

Mapping High Dimensional Sparse Customer Requirements into Product Configurations

Yao Jiao¹, Yu Yang^{1,*} and Hongshan Zhang¹

¹Department of Industrial Engineering, College of Mechanical Engineering, Chongqing University, Chongqing, China

^{1,*}yuyang@cqu.edu.cn

Abstract. Mapping customer requirements into product configurations is a crucial step for product design, while, customers express their needs ambiguously and locally due to the lack of domain knowledge. Thus the data mining process of customer requirements might result in fragmental information with high dimensional sparsity, leading the mapping procedure risk uncertainty and complexity. The Expert Judgment is widely applied against that background since there is no formal requirements for systematic or structural data. However, there are concerns on the repeatability and bias for Expert Judgment. In this study, an integrated method by adjusted Local Linear Embedding (LLE) and Naïve Bayes (NB) classifier is proposed to map high dimensional sparse customer requirements to product configurations. The integrated method adjusts classical LLE to preprocess high dimensional sparse dataset to satisfy the prerequisite of NB for classifying different customer requirements to corresponding product configurations. Compared with Expert Judgment, the adjusted LLE with NB performs much better in a real-world Tablet PC design case both in accuracy and robustness.

1. Introduction

Customers express their requirements on product attributes ambiguously and locally due to the lack of expertise [1]. The ambiguous requirements urge designers to develop precise configurators for mapping uncertain information to accurate product specifications. The local requirements form various differentiation on disparate product attributes corresponding to customers' finite and variant domain knowledge. Therefore, mining data of customer requirements from various mediums or direct expressions might lead to fragmental data with high dimensional sparsity, burden the process for mapping customer requirements to product configurations with uncertainty and complexity [2].

Popular methods for mapping customer needs to product configurations heavily rely on the explicit, systematic and structural requirements data [3], due to the lack of expertise, customers may unpleasant and stressful to raise their needs specifically and terminologically [1]. Meanwhile, collecting data for reducing sparsity is time consuming and costly, it is difficult to guarantee the data collected is accurate and consistent [4]. These obstacles make the uncertain, high dimensional sparse dataset of customer requirements arise from many real-world design issues.

In face of limited data availability and fuzzy information, the expert judgement is widely used in practice, as it has no such requirements for data compared with other popular methods for mapping procedures. However, expert judgement raises concern on its repeatability and bias [5]. Some researches emphasize the probabilistic classifiers of machine learning for they are suitable to quantify uncertain information, and robust in diversity [6-7]. Such probabilistic classifiers are inductive



learning techniques, the training datasets from which the classifiers learn the regulations should be under basic prerequisites. Therefore, to apply probabilistic classifiers in product design issues, preprocesses for high dimensional sparse dataset are necessary.

In this study, we apply Naïve Bayes (NB) classifier for mapping customer requirements to product configurations. NB makes use of requirement flexibility, and calculate probability of relevance for all product configurations. This merit makes NB capable and efficient in fuzzy reasoning. Another merit is it converges fast in learning stage, thus NB is suitable for small training dataset which is usual in design issues. However, NB is built on the prerequisite of conditional independent assumption. High dimensional dataset contains so many intertwined relationships and dependencies between attributes, which may cause NB inefficient and fallible, even results in the curse of dimensionality [8].

Therefore, before implementing NB for mapping reasoning, this study uses a dimensionality reduction method, local linear embedding (LLE), to preprocess the high dimensional requirement dataset. Considering the sparsity, some adjustments are applied based on basic LLE to improve its capability for processing sparsity. In addition, this study uses the data from a real-world design case to compare the NB integrated with adjusted LLE to expert judgement in the performance of mapping customer requirements to product configurations.

The rest of this paper is organized as follows. In the section of related works, we review methods for mapping customer needs to product configurations. In the next section, detailed methodology of this study is given out. A real-world Tablet PC design case and its data is shown in the section of data and case, then followed by the results and discussions. At last, we conclude the paper with contribution and future works.

2. Related work

Several approaches are available for mapping customer requirements into appropriate product configuration plans, e.g., configurator design, Kansei engineering, and affective computing [3]. In configurator design process, customers are required to specify product attributes with one product one by one for resulting the desire product [9]. Configurators can be divided into 3 categories, as rule based, model based and case based. Rule based configurators derive configurations in a forward manner [1]. It is firstly developed and applied in by Digital Equipment Corporation with the name called R1. R1 succeeds in application greatly, and accelerates other configurators based on rules, e.g., SICONFEX, MMC-Kon, BLADES, and MICON [3,9]. However, due to the lack of separation of domain knowledge and control system, such rule based systems often suffer from maintainability issues or side effects from rule changing, and fail to accomplish complex design works [6]. Model based configurators take configuration tasks as constraint satisfaction problems (CSP) [10]. Model based systems have advantages of better separation of domain knowledge used, and better robust for complex design tasks [7]. Extensive research of such model based configurators accompanied by the deep-going study of CSP, e.g., the Dynamic CSP contains activity constraints, the Generic CSP for handling constraints in intuitive way, the Composite CSP, and the Mixed and conditional CSP [1,3]. Case based configurators store previous analogous configuration tasks and domain knowledge, then the new configurations are retrieved from the storage [11]. The basic issue in case based systems is how to find similar solved problems and retrieve the best configuration adapting to the new requirements. In summary, challenges for configurators exist in several areas, e.g., knowledge compilation, conflict detection and diagnosis of divide & conquer algorithms, and interactive settings [12]. Besides, configurator design burden customers express their needs in specific parameter domain rather than customer requirement domain [1]. This makes configurator design difficult for mapping ambiguous requirements to specific detail design parameters.

Kansei engineering is introduced by Nagamachi as a translating technology of a customer's feeling of the product to the design parameters. Kansei in Japanese means the psychological feeling and image regarding of customers to the new product, thus Kansei engineering is considered as a methodology within Affective Engineering [13]. Kansei engineering starts with the choice of domain to describe the overall idea behind an assembly of products for further evaluation. The second step is the selection of

kansei words from spanning the Semantic Space which is the collection of words that describe the domain, e.g., emotions, moods, and impressions. Then, similar to the spanning of Semantic Space, the third step is to span the space of product properties. In the next, synthesis step, Semantic Space and the space of product properties are linked together, to form the database that quantifies the connections between those two spaces. Kansei engineering is very helpful for customers to select the most satisfied product to his/her kansei [14]. However, Semantic Space demands explicit expressions of customer feelings and psychological activities to form the kansei words.

Making customers involve in design process is commonly adopted for processing ambiguous customer requirements [15]. By making customers immerse into design process, designers can acquire customer needs directly from their knowledge, behaviors, and reflections to products, to avoid the false information from traditional market research methods of survey and interview, e.g., focus group, quality function deployment, conjoint analysis, etc. [16]. The lead user design and empathic design are two typical methods for inviting customers to design process. Lead user design encourage those customers, who push a product to its limits and experience needs prior to commons, to distill the needs that satisfy themselves significantly by customizing. Though lead user design is effective for open innovation, those lead users are difficult to find, and may suggest configuration plans for niche products or higher-end products, rather than mainstream products [17]. Empathic design elicits customers' unarticulated requirements by observing customers using products in everyday applications and usage contexts [18]. This foundation makes empathetic design resultful in identifying latent customer needs. However, empathic design depends heavily on designer's power of observation and innovation, and can hardly implement in complex products for the consideration of simulation [17].

Expert judgement is a widely practiced technique for product design in the uncertain and sparse dataset settings, as it has no formal requirement for systematic data which is quite common in design issues [4]. Due to the dependence upon expert knowledge, expertise and experience, repeatability and bias are concerns raised for expert judgement [19]. These concerns demand for the validation to the accuracy and informativeness of expert judgement applications [20]. Most cases of expert judgement reduce to cross validation, whereby experts learn experience from training set, and make judgements on the test set. In addition, taking at least 80% of the entire dataset for training is suggested [21].

3. Methodology

Expert Judgment is a commonly applied method in high dimensional sparse dataset settings, since there is no formal requirement for systematic and structural data. The Figure 1 shows the overview of mapping procedure by using Expert Judgment. In this study, we develop an integrated method by adjusted LLE and NB for mapping unarticulated high dimensional sparse customer requirements to product configurations, where the adjusted LLE preprocess the high dimensional sparse dataset to satisfy the prerequisite of NB for classifying. The overview of this integrated method is shown in Figure 2. Moreover, we compare the proposed method to Expert Judgment in a real-world design case. The next subsections introduce (a) the adjustments for classical LLE to reduce dimensionality with sparsity, (b) the classification process by using NB, and (c) the design for comparison with proposed method and Expert Judgment.

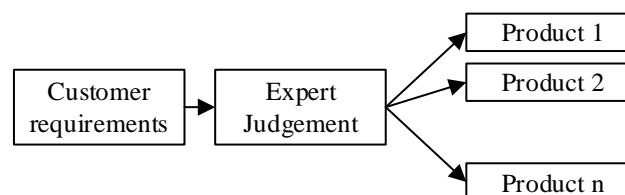


Figure 1. Mapping customer requirements based on Expert Judgement

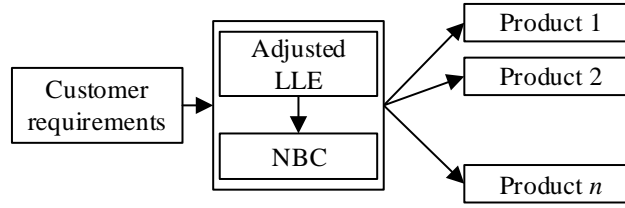


Figure 2. Mapping customer requirements based on adjusted LLE and NB

3.1. Adjusted LLE for sparse data dimensionality reduction

LLE is nonlinear dimensionality reduction method based on manifold learning, and introduced by [22]. For detailed interpretation of LLE, see [22]. LLE has many advantages over other manifold learning methods for dimensionality reduction, because it is able to learn global structure of nonlinear manifolds, which makes it very suitable for pragmatic application in real-world settings. However, it is limit when dealing with sparse data, due to it attempts to retain only local properties of the data [23].

In this study, we adjust LLE by expanding the initial local neighborhood to against the sparsity of customer requirements. The adjusted LLE is described as follows.

For a given high dimensional sparse dataset $X = [x_1, x_2, \dots, x_N]$, $x_i \in R^D$, D is the dimensionality, suppose its low dimensional representation dataset $Y = [y_1, y_2, \dots, y_N]$, $y_i \in R^d$, d is the dimensionality, where $d \ll D$. For each x_i , finds k nearest neighbors. For adjusted LLE, the expanded neighborhood of x_i is the combined neighborhood of nearest neighbors of the initial k nearest neighbors of x_i .

Let $N_k(x_i) = \{x_j\}$ is the set of k nearest neighbors of x_i , $i \neq j$, $j = 1, 2, \dots, k$, the expanded area is $N_k^E(x_i) = \{N_k(x_i), N_k(x_j)\}$. We apply the tangent space discriminant (TSD) to select and retain the sample datapoints. The Eigen-decomposition process of TSD requires at least $m+1$ local sample datapoints, where m is the intrinsic dimensionality of X , in addition, the number of expanded nearest datapoints $k^* = k^2 + k$. Select the m first principal eigenvectors corresponding to smallest k^* nonzero eigenvalues of every sample datapoint, where all the eigenvectors are subject to $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m \geq \dots \geq \lambda_{k^*}$, to form the columns of matrix H , and H_i denotes the tangent direction of x_i . Then, an expanded local sample datapoint can be retained if it satisfies Eq. (1):

$$\left\| (I - H_i H_i^T)(x_i - x_j^*) \right\|_2 < T^*(x_i), \quad (1)$$

where I is the identity matrix, $x_j^* \in \{N_k(x_j)\}$, $j = 1, 2, \dots, k$, $T^*(x_i) = \left(\frac{1}{k^*} \right)^{1/m} \|x_i - x_j^*\|$.

Then the expanded local neighborhood $N_k^{E*}(x_i)$ is formed by all the retained sample datapoints which satisfy Eq. (1). Calculate a sparse matrix of local predictive weights w_{ij} that best reconstruct every datapoint x_i from its k nearest neighbors as:

$$\min \varepsilon(W) = \sum_{i=1}^N \left\| \sum_{j=1}^k w_{ij} (x_i - x_j) \right\|^2 = \sum_{i=1}^N \left\| (x_i - x_j) w_i \right\|^2 = \sum_{i=1}^N (w_i)^T Z_i w_i, \text{ s.t. } \sum_{j=1}^k w_{ij} = 1, \quad (2)$$

where $Z_i = (x_i - x_j)(x_i - x_j)^T$. Based on Lagrangian multiplier methods (LMM),

$$\frac{\partial L}{\partial w_i} = 2 Z_i w_i + \lambda \cdot I_N = 0, \text{ we can derive } w_i = Z_i^{-1} \cdot I_k \left(I_k^T Z_i^{-1} I_k \right)^{-1}.$$

To retain the weight of $N_k^{E*}(x_i)$, the cost function should be minimized as:

$$\min \Phi \left(Y_{N_k^{E*}(x_i)} \right) = \sum_{p \in N_k^{E*}(x_i)} \left\| y_p - \sum_{j=1}^k w_{pj} y_j \right\|^2. \quad (3)$$

Therefore, Y can be computed as:

$$\min \Phi(Y) = \sum_{i=1}^N \sum_{p \in N_k^{E^*}(x_i)} \left\| y_p - \sum_{j=1}^k w_{pj} y_j \right\|^2 = \sum_{i=1}^N \sum_{p \in N_k^{E^*}(x_i)} \left\| Y_i (I_p - W_p) \right\|^2 = \sum_{i=1}^N \text{Tr}(Y_i M_i Y_i^T), \quad (4)$$

where I_p is the p th column of I , $M_i = (I - W_i)(I - W_i)^T$.

Suppose $S = [S_1, S_2, \dots, S_N]$, $Y_i = Y S_i$, $i = 1, 2, \dots, N$, where $S_{ij} = 1$ ($i = j$) or $S_{ij} = 0$, and $M = \text{diag}(M_1, M_2, \dots, M_N)$, $B = S M S^T$, thus Eq. (4) can be rewritten as:

$$\min \Phi(Y) = \sum_{i=1}^N \text{Tr}(Y_i M_i Y_i^T) = \sum_{i=1}^N \text{Tr}(Y B Y^T), \text{ s.t. } Y Y^T = I. \quad (5)$$

Based on LMM, $B Y^T = \lambda Y^T$, Y is the eigenvectors corresponding to the d smallest nonzero eigenvalues of B . Therefore, the representation dataset $Y = [y_1, y_2, \dots, y_N]$, $y_i \in R^d$ replaces X of original customer requirement to be the input of NB process.

3.2. NB for classifying process

In this NB process, dataset Y is the input for replacing initial customer requirements dataset X , to meet the prerequisite of NB. To map customer requirements to the optimal configuration plan, $P(\bar{U} | y_1, y_2, \dots, y_N)$ is needed, where \bar{U} denotes a configuration plan, in addition, then:

$$P(\bar{U} | y_1, y_2, \dots, y_N) = P(\bar{U}) P(y_1, y_2, \dots, y_N | \bar{U}) (P(y_1, y_2, \dots, y_N))^{-1} \propto P(\bar{U}) \cdot \prod_{k=1}^N P(y_k | \bar{U}). \quad (6)$$

Let u_i denotes the i th product configuration, [24] estimate the probabilities of $P(\bar{U})$ and $P(y_k | \bar{U})$ by applying maximum likelihood estimation in the learning stage for NB, as:

$$P(\bar{U} = u_i) = \frac{|u_i|}{(\sum_i |u_i|)^{-1}}, \text{ and } P(y_k | \bar{U} = u_i) = \frac{|y_k \cap u_i|}{(|u_i|)^{-1}}, \quad (7)$$

where $|u_i|$ denotes the number of product configuration u_i , and $|y_k \cap u_i|$ is the number of cases of the choosing of u_i . The learning stage can be applied by using existing customer requirements representation dataset Y and corresponding selected configuration plans.

The NB is trained to select the configuration plan with highest probability, as:

$$\bar{U}^* = \arg \max_{\bar{U}} P(\bar{U} | y_1, y_2, \dots, y_N) = \arg \max_{\bar{U}} P(\bar{U}) \cdot \prod_{k=1}^N P(y_k | \bar{U}). \quad (8)$$

3.3. Cross validation and performance measurements

Ten-fold cross validation is applied in this study to verify the effectiveness of the methods proposed over Expert Judgment. For a given high dimensional sparse customer requirements dataset, nine tenths of the data are randomly selected as the training set, and the remained one tenth act as the testing data. Such cross validation procedure is carried out for ten rounds. In each round, the training data is learned by experts, simultaneously, preprocessed by adjusted LLE and learned for NB, then the performances of Expert Judgment and adjusted LLE with NB are compared under two performance measurements for the testing data, they are: (a) The accuracy rate, and (b) the F-measure. In the following content, the measurements for accuracy and F-measure are given out.

Let π denotes the overall accuracy of all classes, m denote the number of class, TC_i denotes the number of classifications that correctly assigned to class i , FC_i denotes the number of classifications that not belong to class i , then, the overall accuracy rate is given by:

$$\pi = \left(\sum_{i=1}^m (TC_i + FC_i) \right)^{-1} \sum_{i=1}^m TC_i. \quad (9)$$

However, accuracy rate omits the false negative rate when measures the performance [1]. Thus, the F-measure is applied to overcome this issue. F-measure is originally defined for binary classification case. This study adopts macro-average F-measure proposed by [25] for multiple classification cases. F-measure ranges from 0 to 1, with higher value indicates better classification quality.

Let ρ_i denotes the recall rate of class i , FI_i denotes the number of individuals which are not assigned to class i while they actually belong to. Then, the recall rate is given by:

$$\rho_i = TC_i \cdot (TC_i + FI_i)^{-1} \quad (10)$$

Let π_i denotes the accuracy for class i , The macro-average F-measure F_{macro} is calculated as:

$$F_{macro} = m^{-1} \sum_{i=1}^m \left(\frac{2\rho_i\pi_i}{\rho_i + \pi_i} \right) \quad (11)$$

4. Case and data

Data from a real-world customized tablet PC design project is used to testify the effectiveness of the method proposed in this study. The design project is supported by a consumer electronics company in south China. There are 126 attributes to be specified by customers which corresponding to the requirements. Number of options alternatives for attributes ranges from 3 to 24, thus the customer requirements construct a 126×24 matrix. Part of the attributes and their options are listed in Table 1.

Table 1. Part of the attributes and corresponding options alternatives.

Attribute	Attribute Options			
Operation system	Android	Windows	Android +windows	
Screen size	7.9 inch	8.4 inch	9.7 inch	10.1 inch
CPU cores	2 cores	4 cores	6 cores	8 cores
CPU brand	Intel	Nvidia	MTK	Qualcomm
Color	Black	Gray	Yellow/gold	White
Storage volume	16G	32G	64G	128G
Aspect ratio	3:2	4:3	16:9	16:10
HDMI interface	None	Micro HDMI	Standard HDMI	

1030 customers were interviewed in a survey in two stages. At the first stage, customers were required to express their requirements to the tablet PC, and specify their expression under the assistance of designers. After that, they were shown a set of 24 different tablet PCs, and were required to make selection based on their requirements and preferences. Thus, there are 24 classes for classifying in this case. The customer requirements dataset and corresponding final selection will be used as the training and testing data for the ten-fold cross validation and performance measurements.

Expert Judgments for each round is decided by an expert team with eight experts. The expert team contains four senior designers in electronic products, two professors mainly research on consumer products design, and two executive editors from magazines focus on electronics products.

Sparsity is defined as the proportion of non-zero elements in a matrix [26]. The useful elements take account for 27.15% of the requirements matrix, but considering zero elements in the useful portion as customers do not take all the provided options, the sparsity is as high as 0.8456.

5. Results and discussion

Before comparing the performance of adjusted LLE with NB to expert judgement, this study has conducted a comparison of the number of extracted features (Num. of ET) and the computing time for classic LLE and adjusted LLE under the case situation. The results are shown in the Table 2.

Table 2. Comparison of LLE and adjusted LLE for extracting features.

Algorithm	Num. of ET	Computing time
LLE	32	2.742
Adjusted LLE	6	1.411

The results in Table 2 indicate that due to the expansion of initial nearest neighbors, the adjusted LLE can induct rules at the very beginning for reduction. While, the classic LLE costed more computational time because it fails to learn inductions in the first trials, the re-trials for finding enough initial datapoints to construct inductive rules are time consuming. Also, the number of features extracted by LLE is more than adjusted LLE, which implies the poor performance of LLE for reducing high dimensional spare data over the adjusted LLE proposed in this study.

Table 3. The accuracy rate and F-measure for two methods in every round.

Round	Adjusted LLE with NB		Expert Judgment	
	Accuracy rate (%)	F-measure (%)	Accuracy rate (%)	F-measure (%)
1	81.55	84.17	49.51	55.28
2	79.61	82.50	52.43	57.08
3	82.52	85.00	45.63	52.92
4	83.50	85.83	53.40	51.39
5	81.55	84.17	47.57	53.75
6	79.61	83.33	50.49	56.94
7	80.58	83.33	52.43	57.78
8	81.55	85.00	41.75	48.47
9	82.52	85.83	43.69	50.69
10	82.52	84.17	48.54	55.42

Table 4. Statistical analysis for accuracy and F-measure.

Measurement	Mean	Std. deviation	Kurtosis
Adjusted LLE with NB			
Accuracy rate	0.816	-0.748	-0.748
F-measure	0.843	-0.751	-0.751
Expert Judgment			
Accuracy rate	0.485	-0.805	-0.805
F-measure	0.540	-0.806	-0.806

Table 3 shows the results of accuracy and F-measure for each round of the adjusted LLE with NB and Expert Judgement based on the case data. Table 4 shows statistical analysis based on the results data shown in Table 3. All the results indicate that adjusted LLE with NB proposed in this study performance better than Expert Judgment in mapping high dimensional sparse customer requirements to product configurations in our research.

In each round, adjusted LLE with NB outputs higher accurate result (see ‘Accuracy rate’ in Table 3) and higher classification quality (see ‘F-measure’ in Table 3). Statistical analysis for the results indicates that adjusted LLE with NB are more robust than Expert Judgment. This verdict can be revealed from two aspects, (a) results of accuracy and F-measure of adjusted

LLE and NB are less dispersive (see ‘Std. deviation’ in Table 4), (b) Expert Judgment is more possible to generate outliers in classification process (see ‘Kurtosis’ in Table 4).

Expert Judgment is easy to implement and is widely used, but this study raises three major issues for implementing Expert Judgment in high dimensional sparse customer requirements settings of product design as follows:

- Experts selection. Experience of expert impact on the performance of Expert Judgment greatly. Thus, the first obstacle for Expert Judgment is the selection for composing members of an expert team, to reduce impact of differentiation in expertise. This study selects eight experts with ample expertise, however, the results of Expert Judgment are more dispersive than machine learning method.
- Low accuracy. The average accuracy rate of Expert Judgment is below 0.5 for the high dimensional sparse customer requirements in this study. It challenges the reliability of expert judgment for processing unsystematic and nonstructural dataset with sparsity. The low accuracy of Expert Judgment in this study reminds the necessity of analysis of suitability for its application in real-world design issues.
- High cost. The expert team costs at least three days for each round in learning from training dataset and determining classes for testing dataset, and the Expert Judgment in this study lasts more than one month. Though Expert Judgment is easy to implement, in front of high dimensional sparse customer requirements, it is costly, time consuming, and disadvantageous to make advantages for product design.

To the contrary, adjusted LLE with NB overcomes the major issues of Expert Judgment, and has several advantages. Firstly, the outputs of adjusted LLE with NB are of high accuracy and robust, make it reliable to process high dimensional sparse customer requirements. Secondly, adjusted LLE and NB is not difficult to implement pragmatically. Moreover, experts are not necessary for computing and reasoning in the procedures of adjusted LLE and NB, which saves time and cost due to human factors.

6. Conclusion

In this study, we propose an integrated method by adjusted LLE and NB for mapping high dimensional sparse customer requirements to product design configurations. And we compare this method with widely used expert judgement in coping with a real-world Tablet PC design case with unarticulated high dimensional sparse customer requirements. The results indicate this method outperforms expert judgement greatly both in accuracy and robustness. The contribution of the paper is summarized as follows:

- We apply NB to map customer requirements to product configurations. Compare with other methods, such as configurator design, Kansei engineering, etc., NB is a probabilistic classifier that more suitable to quantify and process the ambiguous and unarticulated nature of customer requirements. Thus customers are not burdened to express their needs specifically.
- The classical LLE is efficient in dimensionality reduction. In this study, we adjust the classical LLE by expanding initial local neighborhood to guarantee its effectiveness for dealing with sparse dataset, and make raw dataset satisfy the prerequisites of NB. The integration of adjusted LLE and NB performs better in coping with real-world design case than the widely used Expert Judgment under high dimensional sparse customer requirements settings, which is common in design issues with big data.
- The proposed method, adjusted LLE with NB, performs robustly, and no need for expertise or experience from experts to conduct computing and reasoning procedures, as LLE and NB are methods based on machine learning that learn rules by induction. These characteristics make the method reliable in pragmatic application, as well as time and cost saving.

In the future, we will extend this study from several aspects. One of the extensions is analyzing the relationships between sparsity of dataset and accuracy of adjusted LLE and NB under different application settings. Another extension is integrating this method with data

mining techniques for two purposes, one is digging customer latent and affective needs to generate novel product configurations, the other one is improving design efficiency by using big data to achieve market advantages.

Acknowledgements

This study was supported by the National Natural Science Foundation of China under Grant No. 71571023. The authors are grateful to appreciate the reviewers' comments and effort in order to improve the paper.

References

- [1] Wang Y and Tseng M M 2015 *J. Intell. Manuf.* 26 501-9
- [2] Kusiak A 2006 *Int. J. Pro. Res.* 44 4175-91
- [3] Zhou F Ji Y and Jiao R J 2012 *J. Intell. Manuf.* 24 1047-69
- [4] Shepperd M and Cartwright M 2011 *IEEE T. Software Eng.* 27 987-98
- [5] Stumptner M 1997 *AI Commun.* 10 111-25
- [6] Wang Y and Tseng M M 2011 *AI EDAM* 25 185-95
- [7] Wang Y and Tseng M M 2011 *CIRP Ann-M.* 60 175-8
- [8] Keogh E and Mueen A 2011 *Curse of Dimensionality* ed C Sammut and G I Webb (New York: Springer) pp 257-8
- [9] Xie H Henderson P and Kernahan M 2005 *Int. J. Pro. Res.* 43 4455-69
- [10] Felfernig A Hotz L Bagley C and Tiihonen J 2014 *Knowledge-based Configuration: From Research to Business Cases* (Waltham, MA: MK Publications)
- [11] Haug A Hvam L and Mortensen N H 2012 *Comput. Ind.* 63 471-81
- [12] Felfernig A Schubert M and Zehentner C *AI EDAM* 26 53-62
- [13] Nagamachi M and Lokman A M 2011 *Kansei/Affective Engineering* (Boca Raton, FL: CRC Press)
- [14] Nagamachi M 2002 *Appl. Ergon.* 33 289-94
- [15] Tseng M M Jiao R J and Wang C 2010 *CIRP Ann-M.* 59 175-8
- [16] Mattelmäki T and Battarbee K 2002 *Proc. 7th Biennial Participatory Design Conf.* June 23-5 Malmö, Sweden
- [17] Lin J and Seepersad C C 2007 *Proc. 19th Int. Conf. Design Theory and Methodology* September 4-7 Las Vegas
- [18] Leonard D and Rayport J F 1997 *Harvard Bus. Rev.* 75 102-13
- [19] Hughes R T 1996 *Inform. Software Tech.* 38 67-75
- [20] Aspinall W P and Cooke R M 2013 *Risk and Uncertainty Assessment for Natural Hazards* ed J Rougier S Sparks and L J Hill (Cambridge: Cambridge University Press) pp 64-99
- [21] Eggstaff J W Mazzuchi T A and Sarkani S 2014 *Reliab. Eng.* 121 72-82
- [22] Roweis S T and Saul L K 2000 *Science* 290 2323-6
- [23] Wan X Wang D Tse P W Xu G and Zhang Q 2016 *Measurement* 78 138-50
- [24] Russell S J and Norvig P 2009 *Artificial intelligence: A modern approach* (Upper Saddle River, NJ: Prentice Hall)
- [25] Özgür A Özgür L Güngör L 2005 *Text Categorization with Class-based and Corpus-based keyword selection* ed P Yolum T Güngör F Gürgeç and C Özturan (Berlin: Springer-Verlag) pp 606-15
- [26] Daubechies I, Defrise M and De Mol C 2004 *Com. PA Math.* 57 1413-57