Design of a Knowledge Hub of Heterogeneous Multisource Documents to support Public Authorities

Paolo Tagliolato Acquaviva D'Aragona [1,*,†], Lorenza Babbini [2,†], Gloria Bordogna [1, †], Alessandro Lotti[2, †], Annalisa Minelli[2, †] and Alessandro Oggioni [1, †]

[1] CNR – IREA , via Corti 12, Milano, 20133, Italy
[2] INFO/RAC UNEP-MAP c/o ISPRA, DG-SINA , via Vitaliano Brancati 48, Roma, 00144, Italy

Abstract

This contribution outlines the design of a Knowledge Hub of heterogeneous documents related to the Mediterranean Action Plan UNEP-MAP of the United Nations Environment Program [1]. The Knowledge Hub is intended to serve as a resource to assist public authorities and users with different backgrounds and needs in accessing information efficiently. Users can either formulate natural language queries or navigate a knowledge graph automatically generated to find relevant documents. The Knowledge Hub is designed based on state-of-the-art Large Language Models. (LLMs) A user-evaluation experiment was conducted, testing publicly available models on a subset of documents using distinct LLMs settings. This step was aimed to identify the best-performing model for further using it to classify the documents with respect to the topics of interest.

Keywords
Knowledge Hub, Large Language Models, Natural Language Queries, Knowledge graph.

1. Introduction

This contribution reports the feasibility study carried out for the design of a Knowledge Hub (KH) for accessing documents, which is part of the Knowledge Management Platform (KMaP), a platform constituting the unique access point of all knowledge heritage for the United Nations Environmental Program for the Mediterranean Action Plan (UNEP-MAP) [1].

The KH is conceived as an access point to highly heterogeneous multimedia documents distributed on the Web, among the network of United Nations Environmental Program for the Mediterranean Action Plan, about marine studies, political and economic directives, environmental studies and in general as part of UNEP-MAP protocols and activities. For the nature of the contents dealt with in the documents, the hub constitutes a knowledge base for the stakeholders of the Mediterranean Action Plan: The interested public authorities have users with different background knowledge and needs, including politicians, administrators, environmental scientists, projects leaders and citizens, who need to search as well as to navigate the distributed archive.

During the use case analysis, carried out by interviews to some potential stakeholders, it was deemed important that the KH would support users to perform searches by formulating queries in natural language, and would guide them in navigating the collection by providing an organized view of the documents into topics of interest [2].

To this aim, main critical aspects had to be considered to provide feasible solutions: the document collection is highly heterogeneous as far as the genre, some being minutes of meetings while

0000-0002-0261-313X (P. Tagliolato); 0000-0003-3302-6891 (L. Babbini); 0000-0002-6775-753X (G. Bordogna); 0000-0002-4837-4357 (A. Lotti); 0000-0003-1772-0154 (A. Minelli); 0000-0002-7997-219X (A. Oggioni)

others being scientific reports, with highly variable lengths, some documents being of one page while others being reports of hundred pages, in different languages with varying formats (mostly being in *pdf* others in *html* and *jpg*). Finally, the identification of the topics made during the use case analysis revealed that it is not so easy to tell apart which documents belong to a topic, being some of them at the cross-road of several topics.

The approach that we deemed flexible to apply for enabling natural language searches was identified as an Information Retrieval system [3] defined based on Large Language Models (LLMs), and specifically on open source pre-trained LLMs [4].

To aid the organization of the documents into the topics we then retrieved natural language descriptions of the topics by simple keywords and conceived these are natural language queries to be submitted to the collection represented in a continuous bag of words space of a pretrained LLM.

This way, all documents belongs to the topics with a distinct relevance rank. This allowed to build a knowledge graph in which each node represents the ranked list of a topic and each edge between a pair of nodes represents the fuzzy intersection list of the two ranked lists [5].

