ADTB Project: Semantic Search using PostgreSQL

Konstantinos Giantsios
giantsik@csd.auth.gr

Contents

1 Data

The data that were utilized to populate the database are coming from the Semantic Scholar Academic Corpus [1]. In particular a subset of the Semantic Scholar Open Research Corpus (S2ORC) [1] was used. S2ORC is a large corpus of 81.1M English-language academic papers spanning many academic disciplines. Rich metadata, paper abstracts, resolved bibliographic references, as well as structured full text for 8.1M open access papers. For the construction of our subset we have taken only 3k papers that are written in english, have abstract and have a link to open access repository to download the full text.

A slight prepossessing has been applied before I feed forward them to the model to get the semantic vectors and upload them to the database. The text from the title and abstract was lower-cased and cleaned from any control character ('\n', '\t') and latex remnants. Moreover, I have downloaded the full text for only 1k papers to save space. I uploaded to the database

[1] https://www.semanticscholar.org/product/api

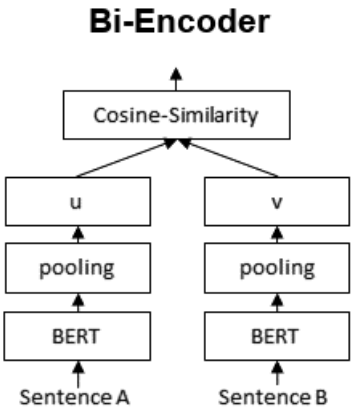

Figure 1: Illustration of a Bi-encoder. [2]

only those papers that have full text. Finally, I have chosen to keep only 4 fields: id, title, abstract, local_link, where local link is the local path to the full text of the paper. The full text of the paper is in pdf format. You can find

2 Populating DB

2.1 Creation of semantic vectors

A well known method to extract semantic vectors from a text is by encoding your text using an model. Bi-Encoders produce for a given text an embedding. In Figure 1 they pass to a BERT independently the sentences A and B, which result in the sentence embeddings u and v. These sentence embedding can then be compared using cosine similarity.

Bi-Encoders are used whenever you need a sentence embedding in a vector space for efficient comparison. Applications are for example Information Retrieval / Semantic Search or Clustering. With a Bi-Encoder, you compute the embedding for each sentence, which takes only 5 seconds [2].

For the creation of our embeddings/vectors we used the all-mpnet-base-v2 model [2]. This model is an all-round model tuned for many use-cases. Trained on a large and diverse dataset of over 1 billion training pairs. Specifically, the pairs included a lot of with academic context and it ranked second in the semantic search performance benchmark [3] held by sbert framework [4], this is the reason why I chose it. Moreover, this model is based on mpnet-base [5]. MPNet is a novel pre-training method that inherits the advantages of BERT and XLNet and avoids their limitations. MPNet leverages the dependency among predicted tokens through permuted language modeling (vs. MLM in BERT), and takes auxiliary position information as

[2] https://huggingface.co/sentence-transformers/all-mpnet-base-v2
[3] https://www.sbert.net/docs/pretrained_models.html
[4] https://www.sbert.net/index.html
[5] https://huggingface.co/microsoft/mpnet-base

Model	Factorization
MLM	$\log P(\text{sentence} \mid \text{the task is [M] [M]}) + \log P(\text{classification} \mid \text{the task is [M] [M]})$
PLM	$\log P(\text{sentence} \mid \text{the task is}) + \log P(\text{classification} \mid \text{the task is sentence})$
MPNet	$\log P(\text{sentence} \mid \text{the task is [M] [M]}) + \log P(\text{classification} \mid \text{the task is sentence [M]})$

Figure 2: An example sentence "the task is sentence classification" to illustrate the conditional information of MLM, PLM and MPNet. [3]

```
sem_search=# SELECT column_name FROM information_schema.columns WHERE TABLE_NAME = 'Paper';
 column_name
-------------
 embedding
 id
 title
 abstract
 local_link
(5 rows)
```

Figure 3: Table Paper Schema

input to make the model see a full sentence and thus reducing the position discrepancy (vs. PLM in XLNet) [3]. In Figure 2 you observe the illustration of conditional information of MLM, PLM and MPNet.

