/208672 [01:57<00:00, 1776.85it/s
Iterate posting for term 'people': 100%|█| 446203/446203 [04:01<00:00, 1845.68it
Iterate posting for term 'what': 100%|█| 542299/542299 [04:29<00:00, 2013.89it/s
-----------------------------------------------------
Current query:  thai daily life
Iterate posting for term 'thai': 100%|█████| 2626/2626 [00:20<00:00, 127.78it/s]
Iterate posting for term 'daily': 100%|█| 81166/81166 [00:23<00:00, 3408.97it/s]
Iterate posting for term 'life': 100%|█| 208672/208672 [01:49<00:00, 1898.03it/s
-----------------------------------------------------
Current query: definition of a sigmet sigmet sigmets airmet turbulence airmets icing air
craft takeoff convective visibility
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:17<00:00, 2212.
Iterate posting for term 'sigmet': 100%|██████| 19/19 [00:00<00:00, 3066.48it/s]
Iterate posting for term 'sigmet': 100%|██████| 19/19 [00:00<00:00, 2209.30it/s]
Iterate posting for term 'sigmets': 100%|█████| 11/11 [00:00<00:00, 2735.04it/s]
Iterate posting for term 'airmet': 100%|██████| 12/12 [00:00<00:00, 2649.73it/s]
Iterate posting for term 'turbulence': 100%|█| 624/624 [00:00<00:00, 3382.70it/s
Iterate posting for term 'airmets': 100%|███████| 7/7 [00:00<00:00, 2781.11it/s]
Iterate posting for term 'icing': 100%|███| 1899/1899 [00:00<00:00, 3368.65it/s]
Iterate posting for term 'aircraft': 100%|█| 14160/14160 [00:04<00:00, 3509.34it
Iterate posting for term 'takeoff': 100%|███| 741/741 [00:00<00:00, 3545.29it/s]
Iterate posting for term 'convective': 100%|█| 344/344 [00:00<00:00, 3598.25it/s
Iterate posting for term 'visibility': 100%|█| 4047/4047 [00:01<00:00, 3520.87it
-----------------------------------------------------
Current query: definition of a sigmet sigmet sigmets airmet turbulence airmets icing air
craft takeoff convective visibility meteorological weather hazardous pilot plural
Iterate posting for term 'definition': 100%|█| 170760/170760 [02:13<00:00, 1282.
Iterate posting for term 'sigmet': 100%|██████| 19/19 [00:00<00:00, 2944.68it/s]
Iterate posting for term 'sigmet': 100%|██████| 19/19 [00:00<00:00, 2906.87it/s]
Iterate posting for term 'sigmets': 100%|██████| 11/11 [00:00<00:00, 2628.01it/s]
Iterate posting for term 'airmet': 100%|██████| 12/12 [00:00<00:00, 2949.06it/s]
Iterate posting for term 'turbulence': 100%|█| 624/624 [00:00<00:00, 2851.27it/s
Iterate posting for term 'airmets': 100%|███████| 7/7 [00:00<00:00, 2407.16it/s]
Iterate posting for term 'icing': 100%|███| 1899/1899 [00:00<00:00, 3114.70it/s]
Iterate posting for term 'aircraft': 100%|█| 14160/14160 [00:04<00:00, 3191.63it
Iterate posting for term 'takeoff': 100%|███| 741/741 [00:00<00:00, 3069.73it/s]
Iterate posting for term 'convective': 100%|█| 344/344 [00:00<00:00, 2651.62it/s
Iterate posting for term 'visibility': 100%|█| 4047/4047 [00:01<00:00, 3268.49it
Iterate posting for term 'meteorological': 100%|█| 996/996 [00:00<00:00, 3359.99
Iterate posting for term 'weather': 100%|█| 87214/87214 [01:03<00:00, 1380.59it/
Iterate posting for term 'hazardous': 100%|█| 6272/6272 [00:02<00:00, 3022.63it/
Iterate posting for term 'pilot': 100%|█| 10105/10105 [00:03<00:00, 3188.34it/s]
Iterate posting for term 'plural': 100%|█| 16213/16213 [00:05<00:00, 2897.88it/s
-----------------------------------------------------
Current query:  sigmet definition
Iterate posting for term 'sigmet': 100%|██████| 19/19 [00:00<00:00, 2101.24it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:31<00:00, 1871.
-----------------------------------------------------
Current query:  sigmet definition
Iterate posting for term 'sigmet': 100%|██████| 19/19 [00:00<00:00, 2972.02it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [02:52<00:00, 991.4
-----------------------------------------------------
Current query: cost of interior concrete flooring flooring concrete floors interior floo
r cost finish polishing waxing gloss
Iterate posting for term 'cost': 100%|█| 221704/221704 [02:26<00:00, 1516.44it/s
Iterate posting for term 'interior': 100%|█| 17121/17121 [00:06<00:00, 2759.20it
Iterate posting for term 'concrete': 100%|█| 21620/21620 [00:06<00:00, 3283.77it
Iterate posting for term 'flooring': 100%|█| 6960/6960 [00:02<00:00, 3378.62it/s
Iterate posting for term 'flooring': 100%|█| 6960/6960 [00:01<00:00, 3522.52it/s
Iterate posting for term 'concrete': 100%|█| 21620/21620 [00:06<00:00, 3573.12it
Iterate posting for term 'floors': 100%|██| 7312/7312 [00:02<00:00, 3405.82it/s]
Iterate posting for term 'interior': 100%|█| 17121/17121 [00:05<00:00, 3420.34it
Iterate posting for term 'floor': 100%|█| 37669/37669 [00:12<00:00, 3038.85it/s]
Iterate posting for term 'cost': 100%|█| 221704/221704 [02:06<00:00, 1755.10it/s
Iterate posting for term 'finish': 100%|█| 21723/21723 [00:07<00:00, 3016.38it/s
Iterate posting for term 'polishing': 100%|█| 1206/1206 [00:33<00:00, 35.83it/s]
Iterate posting for term 'waxing': 100%|██| 1286/1286 [00:00<00:00, 1638.49it/s]
Iterate posting for term 'gloss': 100%|███| 1857/1857 [00:00<00:00, 2079.85it/s]
-----------------------------------------------------
Current query: cost of interior concrete flooring flooring concrete floors interior floo
r cost finish polishing waxing gloss ideas maintenance design stains tips
Iterate posting for term 'cost': 100%|█| 221704/221704 [01:59<00:00, 1850.13it/s
Iterate posting for term 'interior': 100%|█| 17121/17121 [00:44<00:00, 384.50it/
Iterate posting for term 'concrete': 100%|█| 21620/21620 [00:21<00:00, 1022.74it
Iterate posting for term 'flooring': 100%|█| 6960/6960 [00:03<00:00, 2023.09it/s
Iterate posting for term 'flooring': 100%|█| 6960/6960 [00:02<00:00, 3474.82it/s
Iterate posting for term 'concrete': 100%|█| 21620/21620 [00:06<00:00, 3401.35it
Iterate posting for term 'floors': 100%|██| 7312/7312 [00:02<00:00, 2934.82it/s]
Iterate posting for term 'interior': 100%|█| 17121/17121 [00:05<00:00, 3115.00it
Iterate posting for term 'floor': 100%|██| 37669/37669 [01:03<00:00, 594.62it/s]
Iterate posting for term 'cost': 100%|█| 221704/221704 [01:42<00:00, 2155.37it/s
Iterate posting for term 'finish': 100%|█| 21723/21723 [00:06<00:00, 3110.07it/s
Iterate posting for term 'polishing': 100%|█| 1206/1206 [00:00<00:00, 2985.14it/
Iterate posting for term 'waxing': 100%|██| 1286/1286 [00:00<00:00, 3302.98it/s]
Iterate posting for term 'gloss': 100%|███| 1857/1857 [00:00<00:00, 3420.01it/s]
Iterate posting for term 'ideas': 100%|█| 24500/24500 [00:07<00:00, 3292.94it/s]
Iterate posting for term 'maintenance': 100%|█| 26842/26842 [00:10<00:00, 2578.4
Iterate posting for term 'design': 100%|█| 63936/63936 [00:19<00:00, 3356.33it/s
Iterate posting for term 'stains': 100%|██| 5968/5968 [00:02<00:00, 2756.37it/s]
Iterate posting for term 'tips': 100%|███| 41055/41055 [00:47<00:00, 855.92it/s]
-----------------------------------------------------
Current query:  flooring interior concrete cost
Iterate posting for term 'flooring': 100%|█| 6960/6960 [00:02<00:00, 3065.38it/s
Iterate posting for term 'interior': 100%|█| 17121/17121 [00:05<00:00, 3306.81it
Iterate posting for term 'concrete': 100%|█| 21620/21620 [00:06<00:00, 3439.13it
Iterate posting for term 'cost': 100%|█| 221704/221704 [02:13<00:00, 1665.83it/s
-----------------------------------------------------
Current query:  flooring interior concrete
Iterate posting for term 'flooring': 100%|█| 6960/6960 [00:02<00:00, 3401.21it/s
Iterate posting for term 'interior': 100%|█| 17121/17121 [00:04<00:00, 3435.89it
Iterate posting for term 'concrete': 100%|█| 21620/21620 [00:06<00:00, 3137.83it
-----------------------------------------------------
Current query: what is the most popular food in switzerland switzerland swiss popular ap
penzeller vacherin emmental italian boys pasta names
Iterate posting for term 'what': 100%|█| 542299/542299 [04:53<00:00, 1847.85it/s
Iterate posting for term 'most': 100%|█| 783486/783486 [07:33<00:00, 1727.62it/s
Iterate posting for term 'popular': 100%|█| 103940/103940 [01:17<00:00, 1348.44i
Iterate posting for term 'food': 100%|█| 183952/183952 [01:47<00:00, 1710.74it/s
Iterate posting for term 'switzerland': 100%|█| 6118/6118 [00:01<00:00, 3358.99i
Iterate posting for term 'switzerland': 100%|█| 6118/6118 [00:01<00:00, 3612.45i
Iterate posting for term 'swiss': 100%|███| 5282/5282 [00:01<00:00, 3507.37it/s]
Iterate posting for term 'popular': 100%|█| 103940/103940 [01:06<00:00, 1552.70i
Iterate posting for term 'appenzeller': 100%|███| 8/8 [00:00<00:00, 2274.57it/s]
Iterate posting for term 'vacherin': 100%|████| 14/14 [00:00<00:00, 2993.34it/s]
Iterate posting for term 'emmental': 100%|████| 93/93 [00:00<00:00, 3099.07it/s]
Iterate posting for term 'italian': 100%|█| 23902/23902 [00:08<00:00, 2842.25it/
Iterate posting for term 'boys': 100%|██| 20345/20345 [00:06<00:00, 3015.46it/s]
Iterate posting for term 'pasta': 100%|███| 8276/8276 [00:02<00:00, 3316.12it/s]
Iterate posting for term 'names': 100%|█| 64406/64406 [01:03<00:00, 1020.54it/s]
-----------------------------------------------------
Current query: what is the most popular food in switzerland switzerland swiss popular ap
penzeller vacherin emmental italian boys pasta names cheese dishes racletten rice.ood pa
stetli
Iterate posting for term 'what': 100%|█| 542299/542299 [05:01<00:00, 1798.32it/s
Iterate posting for term 'most': 100%|█| 783486/783486 [06:52<00:00, 1898.07it/s
Iterate posting for term 'popular': 100%|█| 103940/103940 [00:52<00:00, 1965.24i
Iterate posting for term 'food': 100%|█| 183952/183952 [01:35<00:00, 1929.35it/s
Iterate posting for term 'switzerland': 100%|█| 6118/6118 [00:01<00:00, 3455.46i
Iterate posting for term 'switzerland': 100%|█| 6118/6118 [00:02<00:00, 2449.24i
Iterate posting for term 'swiss': 100%|███| 5282/5282 [00:02<00:00, 1839.96it/s]
Iterate posting for term 'popular': 100%|█| 103940/103940 [01:11<00:00, 1458.97i
Iterate posting for term 'appenzeller': 100%|███| 8/8 [00:00<00:00, 2190.09it/s]
Iterate posting for term 'vacherin': 100%|████| 14/14 [00:00<00:00, 2677.38it/s]
Iterate posting for term 'emmental': 100%|████| 93/93 [00:00<00:00, 3259.03it/s]
Iterate posting for term 'italian': 100%|█| 23902/23902 [00:07<00:00, 3404.76it/
Iterate posting for term 'boys': 100%|██| 20345/20345 [00:05<00:00, 3403.30it/s]
Iterate posting for term 'pasta': 100%|███| 8276/8276 [00:02<00:00, 3430.34it/s]
Iterate posting for term 'names': 100%|█| 64406/64406 [00:48<00:00, 1340.07it/s]
Iterate posting for term 'cheese': 100%|█| 26063/26063 [00:08<00:00, 3057.03it/s
Iterate posting for term 'dishes': 100%|█| 10857/10857 [00:03<00:00, 3182.86it/s
Iterate posting for term 'racletten': 100%|█████| 1/1 [00:00<00:00, 2347.12it/s]
Iterate posting for term 'rice.ood': 100%|██████| 1/1 [00:00<00:00, 2309.64it/s]
Iterate posting for term 'pastetli': 100%|██████| 1/1 [00:00<00:00, 2570.04it/s]
-----------------------------------------------------
Current query:  switzerland popular food what most
```

```
Iterate posting for term 'switzerland': 100%|█| 6118/6118 [00:01<00:00, 3149.30i
Iterate posting for term 'popular': 100%|█| 103940/103940 [01:17<00:00, 1341.87i
Iterate posting for term 'food': 100%|█| 183952/183952 [01:37<00:00, 1894.28it/s
Iterate posting for term 'what': 100%|█| 542299/542299 [04:48<00:00, 1882.87it/s
Iterate posting for term 'most': 100%|█| 783486/783486 [06:52<00:00, 1898.84it/s
```

```
-----------------------------------------------------
Current query:  switzerland popular food
```

```
Iterate posting for term 'switzerland': 100%|█| 6118/6118 [00:02<00:00, 2961.39i
Iterate posting for term 'popular': 100%|█| 103940/103940 [00:31<00:00, 3268.09i
Iterate posting for term 'food': 100%|█| 183952/183952 [01:46<00:00, 1723.86it/s
```

```
-----------------------------------------------------
Current query: how are some sharks warm blooded blooded sharks warm cold shark white pre
dators hotblooded killersâ warmblooded
```

```
Iterate posting for term 'how': 100%|█| 553931/553931 [05:00<00:00, 1844.65it/s]
Iterate posting for term 'some': 100%|█| 586484/586484 [04:40<00:00, 2092.07it/s
Iterate posting for term 'sharks': 100%|██| 3549/3549 [00:01<00:00, 3390.50it/s]
Iterate posting for term 'warm': 100%|██| 48286/48286 [00:35<00:00, 1372.83it/s]
Iterate posting for term 'blooded': 100%|█| 1506/1506 [00:00<00:00, 3292.09it/s]
Iterate posting for term 'blooded': 100%|█| 1506/1506 [00:00<00:00, 3552.74it/s]
Iterate posting for term 'sharks': 100%|██| 3549/3549 [00:01<00:00, 3377.73it/s]
Iterate posting for term 'warm': 100%|██| 48286/48286 [00:42<00:00, 1135.90it/s]
Iterate posting for term 'cold': 100%|██| 71620/71620 [01:09<00:00, 1023.61it/s]
Iterate posting for term 'shark': 100%|███| 4726/4726 [00:01<00:00, 3097.12it/s]
Iterate posting for term 'white': 100%|█| 127977/127977 [01:49<00:00, 1173.75it/
Iterate posting for term 'predators': 100%|█| 5677/5677 [00:02<00:00, 2814.11it/
Iterate posting for term 'hotblooded': 100%|████| 4/4 [00:00<00:00, 3098.86it/s]
Iterate posting for term 'killersâ': 100%|████| 10/10 [00:00<00:00, 2792.67it/s]
Iterate posting for term 'warmblooded': 100%|█| 30/30 [00:00<00:00, 1837.78it/s]
```

```
-----------------------------------------------------
Current query: how are some sharks warm blooded blooded sharks warm cold shark white pre
dators hotblooded killersâ warmblooded whales sharks.â ecto endotherms they
```

