Advanced Engineering Informatics 32 (2017) 237–247

Contents lists available at ScienceDirect

Advanced Engineering Informatics

journal homepage: www.elsevier.com/locate/aei

Full length article

A function-based computational method for design concept evaluation

Jia Hao [a,b,*], Qiangfu Zhao [b], Yan Yan [a]

[a] Beijing Institute of Technology, 5 South Zhongguancun Street, Haidian District, Beijing, China
[b] University of Aizu, Aizu-Wakamatsu City, Fukushima-ken 965-8580, Japan

ARTICLE INFO

Article history:
Received 16 October 2016
Received in revised form 17 January 2017
Accepted 1 March 2017
Available online 27 March 2017

Keywords:
Concept generation
Evaluation metrics
Function basis
Word-embedding

ABSTRACT

Concept generation is an indispensable step of innovation design. However, the limited knowledge and design thinking fixation of designers often impede the generation of novel design concepts. Computational tools can be a necessary supplement for designers. They can generate a big number of design concepts based on an existing knowledge base. For filtering these design concepts, this work presents a computational measurement of novelty, feasibility and diversity based on 500,000 granted patents. First, about 1700 functional terms (terminologies) are mapped to high dimensional vectors (100 dimensional space) by word embedding technique. The resulted database is knowledge base-I (KB-I). Then, we adopt circular convolution to convert patents into high dimensional vectors. The resulted database is KB-II. Based on the two knowledge bases, the computational definitions of novelty, feasibility and diversity are developed. We conduct six experiments based on KB-II, a random dataset and a real product dataset, and the results show that these metrics can be used to roughly filter a big number of design concepts, and then expert-based method can be further used. This work provides a computational framework for measuring the novelty, feasibility and diversity of design concept.

1. Introduction

Concept generation is the most creative and an indispensable step of innovation design. Many researchers have stated that there is a significant correlation between the quality of design concepts and success of final products [4,12]. However, improving the quality of design concepts is not easy, since it largely depends on designers' design thinking process, which is often limited by designers' knowledge and design thinking fixation [23,13]. Therefore, using external knowledge and assistance tools becomes one of the ways to improve the quality of design concepts [12].

The assistance tools for concept generation can be briefly categorized into two branches, including structured method and computational method. The structured method adopts a structured process as guidance for generating design concepts. There are already some typical structured methods. The 6-3-5 [24] method and C-Sketch [26] ensure a group of designers participate into a design process equally. The design-by-analogy [16] and function-mean trees [15] are function-based structure method, and both methods obey the principle of "decomposing function ⇒ finding solution ⇒ integrating solutions". The computational method is a relatively new research direction of generating design concepts, and existing methods are not as many as structured method. Kurtoglu extracted 45 rules of generating design concepts and developed a rule-based method [11]. Yan developed a co-evolutionary based method [9] and Jacquelyn developed a biological knowledge based method [19]. Besides the above methods, in [2] the author summarized 13 existing methods.

One of the common problems for both structured and computational methods is the evaluation of multiple design concepts. For a small number of design concepts, expert-based methods are feasible and effective. However, when we consider computational methods, a huge amount of design concepts would be generated, and the expert-based methods are infeasible, as shown in Fig. 1. Therefore, there is an urgent demand for a computational evaluation method to make the first round filter. After a small number of design concepts are selected, the expert-based methods can be used to make further filter. Considering the fact that functions are the most important information of design concepts, this research triesto develop a function-based computational method for evaluating design concepts based on a huge number of granted patents. To implement this goal, we address two research questions.

* Corresponding author at: Beijing Institute of Technology, 5 South Zhongguancun Street, Haidian District, Beijing, China.

E-mail addresses: haojia632@bit.edu.cn (J. Hao), qf-zhao@u-aizu.ac.jp (Q. Zhao), yanyan331@bit.edu.cn (Y. Yan).

http://dx.doi.org/10.1016/j.aei.2017.03.002
1474-0346/

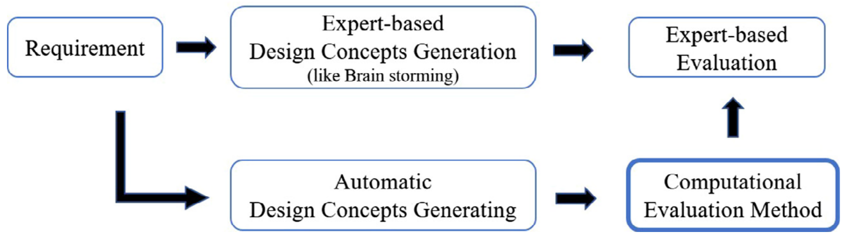

Fig. 1. An illustration of the requirement of computational evaluation method.

1. First, whether it is possible to define computational metrics for measuring design concepts.
2. Second, whether it is effective to use the computational metrics for measuring design concepts.

The rest of the paper is structured as follows. The next section provides some background for this study. The following section explains the proposed method in detail. Section 4 conducts six experiments for verifying the effectiveness of the computational metrics. Section 5 discusses the experimental results. Section 6 summarized this research and outlines some possible future studies.

2. Related works

Some researches have paved the way for this work, and in this section, we make a brief summarization of the research works. Three topics will be addressed, including concept representation, function vocabulary and design concept evaluation.

2.1. Design concept representation

Design concept representation is a critical foundation for computational design concept generation, since structuralization is the foundation of computation. The earliest structured representation of design concept was called function structure [22], which defines all functions and the relationships between functions, and the relationships are defined by flow (material, signal and energy). This kind of model can be constructed based on the function tree [32] and function-means tree [21]. Following this idea, several structured methods were developed, including Function-Behavior-Structure (FBS), Structure-Behavior-Function (SBF) and Function-Behavior-State. These methods only define the structure while ignore the vocabulary, which means the terms of functions are not restricted by a standard vocabulary. With the development of standard vocabulary (like FB and RFB in Section 2.2), Kurtoglu proposed a new structured method for representing design concepts, which is called "Configuration Flow Graph (CFG)" [10]. This method provides both structure representation and standard vocabulary for defining functions and their relationships as well as components for implementing functions.

From the above, we can see that functions are the most important information of composing design concepts. Therefore, it's reasonable to evaluate a design concepts based on the functions of composing the design concept.

2.2. Function vocabulary

This work studies the computational evaluation metrics at the function level. Therefore, function vocabulary is important. In 1984, Pahl constructed a function vocabulary at a very high abstraction level, which includes only five functions (transit, connect, deform, convert and store) and three flows (material, signal and energy)[22]. Following this offshoot, Hundal defined six top-level functions (branch, transit, connect, deform, convert and store) with some detailed definition of sub-functions, and the total number of functions is 44 [7].

In the recent years, Robert and his team built a vocabulary called "Function Basis (FB)" [30], which is extended from Little's work [14]. FB includes eight top-level functions with 24 sub-level functions. Based on FB, Julie integrated it with NIST (National Institute of Standards and Technology)[31] and formulated a new vocabulary called "Reconciled FB (RFB)" [3], which includes 8 top-level functions with 22 sub-level functions. In FB and RFB, the functions are defined one-by-one by human experts. Different with this, Murphy [18] constructed a vocabulary from 65,000 randomly selected patents, and this vocabulary includes about 1700 functions. Since our work is also based on a huge number of granted patents, this vocabulary will be adopted in this research.

2.3. Design concept evaluation

2.3.1. Evaluation metrics

The evaluation of design concept can be divided into process based method and outcome based method [28]. Process based method is to analyze and evaluate the whole cognitive process of generating design concept [12], which faces the difficulty that the inner mechanism of the cognitive process is unobservable. The outcome based method is more feasible than the process based method, and it's a prevalent way of design concept evaluation. The metrics are very important for evaluate design concepts.