A user-evaluation experiment was conducted, testing publicly available LLMs on a subset of documents using distinct settings. This step aimed to identify the best-performing model for further using it for both implementing the information retrieval module answering natural language queries and classifying documents with respect to the topics. The paper reports the steps of design of the KH and its evaluation experiment for selecting the best model to be applied in the future for documents' classification into topics.

2. Knowledge Hub design

The first activity performed was the harvesting of the documents from several potential sources of interest. To this end we relied on the knowledge of a group of experts of the leading institution ISPRA.

2.1. Harvesting Documents' Collection

This step was aimed at identifying the documents sources, i.e., the web sites and archives with potentially interesting documents and at carrying out their characterization with respect to some meaningful dimensions [5].
The documents in these information sources are more than 10000, mainly files, and most of them are in PDF

format. Most information sources (20 out of 24) contain documents, and 3 of these resources also share images and tables, while only 3 out of 24 provide geographical layers. As far as the resources are concerned, they are dedicated to 3 themes: law, regulation and management of the sea (13 out of 24), pollution (7) and biodiversity (2). Finally, 21 of the classified repositories are open to the public, while the remaining 3 are private or have restricted access. From Regional Marine Pollution Emergency Response Centre for the Mediterranean Sea [6], Regional Activity Centre for Specially Protected Areas [7], Regional Activity Centre for Sustainable Consumption and Production [8], Priority Actions Programme/Regional Activity Centre [9], UNEP-MAP library [10] and UNEP library where the author was marked as UNEP-MAP [11], we harvested, through website scraping, all the documents.
For document harvesting, some code has been developed both by CNR-IREA in the R language and from INFO-RAC in Python language [26] freely available under GNU GPL license.
To share the files produced for the harvesting process, a GitHub repository was created [27]. The "scraping" folder contains the R and Python scripts developed for scraping, the output of these files is in the "results" folder.

2.2. Strategies for enabling documents search

Once the collection was available, the methods of representation and indexing of their content have been selected.
It was decided to experiment an up-to-date solution based on state-of-the-art "semantic" indexing methods using continuous bag of words [4]. By this approach users have complete freedom to formulate natural language queries or keywords' queries. In this case the documents are retrieved if their contents are "semantically" close to those of the query.
To this end we experimented several LLMs available publicly on hugging face library [12]. All these models imply the representation and management of the "semantics" of information in a document corpus which has been provided as training set. It must be pointed out that, in this context, the term "semantics" is improperly used since the LLMs identify regular patterns in texts based on heuristic statistical inference; thus, instead of "semantics", the term "relatedness" would be more appropriate. This way they learn how to predict missing words in a sentence, or how to continue a sentence, or to answer a query, and, finally, to retrieve relevant documents in an ad

hoc retrieval task activated by a user query. Such "semantics" models are the most effective in the case one wants a natural language querying interaction, since they can retrieve documents which do not contain the specific query words, but synonymous terms or concepts related with the query concepts.

In our context this approach was the most feasible since we did not have available thesauri for expanding the meaning of terms in the documents, being the documents heterogeneous as far as both their themes and genre. To this end, we have chosen pretrained LLMs that have been set up for the ad hoc retrieval task and based on evolutions of BERT, Bidirectional Encoder Representations from Transformers [13][14] which is the Google state-of-the-art model using a transformer architecture [13], a deep neural network, with self-attention mechanisms, that allows to keep the context of words into account when creating their representation as embeddings, i.e., as vectors of continuous numeric values in a latent semantic space. Once the LLMs have been selected, we defined the architecture of the KH by specifying the preprocessing phase that our corpus of documents should undergo to become a readable input to the selected models. The formats of the input documents, should be simple text with punctuation marks allowing the identification of single words, i.e., tokens; of sentences, ending with punctuation marks like full stop or semicolon, etc.; and of paragraphs, starting with a new line. So, the non-conforming documents consisting in pdf files had to be "translated" into text.

Furthermore, the processing steps have been identified which has implied the selection of the implementation libraries and environment in order to code the whole process.

