systems

MDPI

Article

Machine Reading at Scale: A Search Engine for Scientific and Academic Research

Norberto Sousa, Nuno Oliveira * and Isabel Praça

Research Group on Intelligent Engineering and Computing for Advanced Innovation and Development (GECAD), Porto School of Engineering (ISEP), 4200-072 Porto, Portugal; norbe@isep.ipp.pt (N.S.); icp@isep.ipp.pt (I.P.)
* Correspondence: nunal@isep.ipp.pt

Abstract: The Internet, much like our universe, is ever-expanding. Information, in the most varied formats, is continuously added to the point of information overload. Consequently, the ability to navigate this ocean of data is crucial in our day-to-day lives, with familiar tools such as search engines carving a path through this unknown. In the research world, articles on a myriad of topics with distinct complexity levels are published daily, requiring specialized tools to facilitate the access and assessment of the information within. Recent endeavors in artificial intelligence, and in natural language processing in particular, can be seen as potential solutions for breaking information overload and provide enhanced search mechanisms by means of advanced algorithms. As the advent of transformer-based language models contributed to a more comprehensive analysis of both text-encoded intents and true document semantic meaning, there is simultaneously a need for additional computational resources. Information retrieval methods can act as low-complexity, yet reliable, filters to feed heavier algorithms, thus reducing computational requirements substantially. In this work, a new search engine is proposed, addressing machine reading at scale in the context of scientific and academic research. It combines state-of-the-art algorithms for information retrieval and reading comprehension tasks to extract meaningful answers from a corpus of scientific documents. The solution is then tested on two current and relevant topics, cybersecurity and energy, proving that the system is able to perform under distinct knowledge domains while achieving competent performance.

Keywords: natural language processing; deep learning; question answering system; reading comprehension; information retrieval; machine reading at scale

Citation: Sousa, N.; Oliveira, N.; Praça, I. Machine Reading at Scale: A Search Engine for Scientific and Academic Research. *Systems* **2022**, *10*, 43. https://doi.org/10.3390/systems10020043

Academic Editor: Fernando De la Prieta, Sara Rodriguez, Juan M. Corchado and Vicent Botti

Received: 19 February 2022
Accepted: 2 April 2022
Published: 5 April 2022

1. Introduction

As of today, the exponential growth of the World Wide Web, resulting from the advent of technology, has generated huge amounts of data that, although having the potential of being beneficial for overall society, also contributes to the phenomenon of severe information overload [1]. In fact, and despite recent concerns, the problem of information overload is not new at all, with Klapp in [2] raising awareness in that regard over three decades ago. Nevertheless, as we now live in a digital era, there are several challenges to tackle when dealing with such amounts of data. For instance, information is now spread into a great variety of formats, such as emails, wikis and social media posts, that can be accessed through multiple communication channels, making it even harder to find what we are looking for when searching for a particular topic [3]. In an attempt to mitigate this issue, search engines have been proposed as a de facto tool for providing simplified/efficient access to information, with Bing and Google gaining huge popularity when it comes to web-based search [4].

Since a vast majority of information online can be found in textual representations, and as most search engines work on the basis of text-based queries, there is a need to not only accurately determine the search intent of such queries but also to appropriately

represent the semantic meaning of documents [4–6]. However, the ability of reading a text and then answering questions about it is considered to be a very difficult task for machines [7]. In that sense, novel developments in Natural Language Processing (NLP), such as the introduction of new Reading Comprehension (RC), and transformer-based language models, such as BERT [8], RoBERTa [9] and, even more recently, GPT-3 [10], have resulted in quite substantial contributions to the field. Nevertheless, these Deep Learning (DL) algorithms, based on the transformer architecture [11], cannot be directly applied to huge amounts of text due to constraints related to computational capabilities, requiring first some sort of information filtering so that only the relevant text gets analyzed. To overcome such limitations, Information Retrieval (IR) methods are usually applied to measure the relevance of a given document to a given question, narrowing down the search space and making the query more efficient [12].

The scientific community itself is not indifferent to the problem of information overload. As a matter of fact, according to the AI Index Report 2021 [13] published by Zhang et al., Artificial Intelligence (AI) alone has been the subject of over 120 thousand peer-reviewed publications by 2019, 12 times as much as the number recorded in the year 2000. Therefore, as the available number of scientific publications accumulates due to the increasing number of publications per year, there is, as of today, the need for an efficient way of navigating through all of that information, reducing the efforts of brute-force filtering when researching for a particular subject [14]. Intelligent solutions for this problem are emerging, with examples such as Semantic Scholar [15] showcasing the usefulness of AI in this field. Semantic scholar builds on top of existing search engines, allowing metadata-based article searches, but adding numerous interesting features. It sorts results based on author information and citations, with leaderboards for most influential authors and most cited works, as well as presenting an AI-generated summary of each article.

This work, however, lowers the entry bar on scientific knowledge even more by proposing a Question Answering (Q&A) system, in which one can place a domain-related question and expect a set of candidate answers retrieved from a corpus of scientific publications [6]. Moreover, by combining IR and RC methods, the system aims to provide a comprehensive matching between user intents expressed in natural language and the true semantic meaning of scientific documents, resulting in a more effective search process. The system is showcased in the context of two different domains, cybersecurity and energy, to demonstrate not only the ability of answering significant research questions but also the algorithm's generalization capabilities.

This work is organized into multiple sections that can be described as follows. Section 2 provides an overview of the state-of-the-art on information retrieval and reading comprehension algorithms. Section 3 describes the solution proposed in this work, providing fine-grained details regarding both the software architecture and AI algorithms. Section 4 describes the results obtained while applying the proposed solution to two different case studies. Section 5 provides a summary of the main conclusions to be drawn from this work, delineating further research lines.

2. Related Work

Intelligent Q&A systems touch upon multiple subtopics of the NLP domain such as reading comprehension and information retrieval, of which the Internet and search engines are a great example [16].

Information retrieval sees multiple different approaches using semantic matching, term matching and word embedding, where distinct chunks of text are matched through similar meanings. In [17], Nimmani et al. applied IR to the domain of software engineering, and more specifically, to aid in Change Impact Analysis (CIA). The authors combined the Bag of Words (BoW) method and Long-Short Term Memory (LSTM) networks, achieving better accuracy than current methods. Several optimization algorithms, such as AdaGrad, Adam and RMSprop, were experimented with and compared, achieving the best precision and recall results, at 98.1% and 98.5%, respectively.

In [18], Yoon et al., proposed a two-fold approach for sentence-level answer selection. First, a language model pretrained on a large-scale corpus was used to compute vector representations of input text. Then, the authors enhanced the compare-aggregate model by proposing a novel latent clustering method to compute additional information within the target corpus and by changing the objective function from listwise to pointwise. The proposed approach was tried out on the WikiQA and TREC-QA datasets, achieving Mean Average Precision (MAP) values of 76.4% and 87.5%, respectively.

In [19], Shtekh et al. investigated how text segmentation can help in information retrieval. The splitting of the text into semantically homogeneous blocks allows the detection of segment boundaries in documents [20]. The results show that, although offering an improvement in simpler models such as word2vec, going from 81.7% to 82.4% accuracy, in more modern models such as doc2vec the results tend to be inconclusive.

In [21], Alkılınç et al. performed an analysis on current information retrieval methods applied to old datasets, and commented upon their performance. The datasets used were Cranfield, Cacm and Medline, which are datasets containing different numbers of documents and queries created across several domains [22]. Preprocessing steps included tokenization, case folding and stemming. The authors applied four different models and although Divergence from Independence (DFIC) attained a better efficiency overall, different models can be the most effective for different datasets, supporting the theory that no single model has the best effectiveness [23].

