Trend Mining for Predictive Product Design[1]

Conrad S. Tucker
Assistant Professor
Mem. ASME
213-N Hammond Building,
Engineering Design and Industrial Engineering,
The Pennsylvania State University,
University Park, PA 16802
e-mail: ctucker4@psu.edu

Harrison M. Kim[2]
Assistant Professor
Mem. ASME
104 S. Mathews Avenue,
Industrial and Enterprise Systems Engineering,
University of Illinois,
Urbana-Champaign Urbana,
IL 61801
e-mail: hmkim@illinois.edu

The Preference Trend Mining (PTM) algorithm that is proposed in this work aims to address some fundamental challenges of current demand modeling techniques being employed in the product design community. The first contribution is a multistage predictive modeling approach that captures changes in consumer preferences (as they relate to product design) over time, hereby enabling design engineers to anticipate next generation product features before they become mainstream/unimportant. Because consumer preferences may exhibit monotonically increasing or decreasing, seasonal, or unobservable trends, we proposed employing a statistical trend detection technique to help detect time series attribute patterns. A time series exponential smoothing technique is then used to forecast future attribute trend patterns and generates a demand model that reflects emerging product preferences over time. The second contribution of this work is a novel classification scheme for attributes that have low predictive power and hence may be omitted from a predictive model. We propose classifying such attributes as either standard, nonstandard*, or* obsolete *by assigning the appropriate classification based on the time series entropy values that an attribute exhibits. By modeling attribute irrelevance, design engineers can determine when to retire certain product features (deemed obsolete) or incorporate others into the actual product architecture (standard) while developing modules for those attributes exhibiting inconsistent patterns throughout time (nonstandard). Several time series data sets using publicly available data are used to validate the proposed preference trend mining model and compared it to traditional demand modeling techniques for predictive accuracy and ease of model generation.*
[DOI: 10.1115/1.4004987]

Keywords: trend mining, predictive product design, knowledge discovery, data mining

1 Introduction

Identifying and understanding changes in complex systems are vital to developing efficient models that help to predict future behavior. As data storage capabilities become more efficient and affordable, so do the challenges of extracting meaningful knowledge that may exist within these storage resources. Dynamic systems such as consumer electronics markets, cybersecurity systems, and military network systems, all require reliable and efficient analysis tools for sound decision making objectives.

The ability to model emerging trends has broad applicability in product development, ranging from researching and developing new product technologies to quantifying changes in consumer preferences in highly volatile markets. Traditional demand modeling techniques frequently employed in the product design community typically generate predictive models using data from a single snapshot in time (usually the most currently available data set) and hence may not reflect the evolving nature of product trends. The absence of a temporal demand model for product design presents a challenge to design engineers trying to determine the relevant product attributes to include/exclude in the next generation of products.

To overcome these challenges, we propose a time series model that addresses specific product design problems relating to product preference trend modeling. We introduce a subcategory of data change mining called *Preference Trend Mining* (PTM) that characterizes attribute relevance over time. Once an attribute has been deemed irrelevant, we propose three classification groups based on its historical pattern; *Obsolete attribute*, *Nonstandard attribute*, and *Standard attribute*. This novel classification helps to guide the product architecture by indicating when certain product features should be included or excluded in next generation product designs. A cell phone example is used to demonstrate what each classification option means to design engineers and to the overall success of new product development efforts.

This paper is organized as follows. This section provides a brief motivation and background; Sec. 2 describes previous works closely related to the current research; Sec. 3 describes the methodology; A cell phone case study is presented in Sec. 4 with the results and discussion presented in Sec. 5; Sec. 6 concludes the paper.

2 Related Work

2.1 Demand Modeling Techniques in Product Design. There are several well established demand modeling/customer preference acquisition techniques that have been employed in the product design community such as conjoint analysis, quality function development, discrete choice analysis, supervised machine learning models, to name but a few [1–4]. In this selective literature review, we will limit our discussion to the discrete choice analysis model and the decision tree classification model, in part due to their popularity in the product design community and also due to the research findings in a recent comparative study performed in the product design community [5].

2.1.1 Discrete Choice Analysis. The discrete choice analysis (DCA) approach has been employed extensively in the product design community as an attribute quantification and demand

[1]This work is supported by the National Science Foundation under Awards 0726934 and the Sandia National Labs. Any opinions, findings, and conclusions or recommendations expressed in this work are those of the authors and do not necessarily reflect the views of the National Science Foundation and the Sandia National Labs.
[2]Corresponding author.
Contributed by the Design Theory and Methodology Committee for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received February 10, 2011; final manuscript received August 10, 2011; published online November 11, 2011. Assoc. Editor: Jonathan Cagan.

modeling technique [6–8]. The model measures variations in consumer preferences by employing a random utility function U_{ni} that is comprised of a deterministic part W_{ni} and an unobservable random part ε_{ni}. Although there are many variations of the DCA model, a popular technique employed in the product design community is the multinomial logit (MNL) model. The MNL model assumes that the error terms (ε_{ni}) are independent and identically distributed (i.i.d) and follows a *Gumbel* distribution [9]. Given a set of choice alternatives $i = 1,\ldots,m$, the probability that a customer n would choose alternative i is represented as

$$P_n(i \in C_m) = \frac{e^{W_{ni}/u}}{\sum_{j=1}^{m} e^{W_{nj}/u}} \tag{1}$$

Here $P_n(\mathrm{i} \in C_m)$ is the probability that customer n would choose alternative i within the choice set C_m, $W_{ni} = f(\mathbf{A}_i, \delta_i, \mathbf{S}_n\text{: } \boldsymbol{\beta}_n)$ represents the deterministic part of the utility function U_{ni}, $\mathbf{A}_i$ represents the quantifiable attribute set for choice alternative i, δ_i represents the price for a given product (choice alternative i), $\mathbf{S}_n$ is the sociodemographic attributes of customer n, $\boldsymbol{\beta}_n$ is the unknown coefficients representing a consumer's taste preference, and u is the scaling parameter set to 1, assuming all choice alternatives are equally considered by customer n.

While several variations of the DCA model (e.g., multinomial probit, nested logit, mixed logit, etc.) have been employed in the product design community, they are primarily distinguished from each other by the degree of sophistication with which the unobserved error and heterogeneity in customer preferences are modeled [10–12].

2.1.2 Data Mining Decision Tree Classification. Techniques, such as the C4.5 algorithm, have been employed in the product design domain to solve product concept generation problems involving large scale consumer data [3,5]. This machine learning algorithm gets its foundation from Shannon's classical *Information Entropy* [13]. For the rest of the paper, we will refer to information entropy simply as *Entropy*. An example of entropy in product design terms could represent the uncertainty that exists in distinguishing one choice alternative from another in a choice set within a data set T. The entropy of the set of k choice alternatives can therefore be mathematically represented as [14]

$$\text{Entropy}(T) = -\sum_{i=1}^{k} p(c_i) \cdot log_2 p(c_i)[\text{bits}] \tag{2}$$

Here, $p(c_i)$ represents the probability (relative frequency) of a class variable c_i in the data set T and k represents the number of mutually exclusive class values within the data set (discrete case).