We experimented hybridized techniques, for example, the contents of queries and documents was represented by applying different embedding methods, and the same for the ranking of documents using different similarity measures.

Finally, we identified the most suitable open software for the implementation of the components, the indexing, the retrieval and the classification components of the KH.

Considering that there are a number of open source IR libraries after a review we selected *SentenceTransformer* python framework [15] that makes several Hugging face pretrained models available for sentence embeddings, and we exploited also the python library *NLTK* (Natural Language Toolkit [16]) for managing corpus documents and different tokenization strategies (i.e. the aforementioned subdivisions of documents into chunks, i.e., words, sentences, paragraphs or even n-grams sentences, paragraphs, etc.). For our purposes we deemed meaningful to compute different combinations of pretrained LLMs, documents representations based on different chunks definitions, and matching function either dot product or cosine similarity. Since documents may contain several chunks depending on their length, we experimented several aggregation functions of the chunks relevance scores to compute the overall document relevance score, i.e., the document ranking score. Specifically, we applied a K-NN algorithm aggregation function by increasing the number of the most relevant chunks and by using as metrics the fuzzy document cardinality measure [17].

We have selected the following pre-trained LLMs based on sentence-transformer architectures:

(a) *msmarco-distilbert-cos-v5* [18]: it maps sentences & paragraphs to a 768-dimensional dense vector space and was designed for semantic search. It has been trained on 500k (query, answer) pairs from the MS MARCO Passages dataset(Microsoft Machine Reading Comprehension) which is a large scale dataset focused on machine reading comprehension, question answering, and passage ranking.

(b) *all-MiniLM-L6-v2* [19]: it maps sentences & paragraphs to a 384-dimensional dense vector space and can be used for tasks like clustering or semantic search.

(c) *msmarco-roberta-base-ance-firstp* [20]: this is a port of the ANCE FirstP Model, which uses a training mechanism to select more realistic negative training instances to the sentence-transformers model: it maps sentences & paragraphs to a 768-dimensional dense vector space and can be used for tasks like clustering or semantic search.

(d) *msmarco-bert-base-dot-v5* [21]: it maps sentences & paragraphs to a 768-dimensional dense vector space and was designed for semantic search. It has been trained on 500K (query, answer) pairs from the MS MARCO dataset.

(e) *msmarco-distilbert-base-tas-b* [22]: it is a port of the DistilBert TAS-B Model to sentence-transformers model: It maps sentences & paragraphs to a 768-dimensional dense vector space and is optimized for the task of semantic search.

2.3. Documents classification into topics

As far as the classification of the document corpus into the topics, during the use case analysis the topics were first identified by the seven keywords accounted for

in the UNESCO thesaurus [23], an RDF SKOS concept scheme without definitions, as reported in table 1.

Then we identified "definitions" of each topic keyword in renowned and authoritative sources as reported in table 1, i.e., open domain websites, in the form of textual abstracts. We then enriched the pre-existing thesaurus by adding those definitions in the web of data. The result is available both as linked data and through a SPARQL endpoint [28].

Table 1:
Topics keywords and sources for their definitions as short abstracts

Topics keyword	Definitions Source
Climate change	United Nations
Marine biodiversity	UN
Sustainability and blue economy	UN
Pollution	National Geographic
Marine spatial planning	EU commission
Fishery and aquaculture	FAO
Governance	UN Dev. Progr.

After choosing the best performing model evaluated as explained in the next section, we applied it to classify the whole collection into the topics, by considering the topics' definitions as queries. This way a document can be assigned to multiple topics to a different extent, where in the extent is the relevance score with respect to a topic. The fuzzy intersection of a pair of ranked lists yielded by two topics (computed by their minimum) is the ranked list of documents at the cross-road of both the topics.

This way a knowledge graph can be built in which the nodes are the ranked list of the single topics while the edges are the ranked lists of documents at the cross-road of pairs of topics.

