IOP Conf. Series: Materials Science and Engineering 1174 (2021) 012005 doi:10.1088/1757-899X/1174/1/012005

Artificial Intelligence Tools for Better Use of Axiomatic Design

Haluk Akay, Sang-Gook Kim

Department of Mechanical Engineering, Massachusetts Institute of Technology
Cambridge MA, United States

sangkim@mit.edu

Abstract: Axiomatic Design (AD) provides a powerful thinking framework for solving complex engineering problems through the concept of design domains and diligent mapping and decomposition between functional and physical domains. Despite this utility, AD is yet to be implemented for widespread use by design practitioners solving real world problems in industry and exists primarily in the realm of academia. This is due, in part, to a high level of design expertise and familiarity with its methodology required to apply the AD approach effectively. It is difficult to correctly identify, extract, and abstract top-level functional requirements (FRs) based on early-stage design research. Furthermore, guiding early-stage design by striving to maintain functional independence, the first Axiom, is difficult at a systems level without explicit methods of quantifying the relationship between high-level FRs and design parameters (DPs). To address these challenges, Artificial Intelligence (AI) methods, specifically in deep learning (DL) assisted Natural Language Processing (NLP), have been applied to represent design knowledge for machines to understand, and, following AD principles, support the practice of human designers. NLP-based question-answering is demonstrated to automate early-stage identification of FRs and to assist design decomposition by recursively mapping and traversing down along the FR-DP hierarchical structure. Functional coupling analysis could then be conducted with vectorized FRs and DPs from NLP-based language embeddings. This paper presents a framework for how AI can be applied to design based on the principles of AD, which will enable a virtual design assistant system based on both human and machine intelligence.

1. Introduction

Since first introduced to the ASME and CIRP community in the late 1970s, Axiomatic Design (AD) has provided principles for complex systems design and has been advanced as a powerful methodology for reducing complexity and formalizing the process by which latent needs are translated into concrete functional requirements (FRs) and systematically addressed by design parameters (DPs) [1-3]. The range of AD applications has been very wide and broad. AD has been successfully applied to product design in various fields including automotive [4], electronics [5], manufacturing equipment [6], and MEMS devices [7]. Another popular area of application is manufacturing processes [8] and manufacturing systems design [9]. More recent applications include large, complex socio-technical systems such as healthcare systems [10], enterprise systems, transportation systems, supply chain management, and information system architecture. Although AD has its root in manufacturing and product design, it has also been applied to software design and software development [11].

Despite the value that AD brings to engineering design and production practice, it has not seen widespread adoption in industry, with training in AD remaining limited to mostly academic settings [12]. While AD is a useful thinking framework for expert designers, extensive familiarity with its methodology is required to implement it in design practice. Correctly identifying FRs and abstracting them into top-down hierarchical structure while mapping them to the physical domain is difficult for junior designers with little exposure to this methodology. Even for seasoned designers, identifying functional coupling, extracting FR-DP structures, and analyzing design matrices become challenging when a problem scales up and decomposition goes beyond a few levels along the hierarchy [13].

At the time of AD's conception in the 1970s and until quite recently, the paradigm for computational design aids has been rule-based. However, with the current resurgence of AI bolstered by low-cost computing hardware, novel neural network architectures, and an abundance of digitalized data, the move away from rule-based methods towards models built on learned parameters from data (so-called *Deep Learning*) has proven transformational for certain fields. The field of computer vision, which historically relied on image filters carefully designed by experts, was revolutionized when convolutional neural networks (CNN) trained on examples of handwriting to learn these filters and outperform the status quo [14]. Similarly, the field of natural language processing (NLP) has benefitted tremendously from the application of deep neural networks trained on massive text datasets to learn representations of language [15] which can be used to perform a number of automated language tasks.

