INTERNATIONAL DESIGN CONFERENCE – DESIGN 2022
https://doi.org/10.1017/pds.2022.185

Generative Pre-Trained Transformer for Design Concept Generation: An Exploration

Q. Zhu ✉ and J. Luo

Singapore University of Technology and Design, Singapore

✉ qihao_zhu@mymail.sutd.edu.sg

Abstract

Novel concepts are essential for design innovation and can be generated with the aid of data stimuli and computers. However, current generative design algorithms focus on diagrammatic or spatial concepts that are either too abstract to understand or too detailed for early phase design exploration. This paper explores the uses of generative pre-trained transformers (GPT) for natural language design concept generation. Our experiments involve the use of GPT-2 and GPT-3 for different creative reasonings in design tasks. Both show reasonably good performance for verbal design concept generation.

Keywords: *early design phase, idea generation, generative design, natural language generation, generative pre-trained transformer*

1. Introduction

Design innovation is heavily dependent on high-quality and novel concepts. Concept design activities are divided into two stages: divergence and convergence (Tschimmel, 2012). Designers must generate a wide range of concepts during the divergence stage before any assessment and selection for convergence to be made. Therefore, much research has been conducted to develop methodologies and tools to aid designers to create design concepts, often with the aid of data and computers (Chakrabarti et al., 2011; Han et al., 2018; Han et al., 2020).

Meanwhile, little progress has been made regarding "computer ideation", i.e., computers directly and automatically generating ideas, in contrast to computer-aided ideation, i.e., computers aiding or stimulating human designers to generate ideas. In this study, we introduce a new technique from the field of artificial intelligence (AI)--the generative pre-trained transformer (GPT)--for automated generation of verbal design concepts. GPTs are language models pre-trained on vast quantities of textual data and can perform a wide range of language-related tasks (Radford et al., 2019; Brown et al., 2020). Our work experiments the applicability of different GPT models for both problem-driven reasoning and analogy-driven reasoning for design, with customized datasets.

2. Literature Review

Recent research on design concept generation can be categorized based on three dimensions: the role of method or tool, the form of concept representation, and the targeted design process stage.

A concept generation method or tool can play one of the three roles: as a guide, as a stimulator, or as a generator. A method is considered as guide if it is instructively involved in the generation of design concepts, providing design rules or guidelines to the activities of the designers. For instance, Bonnardel and Didier (2020) proposed two variants of brainstorming, encouraging designers to focus on the evocation of both the design ideas and the constraints related to the design problem. A stimulator provides inspirational stimuli to provoke designers to conceive new concepts. Goldschmidt and Smolkov (2006) discussed how visual stimuli affect problem solving design performance. Jin and Dong (2020) extracted 10 design heuristics as stimuli from RedDot award-wining design concepts to help digital designers overcome design fixation. He et al. (2019) tested the use of word clouds as stimulators to inspire ideation. Meanwhile, some methods can be a guide and a simulator at the same time. Luo et al. (2019) introduced a computer-aided ideation tool InnoGPS to guide the provision of design stimuli from the patent database by their knowledge distance to the design problem or interest. Fargnoli et al. (2006) introduced the morphological matrix to guide designers to navigate and combine alternative solutions to each of multiple functions of a product to generate a variety of designs.

A concept generator represents a fully automated computational agent that creates new concepts for the interest of designers. One example is function-based design synthesis (Chakrabarti et al., 2011). Sangelkar and McAdams (2017) introduced graph grammar to generate the function structures of the design concepts with potential for computational design synthesis. Kang and Tucker (2015) proposed a concept generation method based on function-form synthesis. Other generators perform topology optimization and generative visual design (Vlah et al., 2020). Oh et al. (2019) and Nie et al. (2021) integrate topology optimization and generative adversarial network (GAN) to generate concepts for both aesthetic and engineering performance. Ren et al. (2013), Burnap et al. (2016), and Dogan et al. (2019) use generative models to create new concepts of vehicle form design.

During design activities, concepts can be represented in the forms of abstract diagram, verbal text, or spatial visualization. Guiding or stimulation-based methods can direct designers to generate concepts in either of the three forms, e.g., simple sketches (Shah et al., 2001; Goldschmidt and Smolkov, 2006), mind-mapping graph (Shih et al., 2009; Yagita et al., 2011), functional diagram (Stone et al., 2000), or textual description (He et al., 2019; Sarica et al., 2021). Some guides or stimulators may also lead to multiple forms of concept representation as designers may record their perception of ideas in different ways (Bonnardel and Didier, 2020; Ilevbare, et al., 2013; Yilmaz et al., 2016). On the other hand, for automated tools, the type of concepts to be generated is pre-determined when designing the system, e.g., the graph grammar-based tools represent new concepts in abstract graphs (Campbell, 2009; Sangelkar and McAdams, 2017), while the topology optimization tools generate spatially visualized concepts in the form of 2D images (Oh et al., 2019) or 3D models (Nie et al., 2021).