3. User Evaluation Experiment

We have set up an evaluation experiment of the different LLMs by randomly selecting a subset of 50 documents of the collection, engaging 3 users with three distinct backgrounds (a physicist, an environmental scientist and a biologist) who read these documents and formulated 10-30 queries each and for each query identified the list of their respective relevant documents among the 50 ones.

We evaluated some metrics of retrieval effectiveness. For our purposes we deemed meaningful to compute mean Average Precision (mAP) [25] of different combinations of the 5 pretrained LLMs, documents representations based on different chunks definitions, i.e., sentence, fixed window size and paragraphs, and matching functions (cosine similarity and dot product). The results of the mAP for the tests are reported in the following tables. They differ for the computation of similarity. Table 2 corresponds to cosine similarity, while Table 3 to dot product similarity.

Table 2:
mAP for different LLMs/chunks and cosine similarity

model	chunk type	window size	#ch: 1 (sum)	#ch: 2 (sum)	#ch: 2 (avg)	#ch: 3 (sum)	#ch: 3 (avg)	#ch: 4 (sum)	#ch: 4 (avg)	#ch: 5 (sum)	#ch: 5 (avg)	#ch: 6 (sum)	#ch: 6 (avg)	#ch: 7 (sum)	#ch: 7 (avg)	#ch: 8 (sum)	#ch: 8 (avg)	#ch: 9 (sum)	#ch: 9 (avg)	#ch: 10 (sum)	#ch: 10 (avg)	#ch: all (sum)	#ch: all (avg)	virtual doc	max
(a)	sentence	384	0,60	0,61	0,61	0,64	0,64	0,63	0,63	0,63	0,63	0,63	0,63	0,62	0,62	0,62	0,62	0,62	0,62	0,62	0,62	0,36	0,48	0,54	0,64
	window		0,60	0,61	0,61	0,60	0,61	0,59	0,59	0,57	0,59	0,56	0,58	0,55	0,57	0,55	0,58	0,54	0,57	0,54	0,57	0,38	0,47	0,55	0,61
	paragraph		0,61	0,62	0,61	0,62	0,62	0,61	0,61	0,61	0,61	0,61	0,61	0,60	0,60	0,61	0,61	0,60	0,60	0,60	0,60	0,36	0,47	0,52	0,62
(b)	sentence	256	0,61	0,61	0,61	0,62	0,62	0,63	0,63	0,63	0,63	0,63	0,63	0,62	0,62	0,62	0,62	0,62	0,62	0,62	0,62	0,35	0,53	0,57	0,63
	window		0,61	0,64	0,64	0,62	0,63	0,61	0,62	0,60	0,61	0,59	0,60	0,59	0,60	0,58	0,61	0,58	0,61	0,58	0,62	0,37	0,51	0,60	0,64
	paragraph		0,62	0,64	0,64	0,63	0,63	0,64	0,64	0,63	0,63	0,63	0,63	0,63	0,63	0,62	0,62	0,62	0,62	0,62	0,62	0,35	0,52	0,55	0,64
(c)	sentence	512	0,58	0,59	0,59	0,60	0,60	0,59	0,59	0,59	0,59	0,58	0,58	0,57	0,57	0,56	0,56	0,56	0,56	0,55	0,55	0,28	0,46	0,53	0,60
	window		0,57	0,56	0,57	0,54	0,55	0,52	0,54	0,51	0,55	0,51	0,55	0,50	0,54	0,50	0,54	0,49	0,54	0,49	0,54	0,28	0,51	0,53	0,57
	paragraph		0,58	0,59	0,59	0,60	0,60	0,59	0,59	0,58	0,58	0,57	0,58	0,57	0,57	0,56	0,56	0,56	0,56	0,56	0,56	0,26	0,42	0,49	0,60
(d)	sentence	512	0,56	0,58	0,58	0,58	0,58	0,57	0,57	0,57	0,57	0,57	0,57	0,57	0,57	0,57	0,57	0,55	0,55	0,54	0,54	0,28	0,48	0,53	0,58
	window		0,52	0,55	0,56	0,56	0,55	0,54	0,55	0,52	0,54	0,53	0,55	0,52	0,55	0,52	0,54	0,50	0,54	0,50	0,54	0,28	0,48	0,54	0,56
	paragraph		0,57	0,58	0,58	0,57	0,57	0,57	0,57	0,57	0,57	0,56	0,56	0,55	0,55	0,55	0,55	0,54	0,54	0,54	0,54	0,26	0,49	0,49	0,58
(e)	sentence	512	0,60	0,61	0,61	0,62	0,62	0,64	0,64	0,63	0,63	0,64	0,64	0,63	0,63	0,63	0,63	0,63	0,63	0,62	0,62	0,28	0,47	0,57	0,64
	window		0,56	0,54	0,54	0,54	0,53	0,51	0,53	0,50	0,53	0,50	0,53	0,49	0,52	0,49	0,52	0,48	0,52	0,49	0,52	0,29	0,47	0,55	0,56
	paragraph		0,60	0,60	0,60	0,60	0,60	0,60	0,60	0,60	0,60	0,61	0,61	0,61	0,61	0,60	0,60	0,60	0,60	0,59	0,59	0,26	0,45	0,55	0,61
			0,62	0,64	0,64	0,64	0,64	0,64	0,64	0,63	0,63	0,64	0,64	0,63	0,63	0,63	0,63	0,63	0,63	0,62	0,62	0,38	0,53	0,60	0,64

