Information retrieval (CS60092) term project

FACETED QUERY BY EXAMPLE FOR SCIENTIFIC DOCUMENT RETRIEVAL

Group members (Group 11):

1. Morredigari Likhith Reddy (20CS10037)
2. Soni Aditya Bharatbhai (20CS10060)
3. Rishi Raj (20CS30040)
4. Shreyas Jena (20CS30049)

Introduction

- Given a query paper **Q**, a facet **f** and a set of candidate papers **C**.
- Task is to rank the candidate papers based on similarity with **Q** with respect to **f**.
- Dataset used for evaluation: **CSF-Cube**
- It has 50 query paper-facet pairs each with a set of candidate papers.
- Each candidate paper is assigned a relevance score from {0,1,2,3}.
- The facets are: background/objective, method and result.

CSFCube

- An expert annotated test dataset.
- It contains rated ranking for 50 query paper-facet pairs.
- For each paper, only title and abstract considered.
- Each sentence in abstract is assigned a label from { background, method, result, other}.
- Each query-facet paper has a candidate pool of 100-250 papers.
- Used as evaluation set for all experiments

OBJECTIVE

- Finer grained control on literature search.
- Explore techniques to improve upon state-of-the-art works.
- Find effective ways to find relevance between query and candidates.
- Essentially we require a better document-level representation.
- The representation must also consider the facet being considered.

Background

- **Term-level baselines**: tf-idf and cbow
- **Sentence-level baselines**: SentBERT and SimCSE
- **Abstract-level baselines**: SciBERT,SPECTER and SciNCL
- Only sentence and abstract level baselines are considered

Abstract level baselines

- **SciBERT-scivocab-uncased:**
  - A pre-trained language model trained in a similar way to BERT.
  - Training is done (from scratch) on a corpus of scientific text
  - Uses a different vocabulary called SciVocab.
- **SPECTER and SciNCL:**
  - Dense vector representation for scientific documents
  - SciBERT is fine-tuned using citation data using Contrastive learning
  - Models differ in the way citations are used for training

EXPERIMENTS

- **SciBERT-scivocab-uncased:**
  - Abstract of query/candidate is fed as input
  - CLS embedding is taken as dense vector document representation
- **SPECTER and SciNCL:**
  - Title + [SEP Token] + Abstract is fed as input for query/candidate
  - CLS embedding is taken as dense vector document representation

For all the cases L2 distance between query vector and candidate vector is considered during ranking. Candidates ranked in increasing order of distance

Cases considered

We have 3 ways of using the abstract sentences to capture similarity.

1. **QA-CA (Query abstract + Candidate abstract)**: For both query and candidate, all sentences of abstract are considered during ranked retrieval.
2. **QF-CA (Query facet + Candidate abstract):** For candidate paper same steps as in (1). For query paper, consider only those sentences in abstract which have the same label as the facet being searched for.
3. **QF-CF (Query facet + Candidate facet):** For both query and candidate papers, consider only those sentences in abstract which have same label as facet being searched for.

SENTENCE-LEVEL BASELINES

- **Sentence-BERT**:
  - Takes as input sentences of Query and Candidate papers.
  - Outputs the dense vector representations(embeddings).
  - Two variants models considered: SentBERT NLI and SentBERT Paraphrased.
- **SimCSE**:
  - Takes as input sentences of Query and Candidate papers.
  - Outputs the dense vector representations(embeddings).
  - Two variants models considered: Supervised SimCSE and Unsupervised SimCSE.

Cases considered

We have 3 ways of using the abstract sentences to capture similarity.

1. **QA-CA (Query abstract + Candidate abstract)**: For both query and candidate, all sentences of abstract are considered during ranked retrieval.
2. **QF-CA (Query facet + Candidate abstract):** For candidate paper same steps as in (1). For query paper, consider only those sentences which have the same label as the facet being considered for searching.
3. **QF-CF (Query facet + Candidate facet):** For both query and candidate papers, consider only those sentences which have the same label as the facet being considered for searching.