The forms of generated concepts need to fit with design process stages. Pahl et al. (2007) specified four stages in design processes, including planning and task clarification, conceptual design, embodiment design, and detail design. For designers, visual stimulators like mood boards (Ahmed and Boelskifte, 2006) and spatial concept generators (Oh et al., 2019; Nie et al., 2021) are mainly useful for embodiment and detail design stages and may cause design fixation if applied in earlier stages (Viswanathan et al., 2016). Diagrammatic concepts are represented in a more abstract way that either visualizes the mind map of a design concept (Shih et al., 2009; Yagita et al., 2011) or the relationship between components of function or structure (Stone et al., 2000; Campbell, 2009). They are more suitable for the planning and task clarification as well as conceptual design stages. Meanwhile, text is also a common modality of recording concepts for early design stages. In typical brainstorming sessions, designers exchange preliminary design ideas verbally and frequently. Chiu and Shu (2007) investigated how semantic stimuli presented as words affect concept generation. Sarica et al. (2021) retrieved the terms from a pre-trained technology semantic network as stimuli to generate new concepts in the form of text. Goucher-Lambert & Cagan (2019) and Camburn et al. (2020) collected design ideas written in short text from crowdsourcing campaigns. To date, however, there exist no automated tool that is built to generate verbal design concepts.

To summarize, we present a taxonomy of concept generation methods or tools in Figure 1 by their roles, concept representation forms and suitable design stages. In this research, we focus on verbal generator using the latest natural language generation (NLG) technology. Particularly, we experiment the generative pre-trained transformers from OpenAI to learn design knowledge and reasoning from task-oriented datasets and then generate high fidelity design concept descriptions in natural language.

https://doi.org/10.1017/pds.2022.185 Published online by Cambridge University Press

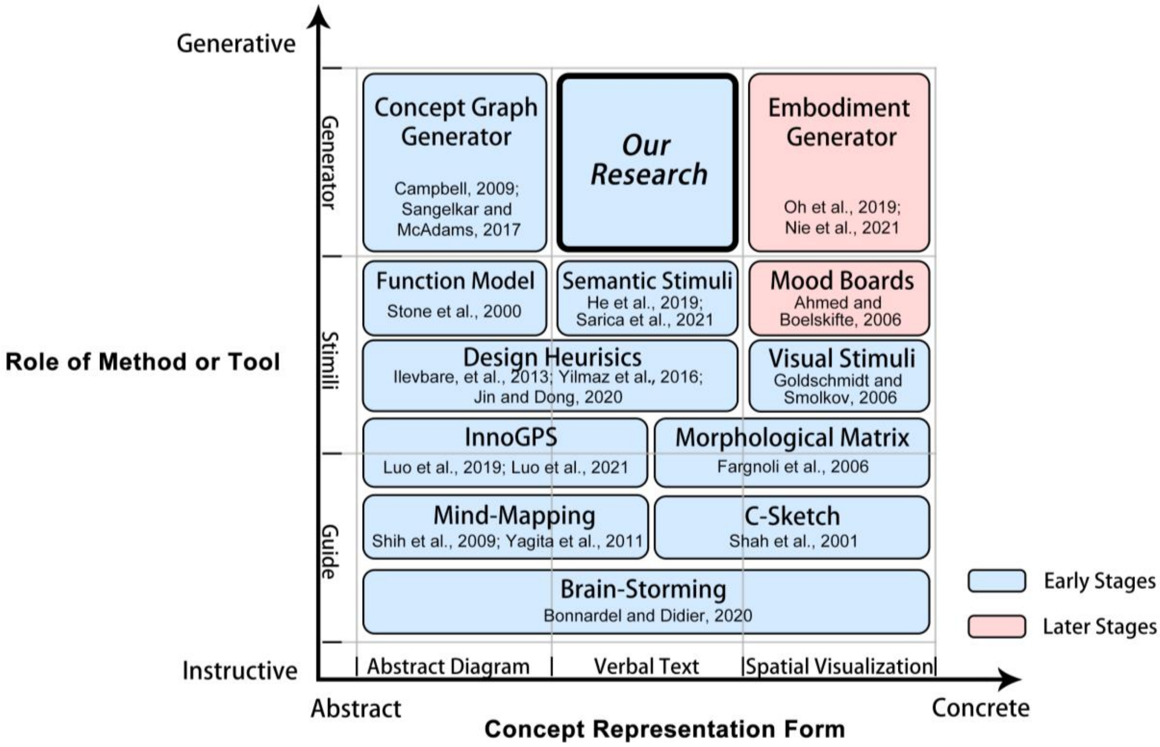

Figure 1. Taxonomy of concept generation methods or tools

3. Natural Language Generation (NLG)

Natural language generation (NLG) is a computer program that generates natural language as output (Gatt and Krahmer, 2018). Recent NLG techniques include machine translation (e.g., Kenny, 2019), text summarization (e.g., Ozsoy et al., 2011), paraphrasing (e.g., Li et al., 2018), and so on.