Table 3:

mAP for different LLMs/chunks and dot-product

model	chunk type	window size	#ch: 1	#ch: 2 (sum)	#ch: 2 (avg)	#ch: 3 (sum)	#ch: 3 (avg)	#ch: 4 (sum)	#ch: 4 (avg)	#ch: 5 (sum)	#ch: 5 (avg)	#ch: 6 (sum)	#ch: 6 (avg)	#ch: 7 (sum)	#ch: 7 (avg)	#ch: 8 (sum)	#ch: 8 (avg)	#ch: 9 (sum)	#ch: 9 (avg)	#ch: 10 (sum)	#ch: 10 (avg)	#ch: all (sum)	#ch: all (avg)	max
(a)	sentence		0,60	0,61	0,61	0,64	0,64	0,63	0,63	0,63	0,63	0,63	0,63	0,62	0,62	0,62	0,62	0,62	0,62	0,62	0,62	0,36	0,48	0,64
	window	384	0,60	0,61	0,61	0,60	0,61	0,59	0,59	0,57	0,59	0,56	0,58	0,55	0,57	0,55	0,58	0,54	0,57	0,54	0,57	0,38	0,47	0,61
	paragraph		0,61	0,62	0,61	0,62	0,62	0,61	0,61	0,61	0,61	0,61	0,61	0,60	0,60	0,61	0,61	0,60	0,60	0,60	0,60	0,36	0,47	0,62
(b)	sentence		0,61	0,61	0,61	0,62	0,62	0,63	0,63	0,63	0,63	0,63	0,63	0,62	0,62	0,62	0,62	0,62	0,62	0,62	0,62	0,35	0,53	0,63
	window	256	0,61	0,64	0,64	0,62	0,63	0,61	0,62	0,60	0,61	0,59	0,60	0,59	0,60	0,58	0,61	0,58	0,61	0,58	0,62	0,37	0,51	0,64
	paragraph		0,62	0,64	0,64	0,63	0,63	0,64	0,64	0,63	0,63	0,63	0,63	0,62	0,62	0,62	0,62	0,62	0,62	0,62	0,62	0,35	0,52	0,64
(c)	sentence		0,57	0,59	0,59	0,61	0,61	0,60	0,60	0,59	0,59	0,59	0,59	0,59	0,59	0,59	0,59	0,59	0,59	0,58	0,59	0,28	0,44	0,61
	window	512	0,60	0,59	0,60	0,57	0,58	0,55	0,57	0,53	0,57	0,53	0,57	0,51	0,55	0,51	0,54	0,50	0,54	0,49	0,53	0,28	0,49	0,60
	paragraph		0,60	0,62	0,62	0,62	0,62	0,61	0,61	0,60	0,60	0,60	0,60	0,60	0,60	0,59	0,59	0,58	0,58	0,58	0,58	0,26	0,42	0,62
(d)	sentence		0,63	0,64	0,64	0,64	0,64	0,63	0,63	0,63	0,63	0,63	0,63	0,62	0,62	0,62	0,62	0,62	0,62	0,62	0,62	0,28	0,47	0,64
	window	512	0,61	0,62	0,63	0,60	0,60	0,58	0,59	0,58	0,60	0,57	0,59	0,54	0,57	0,54	0,57	0,52	0,55	0,51	0,55	0,28	0,48	0,63
	paragraph		0,64	0,64	0,64	0,64	0,64	0,64	0,64	0,64	0,64	0,63	0,63	0,63	0,63	0,63	0,63	0,61	0,61	0,60	0,60	0,26	0,46	0,64
(e)	sentence		0,61	0,63	0,63	0,64	0,64	0,65	0,65	0,65	0,65	0,65	0,65	0,64	0,64	0,64	0,64	0,64	0,64	0,64	0,64	0,28	0,49	0,65
	window	512	0,63	0,61	0,62	0,60	0,60	0,58	0,61	0,56	0,59	0,55	0,59	0,55	0,59	0,54	0,59	0,53	0,58	0,53	0,58	0,29	0,53	0,63
	paragraph		0,61	0,63	0,63	0,63	0,63	0,64	0,64	0,63	0,63	0,63	0,63	0,63	0,63	0,62	0,62	0,62	0,62	0,62	0,62	0,26	0,50	0,64
			0,64	0,64	0,64	0,64	0,64	0,65	0,65	0,65	0,65	0,65	0,65	0,64	0,64	0,64	0,64	0,64	0,64	0,64	0,64	0,38	0,53	0,65