While design has benefited from computational aids such as CAD/CAM tools and use of machine learning for generative design models [16], the field of design has yet to be transformed by AI in the manner by which computer vision and NLP have been. By representing design knowledge for machines to comprehend for storage, manipulation, and retrieval, in a similar way to how language has been represented in NLP, the field of design can truly begin to benefit from methods in AI. In this paper, we propose and show how Axiomatic Design with AI tools provides the necessary framework for formalizing design knowledge representations which has the potential to transform AD thinking from a framework to an accessible methodology.

2. Background

2.1. Axiomatic Design relates the functional domain to the physical domain by modeling interdependencies in design with a matrix framework where functional requirements and design parameters can now be represented as vectors in design space with AI-based Natural Language Processing (NLP) technology. This section provides an overview on these design knowledge representations as well as background on recent advances in AI and NLP.

2.2. Functional Structure in Axiomatic Design

The first step of applying Axiomatic Design (AD) to a problem is identifying the functional requirements (FRs) which must be addressed by design parameters (DPs). AD starts by identifying the highest-level, most overarching FR (what) of a problem, and mapping it to the highest-level DP (how). The highest FR-DP pair need to be decomposed top-down as shown in Figure 1. This FR-DP tree structure is a useful framework for representing design not only because it preserves the hierarchical relationships between perceived needs and conceptualized solutions, but also because it lays a foundation for mapping designs with computational methods.

However, in practice, identifying the key FR and conducting a correct decomposition has been challenging. Differentiating FRs from DPs can be difficult without training in AD theory, and for a complex problem, a multi-level functional hierarchy may need to be abstracted to extract the highest-level FR. As problems scale in breadth, domain expertise across a wide range of fields may be required for a designer to comprehensively identify all necessary FRs and matching DPs for a design concept. A thorough AD analysis is very valuable but very laborious in terms of the human expertise required to complete it.

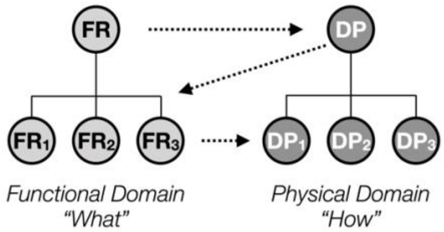

Figure 1: Structured decomposition of functional requirements and design parameters

2.3. Evaluating Functional Coupling

Once the FRs and DPs of a design are identified, their functional coupling needs to be evaluated to assure and maintain the functional independence of the design; this is a critical metric for a good design. One of the fundamental relationships defined in AD is between the functional domain (what) and the physical domain (how). FRs are related to DPs through the design equation, shown in (1), which represents one layer of the hierarchy introduced in Figure 1.

$$\begin{Bmatrix} FR_1 \\ \vdots \\ FR_n \end{Bmatrix} = \begin{bmatrix} A_{11} & \cdots & A_{1n} \\ \vdots & A_{ij} & \vdots \\ A_{n1} & \cdots & A_{nn} \end{bmatrix} \begin{Bmatrix} DP_1 \\ \vdots \\ DP_n \end{Bmatrix} \tag{1}$$

The terms of the design matrix A_{ij} represent the effect each DP has on each FR. Terms along the diagonal where $i = j$ should equal 1 as a given DP_n is designed to address FR_n. For a design exhibiting perfect functional independence, where each DP addresses and affects only one FR, the number of DPs will correspond to the number of FRs, and the matrix A will be diagonal, where $A_{ij} = 0$ when $i \neq j$. Such functionally independent designs are the holy grail of systems engineering. As designs scale in number of FRs and complexity, coupling between the functional and physical domain may result in performance inefficiency, as in the case of pre-industrial revolution steam engines [17], and even catastrophic safety issues when designs are updated without an understanding of functional interdependencies. Conversely, functionally independent systems are much simpler to modify, and often result in better performance even experienced by end users as in the classic AD example of the uncoupled faucet where separate vertical and horizontal levers to control water flow rate and temperature are easier to use than dual knob faucets [18].