2.2 Populating DB and querying

Furthermore, I concatenated the title and abstract and feed-forward the text that was created to get the corresponding embeddings/vectors for every paper. I populated the PostgreSQL database with the 1k papers using 5 fields: id, title, abstract, local_link and embedding, as you can see in Figure 3. Every embedding has 768 dimensions and is normalized.

For the storing of the embedding the creation of my database I have utilised the pgvector plugin [6] [7], which was based in PASE [4]. Using pgvector, I could create an indexing scheme called IVFFlat. IVFFlat uses a clustering algorithm such as k-means to divide vectors in the high-dimensional data space into clusters based on implicit clustering properties. Each cluster has a centroid. IVFFlat traverses the centroids of all clusters to identify the n centroids that are nearest to the vector you want to query. IVFFlat traverses and sorts all vectors in the clusters to which the identified n centroids belong. Then, IVFFlat obtains the nearest k vectors. I have decided to use the docker version of pgvector to be more manageable.

For the querying part, given a query and a k, the app first encodes the query to a vector and then makes a query call to get the k nearest neighbours based on the embedding column. The similarity is calculated using the dot-product. After the retrieval of the k nearest neighbours a re-ranking is applied using a different distance function given by the user. Obviously, if the user selects the dot-product as the scoring function, then no re-ranking is applied. List of

[6] https://github.com/pgvector/pgvector
[7] https://github.com/pgvector/pgvector-python

re-ranking distance functions:

- Bray-Curtis distance $= \sum |u_i - v_i| / \sum |u_i + v_i|$
- Canberra distance $= \sum_i \frac{|u_i - v_i|}{|u_i| + |v_i|}$
- Chebyshev distance $= max_i |u_i - v_i|$
- City Block (Manhattan) distance $= \sum_i |u_i - v_i|$
- Cosine distance $= 1 - \frac{u \cdot v}{||u||_2 ||v||_2}$
- Euclidean distance $= ||u - v||_2$

Finally, I have added the capability to return an explanation (the lines from the texts that contain the corresponding terms / keywords that match what the user user) utilising fuzzy matching with a high threshold (0.9) and taking the neighbouring words (span) of the matched word.

3 Web application

To facilitate the user, I have created an API using the FastAPI [8] python library. To access the User Interface the user have to go to this address: http://127.0.0.1:8000/docs, assuming that you run the app locally.

First, the user will see the starting page of the UI, as you can see in Figure 4. Here, the user has 3 options. The first one is for health checking. The second one is for the searching according to a given query. The third one is for opening the full text of a paper, given a local path.

For the searching/querying in UI, shown in Figure 5 the user will need to click on /search, then click the Try Out button and then the user will insert the query text, the k nearest papers that he wants the app to return and the distance metric for re-ranking. The user can read the results if he scrolls down, as shown in Figure 6. The app returns the relevant papers showing the title, the local path to the full text and an explanation as described previously. Finally, if the user wants to read the full text, he can click on /open_file, then click on Try Out button and insert the local path, as shown in Figure 7. The app will open a new tab in the default browser with the full text of the paper.

4 Installation

It is recommended to have a Linux-based OS, Python 3.7+ and docker.