The first column is the pretrained model used (indicated by the letter used in section 2.2). Second column indicates the chunk type used, either sentence, window/ngram, paragraph; then the size of the input to the model is reported. The other columns report the mAP averaged over all users and all queries by considering different aggregation functions of the chunks relevance scores.

Several column names represent the parameters passed to the aggregation function.

"#ch: *<number>*" is the parameter controlling the number of the best chunks considered for computing the document ranking score. When *<number>*=All, it means that all chunks are taken into account.

The second parameter "avg" is a Boolean controlling if the relevance score is defined as an average of the chunks' scores (in that case the parameter is used), or if it corresponds to their sum (no indication of the parameter appears). More in detail:

"#ch: N (sum)" indicates that the sum of the first N best chunks' scores of each document was computed;

"#ch: N (avg)" indicates that the average of the first N best chunks' scores of each document was computed;

When N=All it means that all the chunks in the documents are considered.

Since documents generally consist of long texts with many chunks we applied also an approach in which the document is represented by a single virtual embedding vector computed as the average of the chunks' vectors. In this case the results of mAP are reported in the column named "*Virtual Doc*" of Table 1.

The last column named "*max*" reports the best mAP obtained by any of the documents' chunks for the given setting in the row.

It can be easily noticed that three distinct models produce the maximum mAP = 0.64 for different settings by using cosine similarity between pairs of embedding vectors. Nevertheless, the most stable model under different input settings (both window and paragraph) and different matching definitions is (b) all-MiniLM-L6-v2.

Table 3 reports the mAP values when changing the similarity metric by using the dot product. In this case the best performing model is (e) msmarco-distilbert-base-tas-b that, when feed with chunks defined by sentences, reaches mAP = 0.65 when taking into account from 4 to 6 best chunks' relevance scores using both the sum or their average.

We thus select this latter model with the setting chunks=sentences, number of chunks per document to consider in the matching from 4 to 6 and either sum of scores or their average.