RESULTS AND BASELINE CODE

1. Baselines code: https://shorturl.at/fuLR8
2. Results spreadsheet: https://shorturl.at/cmpBE

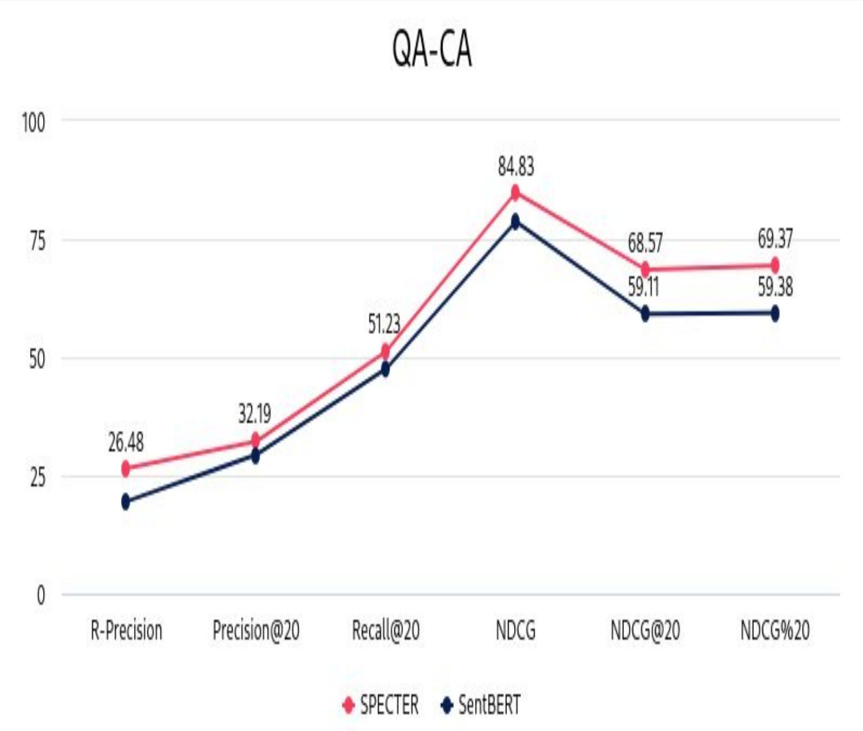
Used GPU for faster inference in baselines.

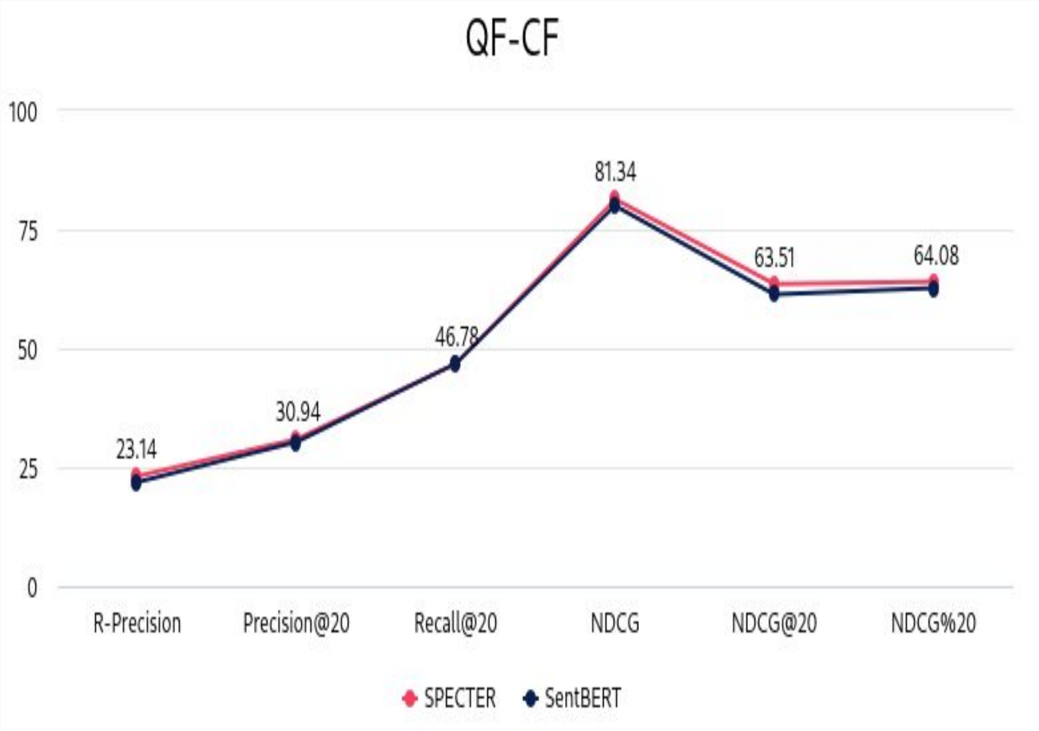
The result metrics (R-Precision, Recall@20 etc.) for each case are computed using the evaluation scripts provided by CSFCube authors.

Analysis

- SPECTER overperforms SciBERT on all metrics, as it uses the SciBERT under the hood and additionally add the citation network information.
- SciNCL has slightly better performance than SPECTER in general.
- SenBERT produces sentence level embeddings which doesn’t take context in account hence SPECTER performs better especially in QA-CA.
- Abstract level baselines performs better on QA-CA while sentence level performs better on QF-CA or QF-CF.
- Large drop in accuracy across faceted and non-faceted query for SciBERT while SciNCL and SPECTER perform consistently well.
- Good performance for the background facet as it contains generic description of a situation.

Analysis


QA-CA
100
75
50
25
0
26.48
32.19
51.23
84.83
68.57
59.11
69.37
59.38
R-Precision
Precision@20
Recall@20
NDCG
NDCG@20
NDCG%20
◆ SPECTER
◆ SentBERT


QF-CF
100
75
50
25
0
23.14
30.94
46.78
81.34
63.51
64.08
R-Precision
Precision@20
Recall@20
NDCG
NDCG@20
NDCG%20
◆ SPECTER
◆ SentBERT

FUTURE WORK

- Use SPECTER and SciNCL architecture and add a few feedforward layers on top. Finetune them using co-citation data from SciDocs.
- Use PARADE dataset, which consist of paraphrase identification pairs to improve the sentence level baselines.
- Setup the SSC Model to possibly use it for sentence level baseline models on dataset which only contains abstract.