```
Iterate posting for term 'how': 100%|█| 553931/553931 [06:13<00:00, 1483.58it/s]
Iterate posting for term 'some': 100%|█| 586484/586484 [06:36<00:00, 1477.67it/s
Iterate posting for term 'sharks': 100%|██| 3549/3549 [00:01<00:00, 1920.11it/s]
Iterate posting for term 'warm': 100%|██| 48286/48286 [00:14<00:00, 3274.26it/s]
Iterate posting for term 'blooded': 100%|█| 1506/1506 [00:00<00:00, 3489.69it/s]
Iterate posting for term 'blooded': 100%|█| 1506/1506 [00:00<00:00, 3488.28it/s]
Iterate posting for term 'sharks': 100%|██| 3549/3549 [00:00<00:00, 3570.50it/s]
Iterate posting for term 'warm': 100%|███| 48286/48286 [00:50<00:00, 964.00it/s]
Iterate posting for term 'cold': 100%|██| 71620/71620 [00:22<00:00, 3200.88it/s]
Iterate posting for term 'shark': 100%|███| 4726/4726 [00:01<00:00, 3412.98it/s]
Iterate posting for term 'white': 100%|█| 127977/127977 [01:15<00:00, 1688.21it/
Iterate posting for term 'predators': 100%|█| 5677/5677 [00:01<00:00, 3313.87it/
Iterate posting for term 'hotblooded': 100%|████| 4/4 [00:00<00:00, 2808.84it/s]
Iterate posting for term 'killersâ': 100%|████| 10/10 [00:00<00:00, 1660.26it/s]
Iterate posting for term 'warmblooded': 100%|█| 30/30 [00:00<00:00, 2974.47it/s]
Iterate posting for term 'whales': 100%|██| 3631/3631 [00:01<00:00, 3251.03it/s]
Iterate posting for term 'sharks.â': 100%|██████| 2/2 [00:00<00:00, 2600.31it/s]
Iterate posting for term 'ecto': 100%|████████| 67/67 [00:00<00:00, 3369.52it/s]
Iterate posting for term 'endotherms': 100%|██| 85/85 [00:00<00:00, 3037.02it/s]
```

```
-----------------------------------------------------
Current query:  blooded sharks warm how some
```

```
Iterate posting for term 'blooded': 100%|█| 1506/1506 [00:00<00:00, 2428.63it/s]
Iterate posting for term 'sharks': 100%|██| 3549/3549 [00:01<00:00, 2981.41it/s]
Iterate posting for term 'warm': 100%|███| 48286/48286 [01:08<00:00, 703.70it/s]
Iterate posting for term 'how': 100%|█| 553931/553931 [05:54<00:00, 1562.61it/s]
Iterate posting for term 'some': 100%|█| 586484/586484 [06:16<00:00, 1558.74it/s
```

```
-----------------------------------------------------
Current query:  blooded sharks warm
```

```
Iterate posting for term 'blooded': 100%|█| 1506/1506 [00:00<00:00, 2310.54it/s]
Iterate posting for term 'sharks': 100%|██| 3549/3549 [00:01<00:00, 3125.69it/s]
Iterate posting for term 'warm': 100%|██| 48286/48286 [00:14<00:00, 3431.96it/s]
```

```
-----------------------------------------------------
Current query: what is durable medical equipment consist of durable equipment dmerc medi
```

```
cal dme medicare carrier regional dmerca dmeops
```

```
Iterate posting for term 'what': 100%|█| 542299/542299 [05:09<00:00, 1754.78it/s
Iterate posting for term 'durable': 100%|█| 8934/8934 [00:02<00:00, 3377.97it/s]
Iterate posting for term 'medical': 100%|█| 147835/147835 [01:47<00:00, 1370.22i
Iterate posting for term 'equipment': 100%|█| 40506/40506 [00:13<00:00, 2924.37i
Iterate posting for term 'consist': 100%|█| 12546/12546 [00:03<00:00, 3416.48it/
Iterate posting for term 'durable': 100%|█| 8934/8934 [00:02<00:00, 3032.55it/s]
Iterate posting for term 'equipment': 100%|█| 40506/40506 [00:11<00:00, 3461.71i
Iterate posting for term 'dmerc': 100%|█████████| 9/9 [00:00<00:00, 2248.96it/s]
Iterate posting for term 'medical': 100%|█| 147835/147835 [02:11<00:00, 1125.55i
Iterate posting for term 'dme': 100%|███████| 590/590 [00:00<00:00, 3273.27it/s]
Iterate posting for term 'medicare': 100%|█| 22024/22024 [00:06<00:00, 3373.26it
Iterate posting for term 'carrier': 100%|█| 13772/13772 [00:04<00:00, 3019.39it/
Iterate posting for term 'regional': 100%|█| 19168/19168 [00:06<00:00, 3153.61it
Iterate posting for term 'dmerca': 100%|████████| 1/1 [00:00<00:00, 2341.88it/s]
Iterate posting for term 'dmeops': 100%|████████| 1/1 [00:00<00:00, 2326.29it/s]
```

```
-----------------------------------------------------
Current query: what is durable medical equipment consist of durable equipment dmerc medi
cal dme medicare carrier regional dmerca dmeops dmea sadmerc dmac items what
```

```
Iterate posting for term 'what': 100%|█| 542299/542299 [05:03<00:00, 1786.81it/s
Iterate posting for term 'durable': 100%|█| 8934/8934 [00:03<00:00, 2627.80it/s]
Iterate posting for term 'medical': 100%|█| 147835/147835 [01:18<00:00, 1883.79i
Iterate posting for term 'equipment': 100%|█| 40506/40506 [00:12<00:00, 3264.13i
Iterate posting for term 'consist': 100%|█| 12546/12546 [00:04<00:00, 3129.71it/
Iterate posting for term 'durable': 100%|█| 8934/8934 [00:02<00:00, 3268.60it/s]
Iterate posting for term 'equipment': 100%|█| 40506/40506 [00:12<00:00, 3157.87i
Iterate posting for term 'dmerc': 100%|█████████| 9/9 [00:00<00:00, 2988.11it/s]
Iterate posting for term 'medical': 100%|█| 147835/147835 [01:35<00:00, 1550.78i
Iterate posting for term 'dme': 100%|███████| 590/590 [00:00<00:00, 3455.85it/s]
Iterate posting for term 'medicare': 100%|█| 22024/22024 [00:37<00:00, 587.25it/
Iterate posting for term 'carrier': 100%|█| 13772/13772 [00:05<00:00, 2579.45it/
Iterate posting for term 'regional': 100%|█| 19168/19168 [00:06<00:00, 2758.79it
Iterate posting for term 'dmerca': 100%|████████| 1/1 [00:00<00:00, 1553.45it/s]
Iterate posting for term 'dmeops': 100%|████████| 1/1 [00:00<00:00, 1757.88it/s]
Iterate posting for term 'dmea': 100%|██████████| 3/3 [00:00<00:00, 1279.27it/s]
Iterate posting for term 'sadmerc': 100%|███████| 4/4 [00:00<00:00, 2769.89it/s]
Iterate posting for term 'dmac': 100%|██████████| 9/9 [00:00<00:00, 1733.42it/s]
Iterate posting for term 'items': 100%|█| 48073/48073 [00:16<00:00, 2906.11it/s]
Iterate posting for term 'what': 100%|█| 542299/542299 [04:57<00:00, 1821.10it/s
```

```
-----------------------------------------------------
Current query:  durable consist equipment medical what
```

```
Iterate posting for term 'durable': 100%|█| 8934/8934 [00:02<00:00, 3304.45it/s]
Iterate posting for term 'consist': 100%|█| 12546/12546 [00:03<00:00, 3376.81it/
Iterate posting for term 'equipment': 100%|█| 40506/40506 [00:34<00:00, 1163.22i
Iterate posting for term 'medical': 100%|█| 147835/147835 [01:21<00:00, 1819.42i
Iterate posting for term 'what': 100%|█| 542299/542299 [05:00<00:00, 1803.39it/s
```

```
-----------------------------------------------------
Current query:  durable consist equipment
```

```
Iterate posting for term 'durable': 100%|█| 8934/8934 [00:03<00:00, 2891.28it/s]
Iterate posting for term 'consist': 100%|█| 12546/12546 [00:03<00:00, 3322.03it/
Iterate posting for term 'equipment': 100%|█| 40506/40506 [00:11<00:00, 3413.36i
```

```
-----------------------------------------------------
Current query: exons definition biology exons introns splicing mrna rna messenger joined
transcription pre nascent
```

```
Iterate posting for term 'exons': 100%|█████| 361/361 [00:00<00:00, 3370.08it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:35<00:00, 1796.
Iterate posting for term 'biology': 100%|█| 14352/14352 [00:04<00:00, 3333.47it/
Iterate posting for term 'exons': 100%|█████| 361/361 [00:00<00:00, 2800.88it/s]
Iterate posting for term 'introns': 100%|███| 379/379 [00:00<00:00, 3426.57it/s]
Iterate posting for term 'splicing': 100%|██| 546/546 [00:00<00:00, 3429.22it/s]
Iterate posting for term 'mrna': 100%|████| 3961/3961 [00:01<00:00, 3551.58it/s]
Iterate posting for term 'rna': 100%|███| 11858/11858 [00:05<00:00, 2309.49it/s]
Iterate posting for term 'messenger': 100%|█| 4961/4961 [00:01<00:00, 3142.17it/
Iterate posting for term 'joined': 100%|█| 17156/17156 [00:05<00:00, 3173.91it/s
Iterate posting for term 'transcription': 100%|█| 6389/6389 [00:01<00:00, 3410.7
Iterate posting for term 'pre': 100%|███| 41889/41889 [00:12<00:00, 3305.27it/s]
Iterate posting for term 'nascent': 100%|███| 320/320 [00:00<00:00, 3462.78it/s]
----------------------------------------------------
Current query: exons definition biology exons introns splicing mrna rna messenger joined
transcription pre nascent nucleus genes molecular biology encoded
Iterate posting for term 'exons': 100%|█████| 361/361 [00:00<00:00, 3296.89it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:34<00:00, 1816.
Iterate posting for term 'biology': 100%|█| 14352/14352 [00:04<00:00, 3122.02it/
Iterate posting for term 'exons': 100%|█████| 361/361 [00:00<00:00, 3159.74it/s]
Iterate posting for term 'introns': 100%|███| 379/379 [00:00<00:00, 3347.90it/s]
Iterate posting for term 'splicing': 100%|██| 546/546 [00:00<00:00, 3276.62it/s]
Iterate posting for term 'mrna': 100%|████| 3961/3961 [00:01<00:00, 3381.34it/s]
Iterate posting for term 'rna': 100%|███| 11858/11858 [00:03<00:00, 3321.46it/s]
Iterate posting for term 'messenger': 100%|█| 4961/4961 [00:01<00:00, 3293.08it/
Iterate posting for term 'joined': 100%|█| 17156/17156 [00:05<00:00, 3235.20it/s
Iterate posting for term 'transcription': 100%|█| 6389/6389 [00:02<00:00, 2479.5
Iterate posting for term 'pre': 100%|███| 41889/41889 [00:13<00:00, 3106.84it/s]
Iterate posting for term 'nascent': 100%|███| 320/320 [00:00<00:00, 3400.89it/s]
Iterate posting for term 'nucleus': 100%|█| 16246/16246 [00:04<00:00, 3332.76it/
Iterate posting for term 'genes': 100%|██| 15086/15086 [00:39<00:00, 382.89it/s]
Iterate posting for term 'molecular': 100%|█| 12148/12148 [00:03<00:00, 3183.92i
Iterate posting for term 'biology': 100%|█| 14352/14352 [00:04<00:00, 3334.08it/
Iterate posting for term 'encoded': 100%|█| 2174/2174 [00:00<00:00, 3160.71it/s]
----------------------------------------------------
Current query:  exons biology definition
Iterate posting for term 'exons': 100%|█████| 361/361 [00:00<00:00, 3322.97it/s]
Iterate posting for term 'biology': 100%|█| 14352/14352 [00:04<00:00, 3119.34it/
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:39<00:00, 1710.
----------------------------------------------------
Current query:  exons biology definition
Iterate posting for term 'exons': 100%|█████| 361/361 [00:00<00:00, 3263.30it/s]
Iterate posting for term 'biology': 100%|█| 14352/14352 [00:04<00:00, 3400.47it/
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:36<00:00, 1774.
----------------------------------------------------
Current query: define visceral? visceral parietal define viscera splanchnic cavity surfa
ce unreasoning serous peritoneum
Iterate posting for term 'define': 100%|█| 25586/25586 [00:07<00:00, 3373.68it/s
Iterate posting for term 'visceral': 100%|█| 2487/2487 [00:00<00:00, 3385.48it/s
Iterate posting for term 'visceral': 100%|█| 2487/2487 [00:00<00:00, 3495.05it/s
Iterate posting for term 'parietal': 100%|█| 2348/2348 [00:00<00:00, 3429.25it/s
Iterate posting for term 'define': 100%|█| 25586/25586 [00:07<00:00, 3461.72it/s
Iterate posting for term 'viscera': 100%|███| 680/680 [00:00<00:00, 3421.71it/s]
Iterate posting for term 'splanchnic': 100%|█| 122/122 [00:00<00:00, 3411.73it/s
Iterate posting for term 'cavity': 100%|█| 14528/14528 [00:32<00:00, 444.97it/s]
Iterate posting for term 'surface': 100%|█| 92185/92185 [00:32<00:00, 2858.32it/
Iterate posting for term 'unreasoning': 100%|█| 33/33 [00:00<00:00, 2638.53it/s]
Iterate posting for term 'serous': 100%|██| 1362/1362 [00:00<00:00, 3386.49it/s]
Iterate posting for term 'peritoneum': 100%|█| 1040/1040 [00:00<00:00, 3461.19it
----------------------------------------------------
Current query: define visceral? visceral parietal define viscera splanchnic cavity surfa
ce unreasoning serous peritoneum pleura cupula hilar instinctive circulation
Iterate posting for term 'define': 100%|█| 25586/25586 [00:37<00:00, 674.58it/s]
Iterate posting for term 'visceral': 100%|█| 2487/2487 [00:01<00:00, 2223.45it/s
Iterate posting for term 'visceral': 100%|█| 2487/2487 [00:00<00:00, 3238.22it/s
Iterate posting for term 'parietal': 100%|█| 2348/2348 [00:00<00:00, 2672.78it/s
Iterate posting for term 'define': 100%|█| 25586/25586 [00:07<00:00, 3359.38it/s
Iterate posting for term 'viscera': 100%|███| 680/680 [00:00<00:00, 3419.73it/s]
Iterate posting for term 'splanchnic': 100%|█| 122/122 [00:00<00:00, 3063.46it/s
Iterate posting for term 'cavity': 100%|█| 14528/14528 [00:04<00:00, 3126.05it/s
Iterate posting for term 'surface': 100%|█| 92185/92185 [00:59<00:00, 1538.46it/
Iterate posting for term 'unreasoning': 100%|█| 33/33 [00:00<00:00, 2086.47it/s]
Iterate posting for term 'serous': 100%|██| 1362/1362 [00:00<00:00, 3093.70it/s]
Iterate posting for term 'peritoneum': 100%|█| 1040/1040 [00:00<00:00, 2949.95it
Iterate posting for term 'pleura': 100%|██| 1090/1090 [00:00<00:00, 2433.81it/s]
Iterate posting for term 'cupula': 100%|██████| 21/21 [00:00<00:00, 1330.38it/s]
Iterate posting for term 'hilar': 100%|█████| 129/129 [00:00<00:00, 2189.11it/s]
Iterate posting for term 'instinctive': 100%|█| 415/415 [00:00<00:00, 1824.38it/
Iterate posting for term 'circulation': 100%|█| 12282/12282 [00:04<00:00, 2912.2
-----------------------------------------------------
Current query:  visceral define
Iterate posting for term 'visceral': 100%|█| 2487/2487 [00:00<00:00, 3379.93it/s
Iterate posting for term 'define': 100%|█| 25586/25586 [00:07<00:00, 3264.42it/s
-----------------------------------------------------
Current query:  visceral define
Iterate posting for term 'visceral': 100%|█| 2487/2487 [00:00<00:00, 3397.90it/s
Iterate posting for term 'define': 100%|█| 25586/25586 [00:07<00:00, 3434.19it/s
-----------------------------------------------------
Current query: tracheids are part of _____. tracheids xylem vessel vessels cells element
s parenchyma cells.water biopolymer end
Iterate posting for term 'tracheids': 100%|█| 125/125 [00:00<00:00, 2548.12it/s]
Iterate posting for term 'part': 100%|█| 289955/289955 [02:44<00:00, 1763.36it/s
Iterate posting for term 'tracheids': 100%|█| 125/125 [00:00<00:00, 3351.54it/s]
Iterate posting for term 'xylem': 100%|███| 1394/1394 [00:00<00:00, 3339.62it/s]
Iterate posting for term 'vessel': 100%|█| 12963/12963 [00:24<00:00, 534.43it/s]
Iterate posting for term 'vessels': 100%|█| 30775/30775 [00:09<00:00, 3333.26it/
Iterate posting for term 'cells': 100%|█| 134812/134812 [01:16<00:00, 1761.46it/
Iterate posting for term 'elements': 100%|█| 43806/43806 [00:13<00:00, 3211.49it
Iterate posting for term 'parenchyma': 100%|█| 882/882 [00:00<00:00, 3391.98it/s
Iterate posting for term 'cells.water': 100%|███| 2/2 [00:00<00:00, 2544.32it/s]
Iterate posting for term 'biopolymer': 100%|██| 43/43 [00:00<00:00, 2746.01it/s]
Iterate posting for term 'end': 100%|█| 155224/155224 [01:53<00:00, 1370.14it/s]
-----------------------------------------------------
Current query: tracheids are part of _____. tracheids xylem vessel vessels cells element
s parenchyma cells.water biopolymer end narrow vascular safranin plants.tracheids plates
Iterate posting for term 'tracheids': 100%|█| 125/125 [00:00<00:00, 3196.94it/s]
Iterate posting for term 'part': 100%|█| 289955/289955 [02:41<00:00, 1796.72it/s
Iterate posting for term 'tracheids': 100%|█| 125/125 [00:00<00:00, 3015.40it/s]
Iterate posting for term 'xylem': 100%|███| 1394/1394 [00:00<00:00, 3344.52it/s]
Iterate posting for term 'vessel': 100%|█| 12963/12963 [00:03<00:00, 3268.05it/s
Iterate posting for term 'vessels': 100%|█| 30775/30775 [00:09<00:00, 3157.49it/
Iterate posting for term 'cells': 100%|█| 134812/134812 [01:28<00:00, 1529.18it/
Iterate posting for term 'elements': 100%|█| 43806/43806 [00:51<00:00, 854.76it/
Iterate posting for term 'parenchyma': 100%|█| 882/882 [00:00<00:00, 3218.86it/s
Iterate posting for term 'cells.water': 100%|███| 2/2 [00:00<00:00, 2781.37it/s]
Iterate posting for term 'biopolymer': 100%|██| 43/43 [00:00<00:00, 2777.21it/s]
Iterate posting for term 'end': 100%|█| 155224/155224 [01:30<00:00, 1715.55it/s]
Iterate posting for term 'narrow': 100%|█| 15241/15241 [00:05<00:00, 3044.82it/s
Iterate posting for term 'vascular': 100%|█| 8988/8988 [00:02<00:00, 3376.89it/s
Iterate posting for term 'safranin': 100%|████| 89/89 [00:00<00:00, 3449.23it/s]
Iterate posting for term 'plants.tracheids': 100%|█| 1/1 [00:00<00:00, 1726.76it
Iterate posting for term 'plates': 100%|█| 12083/12083 [00:03<00:00, 3444.62it/s
-----------------------------------------------------
Current query:  tracheids part
Iterate posting for term 'tracheids': 100%|█| 125/125 [00:00<00:00, 3417.43it/s]
Iterate posting for term 'part': 100%|█| 289955/289955 [03:04<00:00, 1575.00it/s
-----------------------------------------------------
Current query:  tracheids part
Iterate posting for term 'tracheids': 100%|█| 125/125 [00:00<00:00, 3118.85it/s]
Iterate posting for term 'part': 100%|█| 289955/289955 [03:52<00:00, 1248.37it/s
-----------------------------------------------------
Current query: rsa definition key rsa key private public keys ssl pairs algorithm cisco
router
Iterate posting for term 'rsa': 100%|███████| 597/597 [00:00<00:00, 2067.36it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [02:04<00:00, 1370.
Iterate posting for term 'key': 100%|████| 78308/78308 [01:44<00:00, 747.17it/s]
Iterate posting for term 'rsa': 100%|███████| 597/597 [00:00<00:00, 3217.46it/s]
Iterate posting for term 'key': 100%|███| 78308/78308 [00:24<00:00, 3139.59it/s]
Iterate posting for term 'private': 100%|█| 49766/49766 [01:19<00:00, 628.25it/s
Iterate posting for term 'public': 100%|█| 110511/110511 [01:24<00:00, 1300.90it
Iterate posting for term 'keys': 100%|██| 10506/10506 [00:03<00:00, 3114.69it/s]
Iterate posting for term 'ssl': 100%|█████| 1377/1377 [00:00<00:00, 2575.81it/s]
Iterate posting for term 'pairs': 100%|█| 12934/12934 [00:05<00:00, 2542.64it/s]
Iterate posting for term 'algorithm': 100%|█| 2762/2762 [00:01<00:00, 2261.00it/
Iterate posting for term 'cisco': 100%|███| 3257/3257 [00:01<00:00, 2943.98it/s]
Iterate posting for term 'router': 100%|██| 8861/8861 [00:02<00:00, 2989.13it/s]
```