In 2003, Shan proposed four metrics, including novelty, variety, quality and quantity [28,27]. Novelty means the degree of a given design concept is unusual with others. Variety means the degree of dissimilarity of a group of design concepts. Quality means the degree of a given design concept is feasible. Quantity means the total number of a group of design concepts. During the last decade, the four metrics were acknowledged by the research community, although different terminologies were used by different authors. For example, in [19], the authors used "usefulness", which is similar to feasibility. In [36], the author used "originality", which is similar to novelty. In some researches, the term "diversity" is also used to denote variety. The four metrics were also extended. For example, Brent introduced a new metric by combining novelty and variety [20]. In [12], Kurtoglu developed a new metric named "completeness", which defines how well a design concept satisfies the required functions. In all, we can conclude that novelty, variety, quality and quantity are four basic metrics for evaluating design concepts. To keep consistency, this work will use novelty, diversity, feasibility and quantity as the terms to denote these metrics.

2.3.2. Evaluation methods

Design concept evaluation is a Multiple-criteria decision making (MCDM) problem [33], and many different methods from MCDM domain can be used [37,6]. Currently, the researches are focused on expert-based method [33], which adopts one or many experts to grade design concepts from one or more aspects. Based on the scores given by experts, some MCDM methods, like analytic hierarchy process (AHP) [38] and VIKOR [33,38], can be used to calculate a score for each design concept. The traditional methods are feasible in many situations of design concept evaluation. However, two main problems drive this research. The first is that different experts may give different scores to the same design concept due to the experience or knowledge bias; the second is that expert-based methods are infeasible to grade a big amount of design concepts, which are produced by computational design concept generation systems.

3. Methodology

The goal of this research is to develop a computational method for measuring novelty, feasibility and diversity of design concepts. Since quantity can be obtained by simply counting design concepts, we will not consider this metric in the following discussion. To achieve this goal, we have based our research on two premises. The first is that the granted patents, as a public knowledge base, record historical design concepts in the form of nature language. In fact, there are many researches that try to extract information from granted patents for product analysis [34], predicting technical trend [35], finding product opportunities [8], construct function-behavior-structure model [5]. From these works, we can know that granted patents contain much technical information about products [5]. Therefore, it's reasonable to take the granted patents as a knowledge base in this work.

The second is that the functions, which are included in existing design concepts, reflect the novelty, feasibility and diversity to a large extent. As stated in Section 2.1, functions are the main and most important information of composing design concepts. Therefore, it's reasonable to take function information to conduct the evaluation process.

The basic idea underlying this research is to evaluate a given design concept based on a big number of granted patents. Fig. 2 is an illustration of the steps to implement this idea. We have two main steps to evaluate design concepts, including building knowledge base and defining evaluation metrics. The following subsections will detail the techniques.

3.1. Building knowledge bases

The knowledge bases are the foundation for defining evaluation metrics. This subsection illustrates the method for building two different knowledge bases from 500,000 granted patents, and Table 1 shows the details of these patents. In this work, we only use the granted patents from the CPC:F class. This is because this class of granted patents tend to involve more mechanical functions comparing with other class, like CPC:H (Electricity).

Since the goal is to develop a computational measurement, it's also necessary to represent knowledge bases in a computational manner. In this work, two computational knowledge bases are constructed. The first is a computational representation of function vocabulary, named KB-I, and the second is a computational representation of the granted patents, named KB-II. We can regard KB-I and KB-II as two big matrices. Each row of the former matrix is a function, while each row of the latter matrix is a granted patent. The two matrices have the same column size, because the latter matrix is generated based on the former matrix, as illustrated in Section 3.1.2.

3.1.1. Function base: KB-I

In this work, we adopt the function vocabulary constructed in [18], which includes about 1700 functions. The KB-I is a computational representation of all these functions, and each function is converted into a high dimensional vector (the dimension may be 100 or more). One important property of KB-I is that for any two closely related functions, their corresponding points in the high dimensional space are also close with each other.

The calculation of high dimensional vectors of all 1700 functions is typically a machine learning process, which is called "word embedding" [17] in the natural language processing (NLP) domain. Word embedding is a neural network based framework for learning semantic relationships between words based on a huge number of documents. Two different models can be used to implement word embedding, including skip-gram model and continues bag of word (CBOW)[25].

In this work, we learn the high dimensional vectors based on the skip-gram model, which utilizes a huge number of **target-context** pairs as training dataset. Take a cup patent as an example, it includes four functions {contain, hold, filter, seal}. If setting window size (C in Table 2) to 1, we can generate several target-context pairs, like {contain, (hold)}, {hold, (contain, filter)}, {filter, (hold, seal)} and {seal, (filter)}.

Fig. 3 is an implementation structure of the skip-gram model. In this model, the input is a V (total number of functions in vocabulary) dimensional one-hot vector (**target**), in which only one component is 1 and others are 0. The input vector is then transformed into a N (N is far less than V) dimensional vector by $\mathbf{W}$, which is a $V \times N$ matrix representing the connecting weights between input layer and hidden layer. Then, the N dimensional vector is further transformed into another V dimensional vector by $\mathbf{W}'$, which is a $N \times V$ matrix representing the connecting weights between hidden layer and output layer. All values in the V dimensional vector are calculated through softmax regression, and indicate the probability that the input vector (**target**) is related to every other function.

In this skip-gram model, $\mathbf{W}$ and $\mathbf{W}'$ are two parameters that determine the performance, and the whole learning process is to adjust these two parameters for better prediction. At the beginning of the learning process, both $\mathbf{W}$ and $\mathbf{W}'$ are generated randomly, and the predication performance will be very bad. The backward gradient descent (BGD) is used to adjust the two parameters, the readers who are interested in this algorithm can reference to [25] or any other materials about neural network for details.

In this research, the parameter setting of this model is shown in Table 2. It is worthy to note that N is 17 times smaller than V. After feeding 500,000 patents to the model, the learning rated is close to 1×10^{-4}, which means the model has perceived most of the information. Then we got a trained parameter $\mathbf{W}$. We take $\mathbf{W}$ as KB-I and each row of $\mathbf{W}$ as a computational representation of the corresponding function in the vocabulary. The readers who are interested in the data can download it from https://www.dropbox.com/s/tt1okydi17xh2e5/function_space.npy?dl=0[1], and the data only includes 1619 rows since we have removed some functions that appear only a few times.

Based on KB-I, we select 10 commonly used functions, including [import, add, break, fan, wash, heat, branch, select, blow, sort], and calculate their top 10 closest functions to check the rationality of KB-I intuitively. This work adopts cosine similarity to calculate the semantic similarity between function terms. As shown in Table 3, most of the results are consistent with our understanding, although some results are difficult to understand.

3.1.2. Patent base: KB-II

In this section, we convert the 500,000 patents into a computational representation based on KB-I. Since this research focuses on the function level, we regard patent as a combination of multiple different functions. Since the computational representations of

[1] The data is in the python format

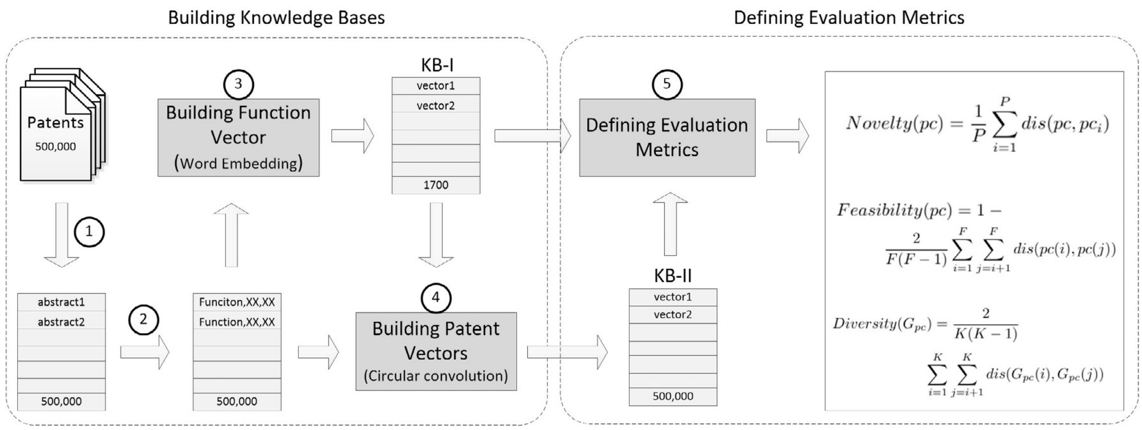

Fig. 2. The basic steps of defining evaluation metrics.