1. Clone Github repo:

```
git clone https://github.com/CoGian/semantic-search-engine.git
```

[8] https://fastapi.tiangolo.com/

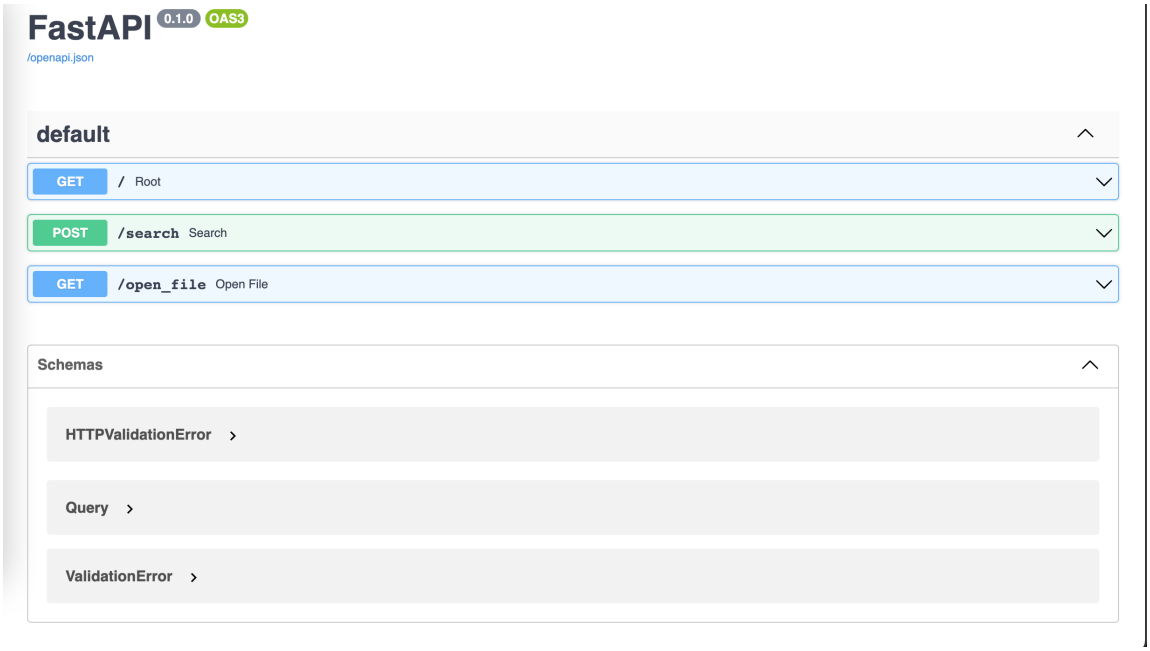

Figure 4: UI starting page

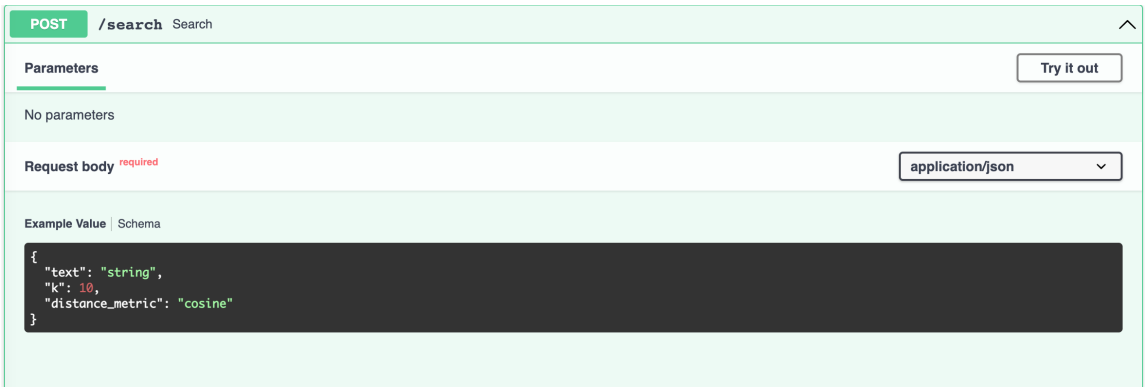

Figure 5: Search/Querying in app

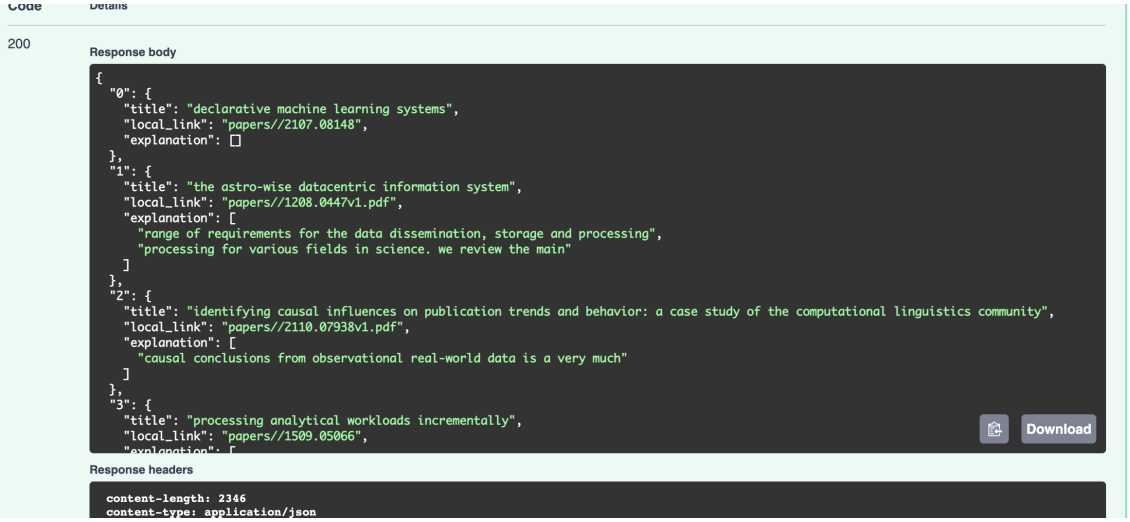

Figure 6: Results from search/querying in app

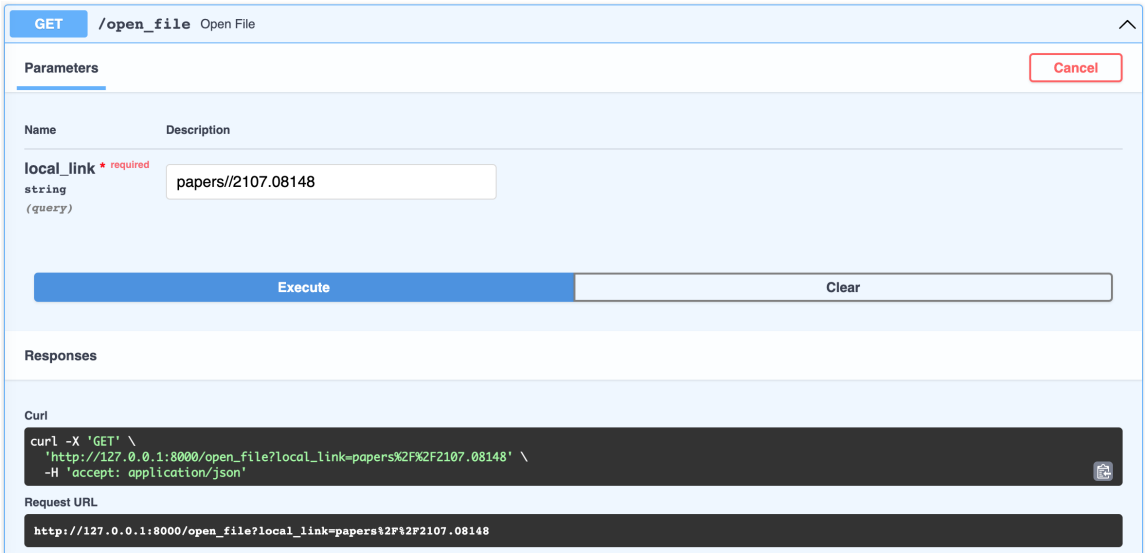

Figure 7: Open file using the app

2. Download & install PostgreSQL
3. Change directory to the repo's directory

   ```
   cd semantic-search-engine
   ```

4. Download postgresql dump and and install all the dependencies needed :

   ```
   bash install.sh
   ```

5. Open terminal inside docker container:

   ```
   docker exec -it sem_search_postgres bash
   ```

6. Connect to postgres:

   ```
   psql -U postgres
   ```

7. Create DB and close connection:

   ```
   CREATE DATABASE sem_search;
   ```

8. Load psql dump and exit docker's container:

   ```
   psql -U postgres sem_search < sem_search_export.pgsql
   ```

9. Run application:

   ```
   source venv/bin/activate
   uvicorn src.api:app --reload
   ```

5 Experiments

Query = "**machine learning deep learning natural language processing**"