```
-----------------------------------------------------
Current query: rsa definition key rsa key private public keys ssl pairs algorithm cisco
router warned pem command prompted generate
```

```
Iterate posting for term 'rsa': 100%|███████| 597/597 [00:00<00:00, 2538.59it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [02:55<00:00, 970.7
Iterate posting for term 'key': 100%|███| 78308/78308 [00:25<00:00, 3109.85it/s]
Iterate posting for term 'rsa': 100%|███████| 597/597 [00:00<00:00, 3528.87it/s]
Iterate posting for term 'key': 100%|████| 78308/78308 [01:23<00:00, 941.73it/s]
Iterate posting for term 'private': 100%|█| 49766/49766 [00:19<00:00, 2544.04it/
Iterate posting for term 'public': 100%|█| 110511/110511 [01:39<00:00, 1106.88it
Iterate posting for term 'keys': 100%|██| 10506/10506 [00:03<00:00, 3382.00it/s]
Iterate posting for term 'ssl': 100%|█████| 1377/1377 [00:00<00:00, 3436.35it/s]
Iterate posting for term 'pairs': 100%|█| 12934/12934 [00:03<00:00, 3394.56it/s]
Iterate posting for term 'algorithm': 100%|█| 2762/2762 [00:00<00:00, 3488.27it/
Iterate posting for term 'cisco': 100%|█████| 3257/3257 [00:41<00:00, 79.27it/s]
Iterate posting for term 'router': 100%|██| 8861/8861 [00:03<00:00, 2547.59it/s]
Iterate posting for term 'warned': 100%|██| 2256/2256 [00:01<00:00, 1597.22it/s]
Iterate posting for term 'pem': 100%|███████| 129/129 [00:00<00:00, 2795.67it/s]
Iterate posting for term 'command': 100%|█| 23842/23842 [00:07<00:00, 3218.56it/
Iterate posting for term 'prompted': 100%|█| 4050/4050 [00:01<00:00, 2599.29it/s
Iterate posting for term 'generate': 100%|█| 13216/13216 [00:05<00:00, 2204.86it
```

```
-----------------------------------------------------
Current query:  rsa key definition
```

```
Iterate posting for term 'rsa': 100%|███████| 597/597 [00:00<00:00, 3442.95it/s]
Iterate posting for term 'key': 100%|████| 78308/78308 [01:42<00:00, 763.76it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:49<00:00, 1552.
```

```
-----------------------------------------------------
Current query:  rsa key definition
```

```
Iterate posting for term 'rsa': 100%|███████| 597/597 [00:00<00:00, 3130.32it/s]
Iterate posting for term 'key': 100%|███| 78308/78308 [01:10<00:00, 1103.59it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [02:02<00:00, 1389.
```

```
-----------------------------------------------------
Current query: who formed the commonwealth of independent states soviet republics common
wealth socialist independent dissolution formed nations union 1991
```

```
Iterate posting for term 'who': 100%|█| 479472/479472 [06:04<00:00, 1315.70it/s]
Iterate posting for term 'formed': 100%|█| 49002/49002 [00:14<00:00, 3439.23it/s
Iterate posting for term 'commonwealth': 100%|█| 5186/5186 [00:01<00:00, 3488.59
Iterate posting for term 'independent': 100%|█| 33631/33631 [00:49<00:00, 681.45
Iterate posting for term 'states': 100%|█| 302247/302247 [03:13<00:00, 1558.53it
Iterate posting for term 'soviet': 100%|██| 7854/7854 [00:02<00:00, 3399.19it/s]
Iterate posting for term 'republics': 100%|█| 675/675 [00:00<00:00, 3330.52it/s]
Iterate posting for term 'commonwealth': 100%|█| 5186/5186 [00:01<00:00, 3420.77
Iterate posting for term 'socialist': 100%|█| 2065/2065 [00:00<00:00, 3463.37it/
Iterate posting for term 'independent': 100%|█| 33631/33631 [00:12<00:00, 2725.5
Iterate posting for term 'dissolution': 100%|█| 1720/1720 [00:00<00:00, 3364.94i
Iterate posting for term 'formed': 100%|█| 49002/49002 [01:28<00:00, 552.03it/s]
Iterate posting for term 'nations': 100%|█| 16471/16471 [00:05<00:00, 2746.22it/
Iterate posting for term 'union': 100%|█| 39192/39192 [00:18<00:00, 2137.55it/s]
Iterate posting for term '1991': 100%|██| 10815/10815 [00:03<00:00, 3432.44it/s]
```

```
-----------------------------------------------------
Current query: who formed the commonwealth of independent states soviet republics common
wealth socialist independent dissolution formed nations union 1991 organization 1945 pro
mote states alliance
```

```
Iterate posting for term 'who': 100%|█| 479472/479472 [05:47<00:00, 1381.05it/s]
Iterate posting for term 'formed': 100%|█| 49002/49002 [00:17<00:00, 2742.45it/s
Iterate posting for term 'commonwealth': 100%|█| 5186/5186 [00:01<00:00, 3427.50
Iterate posting for term 'independent': 100%|█| 33631/33631 [01:00<00:00, 551.68
Iterate posting for term 'states': 100%|█| 302247/302247 [03:09<00:00, 1590.80it
Iterate posting for term 'soviet': 100%|██| 7854/7854 [00:02<00:00, 3315.33it/s]
Iterate posting for term 'republics': 100%|█| 675/675 [00:00<00:00, 3537.66it/s]
Iterate posting for term 'commonwealth': 100%|█| 5186/5186 [00:01<00:00, 3361.36
Iterate posting for term 'socialist': 100%|█| 2065/2065 [00:00<00:00, 3455.05it/
Iterate posting for term 'independent': 100%|█| 33631/33631 [00:09<00:00, 3409.4
Iterate posting for term 'dissolution': 100%|█| 1720/1720 [00:00<00:00, 3511.86i
Iterate posting for term 'formed': 100%|█| 49002/49002 [00:14<00:00, 3285.18it/s
Iterate posting for term 'nations': 100%|█| 16471/16471 [00:39<00:00, 412.64it/s
Iterate posting for term 'union': 100%|█| 39192/39192 [00:13<00:00, 2810.48it/s]
Iterate posting for term '1991': 100%|██| 10815/10815 [00:03<00:00, 3350.47it/s]
Iterate posting for term 'organization': 100%|█| 50742/50742 [00:17<00:00, 2913.
Iterate posting for term '1945': 100%|████| 7674/7674 [00:02<00:00, 3074.63it/s]
Iterate posting for term 'promote': 100%|█| 16153/16153 [00:05<00:00, 3176.79it/
Iterate posting for term 'states': 100%|█| 302247/302247 [03:45<00:00, 1341.63it
Iterate posting for term 'alliance': 100%|█| 6138/6138 [00:01<00:00, 3130.71it/s
-----------------------------------------------------
Current query:  commonwealth independent formed states who
Iterate posting for term 'commonwealth': 100%|█| 5186/5186 [00:01<00:00, 3456.12
Iterate posting for term 'independent': 100%|█| 33631/33631 [00:09<00:00, 3521.5
Iterate posting for term 'formed': 100%|█| 49002/49002 [00:13<00:00, 3579.27it/s
Iterate posting for term 'states': 100%|█| 302247/302247 [03:28<00:00, 1452.76it
Iterate posting for term 'who': 100%|█| 479472/479472 [05:11<00:00, 1541.53it/s]
-----------------------------------------------------
Current query:  commonwealth independent formed
Iterate posting for term 'commonwealth': 100%|█| 5186/5186 [00:02<00:00, 2291.98
Iterate posting for term 'independent': 100%|█| 33631/33631 [00:10<00:00, 3318.5
Iterate posting for term 'formed': 100%|█| 49002/49002 [00:13<00:00, 3515.06it/s
-----------------------------------------------------
Current query: causes of left ventricular hypertrophy hypertrophy ventricular left ventr
icle causes enlargement heart workload chamber pumping
Iterate posting for term 'causes': 100%|█| 136109/136109 [01:42<00:00, 1333.08it
Iterate posting for term 'left': 100%|█| 115188/115188 [01:29<00:00, 1292.80it/s
Iterate posting for term 'ventricular': 100%|█| 3361/3361 [00:01<00:00, 3251.03i
Iterate posting for term 'hypertrophy': 100%|█| 1183/1183 [00:00<00:00, 3223.04i
Iterate posting for term 'hypertrophy': 100%|█| 1183/1183 [00:00<00:00, 3362.34i
Iterate posting for term 'ventricular': 100%|█| 3361/3361 [00:00<00:00, 3495.05i
Iterate posting for term 'left': 100%|█| 115188/115188 [01:36<00:00, 1189.23it/s
Iterate posting for term 'ventricle': 100%|█| 5174/5174 [00:01<00:00, 3306.46it/
Iterate posting for term 'causes': 100%|█| 136109/136109 [01:40<00:00, 1351.58it
Iterate posting for term 'enlargement': 100%|█| 3431/3431 [00:01<00:00, 3085.99i
Iterate posting for term 'heart': 100%|█| 122669/122669 [01:38<00:00, 1242.52it/
Iterate posting for term 'workload': 100%|█| 1241/1241 [00:00<00:00, 3360.50it/s
Iterate posting for term 'chamber': 100%|█| 9307/9307 [00:02<00:00, 3345.65it/s]
Iterate posting for term 'pumping': 100%|█| 3950/3950 [00:01<00:00, 3316.54it/s]
-----------------------------------------------------
Current query: causes of left ventricular hypertrophy hypertrophy ventricular left ventr
icle causes enlargement heart workload chamber pumping 2927 resuscitation.his enlargment
paramedics rvh
Iterate posting for term 'causes': 100%|█| 136109/136109 [01:39<00:00, 1362.70it
Iterate posting for term 'left': 100%|█| 115188/115188 [01:28<00:00, 1301.81it/s
Iterate posting for term 'ventricular': 100%|█| 3361/3361 [00:00<00:00, 3374.58i
Iterate posting for term 'hypertrophy': 100%|█| 1183/1183 [00:00<00:00, 3447.24i
Iterate posting for term 'hypertrophy': 100%|█| 1183/1183 [00:00<00:00, 3469.72i
Iterate posting for term 'ventricular': 100%|█| 3361/3361 [00:00<00:00, 3525.23i
Iterate posting for term 'left': 100%|█| 115188/115188 [01:34<00:00, 1219.60it/s
Iterate posting for term 'ventricle': 100%|█| 5174/5174 [00:01<00:00, 3499.70it/
Iterate posting for term 'causes': 100%|█| 136109/136109 [02:05<00:00, 1081.01it
Iterate posting for term 'enlargement': 100%|█| 3431/3431 [00:01<00:00, 3021.30i
Iterate posting for term 'heart': 100%|█| 122669/122669 [01:35<00:00, 1281.76it/
Iterate posting for term 'workload': 100%|█| 1241/1241 [00:00<00:00, 2417.97it/s
Iterate posting for term 'chamber': 100%|█| 9307/9307 [00:03<00:00, 2546.50it/s]
Iterate posting for term 'pumping': 100%|█| 3950/3950 [00:01<00:00, 3136.59it/s]
Iterate posting for term '2927': 100%|████████| 25/25 [00:00<00:00, 2984.08it/s]
Iterate posting for term 'resuscitation.his': 100%|█| 2/2 [00:00<00:00, 1865.79i
Iterate posting for term 'enlargment': 100%|██| 14/14 [00:00<00:00, 3007.44it/s]
Iterate posting for term 'paramedics': 100%|█| 799/799 [00:00<00:00, 3183.39it/s
Iterate posting for term 'rvh': 100%|█████████| 14/14 [00:00<00:00, 2789.96it/s]
---------------------------------------------------
Current query:  hypertrophy ventricular left causes
```

```
Iterate posting for term 'hypertrophy': 100%|█| 1183/1183 [00:00<00:00, 3323.72i
Iterate posting for term 'ventricular': 100%|█| 3361/3361 [00:00<00:00, 3432.52i
Iterate posting for term 'left': 100%|█| 115188/115188 [01:31<00:00, 1252.08it/s
Iterate posting for term 'causes': 100%|█| 136109/136109 [01:25<00:00, 1585.89it
---------------------------------------------------
Current query:  hypertrophy ventricular left
```

```
Iterate posting for term 'hypertrophy': 100%|█| 1183/1183 [00:00<00:00, 3317.37i
Iterate posting for term 'ventricular': 100%|█| 3361/3361 [00:00<00:00, 3361.80i
Iterate posting for term 'left': 100%|█| 115188/115188 [01:11<00:00, 1613.69it/s
---------------------------------------------------
Current query: lps laws definition lps acronyms abbreviations slang lipopolysaccharides
endotoxins meanings cell.acterial definitions definition
```

