Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 109 (2022) 95–100

www.elsevier.com/locate/procedia

32nd CIRP Design Conference

Semantic knowledge management system for design documentation with heterogeneous data using machine learning

Jack Gammack, Haluk Akay, Ceylan Ceylan, Sang-Gook Kim*

Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA

* Corresponding author. Tel.: +1-617-452-2472 ; *E-mail address:* sangkim@mit.edu

Abstract

Design documentation is presumed to contain massive amounts of valuable information and expert knowledge that is useful for learning from the past successes and failures. However, the current practice of documenting design in most industries does not result in big data that can support a true digital transformation of enterprise. Very little information on concepts and decisions in early product design has been digitally captured, and the access and retrieval of them via taxonomy-based knowledge management systems are very challenging because most rule-based classification and search systems cannot concurrently process heterogeneous data (text, figures, tables, references). When experts retire or leave a design unit, industry often cannot benefit from past knowledge for future product design, and is left to reinvent the wheel repeatedly. In this work, we present AI-based Natural Language Processing (NLP) models which are trained for contextually representing technical documents containing texts, figures and tables, to do a semantic search for the retrieval of relevant data across large corpora of documents. By connecting textual and non-textual data through the use of an associative database, the semantic search question-answering system we developed can provide more comprehensive answers in the context of users' questions. For the demonstration and assessment of this model, the semantic search question-answering system is applied to the Intergovernmental Panel on Climate Change (IPCC) Special Report 2019, which is more than 600 pages long and difficult to read and understand, even by most experts. Users can input custom queries relating to climate change concerns and receive evidence from the report that is contextually meaningful. We expect this method can transform current repositories of design documentation of heterogeneous data forms into structured knowledge-bases which can return relevant information efficiently as well as can evolve to embody manageable big data for the true digital transformation of design.

Keywords: Artificial Intelligence; Machine Learning; Natural Language Processing; Sustainability

1. Introduction

The process of design combines human ingenuity and intuition with the aggregation of knowledge and experience that individuals have built up by designing, building, and testing systems, learning from others' experiences, and educating themselves on phenomena and governing principles. This knowledge base that designers amass is crucial to the evolution of design, as evidenced by the rapid advancement of civilization as information has become more widely accessible.

Design documentation contains rich information detailing past designs that describe functions that are fulfilled, physical objects that are developed, principles that are applied, experiments that are performed, data that is collected and analyzed, known failures and limitations, potential applications, and many other relevant insights. Each document of stored design data holds valuable knowledge that could take an individual or team years to develop independently. The iterative process of design requires applying knowledge and intuition to develop parameters that fulfill certain functions. When the currently attained knowledge and intuition are not enough to develop a solution, one must search past design documentation, as well as other resources, to discover new information. It is important to be able to store and disseminate the body of past design experience to the design community in such a way that users may retrieve relevant information with ease and accuracy in order to improve upon designs and apply findings in their own work.

Due to the massive breadth and depth of topics within the field of design, which span many domains and years, users may not be completely aware of the designs or concepts that exist that may contain solutions to their problems. Users may not be aware of the specific books or journals that they must read, the

10.1016/j.procir.2022.05.220

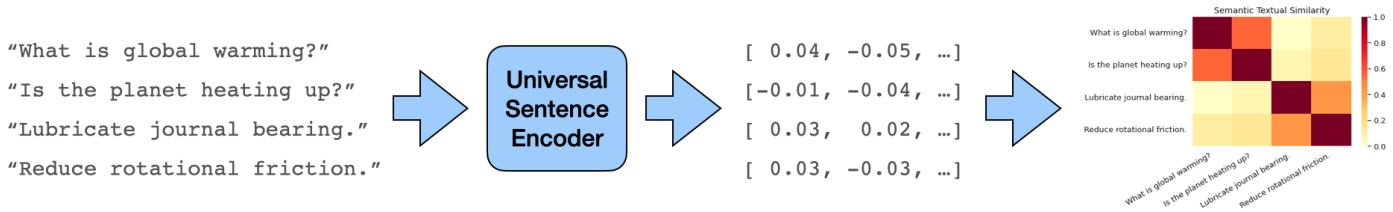

Fig. 1: Semantic similarity of sentence encodings using Google Research’s Universal Sentence Encoder. Adapted from [1].