Identifying and measuring functional coupling is not trivial in practice, however, especially for systems containing many FRs and DPs. Furthermore, many real-world design cases are not easily characterized into the discrete categories of “uncoupled,” “decoupled,” or “coupled.” To address this issue, metrics of functional independence have been developed [19-20] based on the matrix relationship in (1). These metrics seek to characterize not only if a system is coupled or not, but also *how* coupled a system may be. Based on the values of the design matrix A, the metrics of *Reangularity* (R) and *Semiangularity* (S), expressed in (2), (3) reflect the degree to which DPs affect each other, and affect the set of FRs defining the functional domain. R and S values close to 1 indicate ideal functional independence, and values close to 0 indicate worst-case total functional coupling.

$$R = \prod_{\substack{i=1,n-1 \\ j=1+i,n}} \left(1 - \frac{\left(\sum_{k=1}^{n} A_{ki} A_{kj}\right)^2}{\left(\sum_{k=1}^{n} {A_{ki}}^2\right)\left(\sum_{k=1}^{n} {A_{kj}}^2\right)}\right)^{1/2} \qquad S = \prod_{j=1}^{n} \left(\frac{|A_{jj}|}{\left(\sum_{k=1}^{n} A_{kj}^2\right)^{1/2}}\right) \tag{2) (3}$$

The challenge with implementing such metrics to evaluate real-world designs is that design matrix elements are difficult to quantify, especially in the case of heterogeneous units of measure and qualitative FRs in text form. For example, in the classic AD case of faucet design, if one FR pertains to flow rate, measured in units of volume or mass over time, and another FR pertains to temperature, quantifying how a physical DP affects each FR is not straight forward. Furthermore, while simple systems of few FRs and DPs with quantifiable values can be characterized using designers’ intuition, when problems scale up with numerous FRs, accurately computing measurements like R and S can be challenging, which is why such metrics, while theoretically powerful in characterizing systems, have not seen widespread use.

For such challenging cases, AI and NLP models can now be used as a method for representing FRs and DPs in a multi-dimensional vector space where R and S can be quantitatively measured, and thereby functional independence can be quantitatively determined. In our study with NLP modeling, 2^{10} dimensional vector space is used to represent FRs and DPs.

2.4. AI-based Natural Language Processing

In early-stage design, user needs are often communicated with words and sentences, and upstream design efforts such as stakeholder interviews, process proposals, and design specifications are often documented and stored in textual format. In order to analyze such textual data with methods beyond conventional keyword searches relying on rule-based taxonomies, AI methods specifically in natural language processing (NLP) may be applied.

A key functionality in NLP is representing language sequences in a manner which is computationally accessible. Strings of alphabetical characters have no inherent quantitative meaning, but neural networks trained on large document datasets can be trained to embed semantic meaning of language into vectors based on probabilistic modeling tasks [21]. State of the art language models utilize deep neural network architectures trained on billions of words to comprehensively encode language meaning and may be further trained or “fine-tuned” to perform specific tasks. *Bidirectional Encoder Representations from Transformers* (BERT) is one such language model developed by Google AI [15].

BERT is a machine learning-based language model which is able to represent sequences of language (words in sentences) by producing vector representations where words which have similar meaning are placed nearby each other in a multi-dimensional semantic feature space. Vector representations from BERT are contextually dynamic in that the same word’s vector will change slightly depending on the context it exists in (neighboring language). BERT is trained in two steps. The first pre-training step is a general training process and is “unsupervised” meaning the model learns from un-annotated text documents, training on tasks such as predicting the identity of randomly masked words in a sentence, and if two sentences should follow one another consecutively. The second fine-tuning step depends on the application of BERT and is usually supervised learning meaning that example inputs and target output pairs are provided. Due to the manner in which design decomposition can be conducted by extracting the answers to “what” and “how” type questions, BERT fine-tuned on the NLP task of question-answering can be applied as a form of *Hybrid Intelligence* [22] to automate many of the steps in AD. We have previously demonstrated how highest-level FRs can be extracted with question-answering [23], how *Design Reading* can be automated with recursive question-answering to extract and structure large numbers of FRs for a design case [24], and that such extracted functional hierarchies demonstrate agreement with the judgements made by human designers [25]. In this work, we demonstrate how applying such AI-based NLP models can aid with challenging tasks in Axiomatic Design to make it more accessible to the engineering design community and for wider-spread use in industry.