```
Iterate posting for term 'lps': 100%|███████| 457/457 [00:00<00:00, 3488.63it/s]
Iterate posting for term 'laws': 100%|██| 37074/37074 [00:11<00:00, 3107.98it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [02:12<00:00, 1293.
Iterate posting for term 'lps': 100%|███████| 457/457 [00:00<00:00, 2586.89it/s]
Iterate posting for term 'acronyms': 100%|█| 5766/5766 [00:01<00:00, 3158.41it/s
Iterate posting for term 'abbreviations': 100%|█| 6145/6145 [00:01<00:00, 3509.0
Iterate posting for term 'slang': 100%|███| 8723/8723 [00:02<00:00, 3479.22it/s]
Iterate posting for term 'lipopolysaccharides': 100%|█| 86/86 [00:00<00:00, 2855
Iterate posting for term 'endotoxins': 100%|█| 151/151 [00:00<00:00, 3607.79it/s
Iterate posting for term 'meanings': 100%|█| 19539/19539 [00:06<00:00, 3125.16it
Iterate posting for term 'cell.acterial': 100%|█| 4/4 [00:00<00:00, 2190.52it/s]
Iterate posting for term 'definitions': 100%|█| 23262/23262 [00:06<00:00, 3356.9
Iterate posting for term 'definition': 100%|█| 170760/170760 [02:01<00:00, 1408.
---------------------------------------------------
Current query: lps laws definition lps acronyms abbreviations slang lipopolysaccharides
endotoxins meanings cell.acterial definitions definition lipopolysaccharide stand exogen
ous lysis acronymsandslang.com
```

```
Iterate posting for term 'lps': 100%|███████| 457/457 [00:00<00:00, 2141.70it/s]
Iterate posting for term 'laws': 100%|██| 37074/37074 [00:11<00:00, 3123.23it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:28<00:00, 1925.
Iterate posting for term 'lps': 100%|███████| 457/457 [00:00<00:00, 3381.38it/s]
Iterate posting for term 'acronyms': 100%|█| 5766/5766 [00:01<00:00, 3380.63it/s
Iterate posting for term 'abbreviations': 100%|█| 6145/6145 [00:29<00:00, 207.32
Iterate posting for term 'slang': 100%|███| 8723/8723 [00:03<00:00, 2678.39it/s]
Iterate posting for term 'lipopolysaccharides': 100%|█| 86/86 [00:00<00:00, 3268
Iterate posting for term 'endotoxins': 100%|█| 151/151 [00:00<00:00, 2855.26it/s
Iterate posting for term 'meanings': 100%|█| 19539/19539 [00:08<00:00, 2186.02it
Iterate posting for term 'cell.acterial': 100%|█| 4/4 [00:00<00:00, 2158.67it/s]
Iterate posting for term 'definitions': 100%|█| 23262/23262 [00:06<00:00, 3405.8
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:30<00:00, 1893.
Iterate posting for term 'lipopolysaccharide': 100%|█| 108/108 [00:00<00:00, 337
Iterate posting for term 'stand': 100%|██| 33012/33012 [00:51<00:00, 644.54it/s]
Iterate posting for term 'exogenous': 100%|█| 616/616 [00:00<00:00, 1704.43it/s]
Iterate posting for term 'lysis': 100%|█████| 686/686 [00:00<00:00, 1571.09it/s]
Iterate posting for term 'acronymsandslang.com': 100%|█| 206/206 [00:00<00:00, 2
---------------------------------------------------
Current query:  lps laws definition
```

```
Iterate posting for term 'lps': 100%|███████| 457/457 [00:00<00:00, 3146.45it/s]
Iterate posting for term 'laws': 100%|██| 37074/37074 [00:11<00:00, 3222.97it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:46<00:00, 1596.
---------------------------------------------------
Current query:  lps laws definition
```

```
Iterate posting for term 'lps': 100%|███████| 457/457 [00:00<00:00, 2865.43it/s]
Iterate posting for term 'laws': 100%|███| 37074/37074 [01:14<00:00, 497.79it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:42<00:00, 1670.
```

```
-----------------------------------------------------
Current query: what are the three percenters? percenters oath three keepers flag support
ively iii percenter keepersâ war
```

```
Iterate posting for term 'what': 100%|█| 542299/542299 [04:54<00:00, 1842.04it/s
Iterate posting for term 'three': 100%|█| 260225/260225 [02:21<00:00, 1842.10it/
Iterate posting for term 'percenters': 100%|██| 37/37 [00:00<00:00, 3002.77it/s]
Iterate posting for term 'percenters': 100%|██| 37/37 [00:00<00:00, 3203.74it/s]
Iterate posting for term 'oath': 100%|████| 2915/2915 [00:00<00:00, 3245.77it/s]
Iterate posting for term 'three': 100%|█| 260225/260225 [02:39<00:00, 1634.83it/
Iterate posting for term 'keepers': 100%|███| 685/685 [00:00<00:00, 3281.86it/s]
Iterate posting for term 'flag': 100%|███| 13507/13507 [00:27<00:00, 485.33it/s]
Iterate posting for term 'supportively': 100%|██| 6/6 [00:00<00:00, 1877.35it/s]
Iterate posting for term 'iii': 100%|███| 15461/15461 [00:04<00:00, 3099.61it/s]
Iterate posting for term 'percenter': 100%|███| 26/26 [00:00<00:00, 2851.55it/s]
Iterate posting for term 'keepersâ': 100%|████| 19/19 [00:00<00:00, 2619.97it/s]
Iterate posting for term 'war': 100%|███| 80223/80223 [00:27<00:00, 2898.38it/s]
```

```
-----------------------------------------------------
Current query: what are the three percenters? percenters oath three keepers flag support
ively iii percenter keepersâ war american extremist combatants movement resisted
```

```
Iterate posting for term 'what': 100%|█| 542299/542299 [04:48<00:00, 1881.06it/s
Iterate posting for term 'three': 100%|█| 260225/260225 [03:40<00:00, 1181.54it/
Iterate posting for term 'percenters': 100%|██| 37/37 [00:00<00:00, 1878.37it/s]
Iterate posting for term 'percenters': 100%|██| 37/37 [00:00<00:00, 3176.53it/s]
Iterate posting for term 'oath': 100%|████| 2915/2915 [00:01<00:00, 2347.45it/s]
Iterate posting for term 'three': 100%|█| 260225/260225 [03:24<00:00, 1270.76it/
Iterate posting for term 'keepers': 100%|███| 685/685 [00:00<00:00, 2848.91it/s]
Iterate posting for term 'flag': 100%|██| 13507/13507 [00:04<00:00, 3299.88it/s]
Iterate posting for term 'supportively': 100%|██| 6/6 [00:00<00:00, 2867.90it/s]
Iterate posting for term 'iii': 100%|███| 15461/15461 [00:04<00:00, 3344.37it/s]
Iterate posting for term 'percenter': 100%|███| 26/26 [00:00<00:00, 2601.06it/s]
Iterate posting for term 'keepersâ': 100%|████| 19/19 [00:00<00:00, 3173.58it/s]
Iterate posting for term 'war': 100%|███| 80223/80223 [01:05<00:00, 1216.41it/s]
Iterate posting for term 'american': 100%|█| 195413/195413 [01:57<00:00, 1657.24
Iterate posting for term 'extremist': 100%|█| 227/227 [00:00<00:00, 3436.37it/s]
Iterate posting for term 'combatants': 100%|█| 264/264 [00:00<00:00, 3448.10it/s
Iterate posting for term 'movement': 100%|█| 52317/52317 [00:57<00:00, 915.96it/
Iterate posting for term 'resisted': 100%|██| 546/546 [00:00<00:00, 3168.58it/s]
```

```
-----------------------------------------------------
Current query:  percenters three what
```

```
Iterate posting for term 'percenters': 100%|██| 37/37 [00:00<00:00, 2450.21it/s]
Iterate posting for term 'three': 100%|█| 260225/260225 [02:57<00:00, 1464.04it/
Iterate posting for term 'what': 100%|█| 542299/542299 [05:48<00:00, 1554.10it/s
```

```
-----------------------------------------------------
Current query:  percenters three what
```

```
Iterate posting for term 'percenters': 100%|██| 37/37 [00:00<00:00, 1904.86it/s]
Iterate posting for term 'three': 100%|█| 260225/260225 [03:11<00:00, 1356.44it/
Iterate posting for term 'what': 100%|█| 542299/542299 [06:57<00:00, 1298.94it/s
```

```
-----------------------------------------------------
Current query: causes of military suicide suicide military suicides veterans causes prev
entable.een public.n 4,486 2012 2,676
```

```
Iterate posting for term 'causes': 100%|█| 136109/136109 [01:55<00:00, 1181.45it
Iterate posting for term 'military': 100%|█| 43218/43218 [00:16<00:00, 2571.56it
Iterate posting for term 'suicide': 100%|█| 6359/6359 [00:01<00:00, 3363.11it/s]
Iterate posting for term 'suicide': 100%|█| 6359/6359 [00:01<00:00, 3456.75it/s]
Iterate posting for term 'military': 100%|█| 43218/43218 [01:00<00:00, 712.12it/
Iterate posting for term 'suicides': 100%|██| 508/508 [00:00<00:00, 1100.43it/s]
Iterate posting for term 'veterans': 100%|█| 9244/9244 [00:03<00:00, 2540.78it/s
Iterate posting for term 'causes': 100%|█| 136109/136109 [01:28<00:00, 1530.30it
Iterate posting for term 'preventable.een': 100%|█| 1/1 [00:00<00:00, 1240.55it/
Iterate posting for term 'public.n': 100%|████| 16/16 [00:00<00:00, 1840.97it/s]
Iterate posting for term '4,486': 100%|███████| 15/15 [00:00<00:00, 2095.55it/s]
Iterate posting for term '2012': 100%|██| 68099/68099 [00:21<00:00, 3175.89it/s]
Iterate posting for term '2,676': 100%|███████| 19/19 [00:00<00:00, 3161.87it/s]
```

```
-----------------------------------------------------
Current query: causes of military suicide suicide military suicides veterans causes prev
entable.een public.n 4,486 2012 2,676 death bullying genders homicide teens
Iterate posting for term 'causes': 100%|█| 136109/136109 [01:31<00:00, 1487.65it
Iterate posting for term 'military': 100%|█| 43218/43218 [01:07<00:00, 640.77it/
Iterate posting for term 'suicide': 100%|█| 6359/6359 [00:01<00:00, 3244.10it/s]
Iterate posting for term 'suicide': 100%|█| 6359/6359 [00:01<00:00, 3496.90it/s]
Iterate posting for term 'military': 100%|█| 43218/43218 [00:12<00:00, 3433.04it
Iterate posting for term 'suicides': 100%|██| 508/508 [00:00<00:00, 3256.01it/s]
Iterate posting for term 'veterans': 100%|█| 9244/9244 [00:02<00:00, 3311.31it/s
Iterate posting for term 'causes': 100%|█| 136109/136109 [02:23<00:00, 948.17it/
Iterate posting for term 'preventable.een': 100%|█| 1/1 [00:00<00:00, 1633.30it/
Iterate posting for term 'public.n': 100%|████| 16/16 [00:00<00:00, 2082.77it/s]
Iterate posting for term '4,486': 100%|██████| 15/15 [00:00<00:00, 1849.12it/s]
Iterate posting for term '2012': 100%|██| 68099/68099 [00:24<00:00, 2769.22it/s]
Iterate posting for term '2,676': 100%|███████| 19/19 [00:00<00:00, 2331.87it/s]
Iterate posting for term 'death': 100%|█| 72215/72215 [01:09<00:00, 1043.62it/s]
Iterate posting for term 'bullying': 100%|█| 1640/1640 [00:00<00:00, 3378.73it/s
Iterate posting for term 'genders': 100%|█| 1016/1016 [00:00<00:00, 3145.52it/s]
Iterate posting for term 'homicide': 100%|█| 1263/1263 [00:00<00:00, 3247.25it/s
Iterate posting for term 'teens': 100%|███| 6567/6567 [00:02<00:00, 3255.94it/s]
-----------------------------------------------------
Current query:  suicide military causes
Iterate posting for term 'suicide': 100%|█| 6359/6359 [00:01<00:00, 3451.78it/s]
Iterate posting for term 'military': 100%|█| 43218/43218 [00:16<00:00, 2616.39it
Iterate posting for term 'causes': 100%|█| 136109/136109 [01:41<00:00, 1339.70it
-----------------------------------------------------
Current query:  suicide military causes
Iterate posting for term 'suicide': 100%|█| 6359/6359 [00:01<00:00, 3451.28it/s]
Iterate posting for term 'military': 100%|█| 43218/43218 [00:50<00:00, 848.20it/
Iterate posting for term 'causes': 100%|█| 136109/136109 [01:36<00:00, 1406.04it
-----------------------------------------------------
Current query: what is theraderm used for theraderm beckman products skin care lanolin a
ging brand james founded
Iterate posting for term 'what': 100%|█| 542299/542299 [05:30<00:00, 1643.18it/s
Iterate posting for term 'theraderm': 100%|█████| 5/5 [00:00<00:00, 2735.30it/s]
Iterate posting for term 'used': 100%|█| 589843/589843 [06:05<00:00, 1612.97it/s
Iterate posting for term 'theraderm': 100%|█████| 5/5 [00:00<00:00, 2535.55it/s]
Iterate posting for term 'beckman': 100%|█████| 77/77 [00:00<00:00, 2152.82it/s]
Iterate posting for term 'products': 100%|█| 114198/114198 [00:36<00:00, 3138.31
Iterate posting for term 'skin': 100%|█| 135058/135058 [01:50<00:00, 1226.22it/s
Iterate posting for term 'care': 100%|█| 140428/140428 [01:42<00:00, 1370.52it/s
Iterate posting for term 'lanolin': 100%|███| 302/302 [00:00<00:00, 2388.24it/s]
Iterate posting for term 'aging': 100%|█| 10905/10905 [00:03<00:00, 3078.49it/s]
Iterate posting for term 'brand': 100%|██| 44433/44433 [01:10<00:00, 629.12it/s]
Iterate posting for term 'james': 100%|█| 32043/32043 [00:10<00:00, 3204.12it/s]
Iterate posting for term 'founded': 100%|█| 33188/33188 [00:10<00:00, 3310.59it/
-----------------------------------------------------
Current query: what is theraderm used for theraderm beckman products skin care lanolin a
ging brand james founded reversion anti springdale sheepâ moisturizers
Iterate posting for term 'what': 100%|█| 542299/542299 [06:02<00:00, 1497.66it/s
Iterate posting for term 'theraderm': 100%|█████| 5/5 [00:00<00:00, 2605.48it/s]
Iterate posting for term 'used': 100%|█| 589843/589843 [05:33<00:00, 1769.35it/s
Iterate posting for term 'theraderm': 100%|█████| 5/5 [00:00<00:00, 1996.72it/s]
Iterate posting for term 'beckman': 100%|█████| 77/77 [00:00<00:00, 2142.86it/s]
Iterate posting for term 'products': 100%|█| 114198/114198 [01:26<00:00, 1320.28
Iterate posting for term 'skin': 100%|█| 135058/135058 [01:33<00:00, 1441.61it/s
Iterate posting for term 'care': 100%|█| 140428/140428 [01:35<00:00, 1471.24it/s
Iterate posting for term 'lanolin': 100%|███| 302/302 [00:00<00:00, 3322.87it/s]
Iterate posting for term 'aging': 100%|█| 10905/10905 [00:03<00:00, 3387.35it/s]
Iterate posting for term 'brand': 100%|██| 44433/44433 [01:04<00:00, 689.79it/s]
Iterate posting for term 'james': 100%|█| 32043/32043 [00:09<00:00, 3316.21it/s]
Iterate posting for term 'founded': 100%|█| 33188/33188 [00:09<00:00, 3396.11it/
Iterate posting for term 'reversion': 100%|█| 156/156 [00:00<00:00, 3371.68it/s]
Iterate posting for term 'anti': 100%|███| 30776/30776 [01:02<00:00, 490.88it/s]
Iterate posting for term 'springdale': 100%|█| 229/229 [00:00<00:00, 3007.22it/s
Iterate posting for term 'sheepâ': 100%|████| 186/186 [00:00<00:00, 2822.45it/s]
Iterate posting for term 'moisturizers': 100%|█| 559/559 [00:00<00:00, 2806.31it
-----------------------------------------------------
Current query:  theraderm what used
Iterate posting for term 'theraderm': 100%|█████| 5/5 [00:00<00:00, 2601.28it/s]
Iterate posting for term 'what': 100%|█| 542299/542299 [05:42<00:00, 1581.81it/s
Iterate posting for term 'used': 100%|█| 589843/589843 [06:01<00:00, 1631.38it/s
-----------------------------------------------------
Current query:  theraderm what used
Iterate posting for term 'theraderm': 100%|█████| 5/5 [00:00<00:00, 2682.47it/s]
Iterate posting for term 'what': 100%|█| 542299/542299 [05:13<00:00, 1730.00it/s
Iterate posting for term 'used': 100%|█| 589843/589843 [05:27<00:00, 1799.35it/s
-----------------------------------------------------
Current query: what is famvir prescribed for famvir valtrex herpes famciclovir acyclovir
valacyclovir sores treat form.severe acyclovir.here
Iterate posting for term 'what': 100%|█| 542299/542299 [05:04<00:00, 1778.19it/s
Iterate posting for term 'famvir': 100%|██████| 59/59 [00:00<00:00, 3067.49it/s]
Iterate posting for term 'prescribed': 100%|█| 17263/17263 [00:05<00:00, 3339.86
Iterate posting for term 'famvir': 100%|██████| 59/59 [00:00<00:00, 2718.01it/s]
Iterate posting for term 'valtrex': 100%|███| 307/307 [00:00<00:00, 3460.60it/s]
Iterate posting for term 'herpes': 100%|██| 5940/5940 [00:01<00:00, 3499.27it/s]
Iterate posting for term 'famciclovir': 100%|█| 83/83 [00:00<00:00, 2966.62it/s]
Iterate posting for term 'acyclovir': 100%|█| 421/421 [00:00<00:00, 3523.80it/s]
Iterate posting for term 'valacyclovir': 100%|█| 269/269 [00:00<00:00, 3329.13it
Iterate posting for term 'sores': 100%|███| 7094/7094 [00:01<00:00, 3580.10it/s]
Iterate posting for term 'treat': 100%|█| 62913/62913 [00:22<00:00, 2810.00it/s]
Iterate posting for term 'form.severe': 100%|███| 1/1 [00:00<00:00, 2016.49it/s]
Iterate posting for term 'acyclovir.here': 100%|█| 3/3 [00:00<00:00, 1965.77it/s
-----------------------------------------------------
Current query: what is famvir prescribed for famvir valtrex herpes famciclovir acyclovir
valacyclovir sores treat form.severe acyclovir.here prescription zovirax cold breastfeed
ing topicals
Iterate posting for term 'what': 100%|█| 542299/542299 [05:02<00:00, 1792.23it/s
Iterate posting for term 'famvir': 100%|██████| 59/59 [00:00<00:00, 2745.69it/s]
Iterate posting for term 'prescribed': 100%|█| 17263/17263 [00:44<00:00, 386.62i
Iterate posting for term 'famvir': 100%|███████| 59/59 [00:00<00:00, 326.44it/s]
Iterate posting for term 'valtrex': 100%|███| 307/307 [00:00<00:00, 1089.06it/s]
Iterate posting for term 'herpes': 100%|██| 5940/5940 [00:04<00:00, 1221.70it/s]
Iterate posting for term 'famciclovir': 100%|█| 83/83 [00:00<00:00, 3018.06it/s]
Iterate posting for term 'acyclovir': 100%|█| 421/421 [00:00<00:00, 3315.82it/s]
Iterate posting for term 'valacyclovir': 100%|█| 269/269 [00:00<00:00, 3431.03it
Iterate posting for term 'sores': 100%|███| 7094/7094 [00:02<00:00, 2801.72it/s]
Iterate posting for term 'treat': 100%|█| 62913/62913 [00:20<00:00, 3067.35it/s]
Iterate posting for term 'form.severe': 100%|███| 1/1 [00:00<00:00, 2251.37it/s]
Iterate posting for term 'acyclovir.here': 100%|█| 3/3 [00:00<00:00, 2843.60it/s
Iterate posting for term 'prescription': 100%|█| 23735/23735 [00:19<00:00, 1213.
Iterate posting for term 'zovirax': 100%|███| 155/155 [00:00<00:00, 3288.50it/s]
Iterate posting for term 'cold': 100%|███| 71620/71620 [01:17<00:00, 925.98it/s]
Iterate posting for term 'breastfeeding': 100%|█| 3694/3694 [00:01<00:00, 2141.5
Iterate posting for term 'topicals': 100%|████| 52/52 [00:00<00:00, 1467.68it/s]
-----------------------------------------------------
Current query:  famvir prescribed what
Iterate posting for term 'famvir': 100%|██████| 59/59 [00:00<00:00, 2577.19it/s]
Iterate posting for term 'prescribed': 100%|█| 17263/17263 [00:05<00:00, 3245.82
Iterate posting for term 'what': 100%|█| 542299/542299 [05:48<00:00, 1557.48it/s
-----------------------------------------------------
Current query:  famvir prescribed what
Iterate posting for term 'famvir': 100%|██████| 59/59 [00:00<00:00, 2907.68it/s]
Iterate posting for term 'prescribed': 100%|█| 17263/17263 [00:05<00:00, 3343.19
Iterate posting for term 'what': 100%|█| 542299/542299 [06:03<00:00, 1493.35it/s
-----------------------------------------------------
Current query: anthropological definition of environment anthropological study anthropol
```