In [24], Panda et al. proposed a novel IR system based on domain classification named Domain Classification-based IR System (DCIRS). The method is applied to user queries when searching for relevant documents in a corpus. For a given query, the most important keywords are selected and a domain label is given through the employment of Logic Regression and WordNet, respectively. After this initial step, documents within the identified domain with higher keyword match scores are retrieved. The proposed method achieved 93% and 92% recall for random user-placed queries in a corpus of 1000 scientific articles.

In [25], Hayat et al. seeked to solve the issue of broken links on the Internet using information retrieval. The authors proposed a novel pipeline, using a decision tree classification model to extract keywords from a webpage and its broken link. The subsequent search terms were then used to search for the original document with around 92.9% recall.

In [26], Manzoor et al. addressed Conversational Recommender Systems (CRS) that interact with users in natural language. Although most recent research efforts surrounding CRS present neural-based models trained to perform generation-based recommendations, the authors addressed retrieval-base recommendation, a less explored option in current literature. The proposed method combines TF-IDF (Term Frequency–Inverse Term Frequency) and heuristic rules to build a novel retriever-based CRS (RB-CRS). The algorithm was compared with two other methods, DeepCRS and KBR, in a dedicated web page, obtaining better results when judged by human evaluators. On a five-point scale, RB-CRS obtained an average rating of 3.71.

In [27], Shahzad Qaiser et al. employed a TF-IDF ranking system to several web pages in order to compare results. TF-IDF is the most utilized weighting scheme for web searches of information retrieval and text mining [28]. The authors also pointed out TF-IDF's biggest issue, which is not identifying different tenses of words. In the same manner, Neto et al. in [29] employed a modified version of TF-IDF, TF-ISF, applying stemming to reduce the impact of this classification method's weaknesses.

In word embedding, a document's words are mapped as vectors in a continuous vector space, and words with similar meanings will be closer to one another, aiding in dimensionality reduction [30]. In [31], Mikolov et al. demonstrated the application of a skip-gram model, a more computational efficient architecture, to mapping words to a vectorial space, and the same model but focusing on phrases.

On the other hand, reading comprehension has a big focus on attention-based models and its derivatives. In [32], Karpukhin et al. utilized the standard BERT pre-trained model

and a Dense Passage Retriever (DPR) in a dual encoder architecture achieving state-of-the-art results. Their DPR exceeded BM25's capabilities by far, namely more than a 20% increase in top-five accuracy (65.2%). Their results for end-to-end QA accuracy also improved on ORQA, the first open-retrieval question answering system, introduced in [33] by Lee et al., in the natural questions dataset [34].

In [35], Zhou applied several attention mechanisms and inter-layer connection techniques to reading comprehension models in order to merge information from both articles and questions so that answers can be predicted with higher accuracy. Experimental results led the author to believe that the length of provided questions highly impacts the performance of the model. A question length of 60 was selected as the optimal value.

In [36], Shan et al. investigated and compared the performance of different Q&A algorithms based on word-level embedding, sentence-level embedding, and traditional cosine similarity. The approach using attention mechanisms for sentence-level embedding has proven to be superior for the RACE dataset, obtaining the highest accuracy score of 88.3%.

In [37], Matsuyosh proposed the use of an attention-based Long Short-Term Memory (LSTM) model to aid a rule-based question-answering system, by identifying a user's intention behind their questions. This model attained 98% recall and 86% precision.

In [38], Cai et al. analyzed the claim that fine-tuning a model for reading comprehension, such as BERT, improves its results on more specific domains. The authors reached the conclusion that, although this tuning can improve results for certain tasks such as co-reference, question type and boundary probing, for others there is no measurable improvement.

In [39], Xu et al. tackled catastrophic forgetting during neural networks' training for reading comprehension. This phenomenon happens during fine tuning of a model for a specific domain, after pre-training with large out-of-domain datasets, causing the model to perform worse in the source material by the end of it. The authors proposed the incorporation of auxiliary penalty terms in the standard cross entropy loss to regularize the fine-tuning process. Using this approach, the model BERT managed to recover 8.77% of F1 points.

In [40], Hu et al. developed a framework to answer natural language questions on a Q&A system, using a graph-driven perspective. The proposed semantic query graph models the query intention in the natural language question, thus resolving the ambiguity of natural language. Testing with QALD-6 and WebQuestions test sets demonstrated the potential of this framework, achieving a 74% F1-score, in line with other state-of-the-art results.

In [41], Nishida et al. proposed a retrieve-and-read model, based on the bi-directional attention flow (BiDAF) model [42] to tackle reading comprehension problems. The proposed model employs a telescopic setting, where instead of deploying a computationally expensive neural network, a chain of different IR models is used. This novel ensemble achieved state-of-the-art results.

In conclusion, when reviewing the literature it is possible to identify BERT [8] as the cornerstone of reading comprehension's state-of-the-art models. This model influences much of the recent literature, with a big number of works using it or building on top of it, even impacting different approaches that experiment with BERT's attention mechanism, trying to apply it to other models such as LSTMs. By contrast, with information retrieval, there is no clear consensus on only one method or technique. The current literature explores implementations such as logic regression and WordNet. Some data preprocessing steps also receive an honorable mention for their prolific utilization, namely tokenization, case folding and stemming.

3. Proposed Solution

The issue of finding relevant information by means of question-answering across a large number of scientific publications can be framed as a problem of Machine Reading at Scale (MRS). The term was first coined by Cheng et al. in [43], being described as a

two-stepped task, where one should initially retrieve the most relevant documents of a corpus according to a given query before performing an exhaustive scan of such documents in order to extract good candidate answers. Moreover, Cheng et al. in that same work addressed an analogous problem, using more than five million Wikipedia pages as the knowledge base of an open-domain extractive Q&A system. These concerns, such as choosing a proper knowledge base or having the need to support a fully integrated Q&A pipeline, influenced the design of the solution presented in this work.

In that regard, the proposed system was built using the Python programming language on top of Haystack [44], an open-source framework for developing intelligent search systems for large document collections. Haystack takes the recent advances in NLP and provides a bridge between research and industry, allowing complex algorithms to be applied to real world use cases by means of high-level APIs. Moreover, the system's internal architecture encompasses two main components, the front-end, a web-based graphical interface that can be accessed by the users and the back-end, a RESTful API that exposes the use cases of our solution through several endpoints, working on a client–server basis. For the prototyping phase, an SQLite database was selected to serve as a document storage, storing the preprocessed scientific articles. In spite of SQLite presenting some pitfalls in terms of efficiency (in exchange for simplicity), the software was designed so that the database technology can be easily replaced by more robust solutions such as elastic search or FAISS. The back-end side of our application can also be further detailed into two distinct modules, a web-crawler, which was integrated with arXiv.org API to fetch scientific articles in real time, and a search engine, which combines two distinct NLP methods, a retriever and a reader, to build a haystack-like pipeline that is able to find candidate answers in our corpus.

In terms of functionality, the proposed system concerns three core use cases: fetching scientific publications, consulting the database summary and finding candidate answers. These can be detailed as follows:

- **UC1—Fetching Scientific Publications**: This use case is further divided into more fine-grained sub-tasks such as downloading publications from a given source (in this case arXiv.org), preprocessing each document and indexing the resultant data into the document store. The user starts by specifying a given search topic and the maximum number of articles to be downloaded. Then, the crawler tries to find articles related to the specified topic and downloads all of them until the maximum threshold is reached. If the number of articles is inferior to the specified threshold, all articles related to the specified subject are downloaded. After downloading the documents, these are preprocessed—empty lines are removed, consecutive whitespaces are truncated and pdf headers and footers are discarded. The text of each document is also split into several search chunks of 500 words with respect to sentence continuity, so that the search process can be optimal. Finally, each resulting chunk is indexed, along with the document meta-data, in the document database, increasing the knowledge base of the Q&A system. Chunks of the document database that share the same foreign key can be traced back to the original unsplit document that was downloaded and preprocessed.
- **UC2—Consulting Database Summary**: So that the user can keep track of the continuous changes to the available corpus, a summary of the document database content is displayed in the main dashboard of the graphical interface. This summary is comprised of several pieces of information, such as the number of downloaded articles, search chunks and document categories.
- **UC3—Finding Candidate Answers**: This use case is arguably the most important one as it focuses on the answer-finding process by means of intelligent algorithms. The proposed search pipeline works by considering two different components, a retriever and a reader. First, the user poses a question to the system and specifies several search parameters such as a category filter, the number of candidate answers to be displayed, c, and the maximum number of relevant search chunks to be found by the retriever, k. Then, the system executes the retriever, a TF-IDF-based retriever, returning the most

relevant *k* chunks. Finally, the reader, a RoBERTa model, will try to find the best *c* answers in the selected *k* chunks according to a confidence metric.