correct keywords to search, or which questions to ask to the right people to get the answers they need. To improve upon designers’ ability to search and retrieve specific information that is not fully understood from large corpora, such as answers to questions, it is necessary to provide a system that is more sophisticated than simple keyword searching.

Data driven design comes with many heterogeneous forms of data that have historically not been easy to be quantified and represented for machine learning, such as textual language, graphical depictions, and tabular data. These multiple forms of data are crucial to conveying the necessary information to fully pass on the knowledge of the design. When accessing textual information for design purposes, much of the meaning is lost if the graphics and data are not included, especially since this documentation is written with the intention of readers being able to view it all at once.

This work discusses methods in machine learning that may be applied to large corpora of design documentation - proprietary or otherwise - to perform meaningful search that returns relevant design information. The methods allow for relevant heterogeneous design data to be related, stored, and retrieved quickly from an associative database. This framework will allow the current fragmented design knowledge base to grow towards big data for design.

2. Background

2.1. Textual Data Representation

In order for design documentation to be stored, manipulated, and retrieved, it must be represented quantitatively for computational methods [2]. Representation of textual design data is challenging because unlike graphical and numerical data, the elemental alphabetic characters of language do not correspond to an intrinsic quantitative value. The field of Natural Language Processing (NLP) focuses on the representation of language for computational purposes. Prior to the transformation of the field by Artificial Intelligence, NLP language models relied on rule-based approaches to represent text passages and automate systems for tasks such as question-answering [3] with limited performance.

Recent developments in neural network architecture for modeling language sequences, buoyed by the availability of massive datasets and computational power, has transformed the field of NLP with machine learning methods. Feed-forward neural networks have demonstrated how words in a dataset can be represented by a distribution of vectors in a semantic space [4]. Words occurring in similar contexts are represented by vectors with close proximity meaning that the distance between vector representations can be used as a metric for semantic similarity.

Neural network architectures such as the *Transformer* have been introduced [5] and are designed specifically for representing sequences of words. The language model BERT [6] by Google Research uses this architecture to represent language quantitatively, and allows further *Fine Tune* training to modify the model’s outputs depending on application-specific needs. For example, BERT can be fine-tuned on the NLP information retrieval task of *question-answering* by subsequently training the model on crowd-sourced datasets [7] to extractively return answer spans from context passages, based on questions provided.

The Transformer language modeling architecture has also been applied to encode entire sentences into vectors for returning information from long-form documents. Google Research’s Universal Sentence Encoder (USE) [1] leverages these encoded sentence embeddings, pre-trained on many language modeling tasks, to create models that encode any sentence into a high-dimensional vector that represents the semantic meaning of the sentence. These sentence embeddings may be directly compared with each other through vector similarity metrics to produce scores quantifying the similarity of pairs of sentences, as shown in Figure 1. These embeddings may be stored in a database to produce a semantic search system which takes a sentence as input and returns the most semantically similar sentences from the corpus.

By introducing an additional neural layer to estimate the relationship between language of questions and answers, Google Research extended the USE by allowing the model to take a question as an input sentence and return the sentence in the corpus that best resembles a semantic answer to the question. [8] Google’s ”Talk to Books” [9] website applies a similar system to every sentence in the Google Books corpus, creating a publicly available conversational system where users may input a sentence, and the model returns the sentence that most naturally follows in a conversation. By pre-encoding the entire corpus once, these models return outputs extremely quickly to custom user inputs by encoding the input and using a fast nearest-neighbor search algorithm for vector spaces to find the output [10].