```
ogy cultural pometry anthropometry anthropolgies definition anthro classification.efinit
ion
```

```
Iterate posting for term 'anthropological': 100%|█| 254/254 [00:00<00:00, 2068.8
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:30<00:00, 1876.
Iterate posting for term 'environment': 100%|█| 50134/50134 [01:27<00:00, 575.53
Iterate posting for term 'anthropological': 100%|█| 254/254 [00:00<00:00, 1451.2
Iterate posting for term 'study': 100%|█| 86610/86610 [00:33<00:00, 2572.89it/s]
Iterate posting for term 'anthropology': 100%|█| 1703/1703 [00:00<00:00, 3421.01
Iterate posting for term 'cultural': 100%|█| 18454/18454 [00:05<00:00, 3347.33it
Iterate posting for term 'pometry': 100%|███████| 1/1 [00:00<00:00, 2232.20it/s]
Iterate posting for term 'anthropometry': 100%|█| 42/42 [00:00<00:00, 2855.95it/
Iterate posting for term 'anthropolgies': 100%|█| 1/1 [00:00<00:00, 2475.98it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [03:06<00:00, 915.2
Iterate posting for term 'anthro': 100%|██████| 39/39 [00:00<00:00, 2416.97it/s]
Iterate posting for term 'classification.efinition': 100%|█| 1/1 [00:00<00:00, 2
```

```
-----------------------------------------------------
Current query: anthropological definition of environment anthropological study anthropol
ogy cultural pometry anthropometry anthropolgies definition anthro classification.efinit
ion present.â steward wo humans hirst
```

```
Iterate posting for term 'anthropological': 100%|█| 254/254 [00:00<00:00, 3302.9
Iterate posting for term 'definition': 100%|█| 170760/170760 [02:11<00:00, 1294.
Iterate posting for term 'environment': 100%|█| 50134/50134 [01:01<00:00, 809.74
Iterate posting for term 'anthropological': 100%|█| 254/254 [00:00<00:00, 3258.9
Iterate posting for term 'study': 100%|█| 86610/86610 [00:29<00:00, 2956.74it/s]
Iterate posting for term 'anthropology': 100%|█| 1703/1703 [00:00<00:00, 3380.10
Iterate posting for term 'cultural': 100%|█| 18454/18454 [01:14<00:00, 248.08it/
Iterate posting for term 'pometry': 100%|████████| 1/1 [00:00<00:00, 625.83it/s]
Iterate posting for term 'anthropometry': 100%|█| 42/42 [00:00<00:00, 2303.27it/
Iterate posting for term 'anthropolgies': 100%|█| 1/1 [00:00<00:00, 2432.89it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [01:53<00:00, 1503.
Iterate posting for term 'anthro': 100%|██████| 39/39 [00:00<00:00, 2611.81it/s]
Iterate posting for term 'classification.efinition': 100%|█| 1/1 [00:00<00:00, 2
Iterate posting for term 'present.â': 100%|███| 18/18 [00:00<00:00, 2643.65it/s]
Iterate posting for term 'steward': 100%|███| 589/589 [00:00<00:00, 3008.22it/s]
Iterate posting for term 'wo': 100%|████████| 651/651 [00:00<00:00, 3418.51it/s]
Iterate posting for term 'humans': 100%|█| 40601/40601 [00:14<00:00, 2843.42it/s
Iterate posting for term 'hirst': 100%|█████| 129/129 [00:00<00:00, 3390.45it/s]
```

```
-----------------------------------------------------
Current query:  anthropological environment definition
```

```
Iterate posting for term 'anthropological': 100%|█| 254/254 [00:00<00:00, 3544.8
Iterate posting for term 'environment': 100%|█| 50134/50134 [01:31<00:00, 545.67
Iterate posting for term 'definition': 100%|█| 170760/170760 [02:10<00:00, 1303.
```

```
-----------------------------------------------------
Current query:  anthropological environment definition
```

```
Iterate posting for term 'anthropological': 100%|█| 254/254 [00:00<00:00, 3309.1
Iterate posting for term 'environment': 100%|█| 50134/50134 [01:14<00:00, 672.64
Iterate posting for term 'definition': 100%|█| 170760/170760 [02:01<00:00, 1406.
```

```
-----------------------------------------------------
Current query: axon terminals or synaptic knob definition axon synaptic knob terminal sy
napse terminals potentials axons neuron end
```

```
Iterate posting for term 'axon': 100%|████| 2677/2677 [00:00<00:00, 3305.25it/s]
Iterate posting for term 'terminals': 100%|█| 3680/3680 [00:01<00:00, 3256.75it/
Iterate posting for term 'synaptic': 100%|█| 1292/1292 [00:00<00:00, 3438.92it/s
Iterate posting for term 'knob': 100%|████| 2023/2023 [00:00<00:00, 3373.83it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [03:13<00:00, 880.5
Iterate posting for term 'axon': 100%|████| 2677/2677 [00:00<00:00, 2807.55it/s]
Iterate posting for term 'synaptic': 100%|█| 1292/1292 [00:00<00:00, 3111.37it/s
Iterate posting for term 'knob': 100%|████| 2023/2023 [00:00<00:00, 2985.21it/s]
Iterate posting for term 'terminal': 100%|█| 13739/13739 [00:04<00:00, 2852.06it
Iterate posting for term 'synapse': 100%|█| 1472/1472 [00:00<00:00, 3308.42it/s]
Iterate posting for term 'terminals': 100%|█| 3680/3680 [00:01<00:00, 3263.59it/
Iterate posting for term 'potentials': 100%|█| 1213/1213 [00:00<00:00, 3167.62it
Iterate posting for term 'axons': 100%|███| 2219/2219 [00:00<00:00, 3316.96it/s]
Iterate posting for term 'neuron': 100%|██| 5190/5190 [00:02<00:00, 2584.13it/s]
Iterate posting for term 'end': 100%|█| 155224/155224 [01:54<00:00, 1352.74it/s]
-----------------------------------------------------
Current query: axon terminals or synaptic knob definition axon synaptic knob terminal sy
napse terminals potentials axons neuron end cell button neurotransmitters neurotransmitt
er presynaptic
Iterate posting for term 'axon': 100%|████| 2677/2677 [00:00<00:00, 3402.18it/s]
Iterate posting for term 'terminals': 100%|█| 3680/3680 [00:01<00:00, 3458.13it/
Iterate posting for term 'synaptic': 100%|█| 1292/1292 [00:00<00:00, 3630.73it/s
Iterate posting for term 'knob': 100%|████| 2023/2023 [00:00<00:00, 3587.00it/s]
Iterate posting for term 'definition': 100%|█| 170760/170760 [02:57<00:00, 961.0
Iterate posting for term 'axon': 100%|████| 2677/2677 [00:00<00:00, 2890.53it/s]
Iterate posting for term 'synaptic': 100%|█| 1292/1292 [00:00<00:00, 3202.25it/s
Iterate posting for term 'knob': 100%|████| 2023/2023 [00:00<00:00, 2452.29it/s]
Iterate posting for term 'terminal': 100%|█| 13739/13739 [00:04<00:00, 3064.29it
Iterate posting for term 'synapse': 100%|█| 1472/1472 [00:00<00:00, 3350.27it/s]
Iterate posting for term 'terminals': 100%|█| 3680/3680 [00:01<00:00, 3305.51it/
Iterate posting for term 'potentials': 100%|█| 1213/1213 [00:00<00:00, 3288.48it
Iterate posting for term 'axons': 100%|███| 2219/2219 [00:00<00:00, 3349.77it/s]
Iterate posting for term 'neuron': 100%|██| 5190/5190 [00:01<00:00, 3375.07it/s]
Iterate posting for term 'end': 100%|█| 155224/155224 [01:56<00:00, 1336.77it/s]
Iterate posting for term 'cell': 100%|█| 104234/104234 [01:36<00:00, 1082.76it/s
Iterate posting for term 'button': 100%|█| 48631/48631 [00:14<00:00, 3305.70it/s
Iterate posting for term 'neurotransmitters': 100%|█| 2631/2631 [00:00<00:00, 32
Iterate posting for term 'neurotransmitter': 100%|█| 3360/3360 [00:52<00:00, 63.
Iterate posting for term 'presynaptic': 100%|█| 517/517 [00:00<00:00, 1309.45it/
-----------------------------------------------------
Current query:  synaptic knob axon terminals definition
Iterate posting for term 'synaptic': 100%|█| 1292/1292 [00:00<00:00, 2529.49it/s
Iterate posting for term 'knob': 100%|████| 2023/2023 [00:00<00:00, 2312.97it/s]
Iterate posting for term 'axon': 100%|████| 2677/2677 [00:00<00:00, 3342.48it/s]
Iterate posting for term 'terminals': 100%|█| 3680/3680 [00:01<00:00, 3251.96it/
Iterate posting for term 'definition': 100%|█| 170760/170760 [02:20<00:00, 1216.
-----------------------------------------------------
Current query:  synaptic knob axon
Iterate posting for term 'synaptic': 100%|█| 1292/1292 [00:00<00:00, 3277.61it/s
Iterate posting for term 'knob': 100%|████| 2023/2023 [00:00<00:00, 3195.18it/s]
Iterate posting for term 'axon': 100%|████| 2677/2677 [00:00<00:00, 3466.97it/s]
-----------------------------------------------------
Current query: is cdg airport in main paris cdg paris airport gaulle charles de train ta
xi beauvais stations.sual
Iterate posting for term 'cdg': 100%|███████| 190/190 [00:00<00:00, 3386.20it/s]
Iterate posting for term 'airport': 100%|█| 36250/36250 [00:11<00:00, 3105.85it/
Iterate posting for term 'main': 100%|█| 127477/127477 [01:42<00:00, 1248.31it/s
Iterate posting for term 'paris': 100%|██| 13242/13242 [00:46<00:00, 285.54it/s]
Iterate posting for term 'cdg': 100%|███████| 190/190 [00:00<00:00, 2925.86it/s]
Iterate posting for term 'paris': 100%|█| 13242/13242 [00:04<00:00, 3272.40it/s]
Iterate posting for term 'airport': 100%|█| 36250/36250 [00:10<00:00, 3296.82it/
Iterate posting for term 'gaulle': 100%|████| 341/341 [00:00<00:00, 3493.67it/s]
Iterate posting for term 'charles': 100%|█| 17251/17251 [00:05<00:00, 3162.48it/
Iterate posting for term 'de': 100%|████| 40811/40811 [00:13<00:00, 3105.18it/s]
Iterate posting for term 'train': 100%|██| 18994/18994 [01:06<00:00, 285.76it/s]
Iterate posting for term 'taxi': 100%|████| 4360/4360 [00:03<00:00, 1450.48it/s]
Iterate posting for term 'beauvais': 100%|████| 40/40 [00:00<00:00, 2530.35it/s]
Iterate posting for term 'stations.sual': 100%|█| 1/1 [00:00<00:00, 2192.53it/s]
-----------------------------------------------------
Current query: is cdg airport in main paris cdg paris airport gaulle charles de train ta
xi beauvais stations.sual eiffel orly paris.com toureiffel maurois
Iterate posting for term 'cdg': 100%|███████| 190/190 [00:00<00:00, 3443.78it/s]
Iterate posting for term 'airport': 100%|█| 36250/36250 [00:13<00:00, 2637.13it/
Iterate posting for term 'main': 100%|█| 127477/127477 [02:05<00:00, 1014.74it/s
Iterate posting for term 'paris': 100%|█| 13242/13242 [00:03<00:00, 3328.66it/s]
Iterate posting for term 'cdg': 100%|███████| 190/190 [00:00<00:00, 3364.33it/s]
Iterate posting for term 'paris': 100%|█| 13242/13242 [00:03<00:00, 3463.58it/s]
Iterate posting for term 'airport': 100%|█| 36250/36250 [00:10<00:00, 3384.16it/
Iterate posting for term 'gaulle': 100%|████| 341/341 [00:00<00:00, 3744.62it/s]
Iterate posting for term 'charles': 100%|█| 17251/17251 [00:50<00:00, 343.64it/s
Iterate posting for term 'de': 100%|████| 40811/40811 [00:13<00:00, 3009.32it/s]
Iterate posting for term 'train': 100%|█| 18994/18994 [00:06<00:00, 3084.46it/s]
Iterate posting for term 'taxi': 100%|████| 4360/4360 [00:01<00:00, 3386.09it/s]
Iterate posting for term 'beauvais': 100%|████| 40/40 [00:00<00:00, 3266.15it/s]
Iterate posting for term 'stations.sual': 100%|█| 1/1 [00:00<00:00, 1949.03it/s]
Iterate posting for term 'eiffel': 100%|████| 838/838 [00:00<00:00, 3374.40it/s]
Iterate posting for term 'orly': 100%|██████| 144/144 [00:00<00:00, 3346.28it/s]
Iterate posting for term 'paris.com': 100%|█████| 4/4 [00:00<00:00, 2528.59it/s]
Iterate posting for term 'toureiffel': 100%|████| 1/1 [00:00<00:00, 1711.26it/s]
Iterate posting for term 'maurois': 100%|███████| 2/2 [00:00<00:00, 2545.09it/s]
```