3. Method

The key tasks of a designer using Axiomatic Design involve identifying functional requirements (FRs), decomposing FRs from the top-down to produce a hierarchical structure, and determining functional interdependencies to evaluate a designed system. Representations of language from AI-based models can be used to automate the difficult aspects of these tasks.

3.1. Question-Answering for Identifying and Structuring FRs

Detailed designs may be developed by identifying a key FR and zigzagging between the functional and physical domains to identify corresponding DPs and decompose to a hierarchical structure. Depending on the complexity of the problem and the designer’s expertise in AD theory, this can be a difficult process, but AI-based NLP can be leveraged to provide aid.

AD clearly defines the functional domain as the “what” and the physical domain as the “how” of design. For example, FRs are “what we want to achieve” and DPs are “how we achieve them” [1]. While such prompting statements are simple, answering them accurately to identify FRs and DPs can be challenging for those less familiar with AD. However, if sufficient documentation about a design space is available, the extraction and structuring of FRs and DPs can be automated using NLP-based question-answering. Language models such as BERT can be fine-tuned on the task of question-answering, which is a standard information retrieval task in the field of NLP. This task considers a context document and a question as inputs, and probabilistically determines the span of text within the context which has the maximum likelihood of being the answer to the input question, and returns the indices of the answer as output. BERT can be trained on this task with a dataset of crowd-sourced

IOP Conf. Series: Materials Science and Engineering 1174 (2021) 012005 doi:10.1088/1757-899X/1174/1/012005

context-question-answer examples, such as the Stanford Question Answering Dataset (SQuAD) [26] and applied to AD decomposition. By choosing "what" and "how" type questions that build in specificity with information content previously retrieved, high-level FRs and DPs may be decomposed automatically into detailed lower-level FRs, where the questions prompting their retrieval are informed by a previously extracted hierarchy, as visualized in Figure 2.

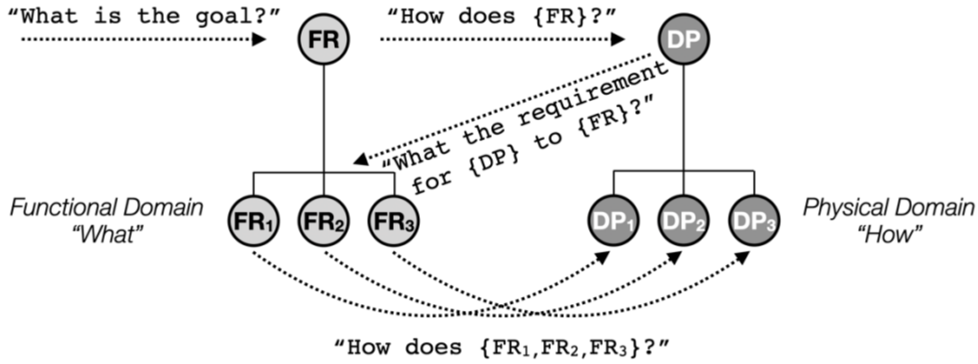

Figure 2: Question-Answering for Axiomatic Design Decomposition

An example of the automated decomposition steps described in Figure 2 can be considered to further elucidate the power of question-answering for application to AD. While actual design specification documents from industry are the ideal target of deploying this automated decomposition method, we will demonstrate an example based on the abstract of an academic paper describing the design of a bio-inspired compliant locking device [27]. Paper abstracts are not only publicly available and distributable, but also can be expected to be rich with high-level FRs and DPs and make a dense testing ground for demonstrating an extraction method.