To determine the attribute (test attribute $\mathbf{X}$) with the greatest ability to reduce the uncertainty of the choice set, each attribute is partitioned into all of its n mutually exclusive outcomes (discrete case). The entropy, given a specific attribute test, is the summation of entropies for each unique value of that attribute [14]

$$\text{Entropy}_x(T) = \sum_{j=1}^{n} \frac{|T_j|}{|T|} \cdot \text{Entropy}(T_j) \tag{3}$$

Here, T_j represents a subset of the training data T that contains one of the mutually exclusive outcomes of an attribute. For example, if the attribute *energy consumption* has three mutually exclusive outcomes (e.g., *low*, *medium*, and *high*), then the training set T, would be partitioned into three data subsets (T_1 would contain all data instances where attribute *energy consumption* is *low* and so on). n represents the number of mutually exclusive outcomes for a given attribute.

The C4.5 decision tree classification algorithm defines the *gain* metric which in essence, is the amount of *uncertainty reduction* that an attribute provides in relation to the class variable. That is, the lower the $\text{Entropy}_x(T)$ for a particular attribute test, the higher the overall $\text{gain}(X)$ metric

$$\text{gain}(X) = \text{Entropy}(T) - \text{Entropy}_x(T) \tag{4}$$

The gain metric was later updated in the C4.5 decision tree algorithm to reduce the bias toward attributes that may contain a greater number of mutually exclusive outcomes and was redefined as [14]

$$\text{Gain Ratio}(X) = \frac{\text{gain}(X)}{-\sum_{j=1}^{n} \frac{|T_j|}{|T|} \cdot \log_2 \frac{|T_j|}{|T|}} \tag{5}$$

One of the assumptions of this model is that the data set can fit into main memory as all data instances are required at least for the first iteration. The definitions of entropy and entropy reduction (gain) are important concepts that serve as the foundation for the attribute irrelevance characterization presented later in this work.

2.1.3 Limitations of Current Demand Modeling Techniques. A recent comparative study in the product design community between the discrete choice analysis and decision tree (DT) classification models reveals that both techniques are quite comparable in terms of model generation and predictive accuracy. However, the decision tree classification model was found to be better suited for large scale data analysis due to multicollinearity issues reported while employing DCA for high dimensional data [5]. The DT model was capable of narrowing down the attribute space to the relevant attributes influencing product choice share. To mitigate the multicollinearity issues of the DCA model, the DT model could serve as a preprocessor, identifying the relevant attributes for the DCA model [5]. Nevertheless, both demand modeling techniques are limited in their ability to characterize evolving product preference trends in the market space due to the static nature of the models. Because the input of each model typically represents an instant in time, design engineers are faced with the challenge of anticipating shifts in product preferences based on personal experience, rather than quantitative customer feedback.

2.2 Time Series Modeling Techniques. In an effort to overcome some of the challenges of static demand models, research into time series modeling techniques have emerged, both in traditional utility theory based research and data mining and machine learning research.

2.2.1 Time Series Utility Function Models. There have been several time series, utility based models proposed in the literature aimed at quantifying the evolution of customer preferences. Mela et al. investigate the *short term, medium term*, and *long term* effects of marketing actions on consumer choice behavior [15]. Mela et al. use first derivative information of the choice share in the multinomial logit model to quantify the time sensitive nature of customer preferences. Jedidi et al. propose a heteroscedastic, varying-parameter joint probit choice and regression quantity model that investigates the tradeoff between promotion and advertising in the marketing domain [16]. Seetharaman proposes a utility-theoretical brand choice model that accounts for four different sources of state dependence, incorporating lagged effects of both consumer choices and marketing variables [17]. Lachaab et al. build upon the temporal discrete choice research by proposing a Bayesian state space framework that incorporates parameter-driven preference dynamics in choice models [18].

While the aforementioned discrete choice analysis models attempt to model evolving consumer preferences, the models are primarily focused on variations in model parameters, rather than the underlying evolution of attribute-class relationships (i.e., how the evolution of a specific attribute influences the dependent/class variable). Furthermore, these time series discrete choice models do not provide engineers with quantifiable measures of attribute relevance/irrelevance to next generation product designs. Since the proposed time series utility based techniques are developed in the marketing domain, they are focused more on the economic impact of customer preferences (evolution of brand preferences, advertising implications, etc.). Consequently, engineers are left with the challenge of determining the optimal attribute combinations for evolving customer preferences without any direct relation to product architecture design.

PTM algorithm that is proposed in this work differs from time series utility based choice models by having the ability to anticipate emerging attribute behavior whether the attribute exhibits a monotonically increasing or decreasing trend, cyclical trend or no trend at all. In addition to this, the PTM algorithm includes a technique to characterize attribute *irrelevance* by classifying attributes based on their time series predictive power. This enables the PTM model helps to guide the product design process by indicating when certain product features should be included or excluded in next generation product designs.

2.2.2 Time Series Data Mining Models. The area of data mining dealing with dynamic information processing is relatively new and has great potential to address many challenging areas of research. *Change Mining* is the umbrella term used to describe research involving data evolution in dynamic data bases [19]. *Data Stream Mining* is a subcategory of change mining that deals more with the continuous flow of data that needs to be analyzed with limited memory complications.

There have been several data mining algorithms proposed to address continuously changing data streams. For example, the very fast decision tree (VFDT) learner employs the Hoeffding statistic to build a decision tree classifier that has similar predictive characteristics as a conventional decision tree learner (for example, the C4.5 or gini based decision tree learners) but with a fraction of the memory requirements [20]. Another example is the concept-adapting very fast decision tree which extends the capabilities of the VFDT by enabling it to accommodate time-sensitive streaming data that may tend to exhibit *concept drift*, a phenomenon in dynamic information processing where the target variable shifts over time and causes the data mining model to diminish in its predictive accuracy [21]. While these models have the ability to handle incoming data streams, they are more focused on generating/adapting a model based on incoming data, rather than understanding how the data patterns evolve altogether.

Research domains more interested in data trends, rather than the speed of the data streams also present another interesting area of study. For example, the *RePro* classifier is a data streaming algorithm that applies both proactive and reactive predictions during model generation [22]. The algorithm attempts to alleviate the problems of *concept drift* by anticipating concept changes and making predictions that if incorrect, causes the model to readjust and revert back to a previous model. Another example is the *PreDet* algorithm that fits a polynomial regression model to the monotonically increasing or decreasing time series attribute relevance statistics. The resulting time series model anticipates future attribute patterns that are inherent in the evolving data [19].