3.1. Transformer for NLG

Transformer, first introduced by Vaswani et al (2017), is the state-of-the-art neural network architecture for natural language processing (NLP) and is becoming the dominant for NLG (Topal et al., 2021). Comparing to recurrent neural network (RNN) and long short-term memory (LSTM), which were the most popular neural network architectures until recently, transformer overcomes the vanishing gradient problem (Pascanu et al., 2013) and enables parallel training. With the training data and model architecture become larger in size, it can capture longer sequence features and therefore result in much more comprehensive language understanding and generation (Brown et al., 2020).
Although transformer is a rather new technique in NLP and NLG, some applications have already been seen in different fields. Amin-Nejad et al. (2020) use transformer models to generate structured patient information to augment medical dataset. Fang (2021) use GPT to generate ideas for content creators. However, according to a recent review conducted by Regenwetter et al. (2021), the application of transformers is still a wide-open space for engineering design tasks. Our work fills this gap.

3.2. Generative Pre-trained Transformer (GPT)

Generative pre-trained transformer (GPT) stands for a series of pre-trained language models (PLM) developed by OpenAI (Radford et al., 2019; Brown et al., 2020), which has been the most popular type of transformers in NLG tasks. PLMs are language models that have been trained with a large dataset of textual information and can be applied to deal with specific language-related tasks (Arslan, et al. 2021). For example, BERT was trained with Wiki and books data that contains over 3.3 billion tokens (Kenton & Toutanova, 2019), and is popular in natural language understanding tasks, e.g., text classification. However, BERT as a masked language model can only learn contextual representation of words but not organize and generate language (Duan, et al., 2020), which makes it unsuitable for design concept

https://doi.org/10.1017/pds.2022.185 Published online by Cambridge University Press

generation task. On the other hand, GPTs are autoregressive language models that are trained to predict the next token based on all tokens before it.

GPT-2 uses the two-step training strategy of pre-training and fine-tuning, following Hinton and Salakhutdinov (2006). The workflow is shown in Figure 2(a). During the pre-training step, the model is trained on a text dataset collected from millions of webpages (Radford et al, 2019). For downstream NLP tasks, the pre-trained model needs to be fine-tuned given a customized and task-oriented dataset. The fine-tuned model is trained through repeated gradient updates using a large dataset of corpus of the example task. This process updates the weights of the pre-trained model and stores them for the use of the target task. However, the large dataset suitable for the target NLP task may be unavailable or difficult to collect. GPT-3 is the largest language model so far. It is trained on a mixture of datasets containing 400 billion tokens and has a maximum of 175 billion parameters. Comparing to its precursor, GPT-3 is capable of few-show learning (Brown et al., 2020), in which the model learns from multiple examples of the NLP or NLG task called prompts. In this process, no gradient updates are performed (Brown et al., 2020). The training process of GPT-3 is shown in Figure 2 (b).

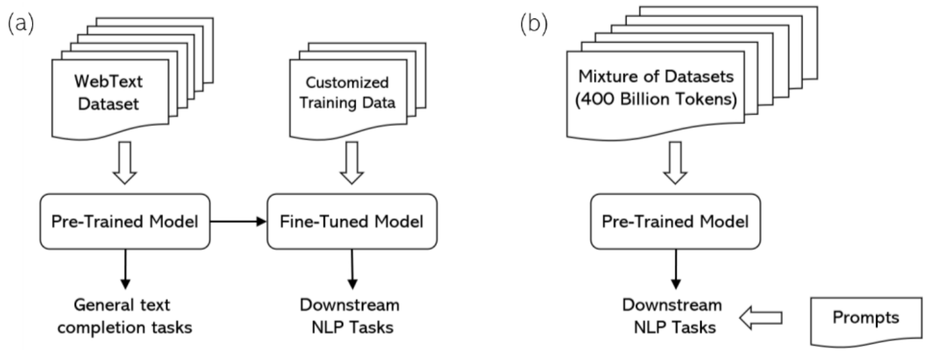

Figure 2. Training and re-training process of GPT-2 (a) and GPT-3 (b)

4. Research Method

In this paper, we experiment the applications of different GPT models in different design concept generation tasks. By customizing the fine-tunning dataset or examples for few-shot learning, GPT can learn and generate concepts based on different reasonings in design. Figure 3 depicts the general framework of our experiments. First, knowledge for the task is the key component of our framework. It is provided through the dataset used for fine-tuning GPT-2 model, or in the examples of design concepts for few-shot learning of GPT-3. Secondly, we take in varied input as for conditional learning. The input should be customized and consistent with the specific reasoning we want the model to learn, and output will be the generated design concept description. For instance, for analogy reasoning, the input will be the source and target domains for analogy mapping. Finally, the transformer in the framework can be either a fine-tuned GPT-2 or the pre-trained GPT-3 for few-shot learning.

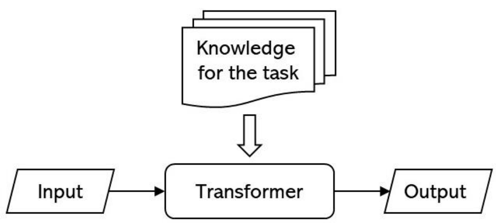

Figure 3. Experimental framework

Table 1 summarizes the settings for two experiments, in which we explore the capability of GPT for generating concepts by problem-driven reasoning and analogy-driven reasoning. The data we use for knowledge acquirement in both experiments is from the repository of RedDot award-wining designs. The first experiment includes three phrases for implementation: preparing the data for the NLG task,

https://doi.org/10.1017/pds.2022.185 Published online by Cambridge University Press

fine-tuning the model with the provided dataset, and testing the performance. The fine-tuning phrase is not included for the second experiment because we will be employing GPT-3 few-shot learning.