5.1 Results for k=5

- distance=dot_product


```
{
  "0": {
    "title": "declarative machine learning systems",
    "local_link": "papers//2107.08148",
    "explanation": [
      "declarative machine learning systems the future"
    ]
  },
  "1": {
    "title": "neural speed reading with structural-jump-lstm",
```

```
    "local_link": "papers//1904.00761v2.pdf",
    "explanation": [
      "neural networks (rnns) can model natural language by sequentially '
          reading'"
    ]
  },
  "2": {
    "title": "pangu-a: large-scale autoregressive pretrained chinese
        language models with auto-parallel computation",
    "local_link": "papers//2104.12369",
    "explanation": [
      "pangu-a: large-scale autoregressive pretrained chinese language
          models with auto-parallel computation",
      "with auto-parallel computation large-scale pretrained language
          models (plms) have become",
      "become the new paradigm for natural language processing (nlp). plms"
          ,
      "have demonstrated strong performances on natural language
          understanding and generation",
      "and generation with \\textit{few-shot in-context} learning. in this
          work, we",
      "practice on training large-scale autoregressive language models
          named pangu-, with"
    ]
  },
  "3": {
    "title": "parts-of-speech tagger errors do not necessarily degrade
        accuracy in extracting information from biomedical text",
    "local_link": "papers//0804.0317",
    "explanation": [
      "of muscorian, a generic text processing tool for extracting protein-
          protein",
      "comparable performance to biomedical-specific text processing tools.
           this result was"
    ]
  },
  "4": {
    "title": "clar: a cross-lingual argument regularizer for semantic role
        labeling",
    "local_link": "papers//2011.04732v1.pdf",
    "explanation": [
      "a given sentence. although different languages have different
          argument annotations,",
      "share common semantic meaning across languages (e.g. adjuncts have
```

```
          more",
      "such linguistic annotation similarity across languages and exploits
          this information",
      "information to map the target language arguments using a
          transformation",
      "the space on which source language arguments lie. by doing",
      "improves srl performance on multiple languages over monolingual and
          polyglot"
    ]
  }
}
```


- distance=cosine, same as previous
- distance=euclidean, same as previous
- distance=braycurtis, same as previous
- distance=canberra Changed from dot-product results,
  position in dot product : position in camberra

  0:0, 3:1, 2:2, 4:3, 1:4
- distance=chebyshev Changed from dot-product results,
  position in dot product : position in chebyshev

  0:0, 3:1, 2:2, 1:3, 4:4
- distance=cityblock, same as dot product results

In my opinion, I believe that the only irrelevant paper that has returned as result is the first one, because it doesn't refer to natural language processing. Therefore, we can agree that we have a 80% precision.