Although the aforementioned change mining algorithms generate models using time series data, they suffer from a limitation similar to the DCA models described above. That is, their inability to quantify the irrelevant attributes in the resulting model. Furthermore, the change mining algorithms fail to model seasonality which can have dramatic effects on the model predictive accuracy. The PTM algorithm that we propose in this work differs from the PreDet and other change mining algorithms by having the ability to anticipate emerging attribute behavior whether the attribute exhibits a monotonically increasing or decreasing trend, cyclical trend or no trend at all. In addition to this, the aforementioned change mining algorithms do not suggest approaches to characterize attributes that may exhibit weaker predictive power over time. We propose an approach to handle the notion of attribute *irrelevance* by classifying attributes based on their time series predictive power. This enables the PTM model to quantify attributes that may be experiencing changes in the distribution of the attribute values themselves or novel/emerging attributes. The goal of the proposed PTM algorithm is to enable design engineers to understand changing customer preferences and anticipate emerging product designs trends in a timely and efficient manner.

3 Methodology

Figure 1 presents the overall flow of the preference trend mining algorithm, starting with the acquisition of *n* time-stamped data sets. For each time stamped data set (*t*) and subsequent data subset (*j*), the interestingness measure (IM) is calculated for each attribute (*i*) until the final attribute (*k*). There have been many proposed measures for evaluating attribute *interestingness* (relevance) such as the *information gain* metric, *gini* index, *Cosine* measure, *support* measure, *confidence* measure, to name but a few [23,24]. In this work, we will limit our definition of attribute *interestingness* to an attribute's ability to reduce the nonhomogeneity of the class variable. In Sec. 3.2, we will highlight the inconsistencies that exist among different definitions of *relevance* and propose an approach to mitigate these inconsistencies by evaluating attribute interestingness through time. That is, an attribute that is truly relevant, will have consistently high relevance scores throughout time and vice versa.

For each time step in Fig. 1, we calculate the IM for each attribute and then employ a seasonal time series predictive model to forecast the trend patterns (monotonically increasing, decreasing or seasonal trend patterns) for each attribute. The attribute with the highest predicted IM is selected as the split attribute for the future (unseen) time period and all time stamped data sets are partitioned based on the unique values of this attribute. The process continues until a homogenous class value exists in the model. The flow diagram in Fig. 1 ends with the classification of attributes (as either obsolete, standard, or nonstandard) that are omitted from the resulting model.

Sections 3.1–3.3.2 of the paper will expound on the steps of the flow diagram in Fig. 1.

3.1 Discovering Emerging Trends for Product Design. Trends within a data set can be characterized as monotonically increasing or decreasing, seasonal (where data exhibit some type of cyclical behavior) or both. There may also be

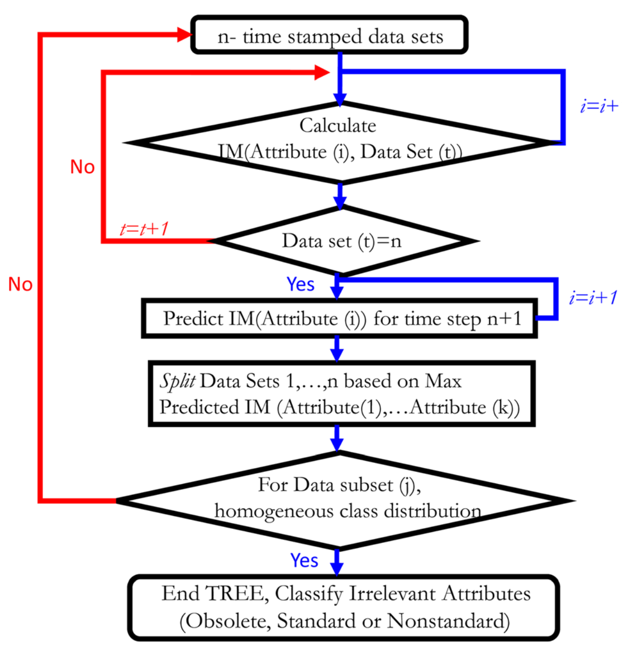


Fig. 1 Overall flow of preference trend mining methodology

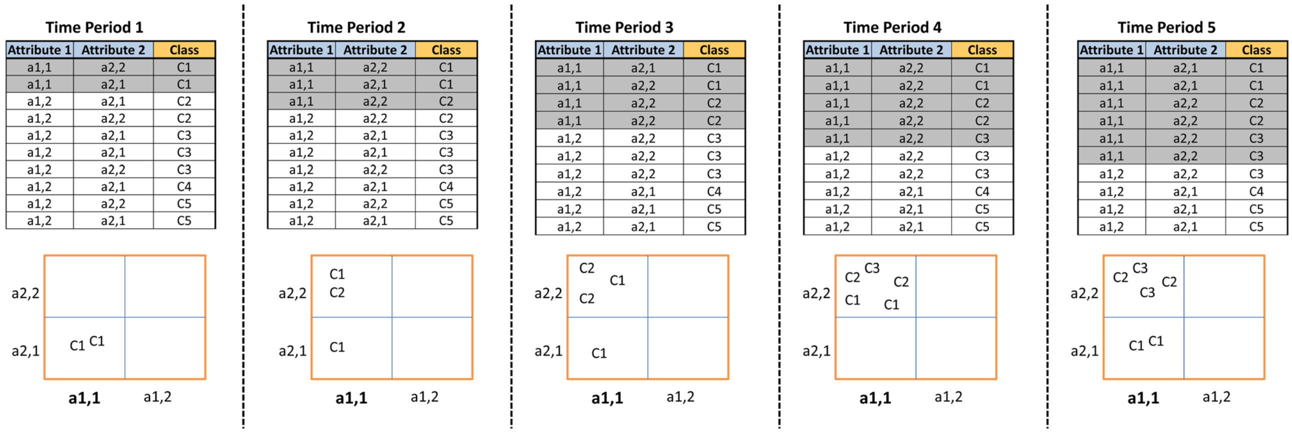


Fig. 2 Attribute-class distributions over time (attribute a1,1 is highlighted although both attribute patterns change over time)

instances where the time series data set does not exhibit a discernable pattern suitable for statistical modeling. In the context of product design, we will consider each of these different preference trend scenarios in our methodology. The time series data set represented in Fig. 2 will be used to illustrate the notion of attribute trends within a raw data set. Figure 2 comprises of 5 time periods. Attribute 1 comprises of two unique values $\{a_{1,1}, a_{1,2}\}$ and similarly for attribute 2 $\{a_{2,1}, a_{2,2}\}$. The last column in Fig. 2 represents the class (dependent) variable which has five mutually exclusive outcomes $\{c_1, c_2, c_3, c_4, c_5\}$. As we observe from time period t_1 to t_5, the number of instances of attribute 1's value $a_{1,1}$ increases from 2 at time period t_1 to 6 at time period t_5. Looking closer at the square graphs in Fig. 2, we observe that at time period t_1, although attribute 1's $a_{1,1}$ value only has a total count of 2, it represents a homogenous distribution of class value c_1 (lower left quadrant in time period t_1). Moving through time to time step t_5, we observe that the same attribute value $a_{1,1}$ has a count of 6 but with a nonhomogeneous distribution of the class variable (the lower left quadrant in time series t_5 has a mixture of c_1, c_2, and c_3). The change in the predictive power of each attribute can be quantified by calculating the attribute IM over time which in this case is the *gain ratio*. Figure 3 presents a visual representation of each attribute's gain ratio over time. In Fig. 3, although attribute 1 starts out with a higher gain ratio (predictive power) than attribute 2, by time period 4, attribute 2 has over taken attribute 1 in *relevance* to the class variable. If we had generated a predictive model at time period 3, we would not have realized the emerging preference trend of attribute 2. To overcome these challenges, we employ the Holt-Winters exponential smoothing model that uses a weighted averaging technique, taking into account the local level, the trend, and the seasonal components of the time series data [25,26].

3.1.1 Holt-Winters Exponential Smoothing Model. Holt-Winters is a nonparametric, exponential smoothing model that can be used to forecast each attribute's predictive power for the kth step ahead so that emerging preference trends can be anticipated in the market space. Nonparametric statistical tests may be preferred in machine learning scenarios due to the relaxation of the normality assumption that many parametric statistical trend tests require [27]. Since we assume no prior knowledge of the distribution of the incoming data, a relaxation of the data normality constraint is preferred. The (k) step ahead forecasting model is defined as

$$\hat{y}_t(k) = L_t + kT_t + I_{t-s+k} \tag{6}$$

where

Level L_t (the level component)

$$L_t = \alpha(y_t - I_{t-s}) + (1 - \alpha)(L_{t-1} + T_{t-1}) \tag{7}$$

Trend T_t (the slope component)

$$T_t = \gamma(L_t - L_{t-1}) + (1 - \gamma)T_{t-1} \tag{8}$$

Season I_t (the seasonal component)

$$I_t = \delta(y_t - L_t) + (1 - \delta)I_{t-s} \tag{9}$$

Here, y_t represents the data point at the most recent time period (t), $\hat{y}_t(k)$ represents the kth time step ahead forecasted value beyond y_t (i.e., $\hat{y}_t(k) = y_{t+k}$), s represents the frequency of the seasonality (monthly, quarterly, yearly, etc.)

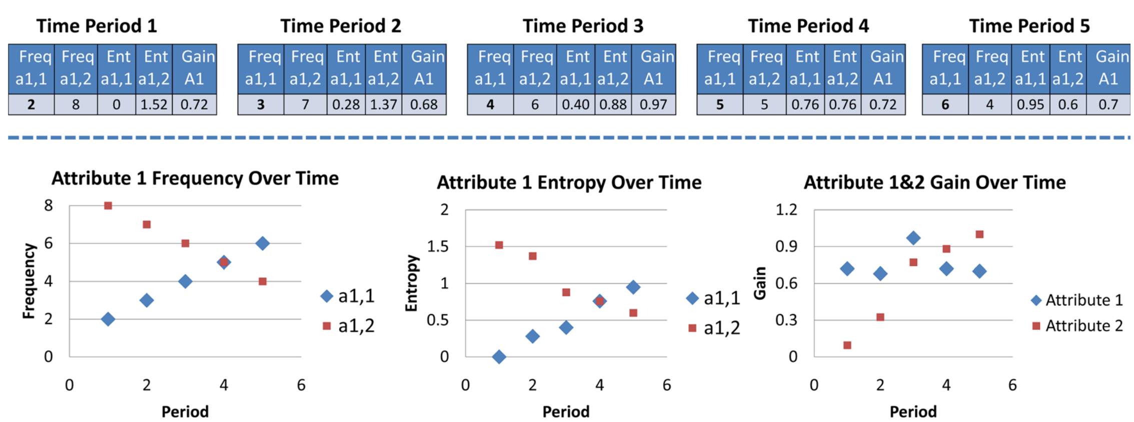


Fig. 3 Characterizing attribute preference trend over time

The smoothing parameters α, γ, and δ are in the range $\{0,1\}$ and are estimated by minimizing the sum of squared errors for one time step ahead [25,26].

Several well established statistical techniques (both parametric and nonparametric) exist for modeling time series data including the seasonal-trend decomposition procedure based on loess regression, variations of the Box-Jenkins models which include the autoregressive moving average and autoregressive integrated moving average, to name but a few [28,29]. Research studies on the predictive accuracies of these models reveal no conclusive evidence to suggest one model being superior for all data structures [29].

Based on the results in Fig. 3, we can observe that attribute 2 would be selected as the relevant attribute in time period 6 (since at each iteration, we always select the attribute with the highest gain ratio). Under the gain ratio definition of attribute relevance, attribute 1 would now be considered *irrelevant* at iteration 1 of the decision tree induction algorithm. Based on the irrelevance characterizations presented in Sec. 3.2, attribute 1 could either be an *obsolete attribute*, a *nonstandard attribute*, or a *standard attribute*. In order to determine the assignment of attribute 1, the temporal behavior of each mutually exclusive value of attribute 1 ($a_{1,1}$ and $a_{1,2}$) needs to be determined. Section 3.2 details the proposed attribute quantification methodology.

3.2 Quantifying Attribute Relevance. One of the major challenges in predictive model generation is understanding the design implications of the resulting model in terms of attribute *relevance* or *irrelevance*. To understand some of the challenges that arise in demand models, the following example is presented.

Let us define a set of attributes $\{A_1,\ldots, A_5\}$ each with a set of mutually exclusive outcomes $a_{i,j}$, where i corresponds to the specific attribute A_i, and j corresponds to the attribute value. For simplicity, let us assume that $j=2$ for all attributes. We also define a *class* variable that is conditionally dependent on one or several of the defined attributes. The class variable is also binary with values $\{c_1, c_2\}$.

Figure 4 is a visual representation of a resulting data mining decision tree structure employing the gain ratio metric described in Sec. 2.1.2. The following decision rules can be obtained by traversing down each unique path of the tree in Fig. 4.

1. If $A_2=a_{2,1}$ and $A_5=a_{5,1}$ then Class $=c_1$
2. If $A_2=a_{2,1}$ and $A_5=a_{5,2}$ and $A_3=a_{3,1}$ then Class $=c_1$
3. If $A_2=a_{2,1}$ and $A_5=a_{5,2}$ and $A_3=a_{3,2}$ then Class $=c_2$
4. If $A_2=a_{2,2}$ then Class $=c_2$

Looking at the four decision rules above, we observe that attributes A_1 and A_4 are not part of the model. Some immediate questions arise based on these findings:

1. What does the absence of attributes A_1 and A_4 tell design engineers about their *relevance* to future product designs?
2. How long into the future will the current decision rules be *valid*? (i.e., maintain high predictive capability)
3. Are there any emerging attribute trends that are not represented by the decision tree that may be useful to design engineers?

To address these research questions concerning *attribute relevance/irrelevance*, let us first introduce several well established definitions of attribute relevance that exist in the literature [30,31].

Definition 1. *An attribute A_i is said to be relevant to a concept (decision rule) C if A_i appears in every Boolean formula that represents C and irrelevant otherwise.*

Definition 2. *A_i is relevant iff there exists some attribute value a_{ij} and class value c_i for which $p(A_i=a_{ij})>0$ such that $p(Class=c_i \mid A_i=a_{ij}) \neq p(Class=c_i)$*

Definition 3. *A_i is relevant if each unique value varies systematically with category (class) membership*

Definition 4. *A_i is relevant iff there exists some a_{ij}, c_i, and s_i for which $p(A_i=a_{ij})>0$ such that $p(Class=c_i,\ S_i=s_i \mid A_i=a_{ij}) \neq p(Class= c_i,\ S_i=s_i)$*, where S_i represents the set of all attributes not including A_i

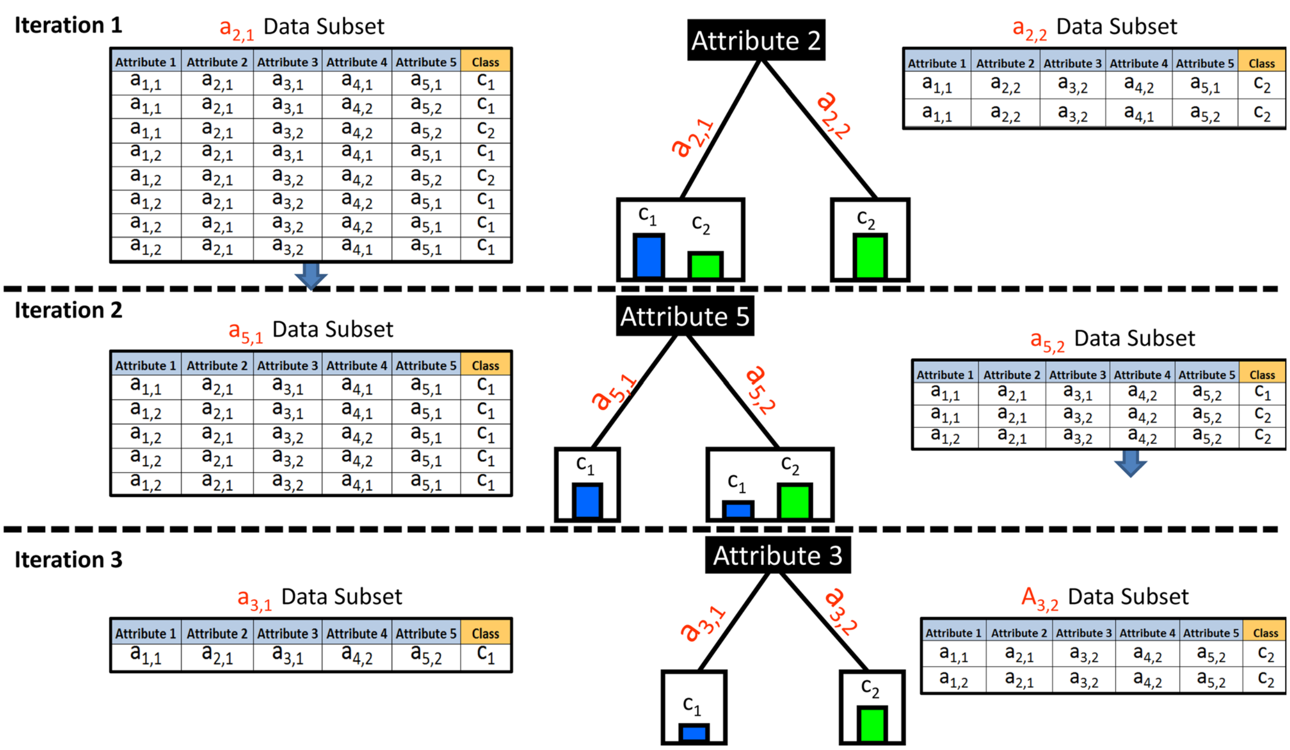


Fig. 4 Example decision tree result for product design

Definition 5. A_i *is* **strongly relevant** *iff there exists some* a_{ij}, c_i *and* s_i *for which* $p(A_i = a_{ij}, S_i = s_i) > 0$ *such that* $p(Class = c_i \mid A_i = a_{ij}, S_i = s_i) \neq p(Class = c_i \mid S_i = s_i)$

Based on the results from Table 1, there exists the possibility that an attribute evaluation metric may omit relevant attributes in the model due to inconsistencies in how attribute relevance is defined [30]. For design engineers, omitting a key attribute due to an irrelevance characterization could mean the subsequent failure of a product as customer needs may not be fully captured. We aim to minimize the inconsistencies in attribute characterization by looking at the problem from a time series perspective. That is, attributes that are truly relevant to a product design should consistently show up in the predictive models through many time steps and attributes that are indeed irrelevant to a product design would remain absent in the predictive model over time.

Section 3.3 relates the concepts of attribute relevance to product design where we expand on the definition of attribute relevance-irrelevance to aid design engineers determine when to include or exclude certain attributes for next generation product design.

3.3 Characterizing Attribute Irrelevance in Product Design. For design engineers, determining how attributes within a given data set influence future consumer purchasing decisions is paramount and could mean the market success or failure of a new product. The definitions of attribute relevance presented in Sec. 3.2 may not capture all of the concepts relating to product design. For example, in the decision tree in Fig. 4, we have determined that attributes A_1 and A_4 are not part of the decision tree and are therefore considered *irrelevant* based on the pertaining definitions of attribute relevance presented in Sec. 3.2. That is, their inclusion/exclusion does not significantly influence the values of the class variable. Should attributes A_1 and A_4 therefore be omitted from future product designs and if so, what consequences would this have in the consumer market space?

To address these issues in product design, we propose several subcategories of attribute *irrelevance* with the goal of ensuring that vital attributes are not omitted from a product design simply based on an irrelevance characterization.

1. Obsolete attribute (OA)*:* An attribute A_i is defined as obsolete if it has been deemed irrelevant at iteration j (given time periods $t_1,\ldots,t_n$) and its inclusion/exclusion over time does not *systematically influence* the values of a class variable. The measure of systematic influence is determined by the time series entropy trend of A_i. If A_i exhibits a monotonically increasing entropy trend (determined by the Mann-Kendall trend detection test introduced in Sec. 3.3.1), then this indicates that attribute A_i is consistently losing predictive power over time. If an attribute falls under this classification at the end of a given time series, it can be omitted from the next generation product designs as seen in Fig. 5.
2. Standard attribute (SA)*:* An attribute A_i is defined as standard if it has been deemed irrelevant at iteration j (given time periods $t_1,\ldots,t_n$) and its inclusion/exclusion over time systematically influences the values of a class variable. As with the previous definition, the measure of systematic influence will be quantified based on the time series entropy trend of A_i. If A_i exhibits a monotonically decreasing entropy trend (determined by the Mann-Kendall trend detection test introduced in Sec. 3.3.1), then this indicates that attribute A_i is consistently gaining predictive power over time (despite its initial irrelevant characterization). If an attribute falls under this classification at the end of a given time series, it should be considered vital to a product design, despite its seemingly irrelevant characterization as seen in Fig. 5. An example of such an attribute would be an airbag in an automobile. Since almost every vehicle is now equipped with an airbag, customers may not consider this attribute while making a vehicle purchase because it is assumed to be a standard to the vehicle. If, however, the airbag were removed from the vehicle design, this may significantly alter a customer's purchasing decision.
3. Nonstandard attribute (NA)*:* An attribute A_i is defined as nonstandard if it has been deemed irrelevant at iteration j (given time periods $t_1,\ldots,t_n$), and its inclusion/exclusion does not reveal a discernible relation to the class variable. This is determined by the absence of a monotonically increasing or decreasing entropy trend as determined by the Mann-Kendall trend detection test introduced in Sec. 3.3.1. Attributes that may exhibit this type of behavior in product design may be novel attributes that consumers may not yet fully be aware of or existing attributes that have variations within the market space. Such attributes should not be overlooked and may either turn out to be a short term consumer hype or may eventually become standard expectations. Consequently, we propose that modular components be designed for attributes exhibiting this type of pattern (as seen in Fig. 5) as these modules can be upgraded or eliminated all together based on future market demands.

Table 1 Attribute characterization based on attribute definition

Attribute	D 1	D 2	D 3	D 4	D 5
Attribute 1	—	x	—	x	—
Attribute 2	x	x	—	x	—
Attribute 3	—	x	—	x	x
Attribute 4	—	x	—	x	—
Attribute 5	x	x	—	x	x

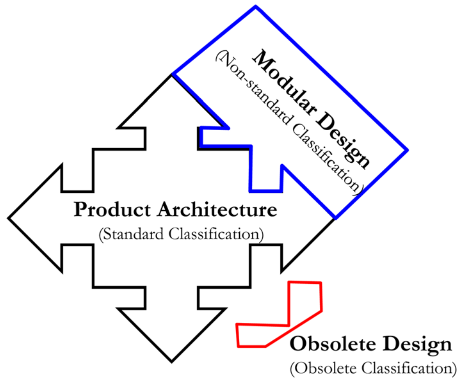


Fig. 5 Product design implications of attribute irrelevance classification

3.3.1 Mann-Kendall Trend Detection. To detect trends for each Attribute A_i that has been deemed *irrelevant* at iteration j, we employ the nonparametric Mann-Kendall statistic [32,33]. The Mann Kendall trend test does not provide us with the magnitude of the trend, if one is detected. Rather, it simply quantifies the presence/absence of a trend which is all we need to classify each attribute within the data set. The Mann-Kendall test is based on the statistic S defined as [27]

$$S = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} sgn(x_j - x_i) \tag{10}$$

Here, n represents the total number of time series data points, x_j represents the data point one time step ahead and x_i represents the current data point

$$sgn = \begin{cases} 1 & \text{if } (x_j - x_i) > 0 \\ 0 & \text{if } (x_j - x_i) = 0 \\ -1 & \text{if } (x_j - x_i) < 0 \end{cases} \tag{11}$$

The corresponding Kendall's Tau is related to the S statistic as follows:

$$\tau = \frac{S}{\frac{1}{2}n(n-1)} \tag{12}$$

The null hypothesis is that there is no trend within the data. Therefore, if the resulting p-value is less than the significance level ($\alpha = 0.05$), we reject the null hypothesis and assume a positive (positive τ) or negative (negative τ) trend. For more complex trend patterns that may also exhibit seasonality, the seasonal Kendall test can be employed [34].

The characterization of attribute irrelevance (as either obsolete, nonstandard, or standard) is determined by looking beyond a single data set and generating models based on multiple time steps that quantify attribute relevance/irrelevance over time. Given a time series data set t_1 to t_n as illustrated in Fig. 6, we analyze each data set from t_1 to t_n and based on the gain ratio relevance definition, characterize the test attribute A_i as either relevant or irrelevant at iteration j. If an attribute is deemed irrelevant, we then employ the Mann-Kendall test to analyze the histories of each attribute entropy value from t_1 to t_n. An attribute value exhibiting increasing predictive power (lower entropy) over time would be deemed potentially useful in future iterations. The resulting characterization of the predictive model generated in time period t_{n+1} will therefore assign an attribute irrelevance characterization based on the trends of the historical entropy data.

Each of the attribute irrelevance definitions will be represented as a binary variable; 1 implies that an attribute is characterized as either Obsolete, Nonstandard, or Standard at a given iteration j and 0, otherwise. At each iteration, an attribute deemed irrelevant can only assume one of the three possible irrelevant characterizations. The final classification of an irrelevant attribute is assigned after the final iteration m. The final iteration m is reached after a homogeneous class distribution is attained for one of the subsets of the data (a leaf node in the decision tree structure). A variable is defined for each irrelevant characterization ($OA_{t=1,\ldots,n}$, $NS_{t=1,\ldots,n}$, and $SA_{t=1,\ldots,n}$) and its value, determined by summing across all iterations ($j = 1,\ldots,m$) as described below

$$OA_{t=1,\ldots,n} = \sum_{j=1}^{m} OA_j \cdot \frac{T_j}{T} \tag{13}$$

$$NS_{t=1,\ldots,n} = \sum_{j=1}^{m} NS_j \cdot \frac{T_j}{T} \tag{14}$$

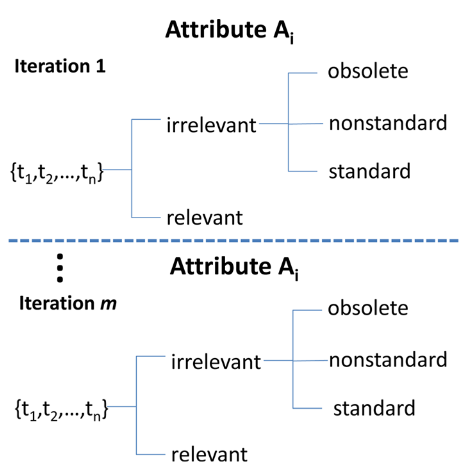


Fig. 6 Attribute (A_i) characterization (relevant and irrelevant categorization) from iteration 1 to iteration m (each iteration contains a total of n time series data sets)

$$SA_{t=1,\ldots,n} = \sum_{j=1}^{m} SA_j \cdot \frac{T_j}{T} \tag{15}$$

Here, T_j represents the number of data instances used to calculate the gain ratio statistics at iteration j and T represents the total number of data instances in the entire data set.