```
-----------------------------------------------------
Current query:  cdg paris airport main
```

```
Iterate posting for term 'cdg': 100%|███████| 190/190 [00:00<00:00, 3286.39it/s]
Iterate posting for term 'paris': 100%|█| 13242/13242 [00:03<00:00, 3365.31it/s]
Iterate posting for term 'airport': 100%|█| 36250/36250 [00:11<00:00, 3137.92it/
Iterate posting for term 'main': 100%|█| 127477/127477 [01:58<00:00, 1077.73it/s
```

```
-----------------------------------------------------
Current query:  cdg paris airport
```

```
Iterate posting for term 'cdg': 100%|███████| 190/190 [00:00<00:00, 3503.86it/s]
Iterate posting for term 'paris': 100%|██| 13242/13242 [00:49<00:00, 269.67it/s]
Iterate posting for term 'airport': 100%|█| 36250/36250 [00:10<00:00, 3394.53it/
```

```
-----------------------------------------------------
Current query: example of monotonic function monotonic function domain x property.econom
ic consider examplesuppose example montonic nonincreasing
```

```
Iterate posting for term 'example': 100%|█| 234474/234474 [03:00<00:00, 1300.87i
Iterate posting for term 'monotonic': 100%|███| 68/68 [00:00<00:00, 2789.42it/s]
Iterate posting for term 'function': 100%|█| 89643/89643 [01:07<00:00, 1335.75it
Iterate posting for term 'monotonic': 100%|███| 68/68 [00:00<00:00, 2944.62it/s]
Iterate posting for term 'function': 100%|█| 89643/89643 [00:36<00:00, 2481.26it
Iterate posting for term 'domain': 100%|█| 13475/13475 [01:20<00:00, 168.29it/s]
Iterate posting for term 'x': 100%|█████| 81505/81505 [00:31<00:00, 2550.63it/s]
Iterate posting for term 'property.economic': 100%|█| 1/1 [00:00<00:00, 1535.81i
Iterate posting for term 'consider': 100%|█| 48572/48572 [01:33<00:00, 521.06it/
Iterate posting for term 'examplesuppose': 100%|█| 2/2 [00:00<00:00, 1917.40it/s
Iterate posting for term 'example': 100%|█| 234474/234474 [02:23<00:00, 1632.72i
Iterate posting for term 'montonic': 100%|██████| 1/1 [00:00<00:00, 1822.03it/s]
Iterate posting for term 'nonincreasing': 100%|█| 6/6 [00:00<00:00, 2303.30it/s]
```

```
-----------------------------------------------------
Current query: example of monotonic function monotonic function domain x property.econom
ic consider examplesuppose example montonic nonincreasing nondecreasing decreasing isnot
concave convex.ote
```

```
Iterate posting for term 'example': 100%|█| 234474/234474 [02:10<00:00, 1799.26i
Iterate posting for term 'monotonic': 100%|███| 68/68 [00:00<00:00, 3229.35it/s]
Iterate posting for term 'function': 100%|█| 89643/89643 [01:32<00:00, 969.44it/
Iterate posting for term 'monotonic': 100%|███| 68/68 [00:00<00:00, 3350.28it/s]
Iterate posting for term 'function': 100%|█| 89643/89643 [01:45<00:00, 849.48it/
Iterate posting for term 'domain': 100%|█| 13475/13475 [00:04<00:00, 2881.87it/s
Iterate posting for term 'x': 100%|█████| 81505/81505 [01:13<00:00, 1107.88it/s]
Iterate posting for term 'property.economic': 100%|█| 1/1 [00:00<00:00, 1782.53i
Iterate posting for term 'consider': 100%|█| 48572/48572 [00:15<00:00, 3094.87it
Iterate posting for term 'examplesuppose': 100%|█| 2/2 [00:00<00:00, 1872.46it/s
Iterate posting for term 'example': 100%|█| 234474/234474 [03:12<00:00, 1218.52i
Iterate posting for term 'montonic': 100%|███████| 1/1 [00:00<00:00, 858.26it/s]
Iterate posting for term 'nonincreasing': 100%|█| 6/6 [00:00<00:00, 2031.30it/s]
Iterate posting for term 'nondecreasing': 100%|█| 7/7 [00:00<00:00, 2435.92it/s]
Iterate posting for term 'decreasing': 100%|█| 5440/5440 [00:01<00:00, 3107.39it
Iterate posting for term 'isnot': 100%|███████| 41/41 [00:00<00:00, 3118.10it/s]
Iterate posting for term 'concave': 100%|█| 1221/1221 [00:00<00:00, 2578.72it/s]
Iterate posting for term 'convex.ote': 100%|████| 1/1 [00:00<00:00, 2053.01it/s]
```

```
-----------------------------------------------------
Current query:  monotonic function example
```

```
Iterate posting for term 'monotonic': 100%|███| 68/68 [00:00<00:00, 3413.51it/s]
Iterate posting for term 'function': 100%|█| 89643/89643 [01:27<00:00, 1029.38it
Iterate posting for term 'example': 100%|█| 234474/234474 [02:58<00:00, 1310.21i
-----------------------------------------------------
Current query:  monotonic function example
Iterate posting for term 'monotonic': 100%|███| 68/68 [00:00<00:00, 2781.86it/s]
Iterate posting for term 'function': 100%|█| 89643/89643 [00:30<00:00, 2931.75it
Iterate posting for term 'example': 100%|█| 234474/234474 [03:09<00:00, 1234.57i
-----------------------------------------------------
Current query: what is physical description of spruce spruce description conical colorad
o blue tree maturing trees christmas feet
Iterate posting for term 'what': 100%|█| 542299/542299 [06:40<00:00, 1353.18it/s
Iterate posting for term 'physical': 100%|█| 77532/77532 [00:22<00:00, 3383.74it
Iterate posting for term 'description': 100%|█| 40874/40874 [00:43<00:00, 936.29
Iterate posting for term 'spruce': 100%|██| 1771/1771 [00:00<00:00, 2065.86it/s]
Iterate posting for term 'spruce': 100%|██| 1771/1771 [00:00<00:00, 3424.91it/s]
Iterate posting for term 'description': 100%|█| 40874/40874 [00:13<00:00, 3026.1
Iterate posting for term 'conical': 100%|█| 1185/1185 [00:00<00:00, 3146.10it/s]
Iterate posting for term 'colorado': 100%|█| 18486/18486 [00:05<00:00, 3295.92it
Iterate posting for term 'blue': 100%|██| 58858/58858 [00:18<00:00, 3229.14it/s]
Iterate posting for term 'tree': 100%|███| 41959/41959 [01:14<00:00, 560.75it/s]
Iterate posting for term 'maturing': 100%|██| 716/716 [00:00<00:00, 3115.88it/s]
Iterate posting for term 'trees': 100%|█| 30745/30745 [00:09<00:00, 3121.36it/s]
Iterate posting for term 'christmas': 100%|█| 12028/12028 [00:03<00:00, 3199.52i
Iterate posting for term 'feet': 100%|█| 102013/102013 [01:52<00:00, 905.77it/s]
-----------------------------------------------------
Current query: what is physical description of spruce spruce description conical colorad
o blue tree maturing trees christmas feet utah habit therapist.ecoming tall moench
Iterate posting for term 'what': 100%|█| 542299/542299 [06:14<00:00, 1448.38it/s
Iterate posting for term 'physical': 100%|█| 77532/77532 [01:10<00:00, 1092.12it
Iterate posting for term 'description': 100%|█| 40874/40874 [00:12<00:00, 3354.1
Iterate posting for term 'spruce': 100%|██| 1771/1771 [00:00<00:00, 3358.00it/s]
Iterate posting for term 'spruce': 100%|██| 1771/1771 [00:00<00:00, 3623.34it/s]
Iterate posting for term 'description': 100%|█| 40874/40874 [00:45<00:00, 895.60
Iterate posting for term 'conical': 100%|█| 1185/1185 [00:00<00:00, 2857.69it/s]
Iterate posting for term 'colorado': 100%|█| 18486/18486 [00:05<00:00, 3180.98it
Iterate posting for term 'blue': 100%|██| 58858/58858 [00:19<00:00, 3043.75it/s]
Iterate posting for term 'tree': 100%|██| 41959/41959 [00:12<00:00, 3311.59it/s]
Iterate posting for term 'maturing': 100%|██| 716/716 [00:00<00:00, 2911.10it/s]
Iterate posting for term 'trees': 100%|██| 30745/30745 [01:03<00:00, 484.80it/s]
Iterate posting for term 'christmas': 100%|█| 12028/12028 [00:05<00:00, 2402.11i
Iterate posting for term 'feet': 100%|█| 102013/102013 [00:32<00:00, 3145.35it/s
Iterate posting for term 'utah': 100%|█████| 9266/9266 [00:53<00:00, 174.15it/s]
Iterate posting for term 'habit': 100%|███| 5646/5646 [00:02<00:00, 2272.63it/s]
Iterate posting for term 'therapist.ecoming': 100%|█| 1/1 [00:00<00:00, 2493.64i
Iterate posting for term 'tall': 100%|██| 26547/26547 [00:08<00:00, 2999.66it/s]
Iterate posting for term 'moench': 100%|██████| 36/36 [00:00<00:00, 3133.20it/s]
-----------------------------------------------------
Current query:  spruce description physical what
Iterate posting for term 'spruce': 100%|██| 1771/1771 [00:00<00:00, 3292.61it/s]
Iterate posting for term 'description': 100%|█| 40874/40874 [00:12<00:00, 3268.8
Iterate posting for term 'physical': 100%|█| 77532/77532 [01:08<00:00, 1139.13it
Iterate posting for term 'what': 100%|█| 542299/542299 [05:18<00:00, 1701.63it/s
-----------------------------------------------------
Current query:  spruce description physical
Iterate posting for term 'spruce': 100%|██| 1771/1771 [00:00<00:00, 3135.07it/s]
Iterate posting for term 'description': 100%|█| 40874/40874 [00:14<00:00, 2747.5
Iterate posting for term 'physical': 100%|█| 77532/77532 [01:15<00:00, 1026.74it
-----------------------------------------------------
Current query: hydrogen is a liquid below what temperature hydrogen liquid temperature b
elow degrees 434.49 studysoup 423.17 f.due what
Iterate posting for term 'hydrogen': 100%|█| 22860/22860 [00:07<00:00, 3040.73it
Iterate posting for term 'liquid': 100%|█| 48415/48415 [01:12<00:00, 669.32it/s]
Iterate posting for term 'below': 100%|█| 206486/206486 [02:02<00:00, 1687.97it/
Iterate posting for term 'what': 100%|█| 542299/542299 [05:19<00:00, 1695.56it/s
Iterate posting for term 'temperature': 100%|█| 121196/121196 [01:32<00:00, 1304
Iterate posting for term 'hydrogen': 100%|█| 22860/22860 [00:07<00:00, 3143.30it
Iterate posting for term 'liquid': 100%|█| 48415/48415 [01:10<00:00, 687.37it/s]
Iterate posting for term 'temperature': 100%|█| 121196/121196 [01:49<00:00, 1105
Iterate posting for term 'below': 100%|█| 206486/206486 [02:07<00:00, 1624.62it/
Iterate posting for term 'degrees': 100%|█| 68855/68855 [01:31<00:00, 753.63it/s
Iterate posting for term '434.49': 100%|████████| 3/3 [00:00<00:00, 1799.62it/s]
Iterate posting for term 'studysoup': 100%|███| 75/75 [00:00<00:00, 2921.67it/s]
Iterate posting for term '423.17': 100%|████████| 6/6 [00:00<00:00, 2300.77it/s]
Iterate posting for term 'f.due': 100%|█████████| 1/1 [00:00<00:00, 2091.92it/s]
Iterate posting for term 'what': 100%|█| 542299/542299 [06:11<00:00, 1458.13it/s
```

```
-----------------------------------------------------
Current query: hydrogen is a liquid below what temperature hydrogen liquid temperature b
elow degrees 434.49 studysoup 423.17 f.due what 423.18 gas notetaker liquidation 183
```

```
Iterate posting for term 'hydrogen': 100%|█| 22860/22860 [00:07<00:00, 3238.50it
Iterate posting for term 'liquid': 100%|█| 48415/48415 [01:13<00:00, 662.54it/s]
Iterate posting for term 'below': 100%|█| 206486/206486 [02:11<00:00, 1571.67it/
Iterate posting for term 'what': 100%|█| 542299/542299 [05:57<00:00, 1515.45it/s
Iterate posting for term 'temperature': 100%|█| 121196/121196 [01:33<00:00, 1299
Iterate posting for term 'hydrogen': 100%|█| 22860/22860 [00:08<00:00, 2673.79it
Iterate posting for term 'liquid': 100%|█| 48415/48415 [00:15<00:00, 3151.30it/s
Iterate posting for term 'temperature': 100%|█| 121196/121196 [01:42<00:00, 1183
Iterate posting for term 'below': 100%|█| 206486/206486 [03:33<00:00, 965.48it/s
Iterate posting for term 'degrees': 100%|█| 68855/68855 [00:20<00:00, 3441.63it/
Iterate posting for term '434.49': 100%|████████| 3/3 [00:00<00:00, 2785.68it/s]
Iterate posting for term 'studysoup': 100%|███| 75/75 [00:00<00:00, 2716.03it/s]
Iterate posting for term '423.17': 100%|████████| 6/6 [00:00<00:00, 2789.69it/s]
Iterate posting for term 'f.due': 100%|█████████| 1/1 [00:00<00:00, 2114.06it/s]
Iterate posting for term 'what': 100%|█| 542299/542299 [06:36<00:00, 1366.65it/s
Iterate posting for term '423.18': 100%|████████| 2/2 [00:00<00:00, 1416.28it/s]
Iterate posting for term 'gas': 100%|████| 75249/75249 [01:17<00:00, 973.27it/s]
Iterate posting for term 'notetaker': 100%|███| 52/52 [00:00<00:00, 3094.15it/s]
Iterate posting for term 'liquidation': 100%|█| 691/691 [00:00<00:00, 3187.24it/
Iterate posting for term '183': 100%|█████| 1310/1310 [00:00<00:00, 3194.60it/s]
```

```
-----------------------------------------------------
Current query:  hydrogen liquid temperature below what
```

```
Iterate posting for term 'hydrogen': 100%|█| 22860/22860 [00:06<00:00, 3284.80it
Iterate posting for term 'liquid': 100%|█| 48415/48415 [00:18<00:00, 2663.85it/s
Iterate posting for term 'temperature': 100%|█| 121196/121196 [01:44<00:00, 1163
Iterate posting for term 'below': 100%|█| 206486/206486 [03:29<00:00, 985.56it/s
Iterate posting for term 'what': 100%|█| 542299/542299 [06:51<00:00, 1316.89it/s
```

```
-----------------------------------------------------
Current query:  hydrogen liquid temperature
```

```
Iterate posting for term 'hydrogen': 100%|█| 22860/22860 [00:09<00:00, 2320.71it
Iterate posting for term 'liquid': 100%|█| 48415/48415 [01:16<00:00, 635.09it/s]
Iterate posting for term 'temperature': 100%|█| 121196/121196 [01:58<00:00, 1018
```

```
-----------------------------------------------------
Current query: difference between a mcdouble and a double cheeseburger mcdouble cheesebu
rger double cheeseburgers cheese slices burgers cheesier slice mcdonaldâ
```