As a result of this recursive application of question-answering, we can automatically produce an AD decomposition and a set of structured FRs and DPs without any prior taxonomical information about this design. Due to the extractive nature of the information retrieval, the grammatical syntax of some of the questions juxtaposed with FRs are nonstandard, such as "How does stability?" but they elicit DPs from the context, i.e., "split locking mechanism." With this method, design documentation can be processed to determine an Axiomatic Design functional structure, allowing designers to experience the benefits of this analysis without expertise in AD theory that would have been required to manually produce these decompositions.

Table 1: Recursive Question-Answering Example of Axiomatic Design Decomposition

Design context document [27]: A device reminiscent of the mammalian spine has been designed and built with the ability to lock each individual joint in a string of ball joints. The assembly may be controlled in a manner similar to other hyper-redundant robots, with the added advantage of locking in a straight or axial position. Locking is achieved by orienting two mating collars in a singular configuration that forces compression against neighboring collars and prohibits bending or rotation. Locking is desirable for added strength in supporting objects, as well as for stabilization and power efficiency when bending is not necessary. The split locking mechanism represents a biologically inspired structure with added strength and stability for use in robotics. **Legend:** Functional Requirement Design Parameter	Q: What is the goal? **FR**: added strength in supporting objects Q: How does {added strength in supporting objects}? **DP**: locking Q: What is the requirement for {locking} to {added strength in supporting objects}? **FR1**: stability Q: How does {stability}? **DP1**: split locking mechanism **FR2**: prohibits bending or rotation Q: How does {prohibits bending or rotation}? **DP2**: forces compression against neighboring collars **FR3**: straight or axial position Q: How does {straight or axial position}? **DP3**: locking **FR4**: string of ball joints Q: How does {string of ball joints}? **DP4**: spine

IOP Conf. Series: Materials Science and Engineering 1174 (2021) 012005 doi:10.1088/1757-899X/1174/1/012005

3.2. Axiomatic Design Analysis with Language Vector Embeddings

Once the FRs and DPs of a designed system have been defined, such a system may be evaluated in Axiomatic Design with respect to the first axiom regarding maintenance of functional independence. While the definitions of a "coupled" versus "uncoupled" system are quite clear, identifying functional interdependencies and measuring them can be difficult in practice when units of measure do not align and FRs are qualitative in nature. For such challenging cases, NLP models may be used as a method for representing FRs and DPs in a feature space where interdependencies may be quantitatively measured. We have previously demonstrated [28-29] how metrics of functional independence may be accurately estimated for the classic AD example of coupling in faucet design, and we will show an extension of this case with the following study.

The pre-trained language model BERT (fine-tuned previously on the task of question-answering) may also be used to produce vector representations of language. Consequently, sentence spans describing FRs and DPs may be converted into vector form by averaging the vectors of words making up the descriptions. Such representations are designed to encode the semantic meaning of language into vector space such that more similar words occur closer together. As a result, semantic similarity can be quantified in terms of the cosine similarity between word vectors. For example, the semantic similarity between the words "motor" and "engine" is 0.834, while the similarity between words "motor" and "donut" is 0.383, using vector representations from BERT. If we consider the case of faucet design, we can note that the FRs of the design refer to controlling water (1) "temperature" and (2) "flow rate." If we consider descriptions of DPs for the coupled case in Design A, we can note that both DPs reference temperature ("hot" or "cold") and "flow rate". However, the DPs of uncoupled Design B are also semantically uncoupled in that there is no mention of "flow rate" in DP1 and no mention (explicit or similar to) "temperature" in DP2. To quantify these semantic relationships in the functional domain, we can obtain vectors of each FR and DP description and use these quantitative representations to estimate the design matrix A relating the functional to physical domain in AD. In turn, metrics for coupling *Reangularity (R)* and *Semiangularity (S)* may be computed. We can reproduce our previous study in [29] to visualize how differently phrased descriptions, all of which are similar to those explicitly stated in the left of Figure 3, describing faucet designs. The descriptions of the designs cluster around R and S values accordingly, and the semantic representations reflect the metrics of functional independence from Axiomatic Design.