5.2 Results for k=10

- distance=dot_product


```
{
  "0": {
    "title": "declarative machine learning systems",
    "local_link": "papers//2107.08148",
    "explanation": [
      "declarative machine learning systems the future"
    ]
  },
  "1": {
    "title": "neural speed reading with structural-jump-lstm",
```

```
    "local_link": "papers//1904.00761v2.pdf",
    "explanation": [
      "neural networks (rnns) can model natural language by sequentially '
          reading'"
    ]
  },
  "2": {
    "title": "pangu-a: large-scale autoregressive pretrained chinese
        language models with auto-parallel computation",
    "local_link": "papers//2104.12369",
    "explanation": [
      "pangu-a: large-scale autoregressive pretrained chinese language
          models with auto-parallel computation",
      "with auto-parallel computation large-scale pretrained language
          models (plms) have become",
      "become the new paradigm for natural language processing (nlp). plms"
          ,
      "have demonstrated strong performances on natural language
          understanding and generation",
      "and generation with \\textit{few-shot in-context} learning. in this
          work, we",
      "practice on training large-scale autoregressive language models
          named pangu-, with"
    ]
  },
  "3": {
    "title": "parts-of-speech tagger errors do not necessarily degrade
        accuracy in extracting information from biomedical text",
    "local_link": "papers//0804.0317",
    "explanation": [
      "of muscorian, a generic text processing tool for extracting protein-
          protein",
      "comparable performance to biomedical-specific text processing tools.
           this result was"
    ]
  },
  "4": {
    "title": "clar: a cross-lingual argument regularizer for semantic role
        labeling",
    "local_link": "papers//2011.04732v1.pdf",
    "explanation": [
      "a given sentence. although different languages have different
          argument annotations,",
      "share common semantic meaning across languages (e.g. adjuncts have
```

```
        more",
    "such linguistic annotation similarity across languages and exploits
        this information",
    "information to map the target language arguments using a
        transformation",
    "the space on which source language arguments lie. by doing",
    "improves srl performance on multiple languages over monolingual and
        polyglot"
  ]
},
"5": {
  "title": "opinion mining of text documents written in macedonian
      language",
  "local_link": "papers//1411.4472",
  "explanation": [
    "text documents written in macedonian language the ability to extract
        ",
    "and surveys. in this paper machine learning techniques are used",
    "which are written in macedonian language. the posts are classified"
  ]
},
"6": {
  "title": "training a restricted boltzmann machine for classification by
      labeling model samples",
  "local_link": "papers//1509.01053v1.pdf",
  "explanation": [
    "training a restricted boltzmann machine for classification by
        labeling",
    "classification performance competitive to semi-supervised learning
        if we first train"
  ]
},
"7": {
  "title": "identifying causal influences on publication trends and
      behavior: a case study of the computational linguistics community",
  "local_link": "papers//2110.07938v1.pdf",
  "explanation": [
    "trending tasks and techniques (e.g., deep learning, embeddings,
        generative, and",
    "china on propensity of researching languages beyond english, and the
        "
  ]
},
"8": {
```

```
    "title": "determining whether the non-protein-coding dna sequences are
        in a complex interactive relationship by using an artificial
        intelligence method",
    "local_link": "papers//1708.04019v1.pdf",
    "explanation": [
      "ai research have enabled automatically learning representations of
         high dimensional",
      "data without feature engineering, using deep neural networks.
         therefore, in",
      "whether a dna sequence is natural or artificial. the trained"
    ]
  },
  "9": {
    "title": "deep poetry: a chinese classical poetry generation system",
    "local_link": "papers//1911.08212v1.pdf",
    "explanation": [
      "deep poetry: a chinese classical",
      "classical poetry generation system called deep poetry. existing
         systems for",
      "multi-modal input. unlike previous systems, deep poetry uses neural
         networks"
    ]
  }
}
```


- distance=cosine, same as previous
- distance=euclidean, same as previous
- distance=braycurtis Changed from dot-product results,
  position in dot product : position in braycurtis

  0:0, 1:1, 2:2, 3:3, 4:4, 6:5, 5:6, 8:7, 9:8, 7:9
- distance=canberra Changed from dot-product results,
  position in dot product : position in camberra

  0:0, 3:1, 2:2, 4:3, 1:4, 6:5, 5:6, 8:7, 9:8, 7:9
- distance=chebyshev

  Changed from dot-product results,
  position in dot product : position in chebyshev

  0:0, 3:1, 2:2, 1:3, 6:4, 9:5, 7:6, 5:7, 8:8, 4:9
- distance=cityblock

  Changed from dot-product results,
  position in dot product : position in cityblock

0:0, 1:1, 2:2, 3:3, 4:4, 6:5, 5:6, 7:7, 9:8, 8:9

In my opinion, I believe that here we have 3 irrelevant papers that have returned as result, because they don't refer to natural language processing. Therefore, we can agree that we have a 70% precision. Furthermore, we can observe that changing the distance metric, can change the ranking of the returned results.