```
Iterate posting for term 'difference': 100%|█| 75516/75516 [00:25<00:00, 2910.59
Iterate posting for term 'between': 100%|█| 416673/416673 [05:29<00:00, 1265.48i
Iterate posting for term 'mcdouble': 100%|████| 24/24 [00:00<00:00, 1745.72it/s]
Iterate posting for term 'double': 100%|█| 45518/45518 [00:13<00:00, 3326.85it/s
Iterate posting for term 'cheeseburger': 100%|█| 536/536 [00:00<00:00, 3647.35it
Iterate posting for term 'mcdouble': 100%|████| 24/24 [00:00<00:00, 2185.01it/s]
Iterate posting for term 'cheeseburger': 100%|█| 536/536 [00:00<00:00, 3579.20it
Iterate posting for term 'double': 100%|█| 45518/45518 [00:13<00:00, 3444.17it/s
Iterate posting for term 'cheeseburgers': 100%|█| 124/124 [00:00<00:00, 3243.96i
Iterate posting for term 'cheese': 100%|█| 26063/26063 [00:51<00:00, 509.06it/s]
Iterate posting for term 'slices': 100%|██| 6503/6503 [00:02<00:00, 3185.86it/s]
Iterate posting for term 'burgers': 100%|█| 2597/2597 [00:00<00:00, 3255.53it/s]
Iterate posting for term 'cheesier': 100%|██████| 9/9 [00:00<00:00, 2604.80it/s]
Iterate posting for term 'slice': 100%|███| 9229/9229 [00:02<00:00, 3320.32it/s]
Iterate posting for term 'mcdonaldâ': 100%|█| 937/937 [00:00<00:00, 3349.66it/s]
```

```
-----------------------------------------------------
Current query: difference between a mcdouble and a double cheeseburger mcdouble cheesebu
rger double cheeseburgers cheese slices burgers cheesier slice mcdonaldâ beefier patties
gristle 1.39 1.49
```

```
Iterate posting for term 'difference': 100%|█| 75516/75516 [00:26<00:00, 2874.33
Iterate posting for term 'between': 100%|█| 416673/416673 [04:39<00:00, 1491.21i
Iterate posting for term 'mcdouble': 100%|████| 24/24 [00:00<00:00, 2716.74it/s]
Iterate posting for term 'double': 100%|█| 45518/45518 [00:13<00:00, 3336.56it/s
Iterate posting for term 'cheeseburger': 100%|█| 536/536 [00:00<00:00, 3493.20it
Iterate posting for term 'mcdouble': 100%|████| 24/24 [00:00<00:00, 2914.40it/s]
Iterate posting for term 'cheeseburger': 100%|█| 536/536 [00:00<00:00, 3479.55it
Iterate posting for term 'double': 100%|█| 45518/45518 [01:06<00:00, 681.86it/s]
Iterate posting for term 'cheeseburgers': 100%|█| 124/124 [00:00<00:00, 1180.66i
Iterate posting for term 'cheese': 100%|█| 26063/26063 [00:09<00:00, 2819.02it/s
Iterate posting for term 'slices': 100%|██| 6503/6503 [00:02<00:00, 2824.73it/s]
Iterate posting for term 'burgers': 100%|█| 2597/2597 [00:00<00:00, 3142.71it/s]
Iterate posting for term 'cheesier': 100%|██████| 9/9 [00:00<00:00, 2587.66it/s]
Iterate posting for term 'slice': 100%|███| 9229/9229 [00:02<00:00, 3095.49it/s]
Iterate posting for term 'mcdonaldâ': 100%|█| 937/937 [00:00<00:00, 3226.79it/s]
Iterate posting for term 'beefier': 100%|█████| 41/41 [00:00<00:00, 3285.82it/s]
Iterate posting for term 'patties': 100%|█| 1327/1327 [00:00<00:00, 3288.24it/s]
Iterate posting for term 'gristle': 100%|███| 115/115 [00:00<00:00, 3139.35it/s]
Iterate posting for term '1.39': 100%|██████| 201/201 [00:00<00:00, 2723.06it/s]
Iterate posting for term '1.49': 100%|██████| 298/298 [00:00<00:00, 2979.97it/s]
```

```
-----------------------------------------------------
Current query:  mcdouble cheeseburger double difference between
```

```
Iterate posting for term 'mcdouble': 100%|████| 24/24 [00:00<00:00, 2440.26it/s]
Iterate posting for term 'cheeseburger': 100%|█| 536/536 [00:00<00:00, 3368.00it
Iterate posting for term 'double': 100%|█| 45518/45518 [00:13<00:00, 3313.62it/s
Iterate posting for term 'difference': 100%|█| 75516/75516 [01:20<00:00, 934.50i
Iterate posting for term 'between': 100%|█| 416673/416673 [04:26<00:00, 1561.15i
```

```
-----------------------------------------------------
Current query:  mcdouble cheeseburger double
```

```
Iterate posting for term 'mcdouble': 100%|████| 24/24 [00:00<00:00, 2946.99it/s]
Iterate posting for term 'cheeseburger': 100%|█| 536/536 [00:00<00:00, 3220.36it
Iterate posting for term 'double': 100%|█| 45518/45518 [00:59<00:00, 760.20it/s]
```

In [32]:

```python
# To evaluate the result
def print_results(run, qrel_file='qrel.txt', measures=["map", "ndcg_cut_10", "recall_100
    # Open the qrels file.
    with open(qrel_file, "r") as f:
        msmarco_qrels = pytrec_eval.parse_qrel(f)

    evaluator = pytrec_eval.RelevanceEvaluator(query_relevance=msmarco_qrels, measures=m
    results = evaluator.evaluate(run)
    for measure in sorted(measures):
        print('{:25s}{:8s}{:.4f}'.format(measure, 'all', pytrec_eval.compute_aggregated_
                                         [query_measures[measure]for query_measures in results.
    return results
```

In [21]:

```python
# Store the query result into run files for fusion method
with open("project-part1-bm25.run","w") as f:
    for x in range(len(queries)):
        for i in range(len(bm_25[x])):
            f.write(f"{queries[x][0]} Q0 {bm_25[x][i][0]} {i} {bm_25[x][i][1]} infs7410_

with open("project-part1-expand-10.run","w") as f:
    for x in range(len(queries)):
        for i in range(len(expand_bm25_add_top10[x])):
            f.write(f"{queries[x][0]} Q0 {expand_bm25_add_top10[x][i][0]} {i} {expand_bm

with open("project-part1-expand-15.run","w") as f:
    for x in range(len(queries)):
        for i in range(len(expand_bm25_add_top5[x])):
            f.write(f"{queries[x][0]} Q0 {expand_bm25_add_top5[x][i][0]} {i} {expand_bm2

with open("project-part1-reduce-5-leave.run","w") as f:
    for x in range(len(queries)):
        for i in range(len(reduce_bm25_5_leave[x])):
            f.write(f"{queries[x][0]} Q0 {reduce_bm25_5_leave[x][i][0]} {i} {reduce_bm25

with open("project-part1-reduce-3-leave.run","w") as f:
    for x in range(len(queries)):
        for i in range(len(reduce_bm25_3_leave[x])):
            f.write(f"{queries[x][0]} Q0 {reduce_bm25_3_leave[x][i][0]} {i} {reduce_bm25
```

In [33]:

```
# Run query result
with open("project-part1-bm25.run","r") as f:
    project_bm25 = normalise_run(pytrec_eval.parse_run(f))

# Evaluate models
print('-----------------------------------------')
print('BM25')
project_bm25_result=print_results(project_bm25)

with open("project-part1-expand-10.run","r") as f:
    project_expand10 = normalise_run(pytrec_eval.parse_run(f))
# Evaluate models
print('-----------------------------------------')
print('Expand 10 words')
project_expand10_result=print_results(project_expand10)

with open("project-part1-expand-15.run","r") as f:
    project_expand15 = normalise_run(pytrec_eval.parse_run(f))
# Evaluate models
print('-----------------------------------------')
print('Expand 15 words')
project_expand15_result=print_results(project_expand15)

with open("project-part1-reduce-5-leave.run","r") as f:
    project_leave_5 = normalise_run(pytrec_eval.parse_run(f))
# Evaluate models
print('-----------------------------------------')
print('Reduce down to 5')
project_leave5_result=print_results(project_leave_5)

with open("project-part1-reduce-3-leave.run","r") as f:
    project_leave_3 = normalise_run(pytrec_eval.parse_run(f))
# Evaluate models
print('-----------------------------------------')
print('Reduce down to 3')
project_leave3_result=print_results(project_leave_3)
```

```
-----------------------------------------
BM25
map                     all     0.0030
ndcg_cut_10             all     0.0055
recall_1000             all     0.0665
recip_rank              all     0.0258
-----------------------------------------
Expand 10 words
map                     all     0.0196
ndcg_cut_10             all     0.0286
recall_1000             all     0.2481
recip_rank              all     0.1193
-----------------------------------------
Expand 15 words
```

```
map                     all     0.0178
ndcg_cut_10             all     0.0284
recall_1000             all     0.2438
recip_rank              all     0.0990
-----------------------------------------
Reduce down to 5
map                     all     0.0030
ndcg_cut_10             all     0.0055
recall_1000             all     0.0675
recip_rank              all     0.0260
-----------------------------------------
Reduce down to 3
map                     all     0.0184
ndcg_cut_10             all     0.0276
recall_1000             all     0.1697
recip_rank              all     0.0680
```

The result is printed in the table below for the 3 methods impletemented with different parameters:

	MAP	nDCG	Reciprocal Rank	Recall
BM25	0.0030	0.0055	0.0665	0.0258
Query Expansion (10 Words)	0.0196	0.0286	0.2481	0.1193
Query Expansion (15 words)	0.0178	0.0284	0.2438	0.0990
Query Reduction (5 words)	0.0030	0.0055	0.0675	0.1697
Query Reduction (3 words)	0.0184	0.0276	0.1697	0.0680

For the next step, statistical significant testing is applied to find the best parameter for each model and the best runs will be used to construct the model for rank fusion.

In [52]:

```python
# Query Expansion
query_ids = list(
    set(project_expand10_result.keys()) & set(project_expand15_result.keys()))

expand10_scores = [
    project_expand10_result[query_id]["recall_1000"] for query_id in query_ids]
expand15_scores = [
    project_expand15_result[query_id]["recall_1000"] for query_id in query_ids]

print('Recall:',scipy.stats.ttest_rel(expand10_scores, expand15_scores))

expand10_scores = [
    project_expand10_result[query_id]["map"] for query_id in query_ids]
expand15_scores = [
    project_expand15_result[query_id]["map"] for query_id in query_ids]

print('MAP:',scipy.stats.ttest_rel(expand10_scores, expand15_scores))

expand10_scores = [
    project_expand10_result[query_id]["recip_rank"] for query_id in query_ids]
expand15_scores = [
    project_expand15_result[query_id]["recip_rank"] for query_id in query_ids]

print('Reciprocal Rank:',scipy.stats.ttest_rel(expand10_scores, expand15_scores))

expand10_scores = [
    project_expand10_result[query_id]["ndcg_cut_10"] for query_id in query_ids]
expand15_scores = [
    project_expand15_result[query_id]["ndcg_cut_10"] for query_id in query_ids]

print('nDCG:',scipy.stats.ttest_rel(expand10_scores, expand15_scores))
```

```
Recall: Ttest_relResult(statistic=0.43265879591325024, pvalue=0.6674762904956419)
MAP: Ttest_relResult(statistic=1.1188848212467883, pvalue=0.2695478394689616)
Reciprocal Rank: Ttest_relResult(statistic=1.6596812403828607, pvalue=0.1044283315289314
4)
nDCG: Ttest_relResult(statistic=0.06177360406901148, pvalue=0.95103616172902)
```

In [47]:

```
# Query Reduction
query_ids = list(
    set(project_leave3_result.keys()) & set(project_leave5_result.keys()))

leave3_scores = [
    project_leave3_result[query_id]["recall_1000"] for query_id in query_ids]
leave5_scores = [
    project_leave5_result[query_id]["recall_1000"] for query_id in query_ids]

print('Recall:',scipy.stats.ttest_rel(leave3_scores, leave5_scores))

leave3_scores = [
    project_leave3_result[query_id]["map"] for query_id in query_ids]
leave5_scores = [
    project_leave5_result[query_id]["map"] for query_id in query_ids]

print('MAP:',scipy.stats.ttest_rel(leave3_scores, leave5_scores))

leave3_scores = [
    project_leave3_result[query_id]["recip_rank"] for query_id in query_ids]
leave5_scores = [
    project_leave5_result[query_id]["recip_rank"] for query_id in query_ids]

print('Reciprocal Rank:',scipy.stats.ttest_rel(leave3_scores, leave5_scores))

leave3_scores = [
    project_leave3_result[query_id]["ndcg_cut_10"] for query_id in query_ids]
leave5_scores = [
    project_leave5_result[query_id]["ndcg_cut_10"] for query_id in query_ids]

print('nDCG:',scipy.stats.ttest_rel(leave3_scores, leave5_scores))
```

```
Recall: Ttest_relResult(statistic=4.301781832896046, pvalue=9.894625219719445e-05)
MAP: Ttest_relResult(statistic=2.411270471435657, pvalue=0.020348878443274637)
Reciprocal Rank: Ttest_relResult(statistic=2.684369324978915, pvalue=0.0103552316471493
6)
nDCG: Ttest_relResult(statistic=1.9736043766366775, pvalue=0.055029555058031435)
```

From the statistical testing, query reduction shows there are difference between leaving 5 words and 3 words but query expansion shows no difference between 10 words and 15 words. This indicate that either picking query expansion with 10 words has no difference with picking query expansion with 15 words.

For rank fusion, I have chosen to combine BM25,query expansion with 10 words and query reduction to 3 words to evalutet which rank fusion method is better. Borda, CombSUM and CombMNZ will be implemented and evaluate.

In [60]:

```
runs = [project_bm25,project_expand10,project_leave_3]

print('-----------------------------------------')
print('BM25 + Expand 10 + Leave 3')
print('-----------------------------------------')
print('Fused Borda')
fused_borda_run = borda(runs)
fused_borda_result_bm25 = print_results(fused_borda_run)

print('-----------------------------------------')
print('Fused CombSUM')
fused_combsum_run = combsum(runs)
fused_combsum_result_bm25 =print_results(fused_combsum_run)

print('-----------------------------------------')
print('Fused CombMNZ')
fused_combmnz_run = combmnz(runs)
fused_combmnz_result_bm25 = print_results(fused_combmnz_run)


runs = [project_expand10,project_leave_3]

print('-----------------------------------------')
print('Expand 10 + Leave 3')
runs = [project_expand10,project_leave_3]

print('-----------------------------------------')
print('Fused Borda')
fused_borda_run = borda(runs)
fused_borda_result = print_results(fused_borda_run)

print('-----------------------------------------')
print('Fused CombSUM')
fused_combsum_run = combsum(runs)
fused_combsum_result =print_results(fused_combsum_run)

print('-----------------------------------------')
print('Fused CombMNZ')
fused_combmnz_run = combmnz(runs)
fused_combmnz_result = print_results(fused_combmnz_run)
```

```
-----------------------------------------
BM25 + Expand 10 + Leave 3
-----------------------------------------
Fused Borda
map                      all     0.0186
ndcg_cut_10              all     0.0363
recall_1000              all     0.2056
recip_rank               all     0.0774
-----------------------------------------
Fused CombSUM
map                      all     0.0163
ndcg_cut_10              all     0.0186
recall_1000              all     0.2137
recip_rank               all     0.0788
-----------------------------------------
Fused CombMNZ
map                      all     0.0183
ndcg_cut_10              all     0.0353
recall_1000              all     0.2046
recip_rank               all     0.0844
-----------------------------------------
Expand 10 + Leave 3
-----------------------------------------
Fused Borda
map                      all     0.0258
ndcg_cut_10              all     0.0450
recall_1000              all     0.2409
recip_rank               all     0.0971
-----------------------------------------
Fused CombSUM
map                      all     0.0232
ndcg_cut_10              all     0.0374
recall_1000              all     0.2381
recip_rank               all     0.0932
-----------------------------------------
```

```
Fused CombMNZ
map                     all     0.0252
ndcg_cut_10             all     0.0426
recall_1000             all     0.2394
recip_rank              all     0.1003
```

In [65]:

```
# Fusion inclue BM25 or not
print('----------Borda----------')
query_ids = list(
    set(fused_borda_result.keys()) & set(fused_borda_result_bm25.keys()))

no_bm25_scores = [
    fused_borda_result[query_id]["recall_1000"] for query_id in query_ids]
bm25_scores = [
    fused_borda_result_bm25[query_id]["recall_1000"] for query_id in query_ids]

print('Recall:',scipy.stats.ttest_rel(no_bm25_scores, bm25_scores))

no_bm25_scores = [
    fused_borda_result[query_id]["map"] for query_id in query_ids]
bm25_scores = [
    fused_borda_result_bm25[query_id]["map"] for query_id in query_ids]

print('MAP:',scipy.stats.ttest_rel(no_bm25_scores, bm25_scores))

no_bm25_scores = [
    fused_borda_result[query_id]["recip_rank"] for query_id in query_ids]
bm25_scores = [
    fused_borda_result_bm25[query_id]["recip_rank"] for query_id in query_ids]

print('Reciprocal Rank:',scipy.stats.ttest_rel(no_bm25_scores, bm25_scores))

no_bm25_scores = [
    fused_borda_result[query_id]["ndcg_cut_10"] for query_id in query_ids]
bm25_scores = [
    fused_borda_result_bm25[query_id]["ndcg_cut_10"] for query_id in query_ids]

print('nDCG:',scipy.stats.ttest_rel(no_bm25_scores, bm25_scores))