Functional Requirements of a Faucet System:
FR1: allow control of water temperature
FR2: allow control of water flow rate

Design A (coupled):	**Design B (uncoupled):**
DP1: One valve to control flow rate of hot water	DP1: One lever to control temperature of water
DP2: One valve to control flow rate of cold water	DP2: One lever to control flow rate of water

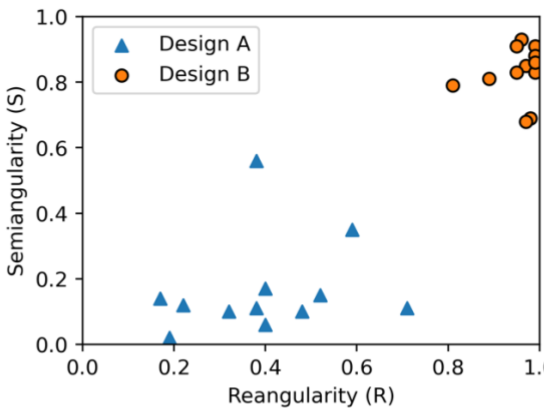

Figure 3: left: images and descriptions of faucet design; right: plot of R and S metrics of coupling for the faucet design based on vectorized descriptions of FRs and DPs from [29]

4. Discussion and Conclusion

Early-stage design can be subjective, requiring expertise to practice. Axiomatic Design provides a powerful framework for resolving customer needs into concrete functional requirements, and mapping these to the physical domain, but actually implementing these steps can be challenging, and training junior designers to practice AD is difficult. While understanding a design space and answering the key questions of AD decomposition is challenging for junior designers, NLP models trained specifically on retrieving information in a question-answering format are able to perform the identification of FRs, mapping to DPs from textual design descriptions, and decomposing them to lower levels. A recursive implementation of Google's language model BERT, fine-tuned for question-answering demonstrated that NLP was applied to automate the fundamental steps of AD. This automated decomposition method can be extended beyond the functional and physical domains to link process variables to early-stage design and create a holistic, digitalized AD-based design knowledge of production steps. The difficulty of manually producing such design decompositions has hindered the implementation of AD in industry, but using the AI-based automated or human-assisted hybrid methods, industry may finally truly benefit from the power of Axiomatic Design thinking.

By producing vector representations of designs based on textual descriptions, semantic similarity can be translated into AD metrics for functional independence. If the detection of functional coupling could be automated using the NLP-based knowledge representation methods shown in this work, functional interdependencies of large-scale systems could be quickly assessed to highlight bad designs and to avoid any potentially catastrophic safety issues.

Axiomatic Design has provided powerful thinking framework for designers but has been also challenging to use and implement at industrial practices. By applying AI-based models to perform the most challenging steps of Axiomatic Design, namely extracting and structuring functional requirements, and evaluating designed systems on the basis of functional independence, we have shown that design knowledge can be represented in a form that machines can understand. We believe that this will enable Axiomatic Design to finally become an accessible method to benefit practitioners of design in a range of industries.

5. Acknowledgements

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

References

[1] Suh NP, 1990. *The Principles of Design*. New York: Oxford University Press.
[2] Suh NP, Bell AC, Gossard DC, 1978. On an axiomatic approach to manufacturing and manufacturing systems.
[3] Jahanmir S, Saka N, Tucker III C, Kim SG, 2015. *Advances in Multidisciplinary Engineering*, ASME Press, NY.
[4] Bae S, Lee JM, Chu CN. Axiomatic design of automotive suspension systems. CIRP Annals. 2002 Jan 1;51(1):115-8.
[5] Schuh G, Rudolf S, Breunig S. Modular platform design for mechatronic systems using axiomatic design and mechatronic function modules. Procedia CIRP. 2016 Jan 1;50:701-6.
[6] Babic B. Axiomatic design of flexible manufacturing systems. International Journal of Production Research. 1999 Mar 1;37(5):1159-73.
[7] Schlipf M, Bathurst S, Kippenbrock K, Kim SG, Lanza G. A structured approach to integrate MEMS and Precision Engineering methods. CIRP Journal of Manufacturing Science and Technology. 2010 Jan 1;3(3):236-47.
[8] Salonitis K. Design for additive manufacturing based on the axiomatic design method. The International Journal of Advanced Manufacturing Technology. 2016 Oct;87(1):989-96.
[9] Gu P, Rao HA, Tseng MM. Systematic design of manufacturing systems based on axiomatic design approach. CIRP Annals. 2001 Jan 1;50(1):299-304.
[10] Peck J, Nightingale D, Kim SG. Axiomatic approach for efficient healthcare system design and optimization. CIRP annals. 2010 Jan 1;59(1):469-72.

[11] Kim SJ, Suh NP, Kim SG. Design of software systems based on axiomatic design. Robotics and Computer-Integrated Manufacturing. 1991 Jan 1;8(4):243-55.
[12] Nordlund M, Tate D, Suh NP. Growth of axiomatic design through industrial practice. In3rd CIRP Workshop on Design and the Implementation of Intelligent Manufacturing Systems 1996 Jun 19 (pp. 77-84).
[13] Brown CA. Teaching axiomatic design to engineers—Theory, applications, and software. Journal of Manufacturing Systems. 2005 Jan 1;24(3):186-95.
[14] LeCun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proceedings of the IEEE. 1998 Nov;86(11):2278-324.
[15] Devlin J, Chang MW, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805. 2018 Oct 11.
[16] Chen W, Ahmed F. PaDGAN: Learning to Generate High-Quality Novel Designs. Journal of Mechanical Design. 2021 Mar 1;143(3).
[17] Cavique M, Fradinho J, Gabriel-Santos A, Mourão A, Gonçalves-Coelho A. Recovery of waste heat from engines: an AD view. InMATEC Web of Conferences 2018 (Vol. 223, p. 01007). EDP Sciences.
[18] Foley JT, Puik E, Cochran DS. The Faucet Reloaded: Improving Axiomatic Design by Example. InMATEC Web of Conferences 2017 (Vol. 127, p. 01009). EDP Sciences.
[19] Rinderle JR. *Measures of functional coupling in design* (Doctoral dissertation, Massachusetts Institute of Technology).
[20] Suh NP, Rinderle JR. Qualitative and quantitative use of design and manufacturing axioms. Cirp Annals. 1982 Jan 1;31(1):333-8.
[21] Bengio Y, Ducharme R, Vincent P, Janvin C. A neural probabilistic language model. The journal of machine learning research. 2003 Mar 1;3:1137-55.
[22] Kim SG, Yoon SM, Yang M, Choi J, Akay H, Burnell E. AI for design: Virtual design assistant. CIRP Annals. 2019 Jan 1;68(1):141-4.
[23] Akay H, Kim SG. Extracting Functional Requirements from Design Documentation using Machine Learning. Procedia CIRP. 2021.
[24] Akay H, Kim SG. Reading functional requirements using machine learning-based language processing. CIRP Annals. 2021 Apr 24.
[25] Akay H, Yang M, Kim SG. Automating Design Requirement Extraction from Text with Deep Learning. ASME IDETC Proceedings. 2021.
[26] Rajpurkar P, Zhang J, Lopyrev K, Liang P. Squad: 100,000+ questions for machine comprehension of text. arXiv preprint arXiv:1606.05250. 2016 Jun 16.
[27] Kern NI, Majewski TJ, Triolo RJ, Kobetic R, Quinn RD. A locking compliant device inspired by the anatomy of the spine. Journal of Mechanical Design. 2009 Jan 1;131(1).
[28] Akay H, Kim SG. Measuring functional independence in design with deep-learning language representation models. Procedia CIRP. 2020 Jan 1;91:528-33.
[29] Akay H, Kim SG. Design transcription: Deep learning based design feature representation. CIRP Annals. 2020 Jan 1;69(1):141-4.