Table 1
Granted patents detail.

Source	http://www.patentsview.org/api/query-language.html
Time	From 1976.01.06 To 2014.09.16
Classification	CPC:F (Mechanical Engineering; Lighting; Heating; Weapons; Blasting Engines; Pumps)
Content	Abstraction
Request URL	http://www.patentsview.org/api/patents/query?q="_and": ["_eq":"cpc_section_id":"F","_gt":"patent_date":"1987-01-13"]&f=["patent_number","patent_date","patent_title", "patent _abstract"]&o="page":1,"per_page":500 *The request condition can be changed *The response is in JSON format
Total Number	500,000

Table 2
Skip-gram parameters setting for generating KB-I.

Parameter	Value	Detail
V	1700	The total number of functions in the vocabulary
N	100	The dimensions of trained vector
C	5	The window size for building target-context pairs

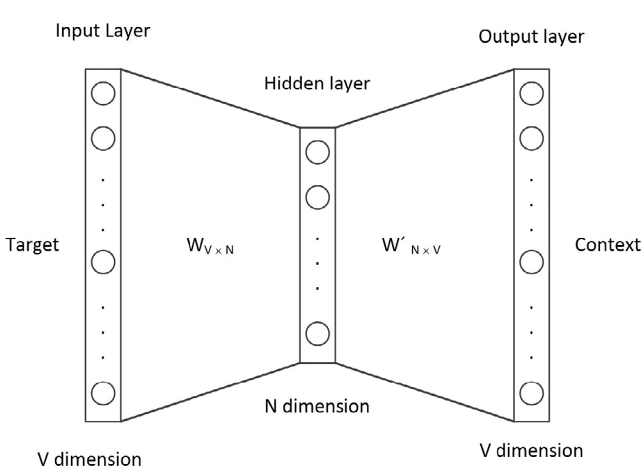

Fig. 3. The skip-gram model.

functions can be obtained from KB-I, the problem is converted into finding a **mathematical vector operation** for combining multiple functions as the computational representation of a patent. The mathematical vector operator should have the following two properties.

1. **Dimension keeping.** The mathematical vector operator should keep the dimension, which means the resulted patent vector has the same number of dimensions with function vector. In this case is $N = 100$;
2. **Discrimination.** The mathematical vector operator should have discrimination ability, which means the combination of two vectors produces a new vector that is different from the original two.

For the above two properties, the first one is easy to understand. This property is a guarantee that the computational cost will not increase rapidly. The second one is necessary although it's uneasy to be understood. For further explanation, We take A and B as two patents, and A includes two functions a_1 and a_2 while B includes two functions b_1 and b_2. If there is another patent C that includes all the four functions, we believe that C represents a novel design concept. Therefore, in the high dimensional space, the vector of C should not close to the vectors of A and B. This requires the result of the operation of A and B is different from themselves. The vector operator Add(+) meets the first property, but fails to meet the second since the Add(+) generates a new vector that is close to the two input vectors. In this research, we adopt Circular Convolution ($\otimes$) as the vector operator, since it meets both of the two properties. In fact, the application of Circular Convolution ($\otimes$) roots from semantic binding by Vectors Symbolic Architectures (VSA) in cognitive science [29]. In the VSA, all symbols (concepts or terms) are represented by high dimensional vectors, which is similar with this research. The combination of low level symbols can generate high level semantic meaning through the Circular Convolution ($\otimes$) operator. Many research works have proved the effectiveness of the VSA [1]. Therefore, we adopt the same operator in this research.

With the Circular Convolution ($\otimes$) operator, we convert all 500,000 patents into high dimensional vectors (100 dimensions in this research). For example, if a patent A includes three functions a_1, a_2 and a_3, we can get the vector representation of A by $(a_1 \otimes a_2) \otimes a_3$. The readers who are interested in the data can download it from https://www.dropbox.com/s/m5n87q2d9c48ua2/patent_space.npy?dl=0,[2] and the data includes 500,000 rows with each row represents a patent.

3.2. Defining evaluation metrics

As illustrated in Section 2.3.1, design concept can be evaluated through four metrics, including novelty, diversity, feasibility and quantity. Novelty and feasibility are used to evaluate a single

[2] The data is in python format

Table 3
Test result of top 10 related functions.

Test Functions	Import	Add	Break	Fan	Wash	Heat	Branch	Select	Blow	Sort
Top 10 Functions	Standardize	Estimate	Fracture	Blow	Rinse	Export	Inflow	Swap	Circulate	Compost
	Violate	Diagnose	Weaken	Duct	Cake	Condense	Bypass	Change	Suck	Lump
	Inventory	Perturb	Rupture	Shroud	Scrub	Dilute	Surge	Transduce	Fan	Crop
	Optimize	Disturb	Collapse	Impel	Slurry	Liquefy	Line	Proportion	Grill	Culture
	Require	Trend	Shear	Cascade	Wet	Sublimate	Purge	Iterate	Hood	Burden
	Pose	Factor	Puncture	Ventilate	Seed	Homogenize	Bank	Parse	Draft	Chop
	Relate	Refine	Fragment	Hood	Dry	Saturate	Feed	Actuate	Draw	Digest
	Manage	Gain	Tear	Baffle	Sieve	Evaporate	Drain	Include	Filtrate	Standardize
	Enhance	Violate	Crush	Suck	Pulp	Ferment	Materialize	Establish	Warm	Weigh
	Exploit	Specify	Damage	Grill	Filtrate	Ionize	Constitute	Regulate	Pasteurize	Heap

design concept while diversity and quantity are used to evaluate a group of design concepts. In this section, we will computationally define novelty, feasibility and diversity based on KB-I and KB-II. Since quantity can be obtained simply by counting the number of design concepts, we will not discuss it from now on.

3.2.1. Novelty

Novelty is the most important metric for evaluating a design concept. This section computationally defines this metric at the function level in the high dimensional space defined by KB-II. A novel design concept has fewer similar design concepts. Therefore, the key is how to measure the similarity between a pair of design concepts. In this work, we use semantic distance to measure the similarity. Since design concepts can be represented by high dimensional vectors in the space defined by KB-II, the semantic distance can be calculated by the distance between vectors. Some prevalent used methods (Euclidean distance, Cosine distance etc.) can be used. Therefore, we define the novelty by Eq. (1).

$$Novelty(pc) = \frac{1}{P}\sum_{i=1}^{P} dis(pc, ec_i) \tag{1}$$

where $Novelty()$ is a function of calculating the novelty value; pc is the current design concept; ec_i is the ith design concept of the top P similar design concepts in KB-II; $dis()$ is a function of calculating the distance between two design concepts in KB-II, and the distance can be any kind of distance in high dimensional space. The cosine distance is adopted in this research. It is worthy to note that, all design concepts are represented by high dimensional vectors.

3.2.2. Feasibility

Feasibility is another important metric for evaluating a design concept. This section computationally defines this metric at the function level in the high dimensional space defined by KB-I.

The functions of a feasible design concept have high co-occurrence possibility than the functions of an infeasible design concept. From the construction process of KB-I, we know that if two functions have high co-occurrence possibility, the semantic distance will be close. Therefore, the feasibility of design concept can be measured by the semantic distance between functions of the design concept. If the average distance is high, we can infer these functions do not often appear together, which means there may be risks that the design concept is infeasible. On the contrary, if the average distance is low, we can infer these functions often appear together, which means the feasibility is good. We define the feasibility by Eq. (2).

$$Feasibility(pc) = 1 - \frac{2}{F(F-1)}\sum_{i=1}^{F}\sum_{j=i+1}^{F} dis(f_i, f_j) \tag{2}$$

where $Feasibility()$ is a function of calculating the feasibility value; pc is the design concept under evaluation; f_i and f_j are the ith and jth function of pc; F is the total number of functions of pc; $dis()$ is a function of calculating the distance between two functions in KB-I; $\frac{2}{F(F-1)}$ is a factor to normalize the value. It is worthy to note that, all design concepts are represented by high dimensional vectors.

3.2.3. Diversity

When conducting concept design, we often generate a group of different design concepts. Different with novelty and feasibility, diversity is a metric to evaluate a group of design concepts. This section computationally defines this metric at the function level in the high dimensional space defined by KB-II.

For a group of design concepts, if the diversity is high, the similarity between these design concepts are low. Therefore, the key is also to measure the similarity between a pair of design concepts. Similar with the definition of novelty, we also use semantic distance of high dimensional vectors in KB-II to measure the diversity. We use Eq. (3) to calculate the value of diversity.

$$Diversity(G) = \frac{2}{K(K-1)}\sum_{i=1}^{K}\sum_{j=i+1}^{K} dis(G_i, G_j) \tag{3}$$

where $Diversity()$ is a function of calculating the diversity value; G is a group of design concepts; K is the total number of design concepts in G; G_i and G_j are the ith and jth design concept in G; $dis()$ is a function of calculating the distance between two design concepts in KB-II; $\frac{2}{K(K-1)}$ is a factor to normalize the value. It is worthy to note that, all design concepts are represented by high dimensional vectors.

4. Experiments

In this section, we conduct several experiments to verify the effectiveness of the above three computational metrics. The datasets are first illustrated, and then the experiments' process and results are illustrated.

4.1. Datasets of experiments

We use three datasets for conducting experiments, including KB-II, a random dataset and a product dataset. In the KB-II, all 500,000 patents are ordered along the time, and stored into 5 txt files with 100,000 patents in each file. In the random dataset, we generate a certain number of virtual patents by randomly combining different functions, and the number of patents depends on the requirement of different experiments. In the product dataset, we collect three kinds of products (Cup, Shaver and Fan), and extract their functions.

4.2. Experiments on novelty

4.2.1. Experiment 1: Historical analysis

This experiment is to verify the capability of novelty for distinguishing patents from different time frames. Intuitively, we believe that the average novelty should increase along the time from a macro viewpoint. For example, the average novelty of all patents between 2010 and 2014 may be greater than the average novelty of all patents between 1980 and 1984.

In this experiment, only a subset of KB-II is considered due to the computational cost for computing novelty. Therefore, we extract tree subsets (**named subset A, subset B and subset C**) from KB-II for calculating novelty, and they include 10,000, 20,000 and 50,000 patents respectively. The three subsets are generated randomly. Take subset B as an example, we iterate all 500,000 patents, and randomly select one from every 25 to extract 20,000 patents from KB-II. After that, we first randomly select one from every 250 patents for extracting a test subset of 2000 patents. Then, we calculate the novelty of each patent according to Eq. (1) by setting $P = 20$. For each of the three subsets, the experiment is conducted 10 times, and all results are averaged.

Figs. 4–6 show the experimental results on the three subsets. The three figures show results from different time-scales, which is determined by the value of groups (20, 10, 5). For example, if the value of groups is 10, we split all the 2000 results into 10 groups (200 values for each group) and average the values of each group.

As we can see from the figures, the novelty shows a consistent uptrend under all the three subsets, which is consistent with our intuition. We also find two interesting phenomenons. The first is the average novelty is very small (between 0.14 and 0.21). This is because a big portion of patents have very low novelty value (close to 0). The second is the increase of subsets' size (10,000, 20,000 and 50,000) decreases the average novelty. This is correct because a bigger subset will make it easy to find closely related patents, which will decrease the novelty value according to Eq. (1).

4.2.2. Experiment 2: Random comparison

This experiment is to verify the capability of novelty for distinguishing existing patents and randomly generated patents. Intuitively, we believe randomly generated patents have greater novelty value comparing with existing patents, because it does not consider feasibility.

In this experiment, we use subset B (see Section 4.2.1) as the space to calculate novelty. We first randomly select one patent from every 250 patents from KB-II, and randomly generate a patent having the same number of functions with the selected patent. By this way, we can get 2000 existing patents and 2000 randomly generated patents respectively. Then the novelties of these patents are calculated according to Eq. (1) by setting $P = 20$. The experiment is conducted 10 times, and the results are averaged.

Figs. 7–9 show the experimental results. The three figures show the same results from different time-scales, which is determined by the value of groups (20, 10, 5). We can see clearly that randomly generated patents have greater novelty value than the selected patents.

What is more, It is very interesting that the trends of the two curves keep consistent. This phenomenon reveals that the number of functions in the patents is a main reason that causes the novelty increases, because the randomly generated patents have the same number of functions with the selected patents.

4.2.3. Experiment 3: Real product

This experiment is to verify the capability of novelty for evaluating real products. We formulate the product dataset by extracting the main functions of 3 commonly used products and their variants. In Section 4.2.2, we find the number of functions influences novelty. Therefore, we verify the novelty by a group of product variants with the same number of functions. In the dataset, each variant has several basic functions, and one special function. Table 4 shows details of the product dataset.

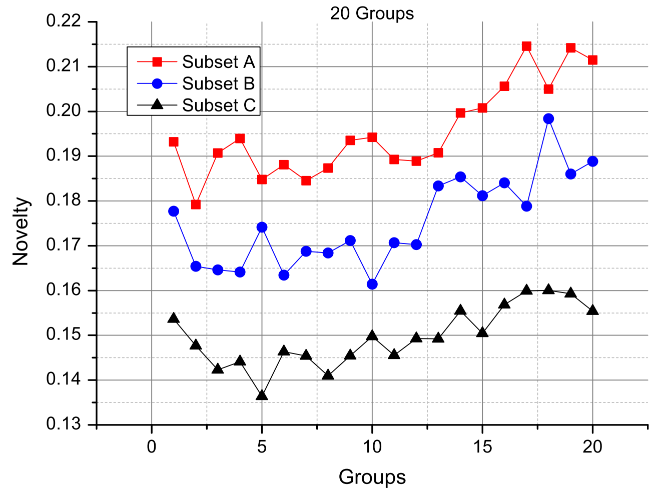

Fig. 4. The average novelty value of different time frames (20 Groups).

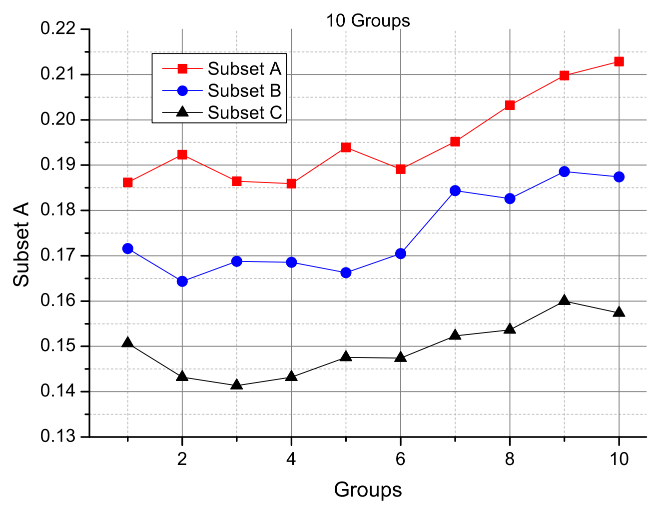

Fig. 5. The average novelty value of different time frames (10 Groups).

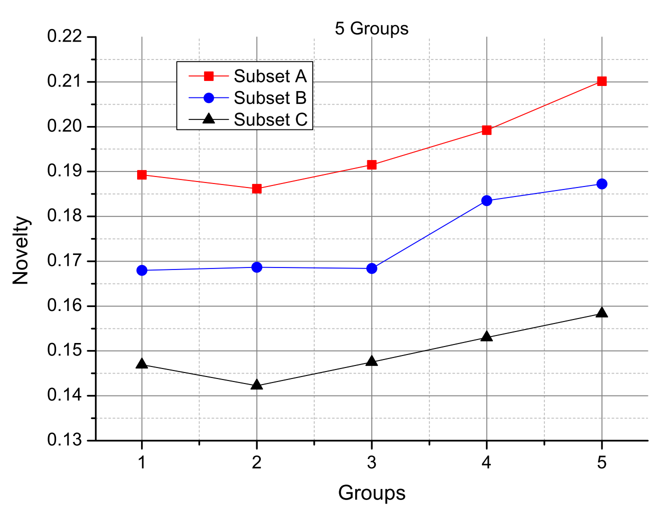

Fig. 6. The average novelty value of different time frames (5 Groups).

In this experiment, the novelty of each product variant is calculated based on the whole KB-II. Figs. 10–12 show the experimental results of cup, shaver and fan respectively. From the results, we can

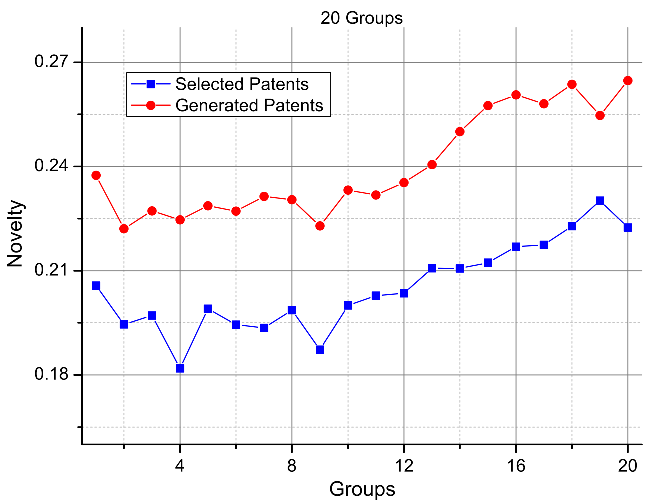

Fig. 7. Novelty comparison between KB-II and random dataset (20 Groups).

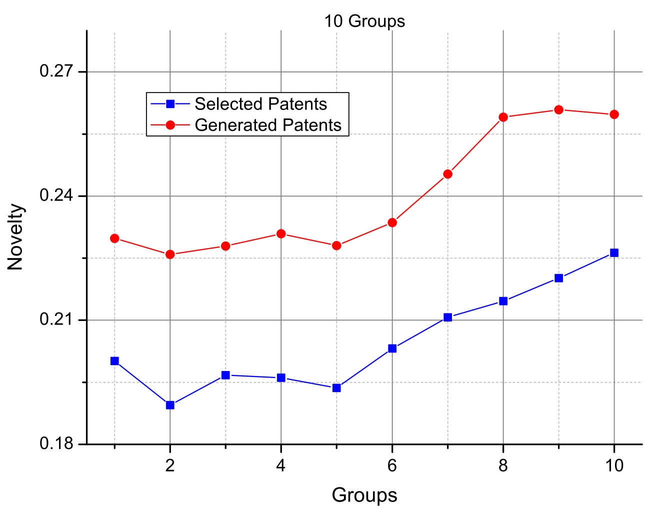

Fig. 8. Novelty comparison between KB-II and random dataset (10 Groups).

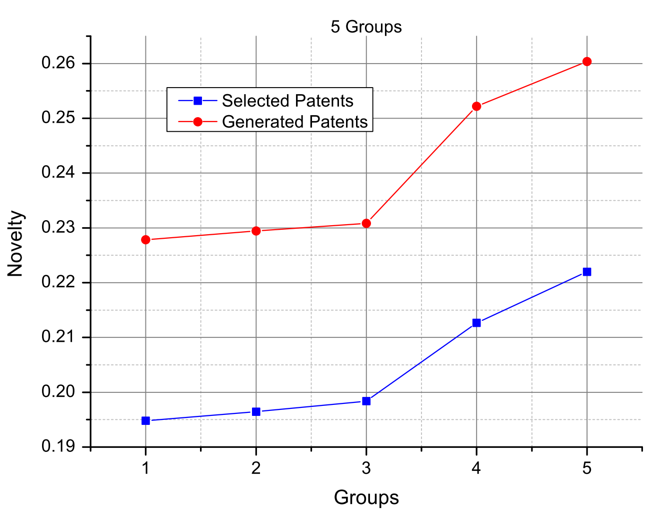

Fig. 9. Novelty comparison between KB-II and random dataset (5 Groups).

see that most of the novelty values meet our understanding, although some results seem unreasonable. For example, in Fig. 10, 'filter', 'carry', 'scale' and 'seal' have low novelty since they are very common in many real cups. The 'time', 'heat' and 'stretch' have high novelty because they are uncommon functions of cups. The unreasonable point is that the novelty of 'insulation' is also high since it's a common function of cups.

4.3. Experiments on feasibility

4.3.1. Experiment 4: Random comparison

This experiment is to verify the capability of feasibility for distinguishing existing patents and randomly generated patents. Intuitively, we believe the existing patents have greater feasibility comparing with randomly generated patents.

In this experiment, we randomly select one from every 250 patents from KB-II, and randomly generate a patent having the same number of functions with the selected patent. By this way, we can get 2000 existing patents and 2000 randomly generated patents respectively. Then the feasibilities of these patents are calculated according to Eq. (2). The experiment is conducted 10 times, and the results are averaged.

Fig. 13–15 show the experimental results from three different time-scales. As we can see from these figures, the selected patents have obvious greater feasibility than the randomly generated patents, which is identical with our intuition. Different with the experiment in Section 4.2.2, the two curves have different trends. The feasibilities of generated patents keep identical while the feasibilities of selected patents show an obvious uptrend.

4.3.2. Experiment 5: Real product

This experiment is to verify the capability of feasibility for evaluating real products. We use the same dataset used in Section 4.2.3. In this experiment, the feasibility is calculated based on Eq. (2). The results are shown in Figs. 16–18.

From the results, we can find that common functions tend to have high feasibility. For example, the 'carry', 'seal' and 'insulate' for a cup; the 'waterproof', 'slip' and 'float' for a shaver; the 'oscillate', 'handle' and 'mute' for a fan. We also find some uncommon functions to have high feasibility, and these have the potential to generate novel design concepts, like 'stretch' for cup, 'mist' for a fan.

4.4. Experiments on diversity

4.4.1. Experiment 6: Real product

This experiment is to verify the capability of diversity for evaluating different groups of product concepts. We also use the product dataset constructed in Section 4.2.3, but the dataset is revised for this experiment. As shown in Table 5, we select six variants from Table 4 for each product as group2. Three variants of group1 are selected from Table 4, and other three are generated by replace the last function with its similar function. For example, 'carry' is similar with 'handle' and 'scale' is similar with 'measure', etc.

In this experiment, we calculate the diversity by Eq. (3) and in this case $K = 6$. As shown in Fig. 19, for each of the three products, the diversity of group1 is smaller than that of group2. Therefore, we can infer that the metric of diversity is capable of differentiating the two groups.

5. Discussion

In this work, to meet the challenge proposed in the introduction section, we define the novelty, feasibility and diversity of design concepts at the function level based on a huge number of patents. To verify the three metrics, we conduct six experiments. This section discusses the experimental results.

Table 4
Product dataset: 3 products and their variants.

Product	Variants	Functions	Explanation
Cup	Variant-1	Hold, contain, carry	Easy to carry
	Variant-2	Hold, contain, filter	Filter tea or others
	Variant-3	Hold, contain, scale	Measure the volume
	Variant-4	Hold, contain, seal	Prevent leakage
	Variant-5	Hold, contain, insulate	Prevent hot
	Variant-6	Hold, contain, heat	Heat the water
	Variant-7	Hold, contain, time	Timekeeping
	Variant-8	Hold, contain, stretch	Compress when not in use
Shaver	Variant-1	Trim, hold, rotate	Rotate to generate flow
	Variant-2	Trim, hold, clean	Self-clean
	Variant-3	Trim, hold, indicate	Show some statuses
	Variant-4	Trim, hold, shake	Shake when cutting
	Variant-5	Trim, hold, charge	Charge the battery
	Variant-6	Trim, hold, waterproof	Washable
	Variant-7	Trim, hold, float	Adjust angle to fit face
	Variant-8	Trim, hold, slip	Anti slip
Fan	Variant-1	Blow, rotate, adjust, oscillate	Swing the head
	Variant-2	Blow, rotate, adjust, time	Timekeeping
	Variant-3	Blow, rotate, adjust, mute	Decrease noise
	Variant-4	Blow, rotate, adjust, handle	Carry by hand
	Variant-5	Blow, rotate, adjust, charge	Charge battery
	Variant-6	Blow, rotate, adjust, mist	Increase humidity
	Variant-7	Blow, rotate, adjust, purge	Clean air
	Variant-8	Blow, rotate, adjust, illuminate	Light

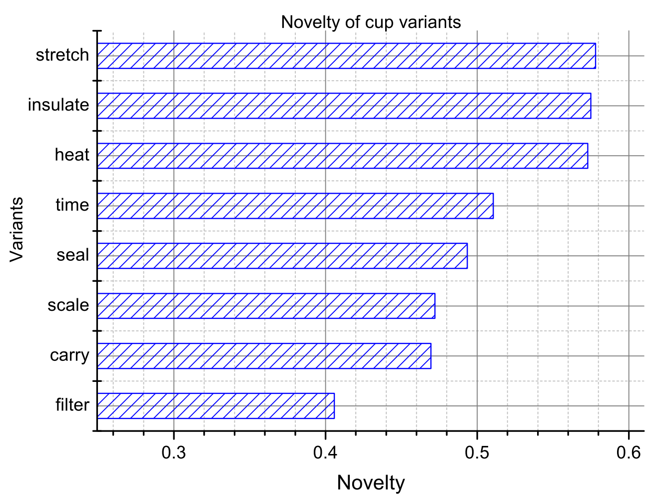

Fig. 10. Novelty of cup variants (computed based on 500,000 patents).

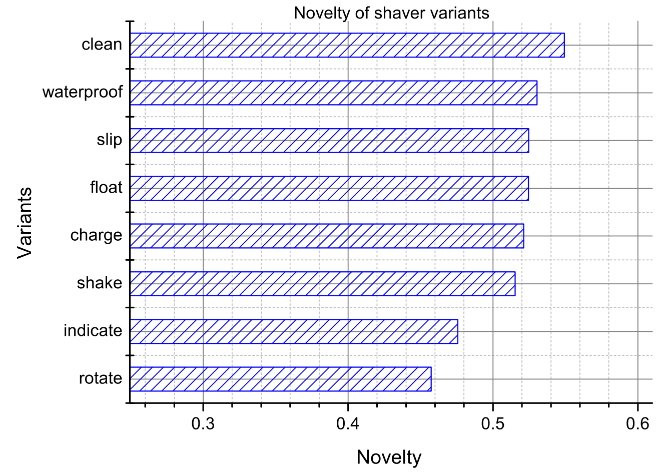

Fig. 11. Novelty of shaver variants (computed based on 500,000 patents).

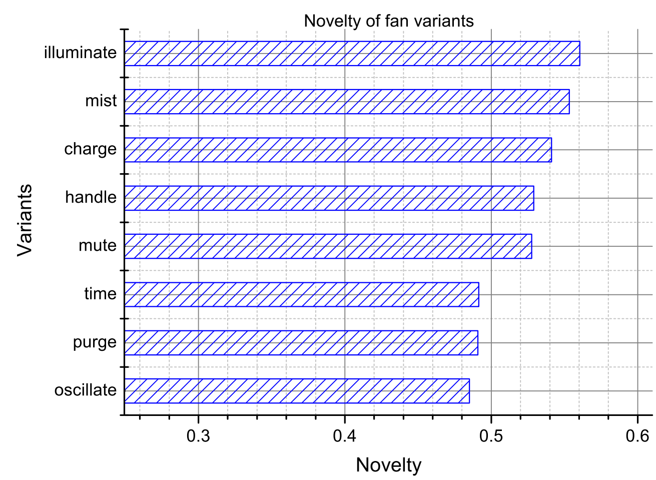

Fig. 12. Novelty of fan variants (computed based on 500,000 patents).

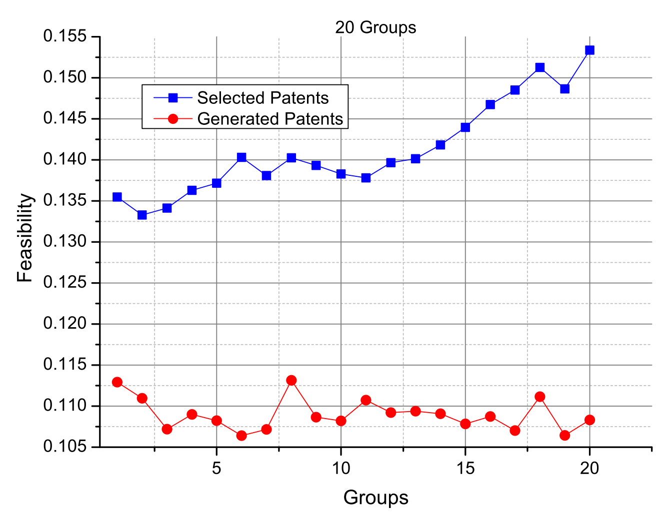

Fig. 13. Feasibility comparison between KB-II and random dataset (20 Groups).

Based on the experimental results, we can see the metric of novelty can be used to roughly measure design concepts. First, this metric can reflect the novelty trend of granted patents from a statistical perspective (experiment 1). Second, this metric is capable of differentiating the existing patents and randomly generated virtual patents (experiment 2). Third, this metric can rank real products in terms of their novelty reasonably, although a few product variants get unreasonable rank (experiment 3). To sum up, we can use this metric to roughly filter a big number of design concepts. We believe this metric can be helpful from the statistical perspective.

Based on the experimental results, we find the metric of feasibility is useful to roughly measure design concepts. First, this metric can differentiate the existing patents and randomly generated virtual patents from a statistical perspective (experiment 4). Second, this metric also can give reasonable ranking of real products in terms of their feasibility (experiment 5). Similar with novelty, there are also some unreasonable components in the experimental result. However, we think it can be used to filter a big number of design concept for the first round.

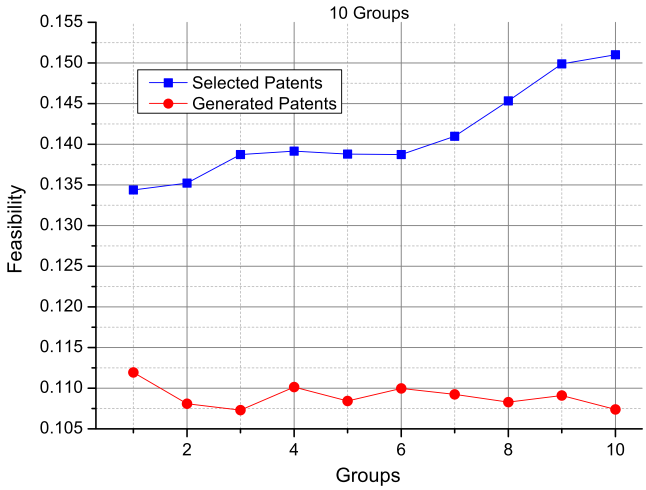

Fig. 14. Feasibility comparison between KB-II and random dataset (10 Groups).

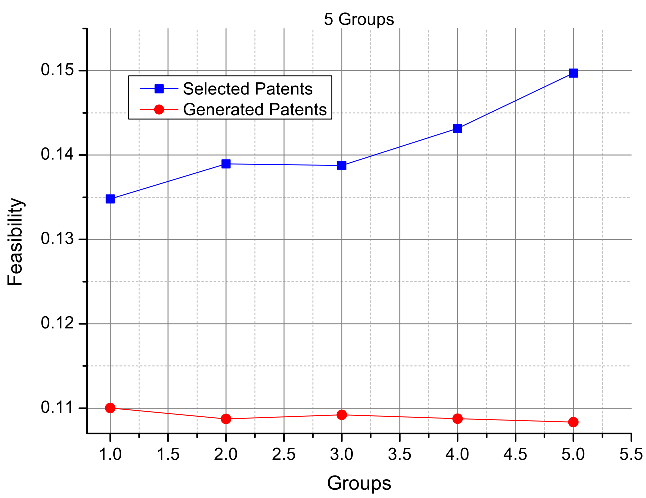

Fig. 15. Feasibility comparison between KB-II and random dataset (5 Groups).

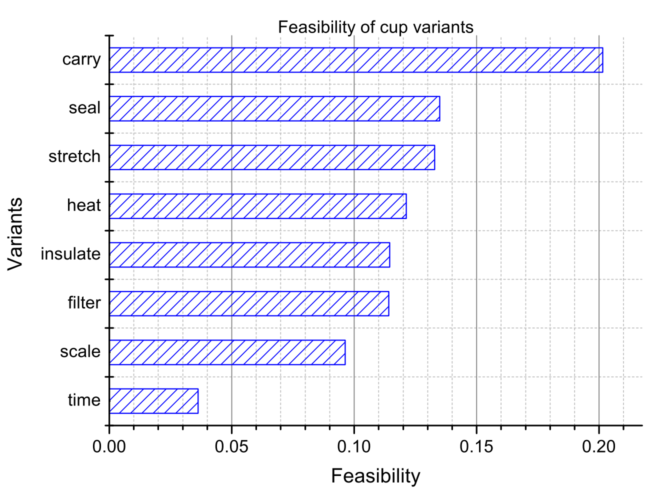

Fig. 16. Feasibility of cup variants.

Based on the experimental result, we find the metric of diversity is capable to measure a group of design concepts. This metric is test with real products, we can see that this metric can differentiate the two groups of design concepts in terms of their diversity

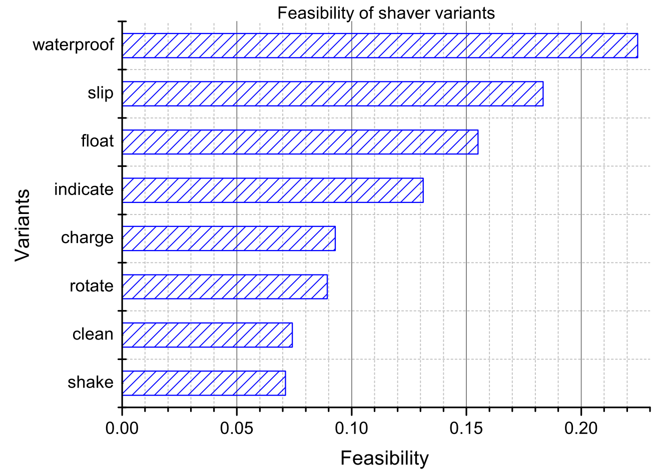

Fig. 17. Feasibility of shaver variants.

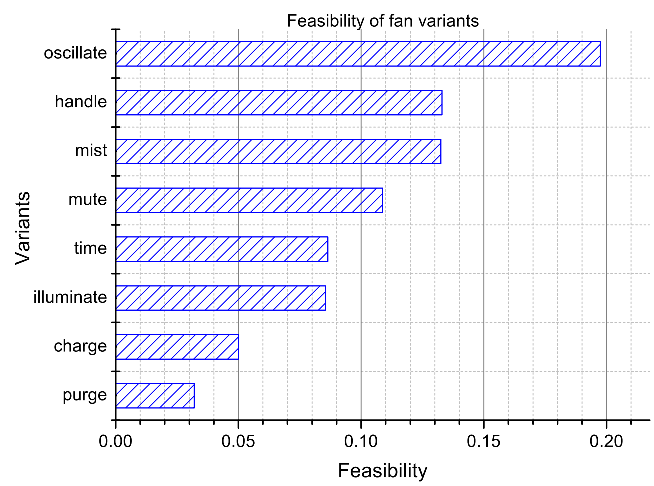

Fig. 18. Feasibility of fan variants.

Table 5
Product dataset: 3 products with two groups of variants.

Product	Variants	Group1	Group2
Cup	Variant-1	Hold, contain, carry	Hold, contain, carry
	Variant-2	Hold, contain, handle	Hold, contain, filter
	Variant-3	Hold, contain, scale	Hold, contain, scale
	Variant-4	Hold, contain, measure	Hold, contain, heat
	Variant-5	Hold, contain, insulate	Hold, contain, stretch
	Variant-6	Hold, contain, protect	Hold, contain, insulate
Shaver	Variant-1	Trim, hold, indicate	Trim, hold, clean
	Variant-2	Trim, hold, show	Trim, hold, indicate
	Variant-3	Trim, hold, shake	Trim, hold, shake
	Variant-4	Trim, hold, vibrate	Trim, hold, charge
	Variant-5	Trim, hold, waterproof	Trim, hold, slip
	Variant-6	Trim, hold, wash	Trim, hold, float
Fan	Variant-1	Blow, rotate, adjust, oscillate	Blow, rotate, adjust, oscillate
	Variant-2	Blow, rotate, adjust, swing	Blow, rotate, adjust, mute
	Variant-3	Blow, rotate, adjust, handle	Blow, rotate, adjust, handle
	Variant-4	Blow, rotate, adjust, carry	Charge
	Variant-5	Blow, rotate, adjust, purge	Blow, rotate, adjust, purge
	Variant-6	Blow, rotate, adjust, clean	Blow, rotate, adjust, time

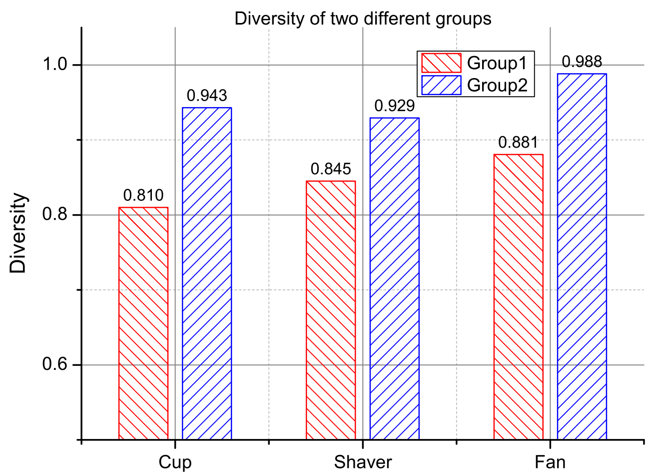

Fig. 19. Diversity comparison of two different groups.

(experiment 6). This metric is useful when we filter a big number of design concepts by novelty and feasibility, because it can keep the diversity of the selected results.

On the whole, we can see that the three metrics will helpful to make the first round filter of a big number design concepts. These metrics will select a small number of design concepts, so that expert-based evaluation method can be used to make further filter.

6. Conclusion

In an effort to develop three computational metrics for evaluating design concepts, this work defines novelty, feasibility and diversity based on 500,000 granted patents. By adopting word embedding technique from the natural language processing domain, about 1700 functions are represented by high dimensional vectors (KB-I), and further circular convolution is used as an operator for converting all 500,000 patents into high dimensional vectors (KB-II). The KB-I and KB-II provide the foundation for defining the novelty, feasibility and diversity. The experimental results show that these metrics can be used to roughly filter a big number of design concepts. Despite the progress of this research, there are some issues need to be addressed in our future work.

1. First, this work only focuses on the function level, and this is only a partial information contained in a design concept. In the future, the structure and behavior information should be considered for defining novelty, feasibility and diversity.
2. Second, this work only used randomly selected patents (10,000, 20,000, 50,000) for calculating the metrics due to the high computation cost when considers all 500,000 patents. In the future, a fast algorithm, which considering all patents, should be addressed for calculating these metrics.
3. Finally, some statistic researches should be conducted for investigating the characteristics of patents. By this way, the computational definition of these metrics would be more objective.

The method and metrics proposed in this work contribute both to theory and practice. From a theoretical perspective, this work provides a new framework for computationally measuring design concepts. From a practical perspective, the method and metrics can be used directly to develop tools or software for assisting the concept generation process. It is worthy to note that this method is not to replace the expert-based design concept evaluation methods. On the contrary, it is a supplement to expert-based methods. A big amount of design concepts can be filtered by this method to select a small number of design concepts, and then expert-based methods can make further fine filter. We believe this method will be helpful for developing some computational design concept generation systems.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments and thank the strong support provided by National Natural Science Foundation of China (NSFC 51505032), Beijing Natural Science Foundation (BJNSF 3172028) and National Ministries Projects (JCKY2016602B007 2015BAF18B).

References

[1] E. Crawford, M. Gingerich, C. Eliasmith, Biologically plausible, human-scale knowledge representation, Cogn. Sci. (2015).
[2] S.R. Daly, S. Yilmaz, J.L. Christian, C.M. Seifert, R. Gonzalez, Design heuristics in engineering concept generation, J. Eng. Educ. 101 (4) (2012) 601.
[3] M. Dinar, Problem map: a framework for investigating the role of problem formulation in creative design (Ph.D. thesis), 2015.
[4] N. Dylla, Thinking methods and procedures in mechanical design. Mechanical design (Ph.D.), Technical University of Munich, 1991.
[5] G. Fantoni, R. Apreda, F. DellOrletta, M. Monge, Automatic extraction of function–behaviour–state information from patents, Adv. Eng. Inform. 27 (3) (2013) 317–334.
[6] X. Geng, X. Chu, Z. Zhang, A new integrated design concept evaluation approach based on vague sets, Expert Syst. Appl. 37 (9) (2010) 6629–6638.
[7] M. Hundal, A systematic method for developing function structures, solutions and concept variants, Mech. Mach. Theory 25 (3) (1990) 243–256.
[8] G. Jin, Y. Jeong, B. Yoon, Technology-driven roadmaps for identifying new product/market opportunities: use of text mining and quality function deployment, Adv. Eng. Inform. 29 (1) (2015) 126–138.
[9] Y. Jin, W. Li, Design concept generation: a hierarchical coevolutionary approach, J. Mech. Des. 129 (10) (2007) 1012–1022.
[10] T. Kurtoglu, A Computational Approach to Innovative Conceptual Design, ProQuest, 2007.
[11] T. Kurtoglu, M.I. Campbell, J. Gonzalez, C.R. Bryant, R.B. Stone, D.A. McAdams, Capturing empirically derived design knowledge for creating conceptual design configurations, in: ASME 2005 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers, 2005, pp. 249–257.
[12] T. Kurtoglu, M.I. Campbell, J.S. Linsey, An experimental study on the effects of a computational design tool on concept generation, Des. Stud. 30 (6) (2009) 676–703.
[13] J.S. Linsey, I. Tseng, K. Fu, J. Cagan, K.L. Wood, C. Schunn, A study of design fixation, its mitigation and perception in engineering design faculty, J. Mech. Des. 132 (4) (2010) 041003.
[14] A. Little, K. Wood, D. McAdams, Functional analysis: a fundamental empirical study for reverse engineering, benchmarking and redesign, Proceedings of the 1997 Design Engineering Technical Conferences, vol. 97, 1997.
[15] J. Malmqvist, Improved function-means trees by inclusion of design history information, J. Eng. Des. 8 (2) (1997) 107–117.
[16] D.A. McAdams, K.L. Wood, A quantitative similarity metric for design-by-analogy, J. Mech. Des. 124 (2) (2002) 173–182.
[17] T. Mikolov, J. Dean, Distributed representations of words and phrases and their compositionality, Adv. Neural Inform. Process. Syst. (2013).
[18] J. Murphy, K. Fu, K. Otto, M. Yang, D. Jensen, K. Wood, Function based design-by-analogy: a functional vector approach to analogical search, J. Mech. Des. 136 (10) (2014) 101102.
[19] J.K. Nagel, R.L. Nagel, R.B. Stone, D.A. McAdams, Function-based, biologically inspired concept generation, Artif. Intell. Eng. Des. Anal. Manuf. 24 (04) (2010) 521–535.
[20] B.A. Nelson, J.O. Wilson, D. Rosen, J. Yen, Refined metrics for measuring ideation effectiveness, Des. Stud. 30 (6) (2009) 737–743.
[21] B.A. O'Sullivan, Constraint-aided conceptual design (Ph.D. thesis), Citeseer, 1999.
[22] G. Pahl, W. Beitz, Engineering Design: A Systematic Approach, Springer Science & Business Media, 2013.
[23] M. Perttula, P. Sipilä, The idea exposure paradigm in design idea generation, J. Eng. Des. 18 (1) (2007) 93–102.
[24] B. Rohrbach, Kreativ nach regeln–methode 635, eine neue technik zum lösen von problemen, Absatzwirtschaft 12 (19) (1969) 73–75.
[25] X. Rong, word2vec parameter learning explained. arXiv preprint arXiv:1411.2738, 2014.
[26] J.J. Shah, Experimental investigation of progressive idea generation techniques in engineering design, in: ASME DETC Design Theory and Methodology Conference, Atlanta, GA, Sept, 1998, pp. 13–16.
[27] J.J. Shah, S.V. Kulkarni, N. Vargas-Hernandez, Evaluation of idea generation methods for conceptual design: effectiveness metrics and design of experiments, J. Mech. Des. 122 (4) (2000) 377–384.
[28] J.J. Shah, S.M. Smith, N. Vargas-Hernandez, Metrics for measuring ideation effectiveness, Des. Stud. 24 (2) (2003) 111–134.

[29] P. Smolensky, Tensor product variable binding and the representation of symbolic structures in connectionist systems, Artif. Intell. 46 (1–2) (1990) 159–216.
[30] R.B. Stone, K.L. Wood, Development of a functional basis for design, J. Mech. Des. 122 (4) (2000) 359–370.
[31] S. Szykman, J.W. Racz, R.D. Sriram, The representation of function in computer-based design, in: Proceedings of the 1999 ASME design engineering technical conferences (11th international conference on design theory and methodology). Citeseer, 1999.
[32] Y. Tang, X. Liu, Task partition for function tree according to innovative functional reasoning, in: 12th International Conference on Computer Supported Cooperative Work in Design, 2008. CSCWD 2008, IEEE, 2008, pp. 189–195.
[33] V. Tiwari, P.K. Jain, P. Tandon, Product design concept evaluation using rough sets and VIKOR method, Adv. Eng. Inform. 30 (1) (2016) 16–25, http://dx.doi.org/10.1016/j.aei.2015.11.005.
[34] A.J. Trappey, C.V. Trappey, C.Y. Wu, C.-W. Lin, A patent quality analysis for innovative technology and product development, Adv. Eng. Inform. 26 (1) (2012) 26–34.
[35] C.V. Trappey, H.Y. Wu, F. Taghaboni Dutta, A.J. Trappey, Using patent data for technology forecasting: China rfid patent analysis, Adv. Eng. Inform. 25 (1) (2011) 53–64.
[36] J.O. Wilson, D. Rosen, B.A. Nelson, J. Yen, The effects of biological examples in idea generation, Des. Stud. 31 (2) (2010) 169–186.
[37] L.-Y. Zhai, L.-P. Khoo, Z.-W. Zhong, Design concept evaluation in product development using rough sets and grey relation analysis, Expert Syst. Appl. 36 (3) (2009) 7072–7079.
[38] G.-N. Zhu, J. Hu, J. Qi, C.-C. Gu, Y.-H. Peng, An integrated ahp and vikor for design concept evaluation based on rough number, Adv. Eng. Inform. 29 (3) (2015) 408–418.