The presented solution is intentionally generic so that it is simple to replace individual components without affecting the system as a whole. As an example, despite the current implementation of **UC1** targeting arXiv.org as its source, the crawler component can be expanded to integrate with other scientific repositories with little changes to the code base. This mitigates future bottlenecks and prevents the system from depending on a single external source by design, assuring that it is always possible to further enrich the search corpus with the contents of new scientific publications over time. Similarly, the pipeline proposed in the context of **UC3** is also quite broad since both employed algorithms can be smoothly replaced by enhanced versions or further endeavors in NLP's state-of-the-art without requiring substantial changes. On the other hand, and with respect to **UC3**, database management functionalities could be further expanded. While it is interesting to keep track of corpus changes, it is as well useful to perform listings of downloaded articles accordingly to different combinations of search criteria, to manually import new documents and to conduct manual disposal of unwanted articles from the document store.

3.1. Pipeline Description

The pipeline employed in this work is of general purpose as its building blocks are not limited to a specific target domain. The retriever, TF-IDF, is, fundamentally, a statistical measure for any sort of query–document combination; hence, it can be directly applied to any domain without prior fine-tuning. On the other hand, the reader, RoBERTa, requires training examples composed of different question and answer pairs. To overcome such a limitation, we opted to use a model that was pre-trained on the SQuAD dataset [9], a data collection comprising over 100,000 examples of questions posed by crowdworkers on a set of Wikipedia articles [7]. It is a widely used benchmark dataset for training and evaluating general-purpose extractive Q&A machine learning models in current literature [45]. The RoBERTa model employed in our solution, [9], achieved an exact match score of approximately 79.97% and an F1-score of 83.00% under this same testbed. In the conducted experiments, the algorithm also performed quite competently when facing both cybersecurity and energy domains, finding interesting answers to several questions that were placed. A brief description of the theoretical foundations of the employed algorithms, TF-IDF and RoBERTa, is provided in the following sections. Figure 1 describes the employed retriever–reader pipeline.

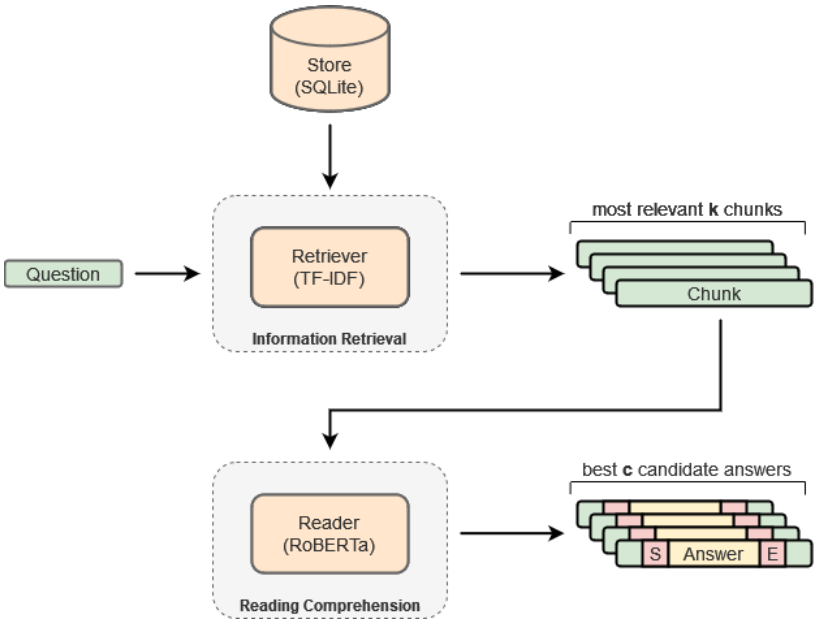


Figure 1. Retriever–reader search pipeline.

3.1.1. Retriever

In order to search through relevant information, a TF-IDF retriever was put in place. It is a numerical statistic that is intended to reflect how important a given word is to a document in a corpus.

$$TFIDF_{i,d} = tf_{i,d} \cdot idf_i \tag{1}$$

In the scientific question and answering domain, it is expected that the queries will have lexical overlap with their answers, making this algorithm a good searcher of relevant information. TF-IDF acts as a low-complexity filter for feeding heavier answer extraction algorithms.

3.1.2. Reader

The other critical step of our proposed pipeline is the question understanding step. Here there is a need to properly understand the question at hand, by being able to properly model it in such a way that it can then be passed through the pipeline and improve the chances of obtaining not only accurate but also relevant answers for the the true intent of the question that was provided initially.

For this step, we use a Framework for Adapting Representation Models (FARM) reader coupled with the RoBERTa language model [46], which works alongside the retriever and parses the candidate documents provided. RoBERTa is an iteration of BERT [8], whose architecture is based on the transformer; see Figure 2. It was also pretrained on a much larger corpus than BERT and as a result, achieves significant performance gains.

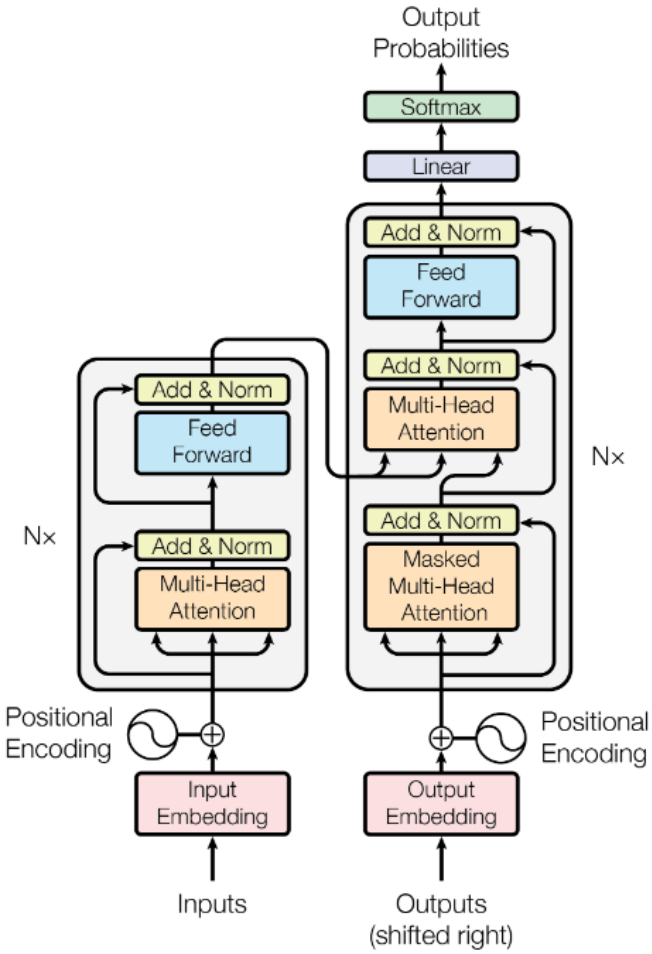


Figure 2. Transformer architecture [11].