4. Conclusions

The originality of the described experience is manifold: first of all, the experimentation of LLMs to index and retrieve a highly heterogeneous collection of documents and their compared evaluation considering different chunk definitions, similarity metrics, and last but not least, by evaluating different aggregation strategies of the chunks relevance scores to compute the overall rank of documents. This last aspect is important in the case the documents are long, consisting of many chunks as in our case.

A second original contribution is the classification of the documents into "fuzzy" overlapping topics, according to a textual description of each topic which is used as a natural language query to retrieve the ranked list of documents belonging to the topic to a given extent. This approach has been deemed feasible to be applied for the implementation of the KH in order to provide public authorities with a tool that can

aid them in searching all documentation they need for the UNEP-MAP program.

Acknowledgements

References

[1] Bordogna, G., Tagliolato, P., Lotti, A., Minelli, A., Oggioni, A., & Babbini, L. (2023). Report 2 - Semantic Information Retrieval – Knowledge Hub. Zenodo. https://doi.org/10.5281/zenodo.10260195

[2] Kadhim, A.I. Survey on supervised machine learning techniques for automatic text classification. *Artif Intell Rev* **52**, 273–292 (2019). https://doi.org/10.1007/s10462-018-09677-1

[3] Manning C.D., Raghavan P., Schütze H., An Introduction to Information Retrieval, Online edition (c) 2009 Cambridge UP, URL https://nlp.stanford.edu/IR-book/pdf/irbookonlinereading.pdf

[4] Zhou, Q. Li, C. Li, J. Yu, Y. Liu, G. Wang, K. Zhang, C. Ji, Q. Yan, L. He et al., A comprehensive survey on pretrained foundation models: A history from bert to chatgpt," arXiv preprint arXiv:2302.09419, 2023.

[5] Kraft, D. H., Bordogna G., Pasi G. Fuzzy Set Techniques in Information Retrieval. (1999).DOI: 10.5281/zenodo.8082923

[6] REMPEC - https://www.rempec.org

[7] SPA/RAC - https://www.rac-spa.org

[8] SCP/RAC - http://www.cprac.org

[9] PAP/RAC - https://paprac.org

[10] https://www.unep.org/unepmap/resources/publications?/resources

[11] https://wedocs.unep.org/discover?filtertype=author&filter_relational_operator=equals&filter=UNEP%2FMAP

[12] Wolf T., Debut L., Sanh V., Chaumond J., et al., HuggingFace's Transformers: State-of-the-art Natural Language Processing, https://arxiv.org/pdf/1910.03771.pdf

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17). Curran Associates Inc., Red Hook, NY, USA, 6000–6010.

[14] Devlin J., Chang M.W., Lee K., Toutanova K., BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Proc. of NAACL-HLT 2019, pp. 4171–4186.

[15] https://github.com/UKPLab/sentence-transformers

[16] https://www.nltk.org/

[17] Yager, R. R. On the fuzzy cardinality of a fuzzy set. *International Journal of General Systems*, *35*(2), 191–206., https://doi.org/10.1080/03081070500422729, 2006

[18] https://huggingface.co/sentence-transformers/msmarco-distilbert-cos-v5

[19] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

[20] https://huggingface.co/sentence-transformers/msmarco-roberta-base-ance-firstp

[21] https://huggingface.co/sentence-transformers/msmarco-bert-base-dot-v5

[22] https://huggingface.co/sentence-transformers/msmarco-distilbert-base-tas-b

[23] http://vocabularies.unesco.org/thesaurus

[24] http://fuseki1.get-it.it/inforac/sparql

[25] Beitzel, S.M., Jensen, E.C., Frieder, O. MAP. In: LIU, L., ÖZSU, M.T. (eds) Encyclopedia of Database Systems. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-39940-9_492 2009

A. Online Resources

[26] https://github.com/INFO-RAC/KMP-library-scraping

[27] https://github.com/IREA-CNR-MI/inforac_ground_truth.

[28] http://rdfdata.get-it.it/inforac/