5.3 Results for k=20

- distance=dot_product


```
{
  "0": {
    "title": "declarative machine learning systems",
    "local_link": "papers//2107.08148",
    "explanation": [
      "declarative machine learning systems the future"
    ]
  },
  "1": {
    "title": "neural speed reading with structural-jump-lstm",
    "local_link": "papers//1904.00761v2.pdf",
    "explanation": [
      "neural networks (rnns) can model natural language by sequentially '
          reading'"
    ]
  },
  "2": {
    "title": "pangu-a: large-scale autoregressive pretrained chinese
        language models with auto-parallel computation",
    "local_link": "papers//2104.12369",
    "explanation": [
      "pangu-a: large-scale autoregressive pretrained chinese language
          models with auto-parallel computation",
      "with auto-parallel computation large-scale pretrained language
          models (plms) have become",
      "become the new paradigm for natural language processing (nlp). plms"
          ,
      "have demonstrated strong performances on natural language
          understanding and generation",
      "and generation with \\textit{few-shot in-context} learning. in this
          work, we",
      "practice on training large-scale autoregressive language models
          named pangu-, with"
    ]
```

```
  },
  "3": {
    "title": "parts-of-speech tagger errors do not necessarily degrade
       accuracy in extracting information from biomedical text",
    "local_link": "papers//0804.0317",
    "explanation": [
      "of muscorian, a generic text processing tool for extracting protein-
         protein",
      "comparable performance to biomedical-specific text processing tools.
          this result was"
    ]
  },
  "4": {
    "title": "clar: a cross-lingual argument regularizer for semantic role
       labeling",
    "local_link": "papers//2011.04732v1.pdf",
    "explanation": [
      "a given sentence. although different languages have different
         argument annotations,",
      "share common semantic meaning across languages (e.g. adjuncts have
         more",
      "such linguistic annotation similarity across languages and exploits
         this information",
      "information to map the target language arguments using a
         transformation",
      "the space on which source language arguments lie. by doing",
      "improves srl performance on multiple languages over monolingual and
         polyglot"
    ]
  },
  "5": {
    "title": "opinion mining of text documents written in macedonian
       language",
    "local_link": "papers//1411.4472",
    "explanation": [
      "text documents written in macedonian language the ability to extract
         ",
      "and surveys. in this paper machine learning techniques are used",
      "which are written in macedonian language. the posts are classified"
    ]
  },
  "6": {
    "title": "training a restricted boltzmann machine for classification by
       labeling model samples",
```

```
    "local_link": "papers//1509.01053v1.pdf",
    "explanation": [
      "training a restricted boltzmann machine for classification by
         labeling",
      "classification performance competitive to semi-supervised learning
         if we first train"
    ]
  },
  "7": {
    "title": "identifying causal influences on publication trends and
       behavior: a case study of the computational linguistics community",
    "local_link": "papers//2110.07938v1.pdf",
    "explanation": [
      "trending tasks and techniques (e.g., deep learning, embeddings,
         generative, and",
      "china on propensity of researching languages beyond english, and the
         "
    ]
  },
  "8": {
    "title": "determining whether the non-protein-coding dna sequences are
       in a complex interactive relationship by using an artificial
       intelligence method",
    "local_link": "papers//1708.04019v1.pdf",
    "explanation": [
      "ai research have enabled automatically learning representations of
         high dimensional",
      "data without feature engineering, using deep neural networks.
         therefore, in",
      "whether a dna sequence is natural or artificial. the trained"
    ]
  },
  "9": {
    "title": "deep poetry: a chinese classical poetry generation system",
    "local_link": "papers//1911.08212v1.pdf",
    "explanation": [
      "deep poetry: a chinese classical",
      "classical poetry generation system called deep poetry. existing
         systems for",
      "multi-modal input. unlike previous systems, deep poetry uses neural
         networks"
    ]
  },
  "10": {
```

```
    "title": "attention module is not only a weight: analyzing transformers
        with vector norms",
    "local_link": "papers//2004.10102",
    "explanation": [
      "recently achieved considerable success in natural language
         processing. hence, attention",
      "bert and a transformer-based neural machine translation system
         include the"
    ]
  },
  "11": {
    "title": "glass-box program synthesis: a machine learning approach",
    "local_link": "papers//1709.08669",
    "explanation": [
      "glass-box program synthesis: a machine learning approach recently
         proposed",
      "inspired by number theory, text processing, and algebra. our results
         "
    ]
  },
  "12": {
    "title": "federated learning for intrusion detection system: concepts,
       challenges and future directions",
    "local_link": "papers//2106.09527v1.pdf",
    "explanation": [
      "federated learning for intrusion detection system:",
      "and privacy of such devices. machine learning and deep learning",
      "server. on the contrary, federated learning (fl) fits in
         appropriately",
      "appropriately as a privacy-preserving decentralized learning
         technique that does not"
    ]
  },
  "13": {
    "title": "data science with vadalog: bridging machine learning and
       reasoning",
    "local_link": "papers//1807.08712",
    "explanation": [
      "data science with vadalog: bridging machine learning and reasoning
         following",
      "corporate knowledge and to draw deep insights using machine learning
         ",
      "data science, typically based on machine learning and statistical
         modelling,",
```

```
      "significant step forward towards combining machine learning and
          reasoning in"
    ]
  },
  "14": {
    "title": "object relational graph with teacher-recommended learning for
        video captioning",
    "local_link": "papers//2002.11566",
    "explanation": [
      "object relational graph with teacher-recommended learning for video
          captioning taking",
      "information from both vision and language is critical for the",
      "meanwhile, we design a teacher-recommended learning (trl) method to
          make",
      "use of the successful external language model (elm) to integrate"
    ]
  },
  "15": {
    "title": "neural feature selection for learning to rank",
    "local_link": "papers//2102.11345v1.pdf",
    "explanation": [
      "neural feature selection for learning to rank learning to",
      "of information retrieval (ir) where machine learning models are
          employed"
    ]
  },
  "16": {
    "title": "3d scene parsing via class-wise adaptation",
    "local_link": "papers//1812.03622",
    "explanation": []
  },
  "17": {
    "title": "tomayto, tomahto. beyond token-level answer equivalence for
       question answering evaluation",
    "local_link": "papers//2202.07654v1.pdf",
    "explanation": []
  },
  "18": {
    "title": "fast domain adaptation for neural machine translation",
    "local_link": "papers//1612.06897",
    "explanation": [
      "fast domain adaptation for neural machine translation neural machine
          translation",
      "of text from one human language into another. the basic",
```

```
      "tasks/benchmarks at least for some language pairs. however, many of",
      "subjective evaluation metric on two language pairs. with our adaptation"
    ]
  },
  "19": {
    "title": "trepan reloaded: a knowledge-driven approach to explaining artificial neural networks",
    "local_link": "papers//1906.08362v2.pdf",
    "explanation": []
  }
}
```