The transformer follows an encoder–decoder architecture, adopting stacked self-attention and point-wise, fully connected layers for both the encoder and decoder, as presented in the left and right sides of Figure 2, respectively. It disregards recurrence and convolutions from the usual encoder–decoder models and instead focuses on several types of attention mechanisms. As an attention function can be described as a mapping of a query and multiple key-value pairs to an output (with all representing numerical vectors), the authors of the transformer [11] found multi-head attention beneficial to be encompassed in the proposed architecture. Multi-head attention provides a way to perform different projections of queries, keys and values, allowing the model to perceive information of multiple representation subspaces at different positions.

RoBERTa was deployed using Deepset's NLP framework, Haystack [44]. Deepset released straight implementations of several popular and well-established models in the NLP literature, some of which are represented in Table 1, in addition to new ones such as TinyRoBERTa where the approach of [47] is applied to the RoBERTa model. These models of Deepset's authorship provide simplified integration with the Haystack framework.

Table 1. Haystack models with SQuAD dataset [7]. as benchmark.

Name	Link	Original	Exact Match	F1-Score
RoBERTa	[9]	[46]	79.9	82.1
TinyRoBERTa	[48]	-	78.7	81.9
TinyBERT	[49]	[47]	71.9	76.36
DistilBERT	[50]	[51]	68.6	72.8

4. Case Study

The usefulness and generalization of this solution allows it to be applied to numerous topics. For this reason, two current and challenging research topics were chosen as a case study—cybersecurity and energy.

A list of cybersecurity-related keywords was compiled, in order to find relevant articles to build the search corpus with. For each search term a number of documents was extracted from arXiv.org, as shown in Table 2. After removing the corrupted/unparsable documents and duplicates, this corpus totaled 821 articles.

Table 2. Cybersecurity corpus composition [6].

Adversarial Attack	200
Attack Detection	175
Cyberphysical Systems	200
Cybersecurity	129
Intrusion Detection Systems	130
Total Used	834
Corrupted	−6
Duplicates	−7
Total Articles in Corpus	821

In the same manner, energy-related keywords were chosen to find relevant articles. The search terms and compiled articles are represented in Table 3. After removing the corrupted/unparsable documents, this corpus totaled 565 articles.

Table 3. Energy corpus composition.

Smart Grids	200
Electricity Markets	156
Energy Forecasting	13
Intelligent Buildings	5
Energy Consumption	197
Total Used	571
Corrupted	−6
Duplicates	0
Total Articles in Corpus	565

Each one of these articles was downloaded and processed as per the pipeline indicated in the previous section. After processing, the articles were split into chunks of 500 words while taking into account sentence continuity.

4.1. Results

The introduced solution has a main dashboard, on the left are located some search configuration sliders and database-related information. In the middle there are two buttons to navigate between the database management and search engine functionalities. The described interface is presented in Figure 3.

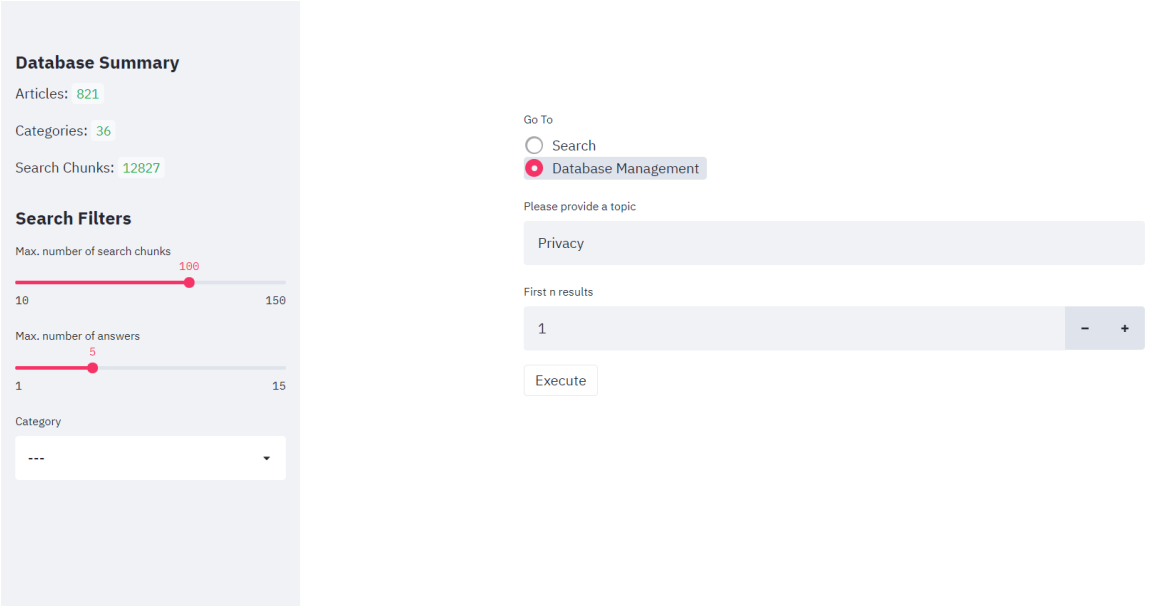


Figure 3. Dashboard [6].

In order to evaluate the system's performance, several research questions were placed empirically. These regard the aforementioned corpus, composed of 821 cybersecurity and 565 energy research papers, built using the system's database management functionality. Additionally, the quality of the responses found are directly connected to the contents of each one. This can be improved by populating the corpus with more articles pertaining to a given topic or adding a new topic entirely. When accessing such functionality, we can specify a given search topic and the maximum number of documents to be downloaded. These will be directly fetched from arXiv.org, preprocessed and indexed alongside their metadata in the document database. For the topic of "Privacy", with a maximum of one article, the result is presented in Figure 4.

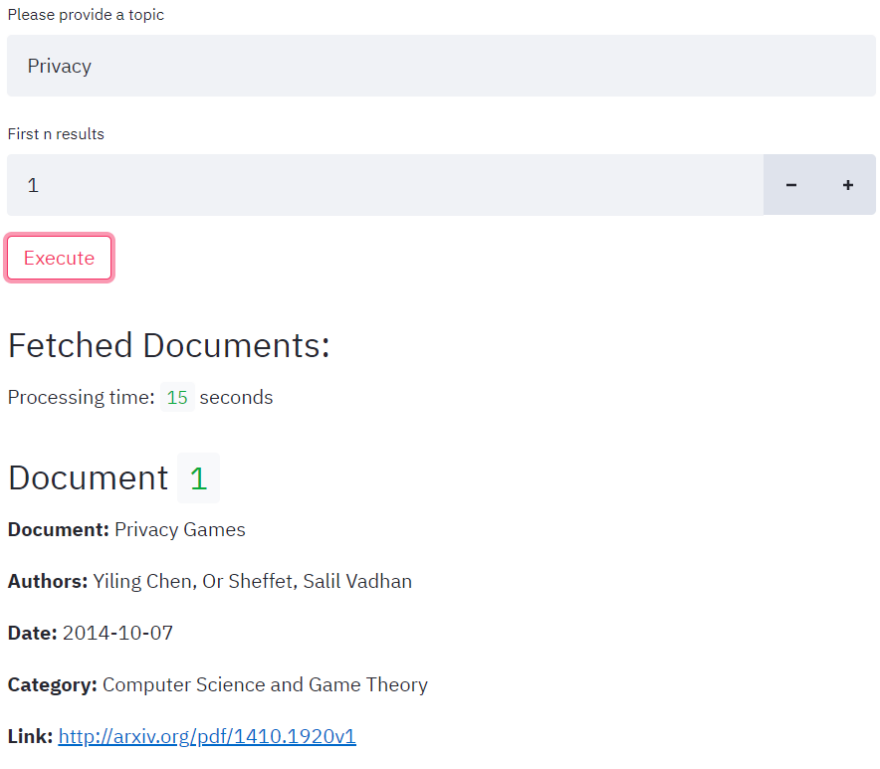


Figure 4. Database menu [6].

4.1.1. Cybersecurity

With the corpus prepared, it is then possible to start asking questions [6]. In this case and by asking: "What are the challenges of AI?", the most interesting candidate answer is presented in Figure 5, due to its high probability (confidence) score. This answer is highlighted in its surrounding context, accompanied by additional information such as title, authors, publishing date, and a link to the article itself.

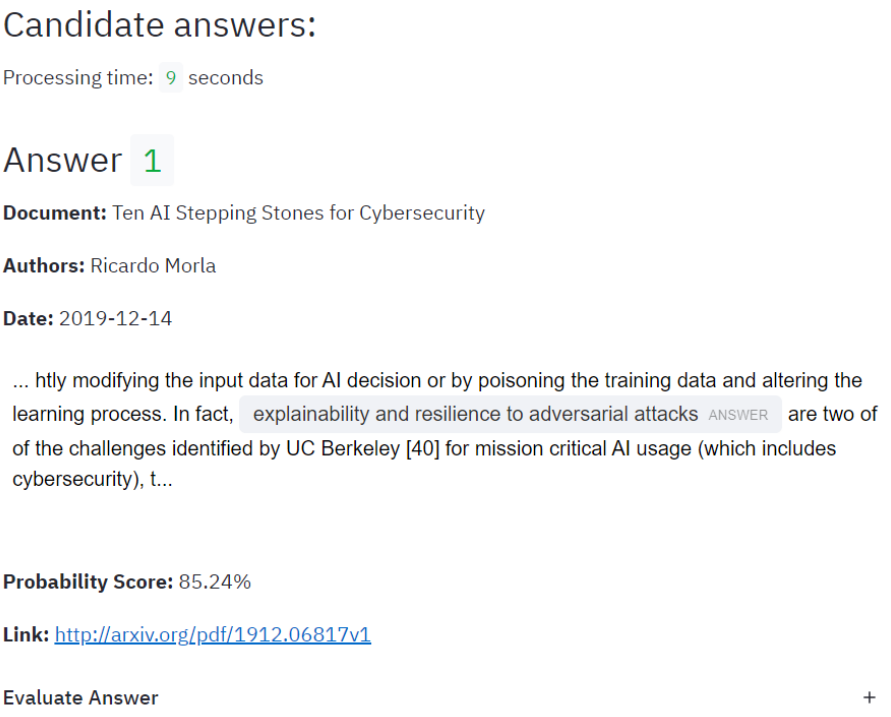
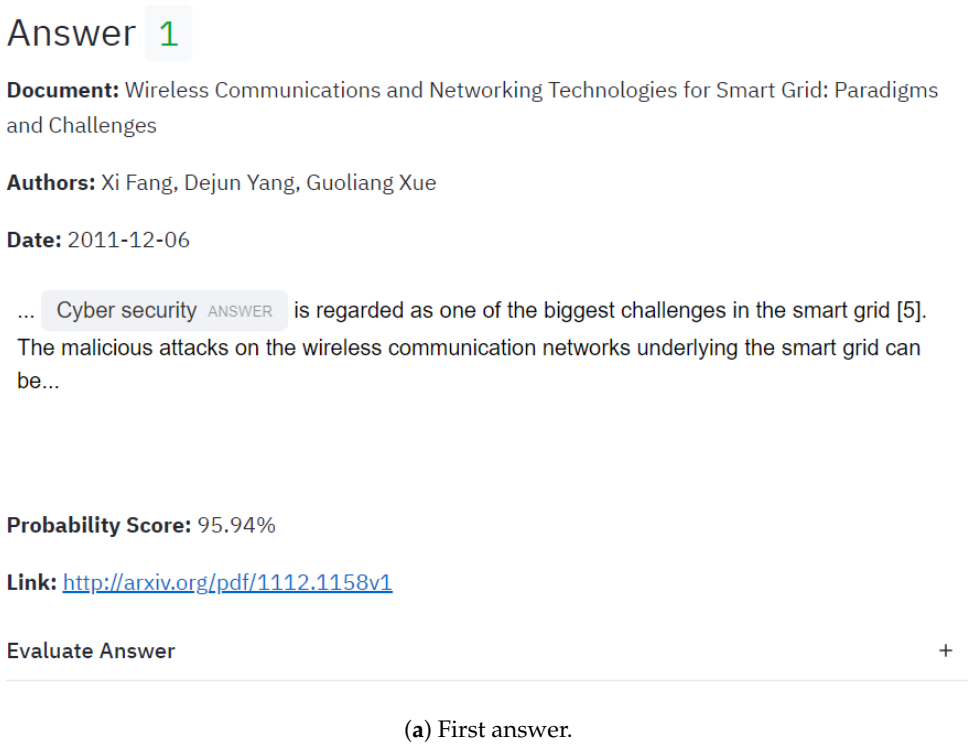


Figure 5. Cybersecurity domain question: What are the challenges of AI [6]?

As the question is vague in nature, and the prepared corpus is geared more towards cybersecurity instead of AI, the obtained answer "explainability and resilience to adversarial attacks" also tended to the cybersecurity side of AI, due to the nature of the used article [52]. Another example is the question "What are the main challenges of cybersecurity research?", which yielded interesting results. The first answer correctly quoted [53] and responded with "lack of adequate evaluation/test environments that utilize up-to-date datasets, variety of testbeds while adapting unified evaluation methods", while the second answer built on the first one with "lack of research methodology standards" [54]. Finally, by asking "Which machine learning models are commonly used?" we obtained "Naïve Bayes, SVM, KNN, and decision trees" from [55] and virtually the same answer, "Support Vector Machine, Decision Trees, Fuzzy Logic, BayesNet and Naïve Bayes", from [56].

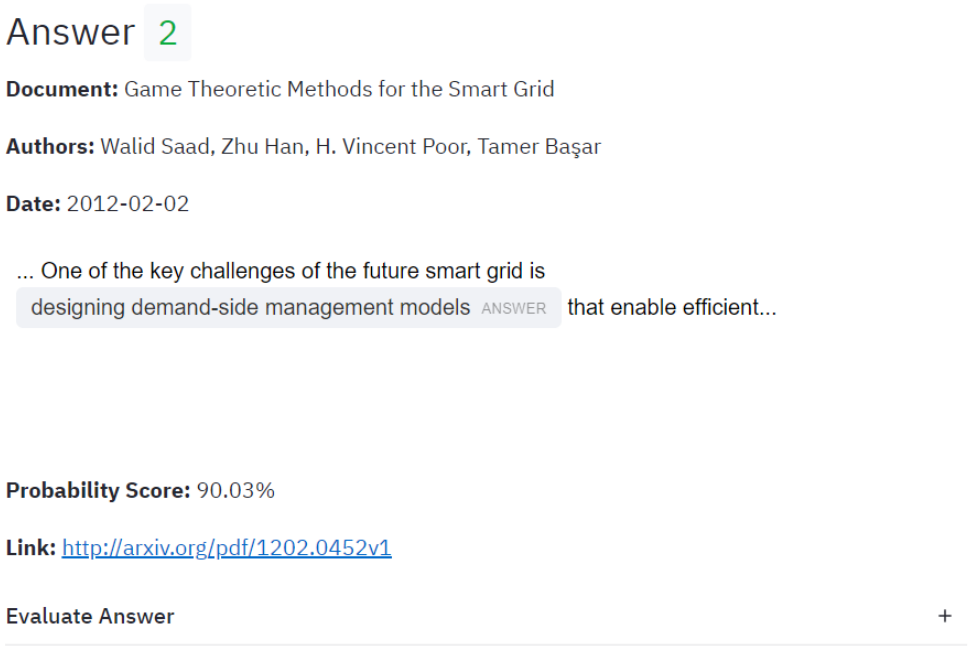
4.1.2. Energy

Similarly, using the energy corpus, when asking "What are the challenges of Smart Grids?" the highest rated answer was "Cybersecurity" [57], with "designing demand-side management models" [58] as a close second, as seen in Figure 6. Although correct, these answers are possibly too narrow in scope to sufficiently answer the question, perhaps indicating a need to further enrich the existing corpus. On the other hand, asking "What are examples of forecasting algorithms?" resulted in the response "ARIMA, SVM, ANN, and adaptive" [59], correctly naming some of the most used models currently in the literature. Following this line of questioning and inquiring more about these algorithms by asking "What are the applications of Neural Networks?" resulted in the answer "price modeling" [60], the main use currently for these algorithms in this domain.



(**a**) First answer.

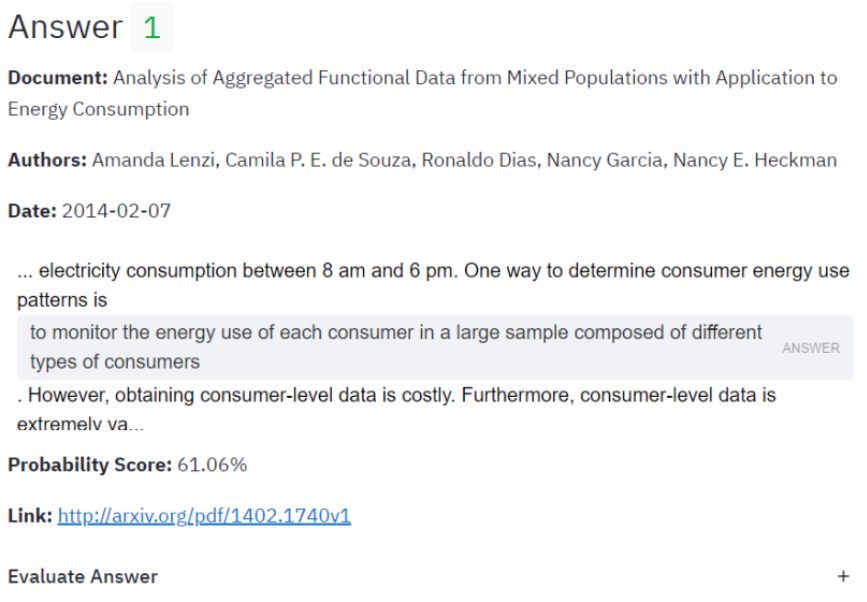
Figure 6. *Cont.*



(**b**) Second answer.

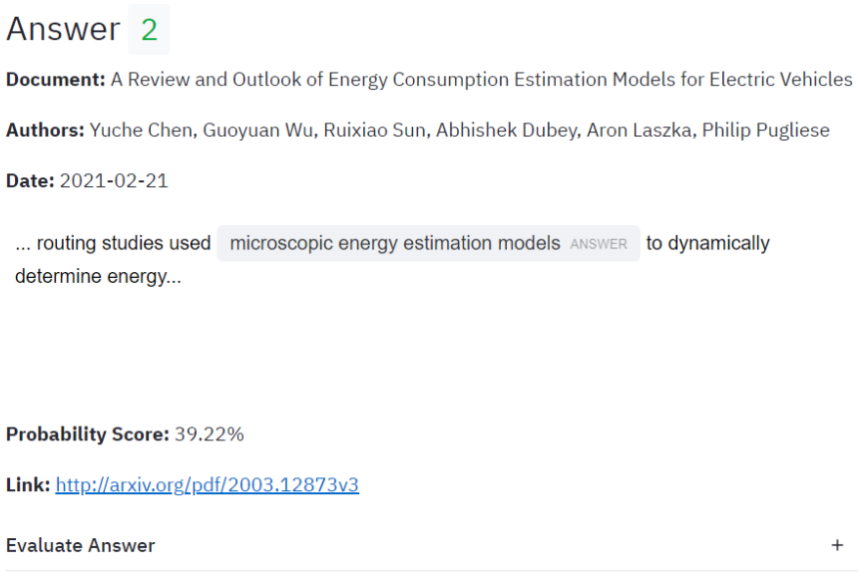
Figure 6. Energy domain question: What are the challenges of Smart Grids?

More specifically, regarding the energy consumption domain, one can ask "How to determine consumers' energy use patterns?", obtaining answers such as "to monitor the energy use of each consumer in a large sample composed of different types of consumers" and "microscopic energy estimation models" quoting [61,62], respectively. These answers are presented in Figure 7.



(**a**) First answer.

Figure 7. *Cont.*



(**b**) Second answer.

Figure 7. Energy domain question: How to determine consumers' energy use patterns?

4.1.3. Complexity

The retriever–reader search pipeline proposed in this work assumes a trade-off between the amount of computational time required to find a specific answer and the number of text chunks to be output by the retriever. As the amount of search chunks is increased, the more likely it is for the reader to find suitable answers; however, more computational power is involved, as it will need to process more blocks of text. In order to better understand this phenomenon, the system response time was tested for 50 to 600 chunks, resorting to a NVIDIA P4000 GPU with 8 Gigabytes of VRAM for hardware support. This analysis is presented in Figure 8.

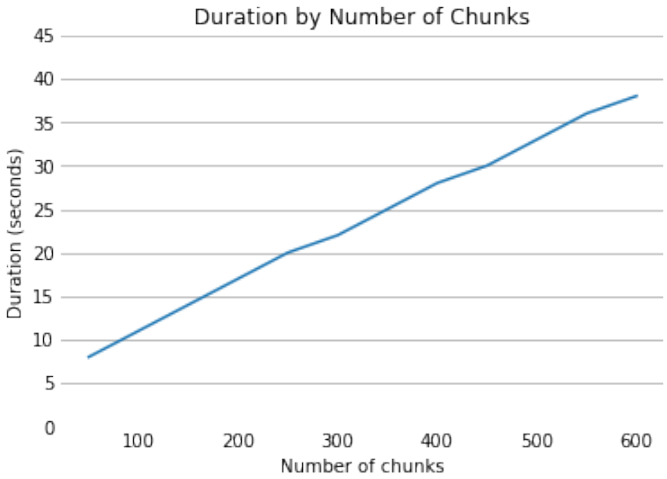


Figure 8. Trade-off between required computational power and the number of chunks to be output by the retriever.

4.1.4. Conclusions

Our solution performed admirably, by compiling two corpuses of articles on the hottest research topics in the selected fields and by finding interesting answers to a set of significant questions regarding applications of AI to cybersecurity and energy, and the main challenges of the current research. Regarding the extractive Q&A pipeline, the RoBERTa

model exhibited a notable adaptation capability since it was not retrained in the scope of either of the domains.

5. Conclusions

Given the amount of scientific articles that are published every year it is hard to find exactly what we are looking for when researching a particular topic. In this work, we presented a software solution that aims to solve this problem while improving on current scientific search engines by allowing searches on the content of the documents, understanding queries in the form of natural language questions and proposing answers found on scientific publications. This not only aims to solve the problem of information overload, but also lowers the entry bar for this advanced type of knowledge, facilitating the navigation through unknown domains by answering simple questions with advanced knowledge. It comprises several advantageous features, such as the continuous update of the search corpus by providing an easy-to-use integration with the arXiv.org API and the ability to find candidate answers extracted from the corpora of downloaded scientific publications by applying a combination of two NLP methods, TF-IDF and RoBERTa. Furthermore, the introduced solution was showcased in the context of cybersecurity and energy, complex fields of science with increasing interest. With a base corpus of 821 and 565 articles for cybersecurity and energy, respectively, the system was able to find proper answers regarding the domains to questions such as "What are the challenges of AI?", "Which machine learning models are commonly used?", "What are the challenges of Smart Grids?" and "What are examples of forecasting algorithms?", showing a great capability of generalization.

As future work, we will implement additional features regarding the document database management, expand the web crawler so that it can work with more scientific repositories and improve the document preprocessing step to make our search engine more efficient. Another research line that can be suggested focuses on the creation of a new Q&A dataset for the scientific context that can serve as a benchmark for novel approaches to solve the problem of information overload in the academia.

Author Contributions: Conceptualization, N.S., N.O. and I.P.; methodology, N.S., N.O. and I.P.; software, N.S. and N.O.; validation, I.P.; investigation, N.S. and N.O.; writing, N.S. and N.O.; visualization, N.S. and N.O.; supervision, I.P.; project administration, I.P.; funding acquisition, I.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work has received funding from the following projects: UIDB/00760/2020 and UIDP/00760/2020.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: In this work, open access publications from arXiv.org were used. These can be found here: arXiv (accessed on 18 February 2022).

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Zhang, W.; Zhao, X.; Zhao, L.; Yin, D.; Yang, G.H.; Beutel, A. Deep Reinforcement Learning for Information Retrieval: Fundamentals and Advances. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Xi'an, China, 25–30 July 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 2468–2471.
2. Klapp, O.E. *Overload and Boredom: Essays on the Quality of Life in the Information Society*; Greenwood Publishing Group Inc.: Westport, CT, USA, 1986.
3. Saxena, D.; Lamest, M. Information overload and coping strategies in the big data context: Evidence from the hospitality sector. *J. Inf. Sci.* **2018**, *44*, 287–297. [CrossRef]

4. Huang, J.T.; Sharma, A.; Sun, S.; Xia, L.; Zhang, D.; Pronin, P.; Padmanabhan, J.; Ottaviano, G.; Yang, L. Embedding-Based Retrieval in Facebook Search. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, 6–10 July 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 2553–2561.
5. Li, H.; Xu, J. Semantic Matching in Search. *Found. Trends Inf. Retr.* **2014**, *7*, 343–469. [CrossRef]
6. Oliveira, N.; Sousa, N.; Praça, I. A Search Engine for Scientific Publications: A Cybersecurity Case Study. In Proceedings of the International Symposium on Distributed Computing and Artificial Intelligence, Salamanca, Spain, 6–8 October 2021; pp. 108–118.
7. Rajpurkar, P.; Zhang, J.; Lopyrev, K.; Liang, P. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; Association for Computational Linguistics: Austin, TX, USA, 2016; pp. 2383–2392. [CrossRef]
8. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, MN, USA, 2–7 June 2019; Association for Computational Linguistics: Minneapolis, MN, USA, 2019; pp. 4171–4186. [CrossRef]
9. Chan, B.; Möller, T.; Pietsch, M.; Soni, T. Deepset Roberta-Base-Squad2. Available online: https://huggingface.co/deepset/roberta-base-squad2 (accessed on 6 May 2021).
10. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners. *arXiv* **2020**, arXiv:2005.14165.
11. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, U.; Polosukhin, I. Attention is All You Need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17), Long Beach, CA, USA, 4–9 December 2017; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 6000–6010.
12. Aggarwal, C.C.; Zhai, C. A Survey of Text Classification Algorithms. In *Mining Text Data*; Springer US: Boston, MA, USA, 2012; pp. 163–222. [CrossRef]
13. Zhang, D.; Mishra, S.; Brynjolfsson, E.; Etchemendy, J.; Ganguli, D.; Grosz, B.J.; Lyons, T.; Manyika, J.; Niebles, J.C.; Sellitto, M.; et al. The AI Index 2021 Annual Report. *arXiv* **2021**, arXiv:2103.06312.
14. Bevendorff, J.; Stein, B.; Hagen, M.; Potthast, M. Elastic ChatNoir: Search Engine for the ClueWeb and the Common Crawl. In Proceedings of the European Conference on Information Retrieval (ECIR), Grenoble, France, 26–29 March 2018.
15. Semantic Scholar. 2022. Available online: https://www.semanticscholar.org/ (accessed on 23 March 2022).
16. Singh, A.K.; Kumar, P.R. A comparative study of page ranking algorithms for information retrieval. *Int. J. Electr. Comput. Eng.* **2009**, *4*, 469–480.
17. Nimmani, P.; Vodithala, S.; Polepally, V. Neural Network Based Integrated Model for Information Retrieval. In Proceedings of the 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 6–8 May 2021; pp. 1286–1289. [CrossRef]
18. Yoon, S.; Dernoncourt, F.; Kim, D.S.; Bui, T.; Jung, K. A Compare-Aggregate Model with Latent Clustering for Answer Selection. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM'19), Beijing, China, 3–7 November 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 2093–2096. [CrossRef]
19. Shtekh, G.; Kazakova, P.; Nikitinsky, N.; Skachkov, N. Applying Topic Segmentation to Document-Level Information Retrieval. In Proceedings of the 14th Central and Eastern European Software Engineering Conference Russia (CEE-SECR'18), Moscow, Russia, 12–13 October 2018; Association for Computing Machinery: New York, NY, USA, 2018. [CrossRef]
20. Du, L.; Buntine, W.; Johnson, M. Topic segmentation with a structured topic model. In Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Atlanta, GA, USA, 9–14 June 2013; pp. 190–200.
21. Alkılınç, A.; Arslan, A. A Comparison of Recent Information Retrieval Term-Weighting Models Using Ancient Datasets. In Proceedings of the 2018 International Conference on Artificial Intelligence and Data Processing (IDAP), Malatya, Turkey, 28–30 September 2018; pp. 1–4. [CrossRef]
22. Sanderson, M. *Test Collection Based Evaluation of Information Retrieval Systems*; Now Publishers Inc.: Hanover, MA, USA, 2010.
23. Petersen, C.; Simonsen, J.G.; Järvelin, K.; Lioma, C. Adaptive Distributional Extensions to DFR Ranking. In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM'16), Indianapolis, IN, USA, 24–28 October 2016; Association for Computing Machinery: New York, NY, USA, 2016; pp. 2005–2008. [CrossRef]
24. Priyadarsini Panda, S.; Prasad Mohanty, J. A Domain Classification-based Information Retrieval System. In Proceedings of the 2020 IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE), Bhubaneswar, India, 26–27 December 2020; pp. 122–125. [CrossRef]
25. Hayat, S.; Li, Y.; Riaz, M. Automatic Recovery of Broken Links Using Information Retrieval Techniques. In Proceedings of the 2nd International Conference on Natural Language Processing and Information Retrieval (NLPIR 2018), Bangkok, Thailand, 7–9 September 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 32–36. [CrossRef]
26. Manzoor, A.; Jannach, D. Generation-Based vs Retrieval-Based Conversational Recommendation: A User-Centric Comparison. In Proceedings of the Fifteenth ACM Conference on Recommender Systems, Amsterdam, The Netherlands, 27 September–1 October 2021; Association for Computing Machinery: New York, NY, USA, 2021; pp. 515–520.
27. Qaiser, S.; Ali, R. Text mining: Use of TF-IDF to examine the relevance of words to documents. *Int. J. Comput. Appl.* **2018**, *181*, 25–29. [CrossRef]