At iteration j, each attribute characterization is weighted based on the proportion (T_j/T) of instances. Therefore, the initial characterization at iteration 1 (containing the entire data set) carries the most weight due to the presence of all instances of the data. The classification of an attribute at time step t_{n+1} is determined by selecting the irrelevant characterization with the highest variable value (($OA_{t=1,\ldots,n}$, $NS_{t=1,\ldots,n}$, and $SA_{t=1,\ldots,n}$)). Given time steps $t_1,\ldots t_n$, the pseudo code for the irrelevant attribute characterization for attribute A_i is as follows:

1. Start: iteration $j = 1$
2. If predicted *Gain Ratio* of Attribute A_i is not the highest, Attribute A_i is considered irrelevant
3. Employ Mann Kendall (MK) trend test for Attribute A_i
4. If MK τ is negative (with p-value < alpha), irrelevant classification = Standard
5. Else If MK τ is positive (with p-value < alpha), irrelevant classification = Obsolete
6. Else If MK τ is positive/negative (with p-value < alpha), irrelevant classification = Nonstandard
7. While data set/subset does not contain a homogeneous class
8. Split the data set into subsets based on the number of mutually exclusive values of the attribute with the highest Gain Ratio from Step 2
9. $j = j+1$ and revert to Step 2 for each data subset
10. End Tree, Classify Irrelevant Attribute A_i based on highest variable value (($OA_{t=1,\ldots,n}$, $NS_{t=1,\ldots,n}$, $SA_{t=1,\ldots,n}$))

3.3.2 Product Concept Demand Modeling. Once the time series decision tree model has been generated and irrelevant attributes characterized, a fundamental question that still remains is how to estimate the demand for the resulting product concepts (unique attribute combinations). If we take for example the resulting product concept {**Hard Drive** = 16 GB, **Interface** = Slider, **Price**=$179} in the left branch of Fig. 9, enterprise decision makers would want to know the overall market demand for this particular product so that potential product launch decisions can be made. With a traditional decision tree model (using a static data set for model generation), the demand for this particular product concept will be a subset of the original training data set used to generate the model (T_m/T, where T_m denotes the number of supporting data instances after m iterations/data partitions) [3]. This is analogous to a product's *choice share* (discrete choice analysis case) which has been used extensively by researchers in the design community to estimate product demand [5,6,8]. Since the proposed trend mining algorithm is making predictions about future product designs, the demand for a resulting product concept is estimated based on the time series trend of the supporting instances T_m using the Holt-Winters forecasting approach presented in Sec. 3.1.1. This will enable to design engineers to anticipate future product demand for the predicted trend mining model.

4 Product Design Example

4.1 Cell Phone Design Study. To validate the proposed trend mining methodology, we test several well known data sets and compare the results of the proposed preference trend mining algorithm with traditional demand modeling techniques. For conciseness, we will present a detailed explanation of the cell phone case study, while only providing the results for the remaining data sets used in our evaluation. The original cell phone case study was based on a University of Illinois online survey of cell phone attribute preferences originally created using the UIUC webtools

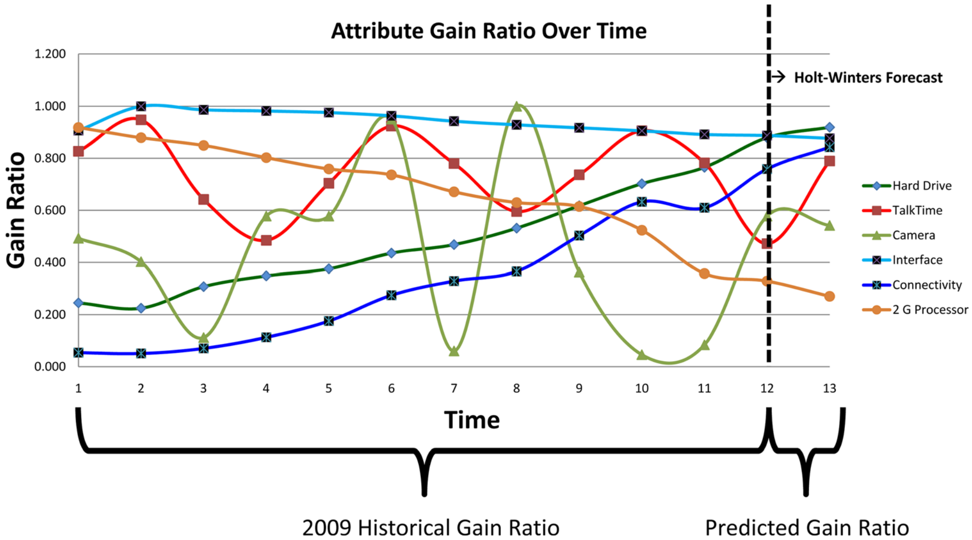


Fig. 7 Time series gain rRatio at iteration 1 (Period 1–12 with Period 13 predicted by employing the Holt-Winters predictive model)

interface [3,4]. To accommodate the time series nature of the proposed methodology, the product design scenario is presented as follows:

Enterprise decision makers within a cell phone company are looking to launch their next generation cell phone early in the first quarter of 2010. To guide their product design decisions, 12 data sets (representing monthly customer preference data for fiscal year 2009) are available through online customer feedback. Based on the time series data, design engineers want to integrate customer preferences directly into the next generation product design. The goal of the new cell phone project is for the functionality of the next generation cell phone design to anticipate the preferences of the customers at the time of product launch; preferences that are constantly evolving within the market space.

For each monthly data set, there are six product attributes and one dependent variable. There are a total of 12,000 instances (customer response) for the entire 12 month time period, partitioned into 1000 instances of customer feedback per month. The attributes, along with their corresponding values are as follows:

Hard Drive: {8 GB, 16 GB, 32 GB}
Talk Time: {3 h, 5 h, 7 h}
Camera: {2.0 MP, 3.1 MP, 5.0 MP}
Interface: {Flip Phone, Slider Phone, Touch Screen Phone}
Connectivity: {Bluetooth, Wifi}
2G Processor: {Limited, Capable}

The class variable is the price category of the given cell phone design within the time series data: **Price:** {\$99, \$149, \$179, \$199, \$249}. The class variable for product design problems can be set by enterprise decision makers regarding the overall enterprise objective. For next generation product design, enterprise decision makers may be interested in quantifying the price customers will be willing to pay, given a combination of product attributes. Other class variables in product design could be product brands, binary purchasing decisions, and environmental impact metrics, to name but a few.