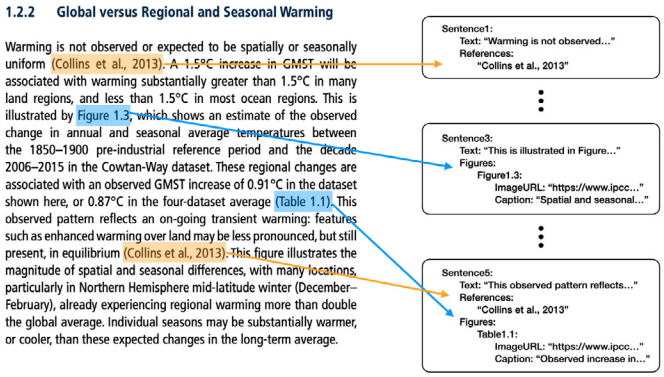

Fig. 2: Creation of associative array database for storing textual data within a document along with references to non-textual data for retrieval.

2.2. NLP for Design

We have examined how textual design data may be utilized to guide structured early-stage design. By using AI-based language models to represent textual descriptions of designed systems, we have demonstrated how semantic vector space corresponds to the design requirements in the functional domain and also how the quality of designs may be characterized with regard to metrics of functional independence [11].

Axiomatic Design was first introduced to CIRP [12] as a principled methodology for structuring functional requirements and mapping them to design concepts in the form of physical solutions. Although these principles can be challenging for non-experts to implement in their design practice, this hierarchical and structured representation of design provides a useful framework for leveraging computational systems for processing design data at the stage of conceptual design [13].

AI-based language models can be used to both quantitatively represent textual data and train on applied tasks such as information retrieval. The specific task of question-answering can be applied to retrieve high-level design information from documentation. Based on Axiomatic Design principles of mapping the functional domain to the physical domain, we have demonstrated how recursive question-answering can be used to extract structured functional requirements and design parameters from otherwise unstructured free text [14]. This process was validated by comparing the NLP-based results to those of real human designers completing identical tasks of identifying requirements from design documentation [15].

However, this past research only processed design data of textual form. Since design documentation differs from other textual documents in that it is heavily reliant on figures, tables, and the relationships between the functional and physical domain, novel methods are needed to make use of this past data. The system presented in this work demonstrates how heterogeneous forms of design documentation can be processed to answer questions and make use of big data in textual and graphical form.

3. Methods

3.1. Data Preparation

In order for the various forms of data within design to be utilized by machine learning algorithms, the data must be pre-processed and organized such that models may understand which data is intrinsically linked. As human readers of design documentation, it may be trivial to understand which graphics are related to certain sections of text simply by reading where figures and tables are referenced. However, computers cannot read and make these associations without an algorithm that can create structures for storing and retrieving these forms of data.

We have created a set of methods for storing these heterogeneous forms of data and their relationships in a document-based associative array. Creating an entry in the database for each design document, we store the relevant sentences contained in the document. We also store the available information for every non-textual form of data within each document, such as figures, tables, and citations. These non-textual forms of data contain information such as captions for figures and tables, URLs for images and webpages, and any other relevant information. An algorithm is created to search the text citing those captions of non-textual data such that the relevant textual data explaining each non-textual data is easily discoverable.

Figure 2 shows an overview of creating the associative array database for an excerpt of text within a document. The use of the database allows data from documents with varying formats to have a common storage and retrieval system. However, as design documentation is not created with a uniform system for referencing non-textual data, it is necessary to alter the algorithms depending on the specific set of documents used in order to store the relevant data.

Since not all data necessary to produce a system for storing and outputting image files are contained within design documentation, some manual intervention is required. For example, URLs for images are rarely included in PDF files of research papers. Ideally, human designers could structure their design documentation such that references to images and tables

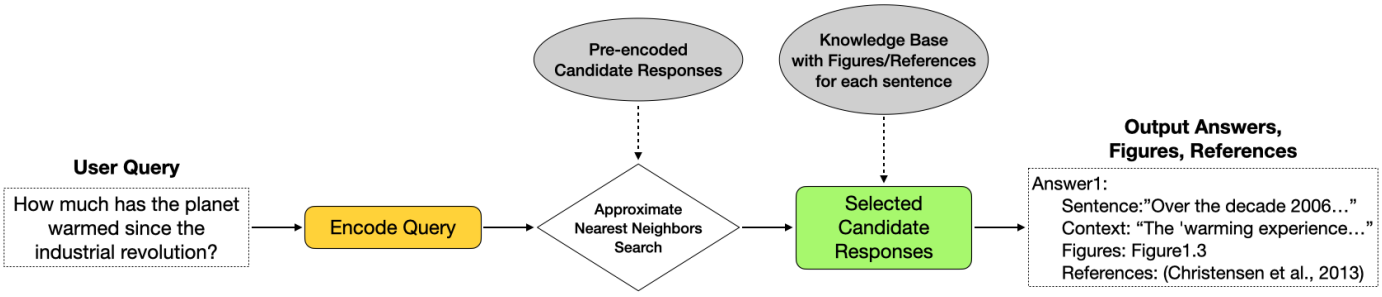

Fig. 3: Answer retrieval from custom user queries using USE-QA and ANN. Adapted from [10].

are readily available to be read by an algorithm with minimal alterations.

3.2. Sentence-Level Semantic Search

Utilizing our database of textual data and references, we apply Google's Universal Sentence Encoder for Question Answering (USE-QA) to encode each sentence in each document into a 512-dimensional vector space as if it were an answer to a question. Each sentence embedding is stored in an index that maintains references to the original sentence in the associative database through a unique identifier. To represent the non-textual elements in each document, such as figures and tables, their captions and citing sentences are also encoded and stored in the index. By inputting a query in natural language, we search this index for the most reasonable answer to the question using an Approximate Nearest Neighbors (ANN) algorithm [16]. Since we encoded the non-textual elements into the searchable index, figures and tables may appear as answers to user queries. The use of ANN and pre-encoded candidate responses provides near-instant answers to questions without having to check every single sentence in the database, which could contain many thousands of sentences.

Figure 3 shows a flowchart of the process of retrieving textual answers as well as related non-textual data sources from custom user queries. Once sentence-level answers are found to the input questions, the non-textual data is accessed via the sentence's unique identifiers. Information in the database may be used to output the data however is best for the desired application.

3.3. Span-Level Question Answering

After retrieving the sentences that appear to contain answers to user queries, we attempt to discover the span-level answers to questions. Google Research's BERT fine-tuned on question-answering data (BERT-QA), described earlier, provides a pre-trained model to perform this function. BERT-QA takes as input a question and a context and returns the most likely starting and ending position of the answer within the context. Due to BERT-QA's complexity and necessity to run the model on both the question and context, it is infeasible to ask the question to every sentence in our corpus every time the user wants to find an answer to a new question. Instead, we may use BERT-QA to find span-level answers within the candidate responses that USE-QA produces. This 2-stage semantic search process greatly reduces run-time and the number of computations [17]. Additionally, we may leverage our associative database to run BERT-QA on all of the textual information that is related to our candidate answer sentences, such as related figure captions or adjacent sentences, to attempt to get better answers to our questions from relevant sources covering both textual and non-textual data.

In our previous papers, a similar method is used to extract functional requirements from text documents. Applying this in conjunction with sentence-level semantic search on a large corpus allows users to quickly filter the amount of text to search and extract design information.

4. Case Study

4.1. Data Preparation

Design documentation, particularly in industry, is generally proprietary and not available to the public. Additionally, design documentation on a specific domain is not usually collected in one place or formatted such that individuals outside of a company may access and alter the data as necessary. Much design documentation that is publicly available, such as patent reports, are not written with the intent of disseminating the knowledge within, but instead are written so as to obscure intricate details that may be necessary to reproduce the design. Due to the lack of available large-scale design data, we test our methods on a long-form research document over a specific topic as a proxy for a design documentation corpus, which discusses design parameters required to achieve certain functions and intends to educate its audience.

We apply our data preparation and semantic search systems to the Special Report on Global Warming of 1.5 °C (SR1.5) by the Intergovernmental Panel on Climate Change (IPCC) [18]. The SR1.5 discusses the "impacts of global warming of 1.5 °C above pre-industrial levels and related global greenhouse gas emission pathways, in the context of strengthening the global response to the threat of climate change, sustainable development, and efforts to eradicate poverty." It is an influential document that is utilized by policymakers, researchers, and the general public for making important decisions on reducing human impact on

the environment and ensuring longevity of civilization. As in the field of design, climate change research involves many graphical depictions with textual descriptions in order to relay information, and it is important for users of the research to be able to retrieve informative answers to questions in order to improve upon their practice.

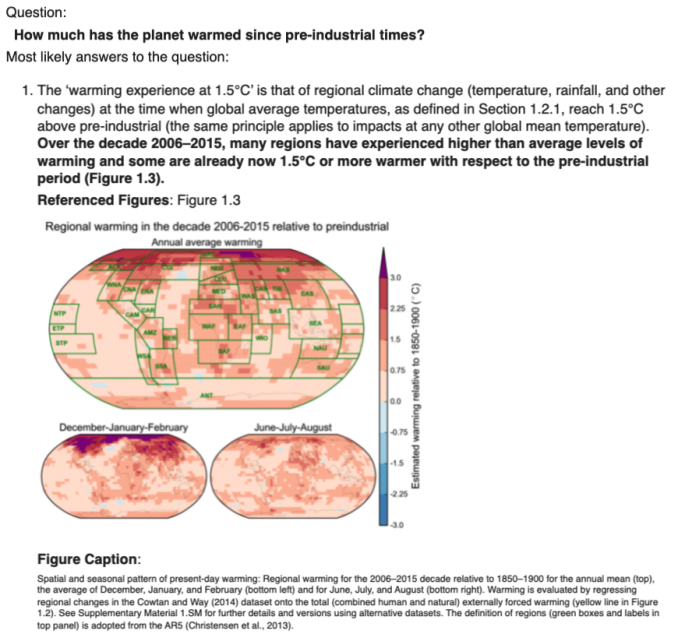

Fig. 4: Question answering for retrieving sentences and related figures from a large corpus with USE-QA.

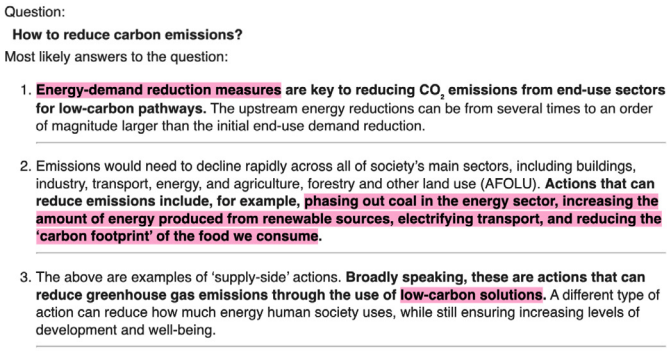

Fig. 5: Span-level identification of answers to questions from a large corpus with BERT-QA after retrieving sentence-level answers.

However, the report is very substantial and contains a wide range of information and recommendations from hundreds of research studies, totaling 7,760 sentences along with 84 figures and 36 tables, which is hard for any one person to read and comprehend. With such a large number of sentences, many relating to topics of global warming, it can be very difficult to use keyword searching to find relevant answers to questions. Simply searching keywords like ”warming”, ”temperature”, and ”increase” can return thousands of possible sentences. In order to make this report and other IPCC documents more accessible, we create a question answering system that allows users to quickly get the top-N sentences and contexts that answer their question, along with relevant figures, tables, and citations for further reading.

To prepare the data for our associative database, we use the Natural Language Toolkit [19] to separate the unstructured text from the corpus into sentences. The SR1.5 has a consistent method for referencing figures, tables, and paper citations, such that references are labeled as ”Figure X.Y” or ”Table X.Y”. We create a substring-searching algorithm to find these references within each sentence. The SR1.5 website contains image URLs for every figure in the report, which we manually add to our database. We use MongoDB [20], a cloud database system for associative arrays that allows fast searching based on unique identifiers, to store our data in a publicly accessible way.

4.2. Semantic Search

To create our semantic search system through question answering, we use USE-QA to encode each sentence as an answer, then store the embedding and unique identifier for the sentence within a data structure. After embedding the user's input question, we utilize Spotify's ANN algorithm [21] to search for the nearest answers. Once the best answers are found, the database is searched to find the attached figures, tables, and citations. We use HTML to output the results to the user.

Figure 4 shows an example of the output when asking a question that users may be interested in. The sentence containing the best answer is displayed in bold print. Since this sentence contains a reference to a figure, the figure URL and caption are found using the database and displayed. We observe that both the sentence and figure are very relevant and directly answer the question while providing additional context.

4.3. Span-Level Question Answering

After retrieving the N most relevant answers to questions, we may search each sentence for a span of text that provides a more direct answer to the user's question using BERT-QA. Figure 5 shows the results of finding span-level answers on how to reduce carbon emissions after searching for the 3 sentences that best answer the question. These sentences and their contexts do not contain any non-textual data references, so none are outputted. We observe that both the sentence-level and span-level answers appear to answer the question directly.

4.4. Limitations

The models used in this case study are trained on general language data, such as the English Wikipedia, to understand what words mean in specific contexts. The vast majority of words in the IPCC report occur in everyday natural language, so pre-trained language models performed fairly well. However, when applying these methods to a corpus of documents that contain very uncommon words, or common words that have unusual meanings, it may be necessary to fine-tune the language model on domain-specific data. Future work may address this need and the performance improvements of fine-tuning.

5. Conclusion

The eventual goal of this work is to develop a knowledge management framework for design. Documentation of past design successes and failures should be rich with learnable information, thinking, decisions, and knowledge that can be used to guide and evaluate current and future generations of product design. Current knowledge management systems provide keyword-based searches and do not capture and manage conceptual design thinking and high-level decisions in a manner machines can understand, store and retrieve when needed. It is also challenging for current NLP systems to represent design documents with heterogeneous data formats. Design data representation is a major bottleneck to create big data in design, together with current design documentation practice in industry which neglects to collect and structure generated knowledge during the design process.

In order to represent design data from textual descriptions, figures and numerical simulations, we developed an AI-based knowledge management system by which designers can seek out and discover answers to specific questions based on a rich history of previous design experiences from similar contexts. By applying AI-based NLP models pre-trained on vast amounts of textual data, we could develop a system for reading design documentation into a vectorized database for performing rapid semantic retrieval of answers to user questions. Utilizing structured relationships between images and the citing descriptions within the text, we have demonstrated the ability to incorporate relevant non-textual graphical data into the searchable text database. In a case study involving long-form documentation, answers and relevant figures from aggregated climate change research reports could be retrieved based on user-inputted queries.

We envision widespread application of our semantic search-based knowledge management system for processing existing fragmented and heterogeneous format design documentation to build a structured knowledge-base. This will help to achieve the true digital transformation of design and the design efforts and knowledge will not be siloed but to be shared within the enterprise to maximize likelihood of success when addressing complex problems. This research is intended to enable the digital transformation of the practice of design.

6. Acknowledgements

This work was supported by MIT/SenseTime Alliance on Artificial Intelligence, and the National Science Foundation (NSF) Leading Engineering for America's Prosperity, Health, and Infrastructure (LEAP HI) program, award number 1854833.

References

[1] D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. St. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, B. Strope, R. Kurzweil, Universal sentence encoder for English, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, 2018, pp. 169–174.

[2] S.-G. Kim, S. M. Yoon, M. Yang, J. Choi, H. Akay, E. Burnell, Ai for design: Virtual design assistant, CIRP Annals 68 (1) (2019) 141–144.

[3] E. Riloff, M. Thelen, A rule-based question answering system for reading comprehension tests, in: ANLP-NAACL 2000 Workshop: Reading Comprehension Tests as Evaluation for Computer-Based Language Understanding Systems, 2000, pp. 13–19.

[4] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Advances in neural information processing systems, 2013, pp. 3111–3119.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in neural information processing systems, 2017, pp. 5998–6008.

[6] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: NAACL-HLT, Vol. 1, 2019, pp. 4171–4186.

[7] P. Rajpurkar, J. Zhang, K. Lopyrev, P. Liang, Squad: 100,000+ questions for machine comprehension of text, in: EMNLP, 2016, pp. 2383–2392.

[8] Y. Yang, S. Yuan, D. Cer, S.-y. Kong, N. Constant, P. Pilar, H. Ge, Y.-H. Sung, B. Strope, R. Kurzweil, Learning semantic textual similarity from conversations, in: Proceedings of The Third Workshop on Representation Learning for NLP, 2018, pp. 164–174.

[9] Google. Talk to books [online] (2018).

[10] Y. Yang, G. H. Ábrego, S. Yuan, M. Guo, Q. Shen, D. Cer, Y. Sung, B. Strope, R. Kurzweil, Improving multilingual sentence embedding using bi-directional dual encoder with additive margin softmax, in: International Joint Conference on Artificial Intelligence, 2019, pp. 5370–5378.

[11] H. Akay, S.-G. Kim, Design transcription: Deep learning based design feature representation, CIRP Annals 69 (1) (2020) 141–144.

[12] N. P. Suh, S. Kim, A. Bell, D. Wilson, N. Cook, N. Lapidot, B. von Turkovich, Optimization of manufacturing systems through axiomatics, Annals of the CIRP 27 (1) (1978) 383–388.

[13] H. Akay, S.-G. Kim, Artificial intelligence tools for better use of axiomatic design, in: IOP Conference Series: Materials Science and Engineering, Vol. 1174, IOP Publishing, 2021, p. 012005.

[14] H. Akay, S.-G. Kim, Reading functional requirements using machine learning-based language processing, CIRP Annals 70 (1) (2021) 139–142.

[15] H. Akay, M. Yang, S.-G. Kim, Automating design requirement extraction from text with deep learning, in: International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Vol. 85390, 2021, p. V03BT03A035.

[16] W. Li, Y. Zhang, Y. Sun, W. Wang, M. Li, W. Zhang, X. Lin, Approximate nearest neighbor search on high dimensional data—experiments, analyses, and improvement, IEEE Transactions on Knowledge and Data Engineering 32 (8) (2019) 1475–1488.

[17] H. Bast, B. Björn, E. Haussmann, Semantic search on text and knowledge bases, Foundations and Trends in Information Retrieval 10 (2-3) (2016) 119–271.

[18] Intergovernmental Panel on Climate Change. Special report on the impacts of global warming [online] (2018).

[19] S. Bird, E. Klein, E. Loper, Natural language processing with Python: analyzing text with the natural language toolkit, " O'Reilly Media, Inc.", 2009.

[20] MongoDB. The application data platform [online] (2021).

[21] E. Bernhardsson. Annoy: Approximate nearest neighbors in c++/python [online] (2018).