# Fusion inclue BM25 or not
# CombSUM
print('----------CombSUM----------')
query_ids = list(
    set(fused_combsum_result.keys()) & set(fused_combsum_result_bm25.keys()))

no_bm25_scores = [
    fused_combsum_result[query_id]["recall_1000"] for query_id in query_ids]
bm25_scores = [
    fused_combsum_result_bm25[query_id]["recall_1000"] for query_id in query_ids]
print('Recall:',scipy.stats.ttest_rel(no_bm25_scores, bm25_scores))

no_bm25_scores = [
    fused_combsum_result[query_id]["map"] for query_id in query_ids]
bm25_scores = [
    fused_combsum_result_bm25[query_id]["map"] for query_id in query_ids]

print('MAP:',scipy.stats.ttest_rel(no_bm25_scores, bm25_scores))

no_bm25_scores = [
    fused_combsum_result[query_id]["recip_rank"] for query_id in query_ids]
bm25_scores = [
    fused_combsum_result_bm25[query_id]["recip_rank"] for query_id in query_ids]

print('Reciprocal Rank:',scipy.stats.ttest_rel(no_bm25_scores, bm25_scores))
```

```
no_bm25_scores = [
    fused_combsum_result[query_id]["ndcg_cut_10"] for query_id in query_ids]
bm25_scores = [
    fused_combsum_result_bm25[query_id]["ndcg_cut_10"] for query_id in query_ids]

print('nDCG:',scipy.stats.ttest_rel(no_bm25_scores, bm25_scores))

# CombSUM
print('----------CombMNZ----------')
query_ids = list(
    set(fused_combmnz_result.keys()) & set(fused_combmnz_result_bm25.keys()))

no_bm25_scores = [
    fused_combmnz_result[query_id]["recall_1000"] for query_id in query_ids]
bm25_scores = [
    fused_combmnz_result_bm25[query_id]["recall_1000"] for query_id in query_ids]
print('Recall:',scipy.stats.ttest_rel(no_bm25_scores, bm25_scores))

no_bm25_scores = [
    fused_combmnz_result[query_id]["map"] for query_id in query_ids]
bm25_scores = [
    fused_combmnz_result_bm25[query_id]["map"] for query_id in query_ids]

print('MAP:',scipy.stats.ttest_rel(no_bm25_scores, bm25_scores))

no_bm25_scores = [
    fused_combmnz_result[query_id]["recip_rank"] for query_id in query_ids]
bm25_scores = [
    fused_combmnz_result_bm25[query_id]["recip_rank"] for query_id in query_ids]

print('Reciprocal Rank:',scipy.stats.ttest_rel(no_bm25_scores, bm25_scores))

no_bm25_scores = [
    fused_combmnz_result[query_id]["ndcg_cut_10"] for query_id in query_ids]
bm25_scores = [
    fused_combmnz_result_bm25[query_id]["ndcg_cut_10"] for query_id in query_ids]

print('nDCG:',scipy.stats.ttest_rel(no_bm25_scores, bm25_scores))
```

```
----------Borda----------
Recall: Ttest_relResult(statistic=4.851502026978419, pvalue=1.721620077073426e-05)
MAP: Ttest_relResult(statistic=2.6693656373114036, pvalue=0.010757680032676574)
Reciprocal Rank: Ttest_relResult(statistic=1.9781073090895183, pvalue=0.0545012026773643
9)
nDCG: Ttest_relResult(statistic=2.350539270786926, pvalue=0.023517504830798588)
----------CombSUM----------
Recall: Ttest_relResult(statistic=5.181223448153183, pvalue=5.899659503993341e-06)
MAP: Ttest_relResult(statistic=2.498856647541443, pvalue=0.016456411895112544)
Reciprocal Rank: Ttest_relResult(statistic=3.5143095242024573, pvalue=0.0010704941938842
706)
nDCG: Ttest_relResult(statistic=1.9241391532248016, pvalue=0.061130650773959544)
----------CombMNZ----------
Recall: Ttest_relResult(statistic=5.341961992372163, pvalue=3.485403788812815e-06)
MAP: Ttest_relResult(statistic=2.8284157255496973, pvalue=0.00714001941105448)
Reciprocal Rank: Ttest_relResult(statistic=3.3001672001869604, pvalue=0.0019763453466265
612)
nDCG: Ttest_relResult(statistic=1.624446676719083, pvalue=0.11176276580448428)
```

	MAP	nDCG	Recall	Reciprocal Rank
Fused Borda (BM25 + Expand 10 + Leave 3)	0.0186	0.0363	0.2056	0.0774
Fused CombSUM (BM25 + Expand 10 + Leave 3)	0.0163	0.0186	0.2137	0.0788
Fused CombMNZ (BM25 + Expand 10 + Leave 3)	0.0183	0.0353	0.2046	0.0844
Fused Borda (Expand 10 + Leave 3)	0.0258	0.0450	0.2409	0.0971

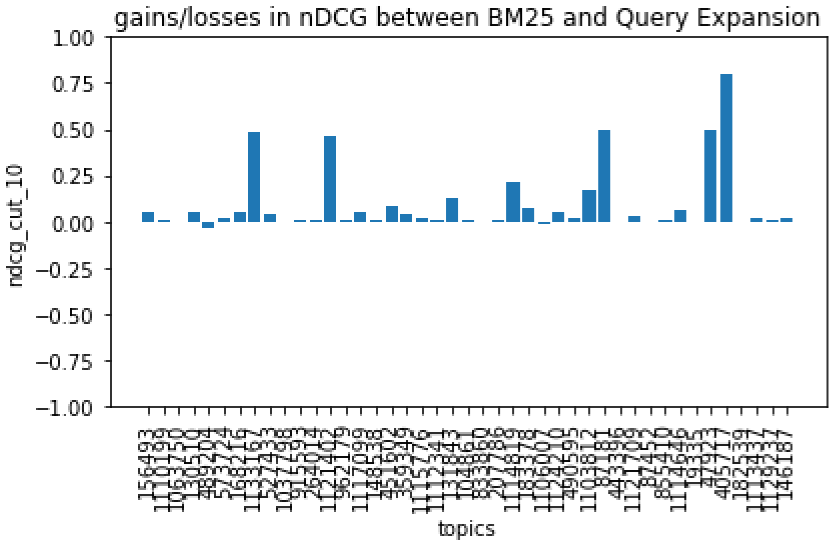
Fused CombSUM (Expand 10 + Leave 3)	0.0232	0.0374	0.2381	0.0932
Fused CombMNZ (Expand 10 + Leave 3)	0.0252	0.0426	0.2394	0.1003

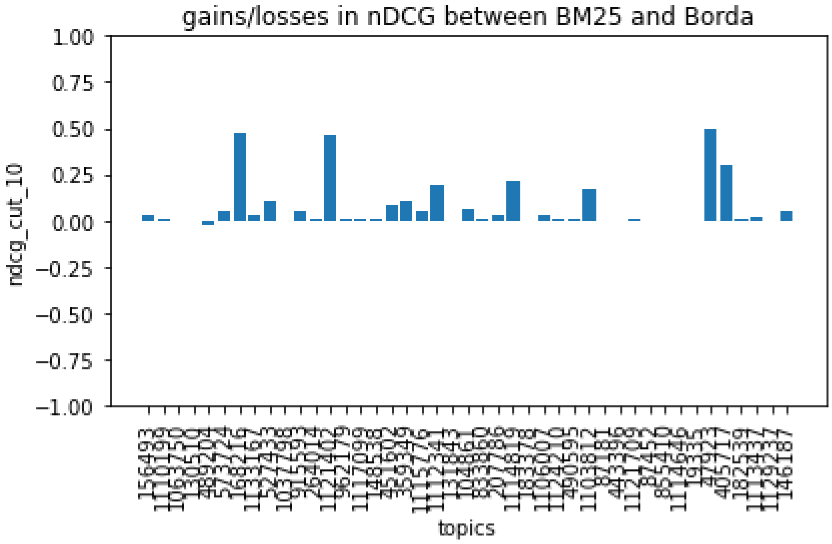
From statistical testing, we found that there are difference bewtween the fusion method keeping BM25 and not keeping BM25. From the table above, we observed generally not keeping BM25 has better performance than keeping BM25 for all fusion rank methods.

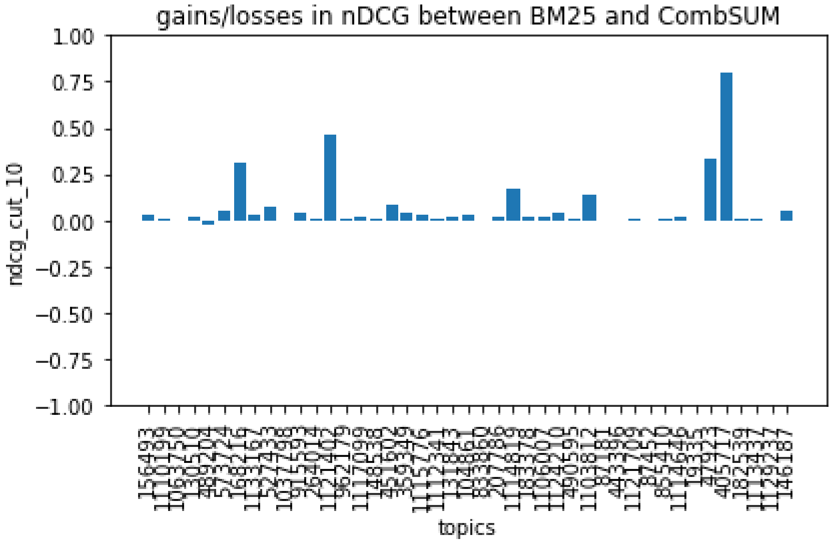
From the graph below, we found that query expasion, Borda, CombSUM and CombMNZ gains from the BM25 model. However, each model gain for different topics.

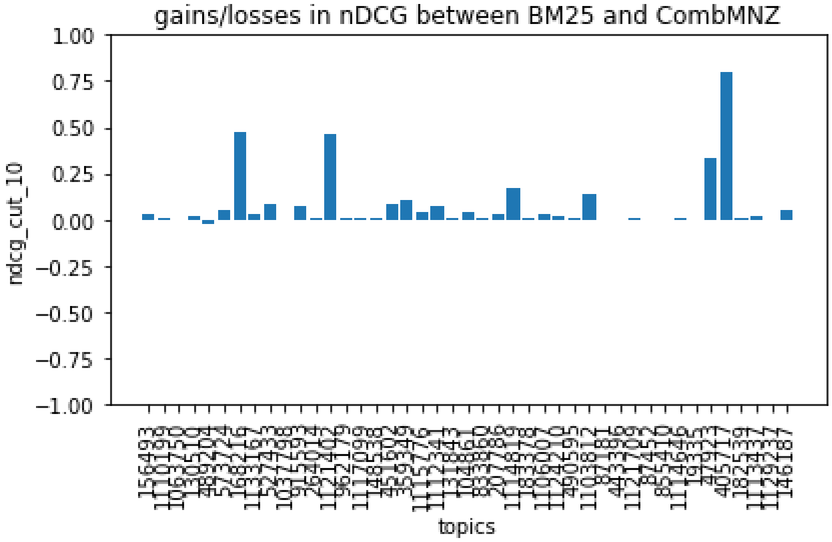
In [76]:

```
# Put your implementations for the gain-loss plots here.
measure = "ndcg_cut_10"

title = "gains/losses in nDCG between BM25 and Query Expansion"

r = dict([(key, value[i]) for key, value in project_bm25_result.items()])

r1 = dict([(key, value[i]) for key, value in project_bm25_result.items()])
r2 = dict([(key, value[i]) for key, value in project_expand10_result.items()])
ind = np.arange(len(r1))
# https://matplotlib.org/3.1.0/gallery/lines_bars_and_markers/barchart.html
plt.bar(ind, np.subtract(list(r2.values()), list(r1.values())))
plt.xticks(ind, list(r.keys()), rotation="vertical")
plt.ylim(-1, 1)
plt.title(title)
plt.ylabel(measure)
plt.xlabel("topics")
plt.tight_layout()
plt.show()


title = "gains/losses in nDCG between BM25 and Borda"

r = dict([(key, value[i]) for key, value in project_bm25_result.items()])

r1 = dict([(key, value[i]) for key, value in project_bm25_result.items()])
r2 = dict([(key, value[i]) for key, value in fused_borda_result.items()])
ind = np.arange(len(r1))
# https://matplotlib.org/3.1.0/gallery/lines_bars_and_markers/barchart.html
plt.bar(ind, np.subtract(list(r2.values()), list(r1.values())))
plt.xticks(ind, list(r.keys()), rotation="vertical")
plt.ylim(-1, 1)
plt.title(title)
plt.ylabel(measure)
plt.xlabel("topics")
plt.tight_layout()
plt.show()

title = "gains/losses in nDCG between BM25 and CombSUM"

r = dict([(key, value[i]) for key, value in project_bm25_result.items()])

r1 = dict([(key, value[i]) for key, value in project_bm25_result.items()])
r2 = dict([(key, value[i]) for key, value in fused_combsum_result.items()])
ind = np.arange(len(r1))
# https://matplotlib.org/3.1.0/gallery/lines_bars_and_markers/barchart.html
plt.bar(ind, np.subtract(list(r2.values()), list(r1.values())))
plt.xticks(ind, list(r.keys()), rotation="vertical")
plt.ylim(-1, 1)
plt.title(title)
plt.ylabel(measure)
plt.xlabel("topics")
plt.tight_layout()
plt.show()

title = "gains/losses in nDCG between BM25 and CombMNZ"

r = dict([(key, value[i]) for key, value in project_bm25_result.items()])

r1 = dict([(key, value[i]) for key, value in project_bm25_result.items()])
r2 = dict([(key, value[i]) for key, value in fused_combmnz_result.items()])
ind = np.arange(len(r1))
# https://matplotlib.org/3.1.0/gallery/lines_bars_and_markers/barchart.html
plt.bar(ind, np.subtract(list(r2.values()), list(r1.values())))
plt.xticks(ind, list(r.keys()), rotation="vertical")
plt.ylim(-1, 1)
plt.title(title)
plt.ylabel(measure)
plt.xlabel("topics")
plt.tight_layout()
plt.show()
```









Then you can edit this cell to provide some analysis about if there is a method that consistently outperform the others on the dev set and the test set.

In [ ]:

```
test_queries = []
with open("test_queries.tsv", "r") as f:
    for line in f.readlines():
        parts = line.split("\t")
        # parts[0] ~> topic id
        # parts[1] ~> query
        test_queries.append((parts[0], parts[1].strip()))

expand_bm25_add_top10_test = []
for i in range(len(queries)):
    expand_bm25_add_top10.append(search(prf_query_expansion_bm25(queries[i][1],5,10), 1.

with open("project-part1-test.run","w") as f:
    for x in range(len(test_queries)):
        for i in range(len(expand_bm25_add_top10_test[x])):
            f.write(f"{test_queries[x][0]} Q0 {expand_bm25_add_top10_test[x][i][0]} {i}
```

	MAP	nDCG	Reciprocal Rank	Recall
BM25	0.0030	0.0055	0.0665	0.0258
Query Expansion (10 Words)	0.0196	0.0286	0.2481	0.1193
Query Expansion (15 words)	0.0178	0.0284	0.2438	0.0990
Query Reduction (5 words)	0.0030	0.0055	0.0675	0.1697

Query Reduction (3 words)	0.0184	0.0276	0.1697	0.0680
Fused Borda (BM25 + Expand 10 + Leave 3)	0.0186	0.0363	0.2056	0.0774
Fused CombSUM (BM25 + Expand 10 + Leave 3)	0.0163	0.0186	0.2137	0.0788
Fused CombMNZ (BM25 + Expand 10 + Leave 3)	0.0183	0.0353	0.2046	0.0844
Fused Borda (Expand 10 + Leave 3)	0.0258	0.0450	0.2409	0.0971
Fused CombSUM (Expand 10 + Leave 3)	0.0232	0.0374	0.2381	0.0932
Fused CombMNZ (Expand 10 + Leave 3)	0.0252	0.0426	0.2394	0.1003

From the result of the dev set, we did see difference if we compare the fusion method with BM25 and query reduction alone. However, if we compare them with query expansion, we did not see much difference in performance. This maybe due to the fact that query expansion is included in the fusion method which contribute to the part retrieving related documents.

To extend the project, we may also adjust the saturation and document length value for the BM25 to see if there are amy improvement on performance of stand alone BM25 and also the fusion method. Furthermore, we can also adjust the index, stemming algorithm and stopwords since we have only removed stopwords in this trial. Choosing a stemming algorithm more also affect the performance of the model due to the training data changes.

In [ ]: