Data-driven Product Functional Configuration: Patent Data and Hypergraph

Wen-Guang Lin

Xiamen University of Technology

Xiao-Dong Liu

Xiamen University of Technology

Ren-bin Xiao (✉ rbxiao@hust.edu.cn)

Huazhong University of Science and Technology <https://orcid.org/0000-0003-2672-7542>

Original Article

Keywords: Hypergraph, Function configuration, Patent analysis, Weighted frequent subgraph algorithm

Posted Date: September 7th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-859859/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Title page

Data-driven product functional configuration: Patent data and hypergraph

Wen-Guang Lin, born in 1985, is currently a lecturer and a master candidate supervisor at *Key Laboratory of Intelligent Manufacturing Equipment, Xiamen University of Technology, China*. He received PhD degree from *Xiamen university, China*, in 2014.

E-mail: linwg@xmut.edu.cn

Xiao-Dong Liu, born in 1995, is currently a master candidate at Key Laboratory of Intelligent Manufacturing Equipment, Xiamen University of Technology, China.

E-mail: Louis_LXD@163.com

Ren-Bin Xiao, born in 1965, is currently a professor and PhD candidate supervisor at School of artificial intelligence and automation, *Huazhong University of Science and Technology, China*. His main research interests include data driven product design, design method, and swarm intelligence.

E-mail: rbxiao@hust.edu.cn

Corresponding author: Ren-Bin Xiao E-mail: rbxiao@hust.edu.cn

ORIGINAL ARTICLE

Data-driven product functional configuration: Patent data and hypergraph

Wen-Guang Lin¹ • Xiao-Dong Liu¹ • Ren-Bin Xiao²

Received September xx, 2021; revised February xx, 202x; accepted March xx, 202x

© Chinese Mechanical Engineering Society and Springer-Verlag Berlin Heidelberg 2017

Abstract: Product functional configuration (PFC) is a common way for the firm to satisfy individual requirements of the customer and be carried out base on market analysis. This study aims to help firms analyze functions and carry out function configuration by the patent data. This paper proposes a patent data-driven product function configuration method based on a hypergraph network. It constructs a weighted network model to optimize the combination of product function quantity and object from the perspective of big data: (1) The functional knowledge contained in the patent is extracted. (2) The functional hypergraph (FH) is constructed according to the co-occurrence relationship of patent and applicant. (3) The function and patent weight are calculated from the perspective of the patent applicant and patent value. (4) The weight calculation model of PFC is built. (5) The weighted frequent subgraph algorithm is used to obtain the optimal function combination list. This method is applied in the innovative design process of a bathroom shower. The result indicates that this method designed in the study has a positive effect on helping firms detach optimal function candidates and develop a multi-function product.

Keywords: Hypergraph, Function configuration, Patent analysis, Weighted frequent subgraph algorithm

1 Introduction

In order to be competitive in the fierce market competition, manufacturers strengthen product innovation and development capability to satisfy various customer needs.

✉ Ren-Bin Xiao
rbxiao@hust.edu.cn

¹ Key Laboratory of Intelligent Manufacturing Equipment, Xiamen University of Technology, Xiamen 361005, China

² School of artificial intelligence and automation, Huazhong University of Science and Technology, Wuhan 40073, China

Commonly, customer will be attracted by multi-functional products rather than products with single function [1-2]. Therefore, firms tend to develop multi-function product (MFP), which can fulfil various needs in one to save space and money for customers. How to effectively design a product and combine many functions into it is one of the most important tasks for firms. In fact, MFP design process is a challenge task for need to solve three problems (2H-W):

- How to obtain functional information?
- How to evaluate the function?
- Which functions are suitable for integrating into one product?

Traditionally, the function configuration often depends on the designer' experience, they bundle suitable functions together to meet customer's needs. Although it is a versality method, it is too subjective and relies on the designer's experience and skill level. In addition, some methods analyze the function needs through questionnaires or surveys, which are time-consuming and the data obtained is limited.

In recent years, with big data technology development, data-driven methods have been applied in product design, including functional analysis and decision support. The quality and reliability of data-driven design results are affected by the data source. The data sources have also been expanded from a single type to various data sources, such as website reviews, machine data, physiological data, or patents. In fact, function configuration needs not only to solve the data sources but also need to be supplemented by reasonable and effective means. Compared to other data sources, patent data has become a vital knowledge source of function-oriented product design, because it is easier to access a large amount of data and connect with product

development closely [3].

With the aim of dominating the market, the enterprise has to develop new functions with multi-functions continually. After developing new products, the patent application is needed to protect enterprise's intellectual property to avoid plagiarism. Many factors can affect the function configuration, such as product types, enterprises' business models and production lead time. These factors have formed a complicated relationship and reflect in patents. The advantages of complex networks in multi-entity relationship analysis are necessary to introduce complex networks for analysis. Recently, network-based patent analysis has attracted much attention among scholars [4]. Luo and Song explored product design space by the patent network [5].

Based on these observations, this study proposes a patent-driven method for product function deployment based on a hypergraph. In addition, this paper innovatively combines data mining technology with networks to propose a new method, which is used to solve multi-functions configuration problems in product development. The innovation of this paper is reflected in the following aspects: 1). Functional knowledge is extracted from patents as nodes, then combined with product patent and the patent applicant as edges to construct a multi-edges hypergraph network; 2). The function node weight is calculated by using the applicant edge, and the patent edge weight is calculated by combining the number of patent citations and the number of patent families; 3). The multi-functional combination weight calculation model is proposed and combined with the network subgraph algorithm to obtain the optimal function combination.

The rest of this paper is organized as follows. The theoretical background and related literature review are discussed in the section 2. In section 3, the research framework for patent-driven PFC is proposed. Then, a weighted FH is constructed in section 4 and followed by the empirical analysis that tests the validity of the proposed method in the application. Finally, conclusions are presented.

2 Literature Review

PFC is a portfolio innovation, which refers to the convergence of different technical features and solutions that are different from existing products [6]. This combination is realized by fusing one or more elements based on the original technology or product [7]. However, the product function configuration is not an arbitrary

superposition of elements, it requires an accurate and comprehensive analysis in advance [8]. Therefore, the current research on technology combination mainly focuses on three aspects: 1) Patent data analysis; 2) Technology opportunity analysis; and 3) Technology convergence analysis.

2.1 Patent data analysis

The purpose of patent mining is to extract valuable data from patents. Patent data can be divided into structured data and unstructured data. Structured data mainly includes application date, citation relationship and classification number. Compared to structured data, patents contain more unstructured data. In previous research, Daim et al analyzed the trend of technology development through patent number, and then select technologies for the firm's R&D [9]. The results showed that the number of patent applications for emergency technologies increased faster than other technologies. Some other scholars clustering patent citation data to find out the potential technology reorganization list [10]. A patent citation is a document cited by applicants or patent office examiners because its content relates to the other patent application. It is generally believed that the more a patent is cited, the higher quality and intrinsic technical value will be. Meanwhile, the number of citations will increase with time since patent publicly, and the value accumulates continuously. Besides, patent classification numbers are often used for technical analysis [11]. The patent classification scheme is a coding system that groups inventions by technical field [12]. The commonly used classification numbers are IPC and CPC. The CPC number has more levels, which has 6 to 7 levels, while the IPC number has only 5 levels. Therefore, more scholars tend to apply CPC number to exploring technological opportunities [13].

Unstructured patent information is mainly composed of textual data. Textual data can provide richer information, it is an important data source for current technology opportunity studies. Blake and Ayyagari obtained popular technology opportunities through literature theme trend analysis [14]. Zhang and Yu conducted research on the technical topic extension by using keyword analysis algorithm [15]. Based on the existing keyword research, Choi et al. integrated dependency syntax and part-of-speech filtering method to collect subject, action, and object (SAO) vocabulary, then to obtain word combinations related to technological opportunities [16]. Moreover, Kwon et al. explores the potential technical opportunities through mining patent texts underlying

semantics [17].

2.2 Technology opportunity analysis

Technology opportunity analysis (TOA) is a method develops from technology forecasting to practicability based on patent data. TOA has attracted the attention of many scholars. And the research is rich and diverse. Through tracking enterprises' technological development, Hou and Yang obtained patents that had not been taken seriously for a long time as potential development targets [18]. In addition, scholars have also studied the formation of patent jungle communities as a technological opportunity [19]. Such as Jin et al. [20] obtained vacuum technology through the technical efficiency matrix, that is, the technical field that has not yet been involved and takes it as the technology expansion objective. Moreover, through the study of scientific literature that directly associated patents, namely goalkeeper patents, as the must obtained technology in the transferring process of scientific theory to industrial applications [21].

To obtain more details, some scholars also use patent texts as research objects to explore more technological opportunities. Wang et al. analyzed the text topics development trend to identify the topics of technological convergence [22]. Yun and Geum applied the latent dirichlet allocation (LDA) algorithm to extract technical topics from patents [23]. Kim et al. tracked the development path through patents semantic similarity in different application times and then provides technology prediction reference [24]. Li et al. combined TRIZ theory and natural language processing technology to evaluate patent creativity level, thus identifying high impact patents [25]. Sheu and Yen identified the harmful resources in patents, to provide a risk reference for TOA [26].

2.3 Technology convergence analysis

Technological convergence refers to the tendency that in the process of technology development, previous unrelated technologies become more closely integrated or even unified. At present, the methods of technology convergence analysis are mainly divided into three categories: two-dimensional map, incidence matrix, and graphs.

For the two-dimensional map analysis method, Lee et al. proposed to obtain keywords by reducing patent data dimension through principal components, and obtain vacuum patent technology accordingly [27]. Lee and Lee constructed a landscape map from patent information as a vector space model to present the configuration of technological components [28]. Seo et al. proposed a

portfolio map with two patent value indices as axes to analyze the patents of target enterprises, in order to discover potential technology opportunities [29].

The incidence matrix is a logical matrix that shows the relationship between two classes of objects, which including the morphology matrix, design structure matrix and vector space matrix. Arciszewski generated new schemes through literature mining, first mined technical keywords in different fields through patent text data, and then combining keywords through the morphological matrix to generate new solutions [30]. Feng et al. calculated the correlation between the technology and the product by designing a structural matrix, then identify the technological development opportunities [31]. A design structure matrix is a tool for analyzing the relationship between different objects. Zheng et al. constructed a pairwise relationship matrix between themes through the co-occurrence, in which the matrix element is the number of co-occurring patents. Vector space model (VSM) is one of the most robust methods for information analysis [32]. Jun et al. introduced a matrix mapping and K-medoids clustering method based on vector matrix to better predict missing technology [33]. Lei et al. proposed a patent analytics method based on VSM to solve the semantics and curse-of-dimensionality loss [34].

In recent years, more and more scholars have adopted graphs to study technology integration. Compared to two-dimensional maps and incidence matrix, the complex network analysis provides a better visualization result, supports more nodes analysis, and implements key technologies acquisition and clustering through a large number of measurement algorithms [35]. Kim et al. identified core technologies from the perspective of technological cross-impacts by analyzing network processes and association rules algorithms methods [36]. Sung et al. used growing cell structure networks to analyze core technologies [37]. Song et al. demonstrated patent keywords through the core-peripheral network, and gravity algorithms demonstrated important technical keywords [38]. Some studies were performed by subgraphs formed from a subset of the vertices of a graph and all the edges connecting pairs of vertices in the subset. Lee et al. used the subgraph unit based on the existing node analysis and uses the QAP algorithm to calculate the correlation between different subgraphs to analyze the technology integration [39]. Lee et al. applied the frequent subgraph algorithm to analyze the correlation of network nodes, and obtained the best technology combination by calculating the confidence and support [40]. Sun et al. formed different patent clusters through text mining technology and studies

the overlapping situation of different communities as a basis for judging technology integration [41].

Although plenty of scholars have conducted studies and made contributions to TOA based on the patent-driven method, the current studies still have some deficiencies as follows:

- The objectives of previous research mainly focus on one type of patent. The convergence objects, whether patent objects or keywords are mostly defined as technical opportunities without distinguishing the differences. At the same time, it rarely involves the excavation of functional requirements. Such research is difficult to provide accurate suggestions for market functional expansion.
- Two necessary steps are missing in the current research, namely calculating and compute the weights of convergent objects in the function configuration and sorting the clusters formed after convergence.
- The analysis tools involved in the current research regard objects as a single relationship, in fact, there are multiple relationships between objects. For example, there is a patent co-occurrence relationship between patent functions and the relationship with the applicant. These relationships will affect the exploration and integration of available functional opportunities.

Therefore, it is reasonable and necessary to develop a novel and efficient method to help firms detach market function opportunities and make function configurations based on patent data. Meanwhile, this study is expected to assist designers in selecting new product functions and the size of functions cluster in one product.

3 Research Framework

Firms often applicate patents for the whole design schemes of products with the multi-functions aim to cost reduction, especially for consumer products [42]. Thus, it is feasible to analyze the function configuration of existing products through patent data. This study proposes a new patent-data-driven product function configuration method based on hypergraph and develops a framework based on the method, as shown in Figure 1. The framework consists of four steps: Patent data collection and mining, function hypergraph construction, function configuration scheduling, and configuration analysis.

Step 1: Patent data collection and mining. According to customer needs and industry life cycle, search terms are conceived to retrieve industry patents. Then download the patents from the website to construct a database on a local

PC. Structured data such as the number of citations, applicants, and application date were obtained by paragraph cutting. Other unstructured data, such as text data, need to be cleaned to remove noisy information such as numbers, symbols, and auxiliary vocabularies.

Step 2: Function hypergraph construction. Multi-part text mining based on TF-IDF (MPTM-TFIDF) algorithm method is used to weigh the words, then select keywords as patent function labels by regular expressions, and capitalization is not distinguished. For those combinations, requires the same sentence appears simultaneously, such as the phrase “cold water” which requires both “cold” and “water” in the same sentence. Then formed a label set with different functions. Besides, use an adjacency matrix to describe the relationship of function and the patent or the applicant. Lastly, based on the matrix, the patent functions hypergraph model is constructed.

Step 3: Function configuration scheduling. Apply applicant edge to weight functions node, and use citation number and patent family size to weight patent edge. A comprehensive calculation model is also constructed to weight function community and then the improved frequent subgraph algorithm was used to identify optical functions combination in the hypergraph network.

Step 4: Function configuration recommendation. According to the existing product functions or customer requirements, target functions are obtained from Step 3. Finally, the accuracy of the configuration results is verified by market analysis.

Figure 1 Patent function configuration framework based on hypergraph

4. Research methodology

4.1 Keyword extraction based on MPTM-TFIDF

TF-IDF is a statistical measure algorithm that evaluates how important a word is to a document in the collection [43]. The formula of TF-IDF is shown in equation (1):

$$w_D^T = TF(T, D) \times IDF(T) \quad (1)$$

$$TF(T, D) = \frac{freq(T, D)}{sizeof(D)} \quad (2)$$

$$IDF(T) = \log_2 \frac{sizeof(N)}{1 + count(T, N)} \quad (3)$$

Where w_D^T donates the weight of term T in document

D . $freq(T, D)$ donates the frequency of terms T in documents D . $sizeof(D)$ donates the number of terms in documents D . $sizeof(N)$ is the number of all documents. $count(T, N)$ donates the number of documents which contain terms T .

In the keyword extraction process, the TF-IDF algorithm is able to clean low weight words, namely, the word frequently shows in documents but is not essential [44]. Nevertheless, the accuracy of the algorithm is low when processing massive text data [45]. Product function words are distributed in different parts of the patent, such as title, abstract, claim, and technical background. The amount of text data in different parts varies greatly.

To this end, this paper proposes the MPTM-TFIDF method. Firstly, to ensure the keyword extraction accuracy, the method used to extract the critical information in all patent titles as the text in this part is much less than other parts. Then, keywords with higher TF-IDF weight in different patents are collected and merge words with the same meaning together when the similarity calculated by wordnet exceeds the threshold. Finally, through the set elements, the similar words in the abstract and technical background of all patents are searched to determine the patent's functions by regular expressions.

(a) Hypergraph (b) K-uniform hypergraph (c) Ordinary graph

Figure 2 Hypergraph, K-uniform hypergraph and graph

Hypergraph can be illustrated as incidence matrix $|V| \times |E|$, and the element is $h(v, e)$. The value of $h(v, e)$ is defined in equation (4):

$$h(v, e) = \begin{cases} 1 & \text{if } v \in e \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

In addition, Figure 3 is used to further explain the relationship between incidence matrix and hypergraph.

(a) Hypergraph case

(b) Incidence matrix

Figure 3 Case of hypergraph and incidence matrix

4.2 Hypergraph Model Construction

In the ordinary graph, an edge connects precisely two vertices that donate one to one relationship [46]. Although the structure is concise, there are limitations in expressing relationships among multi vertices [47]. In contrast, the hyperedge in hypergraph can contain any number of nodes, and a set of hyperedges can exist simultaneously in a hypergraph network. Therefore, this study chooses the hypergraph for analysis.

Definition 1. A hypergraph $H = (V, E)$, where $V = \{v_1, v_2, \dots, v_n\}$ is the finite set of nodes called vertices and $E = \{e_1, e_2, \dots, e_m\}$ is an indexed family of sets called hyperedges in which $e_i \subseteq V$. The degree of a vertex is the number of hyperedges to which it belongs, $d(v) = |\{e: v \in e\}|$, and the size of a hyperedge is its cardinality nodes, $|e_i| = k (1 \leq k \leq n)$. Hypergraph with hyperedges of size k is called k -uniform, while a 2-uniform hypergraph is simply called ordinary graph [48]. Figure 2 gives an example of hypergraphs.

4.3 PFC model construction

As PFC involves not only products and firms, but also complex multi-entity relationships and multilateral relationships. When the patent-driven function configuration method is adopted, these two relationships are transformed into patent relations and applicant relations to form a hypergraph model with functions.

Definition 2. A patent function hypergraph (FH) can be presented as $FH = (F, E)$, where $F = \{f_1, f_2, \dots, f_m\}$ donates the set of function nodes, $E = \{ep, ec\}$ donates the sets of edges, where $ep = \{ep_1, ep_2, \dots, ep_p\}$ donates the set of patent edges and $ec = \{ec_1, ec_2, \dots, ec_c\}$ donates the set of applicant hyperedges.

Since FH has more than one edge, FH is illustrated as many incidences matrices as possible according to the edges. The value of matrix elements can be calculated by equation (4).

4.4 Hypergraph weight calculation

Hypergraph weight calculation includes calculations of function node weight and patent edge weight.

4.4.1 Function node weight calculation based on patent applicant hyperedge.

In order to be the leading product in the market, products need to continuously meet customer needs. Thus, to improve product competitiveness in the market, important functions need to be integrated to increase market

attractiveness. There are currently very few definitions of essential functionality. Therefore, referring to the definition of technological opportunities [49], the important function is defined as follows.

Definition 3. Important functions refer to functions that are widely and quickly accepted by the market.

From the market perspective, important functions are those consumers generally accept. From the perspective of the patent, important functions are objects that are widely and up to date used by the respondent. The more enterprises are involved in the product development, the more critical this function is in the market [50]. The later the feature appears, the more likely it is to catch on.

Therefore, the weight of the function node w_{f_i} in the hypergraph is calculated from the time index of the applicant's hyperedge and function, as shown in equation (5).

$$w_{f_i} = wt_j \cdot \sum_{ec_j^i \in ec} h(f_i, ec_j^i) \quad j = 1, 2, \dots, c \quad (5)$$

where wt_j donates the time index of the function. Considering that the longer the function is available, the less popular it will be in the market. In contrast, new features are more likely to be popular. Therefore, the inter-function index is calculated as shown in equation (6).

$$wt_j = \frac{1}{(T_n - T_{jf} + 1)} \quad (6)$$

where T_n donates the current year and T_{jf} donates the year when the product of function f_i was first applied as a patent. Based on the above formulas, equation (7) is used to calculate the weight of function f_i :

$$w_{f_i} = \frac{\sum_{ec_j^i \in ec} h(f_i, ec_j^i)}{(T_n - T_{jf} + 1)} \quad (7)$$

Due to the significant deviation of the number of patent applicants hyperedge with different function nodes, data normalization is necessary for adjusting values on a large scale to a minor scale. There are different types of normalization in statics. According to patent data characteristics, maximum-minimization normalization is adopted to bring all values into the range [0,1], as shown in equation (8).

$$w'_{f_i} = \frac{w_{f_i} - \min(w_{f_i})}{\max(w_{f_i}) - \min(w_{f_i})} \quad (8)$$

In the formula, $\min(w_{f_i})$ donates the lowest value of the weight range of all functional nodes and $\max(w_{f_i})$ donates the highest value of the weight range of all functional nodes.

4.4.2 The calculation of patent hyperedge weight based on patent quality

Through the analysis of product patents, the function distribution of corresponding products can be extracted. Therefore, the patent hyperedge weight is calculated from the perspective of patent quality. According to the World Intellectual Property Organization [51], the core indexes for calculating patent quality are the citation number and patent family size. The more frequently a patent is cited, the greater its impact [52]. The larger the patent family, the more countries the patent is filed for, and the higher the economic value [53]. Therefore, these two indexes are taken into patent hyper weight calculation, as shown in equation (9).

$$w_{ec_i} = \varphi \cdot fep_i + (1 - \varphi) \cdot cep_i^t \quad (9)$$

where w_{ec_i} donates the weight of the applicant hyperedge ec_i . fep_i donates the number of patent families of patent ep_i . cep_i^t donates citation number per year of patent ep_i , and this reflects the degree to which the patent is valued by peers; φ donates the weight ratio of the two indicators.

It should be noted that the number of patent citations is closely related to the service life of the patent, which means the earlier a patent is applied, the more chances it will be cited, and the number of citations will be higher than that of subsequent patents [54]. In order to eliminate the impact of patent application time, the number of patent citation per year in the whole life cycle is set as the weight calculation index, thus cep_i^t is calculated by equation (10).

$$cep_i^t = \frac{cep_i}{T_n - T_{ep}^i} \quad (10)$$

where cep_i represents the total number of citations of patent ep_i , and T_{ep}^i represents the application year of the patent ep_i .

Through the maximum and minimum standardization process, the calculation formula of the weight w_{ep_i} of ep_i is shown in equation (11).

$$w_{ep_i} = \varphi \frac{fep_i - \min(fep)}{\max(fep) - \min(fep)} + (1 - \varphi) \frac{cep_i^t - \min(cep^t)}{\max(cep^t) - \min(cep^t)} \quad (11)$$

4.5 Function configuration based on hypergraph

The PFC consists of two steps: the evaluation and the acquisition of the function community.

4.5.1 Evaluation of function community

The versatility of one product forms the community in FH. Before PFC, the function community needs to be evaluated comprehensively in advance and filtered through the hypergraph. According to the PFC model, different functions form a community through patent hyperedges. Therefore, the community's evaluation needs to consider the weight of the function node and the patent hyperedge. Suppose function nodes and hyperedges form same community subgraph $FH^o(F^o, ep^o)$, where $F^o = \{f_1^o, f_2^o, \dots, f_o^o\}$ and $ep^o = \{ep_1^o, ep_2^o, \dots, ep_o^o\}$. The weight of the function community w_{FH^o} is calculated by equation (12).

$$w_{FH^o} = \sqrt{\left(\sum_{ep_i^o \in ep^o} w_{ep_i^o} + 1\right) * \frac{(\sum_{f_i^o \in F^o} w_{f_i^o} + 1)}{N_{F^o}}} \quad (12)$$

where $w_{ep_i^o}$ donates the weight of the patent hyperedge ep_i^o . $w_{f_i^o}$ donates the weight of function node f_i^o . And N_{F^o} donates the number of the function community F^o .

According to equation (12), the more functions and hyperedges the function community contains, the higher its weight. That the more functions a product contains, the more customer individual requirements it can meet. Meanwhile, the more products with similar functional combinations, the more critical the function community becomes.

4.5.2 Function community acquisition based on frequent subgraphs

FH contains a large number of subgraphs, and each representing a functioning community. For obtaining the optimal combination of functions, the frequent subgraph algorithm is introduced to select the optimal function community. Currently, there are two commonly used frequent subgraph mining algorithms: Apriori and FP-Growth. Compared with the FP-Growth algorithm, the Apriori algorithm is more mature and widely used [55]. Therefore, this paper adopted Apriori for FH subgraph mining.

Apriori algorithm is generally used to screen subgraphs. However, the existing research on obtaining the optimal subgraph based on the Apriori algorithm ignores the weight of the subgraph,

and the results are not accurate. In this regard, IFSA is proposed in this paper. This algorithm uses the weights of function communities as the basis for subgraph screening.

Definition 4. Improved frequent subgraphs algorithm (RFSA). Given a hypergraph H and a minimum comprehensive weight τ , $S_{up}(H, FH^o)$ represents the weight of the subgraph FH^o in the H , when $S_{up}(H, FH^o) \geq \tau$, FH^o is a frequent subgraph of H . The calculation formula of $S_{up}(H, FH^o)$ is as follows:

$$S_{up}(H, FH^o) = \frac{w_{FH^o}}{w_H} \quad (13)$$

where w_H is the sum of the weights covering all patent communities, and the calculation method is shown as below:

$$w_H = \sum_{FH^o \in H} w_{FH^o} \quad (14)$$

The algorithm is described as follows:

1. //Improved priori Algorithm
2. Function apriori (FH^k, D, τ)
3. //D – Graph Database
4. // FH^k – frequent subgraphs of size ‘k’
5. // τ – minimum wight of subgraph
6. // L_{k+1} - list of graphs whose size is ‘k+1’
7. $FH^{k+1} \leftarrow \text{NULL}$;
8. for all $g_i \in FH^k$:
9. $L_{k+1} \leftarrow$ form as many as graphs possible by combining g_i and g_j : the size of the resultant graph is ‘k+1’
10. for all $g_u \in L_{k+1}$:
11. if [$(S_{up}(H, g_u) \geq \tau)$ AND $(g_u \notin FH^{k+1})$]:
12. $FH^{k+1} \leftarrow FH^{k+1} \cup g_u$
13. else if [FH^{k+1} is NULL]
14. k is optimal size
15. $k \leftarrow k + 1$
16. end
17. end function

Through RFSA, the optimal number of function combination k is obtained, and the groups with better weight from all the same number of function combinations are explored. The algorithm provides a reference for designers to determine the function quantity and optimal function combinations.

5 Case Study

Meeting the needs of every individual customer for bathroom furniture and accessories is a challenging task, especially for showerheads. Developing fashionable and

attractive products is the goal of many enterprises. This section presents a case of the method proposed in the paper. The algorithms are encoded and run using Python software. Patent data is retrieved and downloaded through Patsnap Database (<https://www.patsnap.com/>), a famous commercial patent database service.

Now, there is a firm that mainly produces handle showers with rain functions. This product function is scarce and lack market competitiveness. Therefore, the firm wanted to develop a new multi-function shower and commissioned us to help with patent analysis to detach new function opportunities from the market, and then arrange the function configuration. In the beginning, using the keywords and CPC number in Table 1 to search for patents applied in USPO, and a total of 1358 patents were obtained.

Table 1 U.S. shower patent search results

Retrieval of patent	result
Title or Abstract:(showerhead* OR shower head* OR sprayer*) AND CPC:(B05B1/18) AND Time:(from 19140101 to 20200101)	1358

TF-IDF algorithm is used to calculate the weight of the vocabulary in patent titles, the result is shown in Table 2, and the words with high values are often related to product functions. Therefore, these vocabularies were collected, and the synonyms are merged to 65 function keywords, as shown in Table 3.

Table 2 Information of patents

No	Patent number	Applicant	Title of patent	TF-IDF value
1	US14/360881	DURANT	Concealed top cover-type shower head	"concealed": 0.506, "cover": 0.481, "head": 0.205, "shower": 0.134, "top": 0.481, "type": 0.471
2	US15/334907	DURANT	Combination showerhead with swing button switching	"button": 0.422, "combination": 0.431, "showerhead": 0.275, "swing": 0.603, "switching": 0.376, "with": 0.234
3	US15/871091	JOHANNES	Massage shower head	"head": 0.404, "massage": 0.877, "shower": 0.2613
...	...	...	...	...
1358	US17/040644	KOHLER	Washing unit	"unit": 0.651, "washing": 0.759

Table 3 List of functions

Function code	Description of functions	Function keywords
f01	Adapted for seats	seat* or chair *
f02	Adapted for wash-basins or bath	wash-basins or bath
f03	Washing specific parts of the body	special part
f04	Jet particular shape water	particular shape
f05	Scanner water	scanner
f06	Massage function	massage
f07	Jet annular, tubular or hollow conical form water	annular, tubular or hollow conical form
f08	Moving outlet	moving outlet
f09	Recovery of heat	recovery of heat
f10	Diverting cold water	cold water
f11	Concealed shower	conceal*
f12	Void slippage or dropping	slippage or dropping
f13	Combined shower	combined shower
f14	Fixed shower	fixed shower

f15	Handle shower	handle shower
f16	Void leakage	leakage
f17	Water saving	water saving
f18	Rainfall water	rain
f19	Mist water	mist water
f20	Pulsating water	pulsating water
f21	SPA	SPA
f22	Void blockages	blockages
f23	Self-clean function	self-clean
f24	Lighting	light* or led
f25	Anion Energy	anion*
f26	Stop flow	stop
f27	Kill bacteria	bacteria
f28	Wand	wand
f29	Extendable outlet	extendable outlet
f30	Selective outlet	selective outlet
f31	Suspending or supporting shower	suspending or supporting
f32	Diverter valves	diverter valves
f33	Control by button	button
f34	Adjustable head	adjustable
f35	Electrical control	electrical control
f36	Multi-outlets	outlets
f37	Detach function	detach
f38	Strainers	strainers
f39	Oriented jet	oriented jet
f40	Mixing ratio	mix ratio
f41	Restrict flow	restrict flow
f42	Temperature response	temperature response
f43	Lifting valve	lift valve
f44	Control pressure	control pressure
f45	Mounted faucet	mounted faucet
f46	Control volume	control volume
f47	Heating water	heating
f48	Filter stream	Filters located upstream
f49	Control temperature	control temperature
f50	Generate electricity by motor	motor
f51	Mixture with other material	mixture OR aromatherapy
f52	Jet particular shape	particular shape
f53	Scald	scald
f54	Touchless control	touchless
f55	Adjusting position of head	adjusting the position
f56	Jetting hot air	hot air
f57	Collecting water	collect*

f58	Making radio	radio
f59	Making alarm	alarm
f60	Making music	music
f61	Communication of internet	internet
f62	Communication of Bluetooth	Bluetooth
f63	cleaning hair	hair
f64	Cleaning pets	pet* or cat* or dog*
f65	Using for child	child* OR baby

* Means multiple arbitrary characters

The functions in Table 3 were used to label patents with regular expressions, and the results are shown in Table 4. The table represents that the patent has a corresponding function, otherwise it does not. There are obvious differences in the number of functions of different patents.

Table 4 Function labels of patents

No	Patent number	Function label
1	US14/360881	f11, f14, f30,
2	US15/334907	f15, f30, f37,
3	US15/871091	f06, f14, f18, f46
...		
1358	US17/040644	f15, f17, f57

To further verify the effectiveness of this method, we compared our results with three traditional keywords extraction algorithms by Precision (P), Recall (R) rate and F-measure(F) equation (13-15). Ten patents that contain more than 600 words were randomly selected as test objects, and experts were invited to verify the effect. The results are shown in Figure 4, ROC value of MPTM-TFIDF is significantly better than other algorithms.

$$P = \frac{TP}{TP+FP} \quad (13)$$

$$R = \frac{TP}{TP+FN} \quad (14)$$

$$F = \frac{2 \times P \times R}{P+R} \quad (15)$$

where TP donates true positive, FP donates false positive, and FN donates false negative.

Figure 4 Comparison of four algorithms

Applicant hyperedges are used to calculate the weight of the node. Firstly, set the current year as 2020. Secondly, the number of patent applicants of all functions was calculated

corresponding, and the Min-Max Normalization algorithm is applied to obtain the weight in range (0,1). Finally, the calculation results are shown in Table 5.

Table 5 Weight of function nodes in hypergraph

Function code	First time of function appearance	Time index	Number of applicants	Weight	Weight normalization
f01	2016	0.167	12	2.000	0.135
f02	1972	0.020	32	0.640	0.039
f03	2001	0.048	12	0.571	0.035
f04	1991	0.032	15	0.484	0.029
f05	2011	0.091	7	0.636	0.039
f06	1988	0.029	46	1.353	0.090
f07	2013	0.111	13	1.444	0.096
f08	1972	0.020	55	1.100	0.072
f09	2006	0.063	6	0.375	0.021
f10	2008	0.071	15	1.071	0.070
f11	1945	0.013	53	0.688	0.043
f12	1970	0.019	23	0.442	0.026
f13	1942	0.013	24	0.300	0.016
f14	1981	0.024	220	5.366	0.371
f15	1914	0.009	679	6.287	0.436
f16	1924	0.010	164	1.673	0.112
f17	1989	0.030	71	2.152	0.146
f18	1920	0.010	395	3.873	0.266
f19	1959	0.016	106	1.683	0.113
f20	1981	0.024	31	0.756	0.048
f21	2009	0.077	1	0.077	0.000
f22	1989	0.030	21	0.636	0.039
f23	1959	0.016	19	0.302	0.016
f24	1984	0.026	31	0.816	0.052
f25	1984	0.026	7	0.184	0.008
f26	1982	0.025	35	0.875	0.056
f27	2004	0.056	5	0.278	0.014
f28	1992	0.033	9	0.300	0.016
f29	1990	0.031	81	2.531	0.172
f30	1962	0.017	188	3.133	0.214
f31	1914	0.009	93	0.861	0.055
f32	1914	0.009	75	0.694	0.043
f33	1939	0.012	27	0.325	0.017
f34	1971	0.020	50	0.980	0.063
f35	1983	0.026	36	0.923	0.059
f36	1988	0.029	40	1.176	0.077
f37	1974	0.021	66	1.375	0.091
f38	1961	0.016	67	1.098	0.072
f39	2007	0.067	34	2.267	0.154
f40	1963	0.017	35	0.593	0.036
f41	1983	0.026	39	1.000	0.065
f42	1990	0.031	42	1.313	0.087
f43	2001	0.048	32	1.524	0.101
f44	2013	0.111	129	14.333	1.000
f45	1996	0.038	27	1.038	0.067
f46	1959	0.016	102	1.619	0.108
f47	2013	0.111	6	0.667	0.041
f48	2003	0.053	23	1.211	0.080
f49	1959	0.016	22	0.349	0.019

f50	1978	0.023	20	0.455	0.026
f51	1916	0.009	181	1.708	0.114
f52	1991	0.032	24	0.774	0.049
f53	1942	0.013	78	0.975	0.063
f54	2011	0.091	17	1.545	0.103
f55	1982	0.025	54	1.350	0.089
f56	1994	0.036	16	0.571	0.035
f57	1997	0.040	29	1.160	0.076
f58	1988	0.029	47	1.382	0.092
f59	1994	0.036	27	0.964	0.062
f60	1988	0.029	12	0.353	0.019
f61	2015	0.143	5	0.714	0.045
f62	2010	0.083	21	1.750	0.117
f63	1934	0.011	141	1.602	0.107
f64	1999	0.043	22	0.957	0.062
f65	2013	0.111	3	0.333	0.018

For a more concise visualization of the graph, Figure 5 shows the hypergraph using the Python hypergraph tool, where patent hyperedge is labelled with “ep” and applicant is labelled with “ec”. Meanwhile, functions in Table 3 are

taken as the nodes and the patents and the applicants in Table 2 are both taken as the hyperedges. To distinguish different functions, those functions with higher weights are represented by nodes with a larger radius.

Figure 5 Hypergraph with part of edges and nodes

Since the importance of patent citation number ep and patent family size fep is equal, the weight ratio of the two indicators φ is set at 0.5. Based on patent data, the

weight w_{ec} of patent hyperedge is calculated by ep and fep (as listed in Table 6).

Table 6 The weights of patent edges in the hypergraph

No	Application number	Application time	Patent family	Patent citation	Weight
1	US14/360881	2012	3	0	0.009
2	US15/334907	2016	4	0	0.014
3	US15/871091	2018	2	0	0.005
...	...	...	...	...	...
1354	US17/040644	2019	1	0	0

Note: To save space, only part of the evaluation data is provided

Although there are 1354 patents in total, the number of communities is 753 when patents with the same function are merged in one community. The weights of function communities are identified by equation (12) and listed in Table 7. Many MFP are higher than the product with fewer functions or single function. This indicates MFP is well received by the market.

Table 7 Weight of function community

Functions community	Weight
f51, f40, f47, f35, f44, f49, f15	1.608
f51, f40, f47, f35, f44, f49, f16, f53, f15	1.241
f51, f40, f47, f35, f50, f44, f49, f15	1.066
f15, f18	0.894
...	...
f52, f20	0.069

Note: To save space, only part of the evaluation data is provided

Due to the firm's existing product functions are handling function (f15) and rainfall water function (f18), the RFSA introduced in section 4.5.2 is implemented to help the firm complete PFC. The minimum weight τ is set to 0.2, then the subgraphs including (f35, f60, f18, f58, f15), (f18, f63, f15), (f18, f63, f15, f01), (f18, f15, f17, f41), (f42, f17, f38, f18, f15) and (f18, f19, f15) whose weight exceed τ are

mined and filled in yellow (as shown in Figure 6). Meanwhile, nodes with high weight, such as f44 and f14, are not in these optimal communities. This means the function configuration process depends not only on element weight but also on the degree of correlation between elements, which is a complicated process.

(a) (f35, f60, f18, f58, f15)

(b) (f18, f63, f15)

(c) (f18, f63, f15, f01)

(d) (f18, f15, f17, f41)

Figure 6 Frequent subgraphs of hypergraph network whose weight exceeds the threshold

To verify the accuracy of the proposed method, the number of products on the e-commerce website was counted. Currently, there are more than 7,000 shower products on Amazon (<https://www.amazon.com>). Since there are 753 function communities, each community has fewer than 130 functions (listed in Table 8). However, the functional community is mostly located in more than 200 products. This means that those MFP designs are more popular in the market.

Table 8 Configuration of functions exceeding the threshold

Communities of functions	Number of products	Web of Amazon
(f35, f60, f18, f58, f15)	252	https://www.amazon.com/s?k=shower+music&rh=n%3A680352011&ref=nb_sb_noss
(f18, f63, f15)	381	https://www.amazon.com/s?k=shower+hand+hair&i=tools&rh=n%3A680352011%2Cp_n_feature_seventeen_browse-bin%3A16966680011&dc&qid=1617192575&rmid=16966675011&ref=sr_nr_p_n_feature_seventeen_browse-bin_5
(f18, f63, f15, f01)	381	https://www.amazon.com/s?k=shower+hand+hair&i=tools&rh=n%3A680352011%2Cp_n_feature_seventeen_browse-bin%3A16966680011&dc&qid=1617192575&rmid=16966675011&ref=sr_nr_p_n_feature_seventeen_browse-bin_5
(f18, f15, f17, f41)	221	https://www.amazon.com/s?k=shower+music+flow&rh=n%3A680352011&ref=nb_sb_noss
(f42, f17, f38, f18, f15)	195	https://www.amazon.com/s?k=shower+filter+temperature&rh=n%3A680352011&ref=nb_sb_noss
(f18, f19, f15)	356	https://www.amazon.com/s?k=shower+mist&i=tools&rh=n%3A680352011%2Cp_n_feature_seventeen_browse-bin%3A16966680011&dc&qid=1617192903&rmid=16966675011&ref=sr_nr_p_n_feature_seventeen_browse-bin_5

To further verify the results, one case of MFP put on the shelf with functions f35, f60, f18, f58, and f15 is shown in Figure7(https://www.amazon.com/LINLIM-Handheld-Blue-tooth-Holiday-Creative/dp/B08ZSLRXGX/ref=sr_1_59?dc_hild=1&keywords=shower+music&qid=1617437581&s=ki-tchen-bath&sr=1-59). The case study shows that

multi-functional products are more attractive to customers.

Figure 7 The product with function f35, f60, f18, f58, and f15

6 Conclusions

To solve the 2H-2 problem in the MFP design process, a patent data-driven method based on hypergraph network was proposed. In addition, NLP and association rule algorithms are also applied. Compared to previous studies, the contributions of the research are summarized as follows:

- (1) In this study, the MPTM-TFIDF algorithm was used to extract functional keywords from patent title text, and then these keywords were used to retrieve patent full-text data, with the aim of labelling each patent with function keywords. This method can accurately mine most function data.
- (2) This study constructed FH, in which patents or applicant presents the function and edge presents node. Applicants calculate the weight of node, and the weight of the patent edge is calculated from the number of citations and families. Besides, the community weight calculation model of the function nodes is proposed.
- (3) IFSA is proposed to discover the subgraph with the lowest community weight. This method can quickly obtain the optimal function combination of products.

Finally, the method proposed in this paper is applied to the design of shower products and verified by e-commerce data. In fact, the product function configuration needs to consider many factors, such as fashion factors, regulations, policies and incentives. Therefore, patent information alone is not enough for product design, different types of information are also needed.

7 Declaration

Acknowledgements

The authors sincerely thanks to Professor ** of ** University for his critical discussion and reading during manuscript preparation.

Funding

Supported by the National Natural Science Foundation of China (NSFC) (Grant Nos. 51875220), and China Fujian Province Young and Middle-aged Teacher Education Research Project (Grant Nos. JAT200472).

Availability of data and materials

The datasets supporting the conclusions of this article are included within the article.

Authors' contributions

The author' contributions are as follows: WG-L and RB-X were in charge of the whole trail; XD-L assisted with patent text mining. All authors read and approved the final manuscript.

Competing interests

The authors declare no competing financial interests.

Consent for publication

Not applicable

Ethics approval and consent to participate

Not applicable

References

- [1] M Liu, D Trefler. What's the big idea? Multi-function products, firm scope and firm boundaries. *J Econ Behav Organ*, 2020, 180: 381-406.
- [2] Y Lin, D Cao, Y Han, et al. Guiding conceptual design modeling based on product systems analysis. *Kybernetes*, 2011, 40: 790-799.
- [3] Z J Li, C L Wu, X H Zhu, et al. Design by analogy: achieving more patentable ideas from one creative design. *Chinese Journal of Mechanical Engineering*, 2018, 31(02): 25-34.
- [4] S Jun, SJ Lee. *A Small World Network for Technological Relationship in Patent Analysis*. New York: Springer, 2013.
- [5] J Luo, B Song, L Blessing, et al. Design opportunity conception using the total technology space map. *Ai Edam*, 2018, 32: 449-461.
- [6] F Gallouj, O Weinstein. Innovation in Services. *Research Policy*, 1997, 26: 537-556.
- [7] MM Keupp, O Gassmann. Resource constraints as triggers of

- radical innovation: Longitudinal evidence from the manufacturing sector. *Research Policy*, 2013, 42: 1457-1468.
- [8] C.J.M Jeroen, Van den Bergh. Optimal diversity: Increasing returns versus recombinant innovation. *Journal of Economic Behavior & Organization*, 2008, 68(3-4):565-580.
- [9] T Daim, I Iskin, X Li, et al. Patent analysis of wind energy technology using the patent alert system. *World Patent Information*, 2012, 34(1): 37-47.
- [10] Y Takano, C Mejia, Y Kajikawa. Unconnected component inclusion technique for patent network analysis: Case study of Internet of Things-related technologies. *Journal of Informetrics*, 2016, 10(4): 967-980.
- [11] SJ Lee, S Jun. Key IPC Codes Extraction Using Classification and Regression Tree Structure. *International Symposium Conference on Advanced Intelligent Systems*, Daejeon, Korea, 2014: 101-109.
- [12] S Oh, J Choi, N Ko, et al. Predicting product development directions for new product planning using patent classification-based link prediction. *Scientometrics*, 2020, 125(3): 1833-1876.
- [13] Y Geum, M Kim. How to identify promising chances for technological innovation: Keygraph-based patent analysis. *Advanced Engineering Informatics*, 2020, 46: 101155.
- [14] R Blake, R Ayyagari. Analyzing information systems security research to find key topics, trends, and opportunities. *Journal of Information Privacy and Security*, 2012, 8(3): 37-67.
- [15] J Zhang, W Yu. Early detection of technology opportunity based on analogy design and phrase semantic representation. *Scientometrics*, 2020, 125(1): 551-576.
- [16] S Choi, J Yoon, K Kim, et al. SAO network analysis of patents for technology trends identification: a case study of polymer electrolyte membrane technology in proton exchange membrane fuel cells. *Scientometrics*, 2011, 88(3): 863-883.
- [17] H Kwon, J Kim, Y Park. Applying LSA text mining technique in envisioning social impacts of emerging technologies: The case of drone technology. *Technovation*, 2017, 60-61: 15-28.
- [18] J Hou, X Yang. Patent sleeping beauties: evolutionary trajectories and identification methods. *Scientometrics*, 2019, 120(1): 187-215.
- [19] X Yuan, X Li. A network analytic method for measuring patent thickets: A case of FCEV technology. *Technological Forecasting and Social Change*, 2020, 156: 120038.
- [20] G Jin, Y Jeong, B Yoon. Technology-driven roadmaps for identifying new product/market opportunities: Use of text mining and quality function deployment. *Advanced Engineering Informatics*, 2015, 29(1): 126-138.
- [21] X Li, D Zhao, X Hu. Gatekeepers in knowledge transfer between science and technology: an exploratory study in the area of gene editing. *Scientometrics*, 2020, 124(2): 1261-1277.
- [22] Z Wang, A L Porter, X Wang, et al. An approach to identify emergent topics of technological convergence: A case study for 3D printing. *Technological Forecasting and Social Change*, 2019, 146: 723-732.
- [23] J Yun, Y Geum. Automated classification of patents: A topic modeling approach. *Computers & Industrial Engineering*, 2020, 147: 106636.
- [24] M Kim, Y Park, J Yoon. Generating patent development maps for technology monitoring using semantic patent-topic analysis. *Computers & Industrial Engineering*, 2016, 98: 289-299.
- [25] Z Li, D Tate, C Lane, et al. A framework for automatic TRIZ level of invention estimation of patents using natural language processing, knowledge-transfer and patent citation metrics. *Computer-Aided Design*, 2012, 44(10): 987-1010.
- [26] D D Sheu, M Yen. Systematic analysis and usage of harmful resources. *Computers & Industrial Engineering*, 2020, 145: 106459.
- [27] S Lee, B Yoon, Y Park. An approach to discovering new technology opportunities: Keyword-based patent map approach. *Technovation*, 2009, 29(6-7): 481-497.
- [28] C Lee, G Lee. Technology opportunity analysis based on recombinant search: patent landscape analysis for idea generation. *Scientometrics*, 2019, 121(2): 603-632.
- [29] W Seo, J Yoon, H Park, et al. Product opportunity identification based on internal capabilities using text mining and association rule mining. *Technological Forecasting and Social Change*, 2016, 105: 94-104.
- [30] T Arciszewski. Morphological analysis in inventive engineering. *Technological Forecasting and Social Change*, 2018, 126: 92-101.
- [31] L Feng, Y Niu, J Wang. Development of morphology analysis-based technology roadmap considering layer expansion paths: application of TRIZ and text mining. *Applied Sciences*, 2020, 10(23): 84-98.
- [32] P Zheng, C Chen, S Shang. Towards an automatic engineering change management in smart product-service systems – A

- DSM-based learning approach. *Advanced Engineering Informatics*, 2019, 39: 203-213.
- [33] S Jun, S S Park, D S Jang. Technology forecasting using matrix map and patent clustering. *Industrial Management & Data Systems*, 2012, 122(5): 786-807.
- [34] L Lei, J Qi, K Zheng. Patent Analytics Based on Feature Vector Space Model: A Case of IoT. *IEEE Access*, 2019, 7: 45705-45715.
- [35] C Sternitzke, A Bartkowski, R Schramm. Visualizing patent statistics by means of social network analysis tools. *World Patent Information*, 2008, 30(2): 115-131.
- [36] C Kim, H Lee, H Seol, et al. Identifying core technologies based on technological cross-impacts: An association rule mining (ARM) and analytic network process (ANP) approach. *Expert Systems with Applications*, 2011, 38(10): 12559-12564.
- [37] H Sung, H Yeh, J Lin, et al. A visualization tool of patent topic evolution using a growing cell structure neural network. *Scientometrics*, 2017, 111(3): 1267-1285.
- [38] B Song, J Luo, K Wood. Data-driven platform design: patent data and function network analysis. *Journal of Mechanical Design*, 2019, 141(2): 21101.
- [39] W J Lee, W K Lee, S Y Sohn. Patent network analysis and quadratic assignment procedures to identify the convergence of robot technologies. *PloS one*, 2016, 11(10): e165091.
- [40] W S Lee, E J Han, S Y Sohn. Predicting the pattern of technology convergence using big-data technology on large-scale triadic patents. *Technological Forecasting and Social Change*, 2015, 100: 317-329.
- [41] H Sun, H Du, J Huang, et al. Detecting semantic-based communities in node-attributed graphs. *Computational Intelligence*, 2018, 34(4): 1199-1222.
- [42] V Bruno, D François. The Cost Factor in Patent Systems. *Journal of Industry, Competition and Trade*, 2009, 9(4): 329-355.
- [43] J Murphy, K Fu, K Otto, et al. Function based design by-analogy: a functional vector approach to analogical search. *Journal of Mechanical Design*, 2014, 136: 10110210.
- [44] N Preschitschek, H Niemann, J Leker, et al. Anticipating industry convergence: semantic analyses vs IPC co-classification analyses of patents. *Foresight (Cambridge)*, 2013, 15(6): 446-464.
- [45] G Al-Talib, H Hassan. A study on analysis of SMS classification using TF-IDF Weighting. *International Journal of Computer Networks and Communications Security*, 2013, 1(5): 189-194.
- [46] N Zhang, Y Yang, J Su, et al. Modelling and analysis of complex products design based on supernetwork. *Kybernetes*, 2019, 48(5): 861-887.
- [47] J Zhu, J Zhu, S Ghosh, et al. Social Influence Maximization in Hypergraph in Social Networks. *IEEE Transactions on Network Science and Engineering*, 2019, 6(4): 801-811.
- [48] S G Aksoy, C Joslyn, M C Ortiz, et al. Hypernetwork science via high-order hypergraph walks. *EPJ Data Science*, 2020, 9(1): 16.
- [49] H Noh, Y Song, S Lee. Identifying emerging core technologies for the future: Case study of patents published by leading telecommunication organizations. *Telecommunications Policy*, 2016, 40(10-11): 956-970.
- [50] A Gambardella, M S Giarratana. General technological capabilities, product market fragmentation, and markets for technology. *Research Policy*, 2013, 42(2): 315-325.
- [51] W S Brünger, D Geiß, G Herlan, et al. Quality – Key factor for high value in professional patent, technical and scientific information. *World Patent Information*, 2011, 33(3): 230-234.
- [52] D Harhoff, S Wagner. The Duration of Patent Examination at the European Patent Office. *Management Science*, 2009, 55 (12): 1969-1984.
- [53] C A Cotropia, M A Lemley, B Sampat. Do applicant patent citations matter?. *Research Policy*, 2013, 42(4): 844-854.
- [54] D Chen, W C Lin, M Huang. Using Essential Patent Index and Essential Technological Strength to evaluate industrial technological innovation competitiveness. *Scientometrics*, 2007, 71(1): 101-116.
- [55] T Ramraj, R Prabhakar. Frequent Subgraph Mining Algorithms – A Survey. *Procedia Computer Science*, 2015, 47: 197-204.

Biographical notes

Wen-Guang Lin, born in 1985, is currently a lecturer and a master candidate supervisor at *Key Laboratory of Intelligent Manufacturing Equipment, Xiamen University of Technology, China*. He received PhD degree from *Xiamen university, China*, in 2014.

E-mail: linwg@xmut.edu.cn

Xiao-Dong Liu, born in 1995, is currently a master candidate at *Key Laboratory of Intelligent Manufacturing Equipment, Xiamen University of Technology, China*.

E-mail: Louis_LXD@163.com

Ren-Bin Xiao, born in 1965, is currently a professor and PhD candidate supervisor at *School of artificial intelligence and automation, Huazhong University of Science and Technology, China*. His main research interests include data driven product design, design method, and swarm intelligence.

E-mail: rbxiao@hust.edu.cn