Table 1. Experiment settings

Experiment	Knowledge for the task	Input	Transformer
Problem-driven reasoning	RedDot award-winning design	Problem statement, concept category	Fine-tuned GPT-2
Analogy-driven reasoning	Examples of design-by-analogy concepts from RedDot award-winning design	Target and source domains	GPT-3 few-shot learning

5. Experiments

5.1. Problem-Driven Reasoning

Given GPT-2's capability to generate text based on understanding the context via training and fine-tuning, we experiment its application to generating text of solution ideas for a given problem. Problem-solving in design could be supported by different methods such as analogy and first principle. In this experiment we do not constrain GPT-2's problem-solving approach. The dataset for model fine-tuning is collected from RedDot's official website (https://www.red-dot.org/), including 14,502 product designs from 2011 to 2020 and 1,486 design concepts from 2016 to 2020. Data preparation includes picking out the text description of each design and adding its category name before the description. An example description of a problem-driven design in the RedDot dataset is shown below:

> *"One of the biggest and most common concerns of using public toilets is avoiding dermatosis and bacterial infection that comes from sharing a toilet with others. Clean Seat has a toilet lid that automatically opens when the first sensor (located at the front of the toilet lid) detects a user approaching. A second sensor then detects the person leaving after using the toilet, prompting the toilet lid to close and lock itself. When the lid is locked, the system kickstarts the self-cleaning function of the toilet."*

As shown in the example, the problem is stated in the first sentence, followed by the solution idea description. This structure is common in the descriptions of problem-driven design and ideal for GPT to generate solution idea text as the output in response to a problem text as the input. However, not all award-winning designs are problem-driven and often the description does not begin with a problem description. Our hypothesis is that the model can learn from those problem-driven design descriptions (as the example above) in the total dataset to execute problem-solving tasks, while also learning from design descriptions of other structures during training. A pre-trained GPT-2 of 355M parameters is fine-tuned for 14,000 steps with batch size of 1. Figure 4 reports loss over the fine-tuning steps.

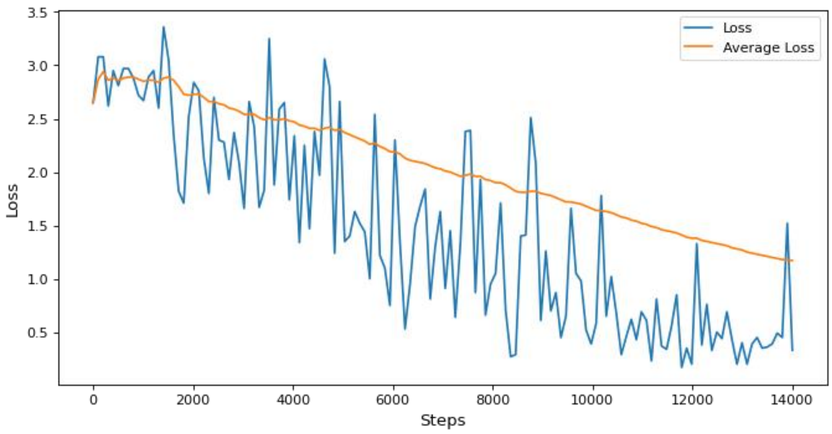

Figure 4. Loss with training steps when fine-tuning with RedDot data

In the generation phrase, the problem statement and the design domain category are both set as input. The problem statement can be a single sentence written by the designer, and the category input can

https://doi.org/10.1017/pds.2022.185 Published online by Cambridge University Press

be selected from the categories listed on RedDot official website. Table 2 reports two solutions generated for each of the two design problems in two experiments (parameters: temperature = 0.85, top-k = 40, and top-p = 1). The first problem statement is from in the example above and the other is from an IF award-winning design (https://ifworlddesignguide.com/entry/324944-ema), in order to test the model's capability to generate solution ideas to known versus unknown problems to the model.

Table 2. Problem-driven reasoning experiment: inputs and generated concepts

Inputs	Generated Concepts
Category: Personal Hygiene **Problem Statement**: One of the biggest and most common concerns of using public toilets is avoiding dermatosis and bacterial infection that comes from sharing a toilet with others.	The concept of this redesign of the traditional single-roof toilet presents a new idea. Instead of storing the toilet and washing it under the seat, it keeps the air in and a circular filter on the surface of the toilet seat, thereby preventing dermatosis from spreading. The built-in water tank is placed at the front of the seat so that it does not interfere with the efficiency of the cleaning process.
	Clean Seat has a toilet lid that automatically opens when the first one is defecated and closes automatically afterwards. In combination with a toilet lid with anti-perspiratory properties, the lid can be fitted over the first defecation und prevent the spread of the bacteria. The design of the toilet lid is very simple and in harmony with the shape of the body. It has a built-in air-flow control that is operated via a single touch of the keypad.
Category: Life Science **Problem Statement:** Current electrocardiograph testing involves a large number of wires, making the process time-consuming for doctors and stressful for children.	As a solution, the Electric Micro-Inspection Tool Show'em! is a wearable micro-test device that alleviates the repetitive testing procedure by simultaneously visualising the heart function of an implanted externa-nation. The externa-nation is a tiny, flexible device made of medical-grade silicone that can be readily attached to the patient's clothing. When an electric current is applied to the device, the externa-nation collects data, which are then translated into a formable electronic document. The user is only required to turn the small switch to trigger the testing procedure.
	The Light Guide is a remote controlled electrocardiograph that makes the process of testing electronic health records easier for both doctors and children. With a Light Guide, children can fully concentrate on examining themselves and their medical condition, which makes the testing of electronic health records more stress-free. The controller is small and remote-controlled so that it can be easily taken along to be tested at home. The information stored in the Light Guide is saved with the child's medical records and is reviewed at a later time. This allows doctors and families to have their medical records reviewed at a glance.

For the known problem with a known solution, GPT-2 can generate many new solution ideas based on the massive knowledge it has acquired from the training. For the unknown problem, it is also capable of comprehending the problem and generating solution ideas. However, as the text lengthens, it is likely to start talking about other features of the design that are unrelated to the given problem. This is not surprising as the design descriptions in the fine-tuning dataset often provide comprehensive elaboration of multiple aspects and the model learnt this well.

5.2. Analogy-Driven Reasoning

Design-by-analogy is the projection of existing reference in a source domain to address a comparable challenge in the target domain (Gentner, 1983; Luo et al., 2021). It is usually considered as a problem-solving approach. However, when a problem is not specified, analogy reasoning can also lead to open-ended design concept generation. This experiment is to test the analogy-driven reasoning of the model for concept generation, particularly for the context when a designer aims to draw analogy from a given source domain to generating design concepts in a given target domain but has not established clear analogy mapping across domains to generate specific new concepts.

As there are insufficient design-by-analogy examples to fine-tune a GPT-2, this experiment employs five analogy-driven reasoning examples selected from RedDot dataset as prompts for GPT-3's few-shot learning. Table 3 shows the source and target domains in each of the five examples for learning. Before each example is inputted, a structured sentence specifying the source and target domains (e.g.,

https://doi.org/10.1017/pds.2022.185 Published online by Cambridge University Press

“Applying accordion to computer mouse”) is inserted so that the GPT-3 may learn to develop ideas based on the input domains. When generating new concepts, we simply need to update the tokens that specify the source and target domains in the input sentence.

Table 3. Analogy-driven reasoning examples used as prompts for GPT-3

Source Domain	Target Domain	Link of the example
Accordion	Computer Mouse	https://www.red-dot.org/project/ambi-48504
Cells	Building	https://www.red-dot.org/project/build-fender-27044
Standing desk	Automobile	https://www.red-dot.org/project/sole-26525
Folding chair	Wheelchair	https://www.red-dot.org/project/fold-light-wheelchair-26521/
Circuit board	Desk	https://www.red-dot.org/project/cabletread-46563

Table 4 reports two drone design concepts generated by GPT-3, by drawing analogy from the given source domains of lantern and origami in two experiments (parameters: temperature = 0.85, top-p = 1, presence_penalty=0.5, frequency_penalty=0.5). The model successfully understands both source domains and applies the lighting feature of the lantern and the folding mechanism of origami to drone design. Without being trained with a large amount of analogical design cases, the generated concepts do adhere to analogical reasoning and learn to build a clear analogical mapping between the source and target domains, even when the two given source domains are culture-related and distant from the technical domain of drone. This is because the GPT-3 is pre-trained on a massive text dataset that includes nearly all possible kinds of scenarios and knowledges and a few examples as prompts are enough for the specific 'memory' to be evoked.

Furthermore, consistent to the pervious experiment, we also tested for the model's performance with the input of the same prompt sentences as in the given examples. It turns out GPT-3 is more likely to repeat the ideas given in the examples given low frequency penalty and presence penalty parameters, but it begins to generate novel texts as these parameters are set to higher values. This reveals another difference between the mechanisms of GPT-2 and GPT-3 models when being applied to design concept generation tasks, which will need further investigation.

Table 4. Analogy-driven reasoning experiment: inputs and generated concepts

Input		Output
Source	**Target**	**Generated Concepts**
Lantern	Drone	Focused on the design of portable drone landing pads, Skylight is conceived as a portable lantern that allows operators to track flying drones for charging or repair. Developed into different designs that can fit with any situation, Skylight has two main forms: docking stations that are focused on portability and public space; and lamp-like structures for open areas.
		D-LIGHT is a decorative drone that is fitted with a lantern inspired by the Chinese ‘paper lantern’. The drone’s movements control the position and movement of the lantern, which is mounted at its centre. The functional design of D-LIGHT focuses on responsive mechanisms for flying and controlling the lantern. This also enables it to be operated wherever necessary.
Origami	Drone	Aerobowl is inspired by the art of origami. The drone folds like an aerial paper aircraft and is able to fold itself up after landing. It can be launched without any human intervention, for example into lakes or rivers to help with rescue operations; it then folds its propellers into lightweight aircraft mode before floating along on the surface of the water like a boat.
		For centuries, paper folding has evolved into various different forms of art. Inspired by this, PaperDrone is an origami-inspired drone that is designed to have its chassis structure be deformable. The main goal of the design is to minimize the folding part while maintaining its strength and rigidity, mainly focusing on the legs for this feature. There are three feathered blades attached to each leg that can rotate in both directions at certain angles. The foldable body allows it to be very compact when folded, thus making easy storage possible.

https://doi.org/10.1017/pds.2022.185 Published online by Cambridge University Press

6. Discussion and Future Works

This article has examined the uses of generative pre-trained transformers to generate design concepts. It is demonstrated that by customizing the training data or examples, GPT can perform conceptual design tasks with a reasonable level of competence. This work opens the path to verbal concept generation using NLG, which may possibly be integrated with other approaches for a broader variety of applications and a higher level of automation. For example, the analogy-driven reasoning experiment necessitates source domain data for concept generation and comparison. This may be supplemented with knowledge graphs that provide domain knowledges with a given knowledge distance. In the current paper, we only provided case studies of using GPT for problem-driven and analogy-driven concept generation, but there are more opportunities to be explored. For example, the concept generation can also be driven by technology domain, i.e., generating concepts that addressing novel needs of a given technology. Furthermore, given suitable datasets, the generation of design concepts based on varied design heuristics will be possible, such as TRIZ (Ilevbare, et al., 2013) or the 77 design heuristics (Yilmaz et al., 2016).

Designers have challenges to generate novel ideas in practice due to a lack of comprehensive knowledge from various fields and the difficulty of properly applying design thinking. GPT's concept generation approach can be useful since it can acquire both knowledge and logic from a wide range of data and generate intelligible concepts for specific design tasks. Designers who use the tool may discover valuable ideas that are beyond their knowledge base and encourage them to think out of the box. In this sense, we expect the method will be an extension of human ideation. Moreover, human-AI collaboration in design concept generation does not necessarily stop at applying what the algorithm may offer. Designers can also draw further inspiration from the given concept and combine inspiration from other sources to make it more suitable, add more details and optimize it for their own project scenario.

However, it should be highlighted that the computer-generated concepts we showcased above were selected from a pool of low-quality results that may not be viable or context relevant. Thus, efficient concept evaluation algorithm is required to filter the automatically generated design concepts. However, common NLG evaluation metrics such as BLEU (Papineni, et al., 2002) and WMD (Kusner, et al., 2015) are designed for assessing the model's capability of generating high-quality language and require ground truth samples to measure the similarity or distance between texts. In design concept generation tasks, in addition to language quality, we aim to gauge the usefulness and novelty of any individual concept where ground truth benchmark is seldomly available. This makes existing NLG metrics unsuitable for concept generation tasks and thus new quantitative evaluation approaches will be needed. Furthermore, extensive experiments to compare different models as well as human assessment should be employed to validate the performances of both the generator and evaluator.

References

Ahmed, S. and Boelskifte, P. (2006), "Investigations of Product Design Engineering Students Intentions and a Users Perception of Product Character", *Proceedings of Nordesign*, Reykjavik, Iceland, pp. 372-381, 2006.

Amin-Nejad, A., Ive, J., & Velupillai, S. (2020), "Exploring transformer text generation for medical dataset augmentation". *In Proceedings of the 12th Language Resources and Evaluation Conference* (pp. 4699-4708).

Arslan, Y., Allix, K., Veiber, L., Lothritz, C., Bissyandé, T. F., Klein, J., & Goujon, A. (2021). "A comparison of pre-trained language models for multi-class text classification in the financial domain". *In Comp. Proc. Web Conf.* 2021 (260-268). https://doi.org/10.1145/3442442.3451375

Bonnardel, N., & Didier, J. (2020), "Brainstorming variants to favor creative design". *Applied Ergo.*, 83, 102987. https://doi.org/10.1016/j.apergo.2019.102987

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., et al. (2020), "Language models are few-shot learners". *Advances in Neural Information Processing Systems 33 (NeurIPS 2020).*

Burnap, A., Liu, Y., Pan, Y., Lee, H., Gonzalez, R., et al, P.Y. (2016), "Estimating and exploring the product form design space using deep generative models", in *IDETC-CIE*, ASME, V02AT03A013. https://doi.org/10.1115/DETC2016-60091

Camburn, B., He, Y., Raviselvam, S., Luo, J. and Wood, K. (2020), "Machine learning-based design concept evaluation", *Journal of Mechanical Design*, 142(3), 031113. https://doi.org/10.1115/1.4045126

Campbell, M., (2009) "A Graph Grammar Methodology for Generative Systems" [Online]. Available: http://repositories.lib.utexas.edu/handle/2152/6258. [Accessed: 10 -11 -2021].

Chakrabarti, A., Shea, K., Stone, R., Cagan, J., Campbell, M., et al. (2011), "Computer-based design synthesis research: an overview", *J. Comput. Inf. Sci. Eng.*, 11(2). https://doi.org/10.1115/1.3593409

Chiu, I. and Shu, L. (2007) "Understanding the use of language stimuli in concept generation", *in IDETC-CIE*, 161-172. https://doi.org/10.1115/DETC2007-35772

Dogan, K.M., Suzuki, H., Gunpinar, E. and Kim, M.-S. (2019), "A generative sampling system for profile designs with shape constraints and user evaluation", *Computer-Aided Design*, 111, 93-112. https://doi.org/10.1016/j.cad.2019.02.002

Duan, J., Zhao, H., Zhou, Q., Qiu, M., & Liu, M. (2020, November). "A Study of Pre-trained Language Models in Natural Language Processing". *In 2020 IEEE International Conference on Smart Cloud (SmartCloud)* (pp. 116-121). https://doi.org/10.1109/SmartCloud49737.2020.00030

Fang, J. (2021), *An Application of Customized GPT-2 Text Generator for Modern Content Creators*, [Master Thesis], UCLA.

Fargnoli, M., Rovida, E. and Troisi, R. (2006), "The morphological matrix: Tool for the development of innovative design solutions", in *4th ICAD*, 1-7. https://doi.org/10.1109/FIE.1998.736828

Gatt, A. and Krahmer, E. (2018), "Survey of the state of the art in natural language generation: Core tasks, applications and evaluation", *Journal of Artificial Intelligence Research*, 61, 65-170. https://doi.org/10.1613/jair.5477

Gentner, D. (1983), "Structure-mapping: A theoretical framework for analogy", *Cognitive science*, 7(2), 155-170. https://doi.org/10.1016/S0364-0213(83)80009-3

Goldschmidt, G. and Smolkov, M. (2006), "Variances in the impact of visual stimuli on design problem solving performance", *Design studies*, 27(5), 549-569. https://doi.org/10.1016/j.destud.2006.01.002

Goucher-Lambert, K. and Cagan, J. (2019) 'Crowdsourcing inspiration: Using crowd generated inspirational stimuli to support designer ideation', Design studies, 61, 1-29. https://doi.org/10.1016/j.destud.2019.01.001

Han, J., Forbes, H., Shi, F., Hao, J. and Schaefer, D. (2020), "A data-driven approach for creative concept generation and evaluation", *in Proceedings of the Design Society: DESIGN Conference*, 167-176. https://doi.org/10.1017/dsd.2020.5

Han, J., Shi, F., Chen, L. and Childs, P.R. (2018), "The Combinator–a computer-based tool for creative idea generation based on a simulation approach", *Design Science*, 4. https://doi.org/10.1017/dsj.2018.7

He, Y., Camburn, B., Liu, H., Luo, J., Yang, M., et al. (2019), "Mining and representing the concept space of existing ideas for directed ideation", *J. Mech. Des.*, 141(12). https://doi.org/10.1115/1.4044399

Hinton, G.E. and Salakhutdinov, R.R. (2006), "Reducing the dimensionality of data with neural networks", *Science*, 313(5786), 504-507. https://doi.org/10.1126/science.1127647

Huang, Q., Gan, Z., Celikyilmaz, A., Wu, D., Wang, J. and He, X. (2019), "Hierarchically structured reinforcement learning for topically coherent visual story generation", *in Proceedings of the AAAI Conference on Artificial Intelligence*, 8465-8472. https://doi.org/10.1609/aaai.v33i01.33018465

Ilevbare, I.M., Probert, D. and Phaal, R. (2013), “A review of TRIZ, and its benefits and challenges in practice”, *Technovation*, Vol. 33 No. 2-3, pp. 30-37. https://doi.org/10.1016/j.technovation.2012.11.003

Jin, X. and Dong, H. (2020), "New design heuristics in the digital era", *in Proceedings of the Design Society: DESIGN Conference*, 607-616. https://doi.org/10.1017/dsd.2020.321

Kang, S.W. and Tucker, C.S. (2015), "Automated concept generation based on function-form synthesis", in *IDETC-CIE*, ASME, V02AT03A008. https://doi.org/10.1115/DETC2015-47687

Kenny, D. (2019), 'Machine translation', In: Baker, M., & Saldanha, G. (Eds.), *Routledge Encyclopedia of Translation Studies Routledge (3rd ed.)*, pp. 305-310. https://doi.org/10.4324/9781315678627

Kenton, J. D. M. W. C., & Toutanova, L. K. (2019). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". *In Proc. of NAACL-HLT* (pp. 4171-4186).

Kusner, M., Sun, Y., Kolkin, N., & Weinberger, K. (2015, June). "From word embeddings to document distances". *In Int’l Conference on Machine Learning* (pp. 957-966). PMLR.

Li, Z., Jiang, X., Shang, L. and Li, H. (2018), "Paraphrase Generation with Deep Reinforcement Learning", *in Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 3865-3878.

Luo, J., Sarica, S. and Wood, K.L. (2019), "Computer-aided design ideation using InnoGPS", in *IDETC-CIE*, ASME, V02AT03A011. https://doi.org/10.1115/DETC2019-97587

Luo, J., Sarica, S. and Wood, K.L. (2021), "Guiding data-driven design ideation by knowledge distance", *Knowledge-Based Systems*, 218, 106873. https://doi.org/10.1016/j.knosys.2021.106873

Nie, Z., Lin, T., Jiang, H. and Kara, L.B. (2021), "Topologygan: Topology optimization using generative adversarial networks based on physical fields over the initial domain", *J. Mech. Des.*, 143(3), 031715. https://doi.org/10.1115/1.4049533

Oh, S., Jung, Y., Kim, S., Lee, I. and Kang, N. (2019), "Deep generative design: Integration of topology optimization and generative models", *J. Mech. Des.*, 141(11). https://doi.org/10.1115/1.4044229

Ozsoy, M.G., Alpaslan, F.N. and Cicekli, I. (2011), "Text summarization using latent semantic analysis", *Journal of Information Science*, 37(4), 405-417. https://doi.org/10.1177%2F0165551511408848

Pahl, Beitz, W., Feldhusen, J., & Grote, K.-H. (2007), "Engineering Design A Systematic Approach", Springer London. https://doi.org/10.1007/978-1-84628-319-2

Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002, July). "Bleu: a method for automatic evaluation of machine translation". *In Proceedings of the 40th annual meeting of the Association for Computational Linguistics* (pp. 311-318).

Pascanu, R., Mikolov, T. and Bengio, Y. (2013), "On the difficulty of training recurrent neural networks", *in International conference on machine learning*, PMLR, 1310-1318.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019), "Language models are unsupervised multitask learners". *OpenAI blog*, 1(8), 9.

Regenwetter, L., Nobari, A.H. and Ahmed, F. (2021), "Deep Generative Models in Engineering Design: A Review", *arXiv preprint* arXiv:2110.10863.

Ren, Y., Burnap, A. and Papalambros, P. (2013), "Quantification of perceptual design attributes using a crowd", in *DS 75-6: Proceedings of the 19th International Conference on Engineering Design (ICED13), Design for Harmonies, Vol. 6: Design Information and Knowledge*, Seoul, Korea, 19-22.08. 2013.

Sangelkar, S. and McAdams, D.A. (2017), "Automated Graph Grammar Generation for Engineering Design With Frequent Pattern Mining", in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, ASME, V02AT03A006. https://doi.org/10.1115/DETC2017-67520

Sarica, S., Song, B., Luo, J. and Wood, K.L. (2021), "Idea generation with technology semantic network", *AI EDAM*, 1-19. https://doi.org/10.1017/S0890060421000020

Shah, J.J., Vargas-Hernandez, N., Summers, J.D. and Kulkarni, S. (2001), "Collaborative Sketching (C-Sketch)—An idea generation technique for engineering design", *The Journal of Creative Behavior*, 35(3), 168-198. https://doi.org/10.1002/j.2162-6057.2001.tb01045.x

Shih, P.C., Nguyen, D.H., Hirano, S.H., Redmiles, D.F. and Hayes, G.R. (2009), "GroupMind: supporting idea generation through a collaborative mind-mapping tool", *in Proceedings of the ACM 2009 international conference on Supporting group work*, 139-148. https://doi.org/10.1145/1531674.1531696

Stone, R.B., Wood, K.L. and Crawford, R.H. (2000), "A heuristic method for identifying modules for product architectures", *Design studies*, 21(1), 5-31. https://doi.org/10.1016/S0142-694X(99)00003-4

Topal, M. O., Bas, A., & van Heerden, I. (2021), "Exploring transformers in natural language generation: GPT, BERT, and XLNET". *International Conference on Interdisciplinary Applications of AI (ICIDAAI)*

Tschimmel, K. (2012), "Design Thinking as an effective Toolkit for Innovation", in *ISPIM Conference Proceedings*, The International Society for Professional Innovation Management (ISPIM), 1.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., et al. (2017), "Attention is all you need", *in Advances in neural information processing systems*, 5998-6008.

Viswanathan, V., Tomko, M. and Linsey, J. (2016), "A study on the effects of example familiarity and modality on design fixation", AI EDAM, 30(2), 171-184. https://doi.org/10.1017/S0890060416000056

Vlah, D., Žavbi, R. and Vukašinović, N. (2020), "Evaluation of topology optimization and generative design tools as support for conceptual design", *in Proceedings of the Design Society: DESIGN Conference*, 451-460. https://doi.org/10.1017/dsd.2020.165

Yagita, H., Tose, A., Nakajima, M., Kim, S.K. and Maeno, T. (2011), "A validation regarding effectiveness of scenario graph", *in IDETC-CIE*, 385-394. https://doi.org/10.1115/DETC2011-48047

Yilmaz, S., Daly, S.R., Seifert, C.M. and Gonzalez, R. (2016), "Evidence-based design heuristics for idea generation", *Design studies*, 46, 95-124. https://doi.org/10.1016/j.destud.2016.05.001