28. Beel, J.; Gipp, B.; Langer, S.; Breitinger, C. Research-paper recommender systems: A literature survey. *Int. J. Digit. Libr.* **2016**, *17*, 305–338. [CrossRef]
29. Neto, J.L.; Santos, A.D.; Kaestner, C.A.; Freitas, A.A. Document Clustering and Text Summarization. In Proceedings of the Fourth International Conference on the Practical Application of Knowledge Discovery and Data Mining, New York, NY, USA, 27–31 August 1998; Mackin, N., Ed.; The Practical Application Company: Woburn, MA, USA, 2000; pp. 41–55.
30. Ge, L.; Moh, T. Improving text classification with word embedding. In Proceedings of the 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, USA, 11–14 December 2017; pp. 1796–1805. [CrossRef]
31. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed representations of words and phrases and their compositionality. *arXiv* **2013**, arXiv:1310.4546.
32. Karpukhin, V.; Oğuz, B.; Min, S.; Wu, L.; Edunov, S.; Chen, D.; Yih, W.T. Dense passage retrieval for open-domain question answering. *arXiv* **2020**, arXiv:2004.04906.
33. Lee, K.; Chang, M.W.; Toutanova, K. Latent retrieval for weakly supervised open domain question answering. *arXiv* **2019**, arXiv:1906.00300.
34. Kwiatkowski, T.; Palomaki, J.; Redfield, O.; Collins, M.; Parikh, A.; Alberti, C.; Epstein, D.; Polosukhin, I.; Devlin, J.; Lee, K.; et al. Natural questions: A benchmark for question answering research. *Trans. Assoc. Comput. Linguist.* **2019**, *7*, 453–466. [CrossRef]
35. Zhou, X. A Study of Machine Reading Comprehension Based on Attention Mechanism. In Proceedings of the 2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP), Xi'an, China, 9–11 April 2021; pp. 1058–1061. [CrossRef]
36. Shan, J.; Nishihara, Y.; Maeda, A.; Yamanishi, R. Extraction of Question-related Sentences for Reading Comprehension Tests via Attention Mechanism. In Proceedings of the 2020 International Conference on Technologies and Applications of Artificial Intelligence (TAAI), Taipei, Taiwan, 3–5 December 2020; pp. 23–28. [CrossRef]
37. Matsuyoshi, Y.; Takiguchi, T.; Ariki, Y. User's Intention Understanding in Question-Answering System Using Attention-based LSTM. In Proceedings of the 2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Honolulu, HI, USA, 12–15 November 2018; pp. 1752–1755. [CrossRef]
38. Cai, J.; Zhu, Z.; Nie, P.; Liu, Q. A Pairwise Probe for Understanding BERT Fine-Tuning on Machine Reading Comprehension. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'20), Xi'an, China, 25–30 July 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 1665–1668. [CrossRef]
39. Xu, Y.; Zhong, X.; Yepes, A.J.J.; Lau, J.H. Forget Me Not: Reducing Catastrophic Forgetting for Domain Adaptation in Reading Comprehension. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–8. [CrossRef]
40. Hu, S.; Zou, L.; Yu, J.X.; Wang, H.; Zhao, D. Answering Natural Language Questions by Subgraph Matching over Knowledge Graphs (Extended Abstract). In Proceedings of the 2018 IEEE 34th International Conference on Data Engineering (ICDE), Paris, France, 16–19 April 2018; pp. 1815–1816. [CrossRef]
41. Nishida, K.; Saito, I.; Otsuka, A.; Asano, H.; Tomita, J. Retrieve-and-Read: Multi-Task Learning of Information Retrieval and Reading Comprehension. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM'18), Turin, Italy, 22–26 October 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 647–656. [CrossRef]
42. Seo, M.; Kembhavi, A.; Farhadi, A.; Hajishirzi, H. Bidirectional attention flow for machine comprehension. *arXiv* **2016**, arXiv:1611.01603.
43. Chen, D.; Fisch, A.; Weston, J.; Bordes, A. Reading Wikipedia to Answer Open-Domain Questions. *arXiv* **2017**, arXiv:1704.00051.
44. Haystack. 2020. Available online: https://haystack.deepset.ai/ (accessed on 6 May 2021).
45. Cambazoglu, B.B.; Sanderson, M.; Scholer, F.; Croft, B. A review of public datasets in question answering research. In *ACM SIGIR Forum*; ACM: New York, NY, USA, 2021; Volume 54, pp. 1–23.
46. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv* **2019**, arXiv:1907.11692.
47. Jiao, X.; Yin, Y.; Shang, L.; Jiang, X.; Chen, X.; Li, L.; Wang, F.; Liu, Q. Tinybert: Distilling bert for natural language understanding. *arXiv* **2019**, arXiv:1909.10351.
48. Chan, B.; Möller, T.; Pietsch, M.; Soni, T.; Bartels, M. Deepset Tinyroberta-Squad2. Available online: https://huggingface.co/deepset/tinyroberta-squad2 (accessed on 25 March 2022).
49. Möller, T.; Risch, J.; Pietsch, M.; Bartels, M. Deepset Tinybert-6L-768D-Squad2. Available online: https://huggingface.co/deepset/tinybert-6l-768d-squad2 (accessed on 25 March 2022).
50. Möller, T.; Risch, J.; Pietsch, M.; Bartels, M. Deepset Bert-Medium-Squad2-Distilled. Available online: https://huggingface.co/deepset/bert-medium-squad2-distilled (accessed on 25 March 2022).
51. Sanh, V.; Debut, L.; Chaumond, J.; Wolf, T. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *arXiv* **2019**, arXiv:1910.01108.
52. Morla, R. Ten AI Stepping Stones for Cybersecurity. *arXiv* **2019**, arXiv:1912.06817.
53. Kayan, H.; Nunes, M.; Rana, O.; Burnap, P.; Perera, C. Cybersecurity of Industrial Cyber-Physical Systems: A Review. *arXiv* **2021**, arXiv:2101.03564.
54. Gardner, C.; Waliga, A.; Thaw, D.; Churchman, S. Using Camouflaged Cyber Simulations as a Model to Ensure Validity in Cybersecurity Experimentation. *arXiv* **2019**, arXiv:1905.07059.

55. Priya, V.; Thaseen, I.S.; Gadekallu, T.R.; Aboudaif, M.K.; Nasr, E.A. Robust Attack Detection Approach for IIoT Using Ensemble Classifier. *Comput. Mater. Contin.* **2021**, *66*, 2457–2470. [CrossRef]
56. Shah, S.A.R.; Issac, B. Performance comparison of intrusion detection systems and application of machine learning to Snort system. *Future Gener. Comput. Syst.* **2018**, *80*, 157–170. [CrossRef]
57. Fang, X.; Yang, D.; Xue, G. Wireless communications and networking technologies for smart grid: Paradigms and challenges. *arXiv* **2011**, arXiv:1112.1158.
58. Saad, W.; Han, Z.; Poor, H.V.; Basar, T. Game-theoretic methods for the smart grid: An overview of microgrid systems, demand-side management, and smart grid communications. *IEEE Signal Process. Mag.* **2012**, *29*, 86–105. [CrossRef]
59. Kaur, D.; Islam, S.N.; Mahmud, M.; Dong, Z. Energy forecasting in smart grid systems: A review of the state-of-the-art techniques. *arXiv* **2020**, arXiv:2011.12598.
60. Rostamnia, N.; Rashid, T.A. Investigating the effect of competitiveness power in estimating the average weighted price in electricity market. *Electr. J.* **2019**, *32*, 106628. [CrossRef]
61. Lenzi, A.; de Souza, C.P.E.; Dias, R.; Garcia, N.L.; Heckman, N.E. Analysis of Aggregated Functional Data from Mixed Populations with Application to Energy Consumption. *Environmetrics* **2014**, *28*, e2414. [CrossRef]
62. Chen, Y.; Wu, G.; Sun, R.; Dubey, A.; Laszka, A.; Pugliese, P. A Review and Outlook of Energy Consumption Estimation Models for Electric Vehicles. *arXiv* **2020**, arXiv:2003.12873.