- distance=cosine, same as dot product results
- distance=euclidean, same as dot product results
- distance=braycurtis Changed from dot-product results,
  position in dot product : position in braycurtis

  0:0, 1:1, 2:2, 3:3, 4:4, 6:5, 5:6, 8:7, 9:8, 10:9, 14:10, 7:11, 11:12, 12:13, 17:14, 13:15, 16:16, 15:17, 18:18, 19:19
- distance=canberra Changed from dot-product results,
  position in dot product : position in canberra

  0:0, 3:1, 2:2, 4:3, 1:4, 6:5, 5:6, 14:7, 8:8, 10:9, 9:10, 12:11, 7:12, 16:13, 17:14, 11:15, 15:16, 13:17, 18:18, 19:19
- distance=chebyshev

  Changed from dot-product results,
  position in dot product : position in chebyshev

  0:0, 3:1, 2:2, 19:3, 15:4, 11:5, 18:6, 13:7, 1:8, 12:9, 6:10, 9:11, 7:12, 16:13, 14:14, 5:15, 8:16, 4:17, 17:18, 10:19
- distance=cityblock

  Changed from dot-product results,
  position in dot product : position in cityblock

  0:0, 1:1, 2:2, 3:3, 4:4, 6:5, 5:6, 10:7, 7:8, 9:9, 8:10, 14:11, 12:12, 17:13, 11:14, 16:15, 13:16, 15:17, 18:18, 19:19

In my opinion, I believe that here we have 8 irrelevant papers that have returned as result, because they don't refer to natural language processing. Therefore, we can agree that we have a 60% precision. Furthermore, we can observe that changing the distance metric, can change even more the ranking of the returned results.

References

[1] Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Michael Kinney, and Daniel S. Weld. S2orc: The semantic scholar open research corpus. In *ACL*, 2020.

[2] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks, 2019.

[3] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mpnet: Masked and permuted pre-training for language understanding. *CoRR*, abs/2004.09297, 2020.

[4] Wen Yang, Tao Li, Gai Fang, and Hong Wei. Pase: Postgresql ultra-high-dimensional approximate nearest neighbor search extension. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, SIGMOD '20, page 2241–2253, New York, NY, USA, 2020. Association for Computing Machinery.