The structure of the data is similar to that presented in Fig. 2 with the attribute names indicated by the first row of each column (except for the last column which represents the class variable, price). In the time series data, the distribution of the attributes as well as the class values associated with each attribute value changes over time.

Up until now, demand modeling in product design had focused on utilizing the most recent data set to generate predictive models about future customer behavior. Our research findings presented in Sec. 5 reveal that such techniques may not fully capture emerging consumer preference trends and may ultimately mislead future product design decisions.

5 Results and Discussion

The results of the cell phone case study introduced in Sec. 4 provide valuable insight into the challenges of designing products for volatile consumer markets. We begin by presenting the time series gain ratio statistics for each attribute (at iteration 1) shown in Fig. 7. In the proposed trend mining methodology, we want to take into consideration all possible scenarios for the attribute gain ratio statistics over time; that is, we want to capture attributes that display a monotonically increasing or decreasing trend, a seasonal trend or no trend at all which we model using the Holt-Winters technique presented in Sec. 3.1. Based on the level of seasonality or trend within the data, the one time step ahead predictions (period 13) are modeled. At period 12 in Fig. 7, we observe that the *Interface* attribute has a higher gain ratio than the *Hard Drive*. However, based on the emerging trends of these two attributes, it can be observed that the Hard Drive attribute will have a higher gain ratio in future time periods, which the Holt-Winters model predicts in time period 13.

New design insights obtained by preference trend mining. In order to understand the product design implications of these findings, let us take a look at the predictive model results that are generated using the most recent data set (period 12). In Fig. 8, the only relevant attributes to the price variable are: *Interface, Connectivity* and *Camera*, with the associated decision rules acquired by traversing down the appropriate paths of the decision tree. In contrast, when the proposed time series preference trend mining algorithm is employed using the data from periods 1 to 12, there

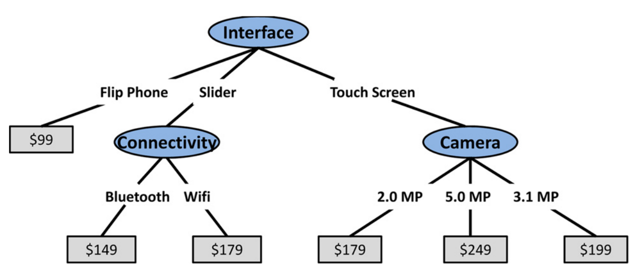
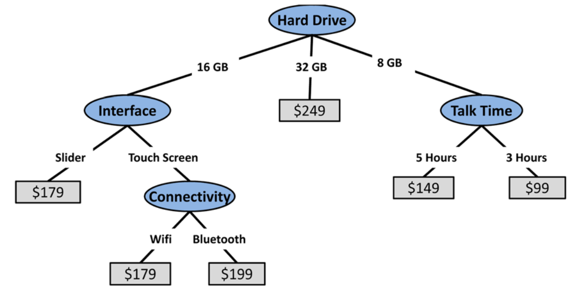


Fig. 8 Decision tree model using Period 12, 2009 data set only for model generation (results attained using Weka 3.6.1 [35])



Tree Path (right to left)	Relevant Attributes	Price	Demand	Irrelevance Characterization: Standard	Nonstandard	Obsolete
1	Hard Drive, Talk Time	$99	180	Camera	2G Processor, Connectivity	Interface
2	Hard Drive, Talk Time	$149	362		2G Processor, Connectivity	Interface, Camera
3	Hard Drive	$249	73	Talk Time, Camera	2G Processor, Connectivity	Interface
4	Hard Drive, Interface Connectivity	$199	412	Talk Time, Camera	2G Processor	
5	Hard Drive, Interface Connectivity	$179	99	Talk Time, Camera	2G Processor	
6	Hard Drive, Interface	$179	208	Talk Time, Camera	2G Processor, Connectivity	

Fig. 9 Trend mining model using Periods 1–12, 2009 data for model generation (results attained using ESOL developed Java Based PTM code compatible with Weka) [35])

are noticeable differences in the resulting attributes that are considered relevant (Fig. 9). From the resulting decision trees in Figs. 8 and 9, we observe that the common attributes between the two models are the interface and connectivity attributes. However, even with the interface attribute being common between the two models, we observe that the *Flip Phone* interface design found in Fig. 8 is not included in Fig. 9, providing engineers with the knowledge that this particular attribute value is not desired in future time periods. Given the differences between these two decision tree structures, entirely different product design decisions may result to address the needs of the market.

Furthermore, for those attributes that are considered irrelevant to the classification of price (and are therefore omitted from the decision tree model in Figs. 8 and 9), design engineers have no direct way of deciding whether these attributes should be omitted from all future cell phone designs. As a reminder, an irrelevant attribute simply means that at iteration j, an attribute does not have the highest gain ratio, not necessarily that it does not have any predictive power whatsoever, as illustrated in Fig. 7. At iteration 1, since the PTM algorithm predicts that the *Hard Drive* attribute will have the highest gain ratio at time period 13 (see Fig. 7), we characterize the remaining attributes as either obsolete, nonstandard, or standard. The entropy histories along with the results from the Mann Kendall trend test in Fig. 10 indicate that the 2G Processor is characterized as obsolete (positive τ values and p value within tolerance limit), while the remaining attributes are characterized as *Nonstandard* (due to p values exceeding the tolerance limit). After subsequent iterations of the PTM algorithm, the

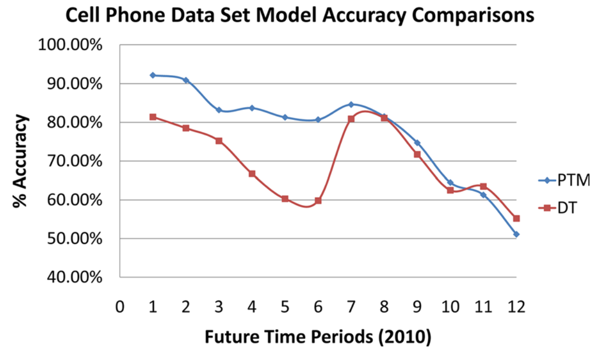


Fig. 11 Comparison of predictive accuracies between the PTM and DT models using 12 unseen time stamped data from 2010) [35])

attributes that do not show up in the tree are therefore classified as shown in Fig. 9, with the accompanying demand (# supporting predicted instances) accompanying each branch of the tree.

5.1 Model Validation. In addition to the structural differences of the resulting decision tree models, there are also noticeable differences in the predictive accuracies. Figure 11 presents the predictive accuracy results between the proposed PTM model and the traditional DT classification model. The predictive accuracies are calculated using 12 monthly data sets from 2010. For each instance in a given monthly data set, the attribute combinations resulting in a class value are tested against the decision tree predictions by traversing down the path of the decision trees in Figs. 8 and 9. If the class value predicted by the decision tree model matches the actual class value in the monthly data set, a value is incremented in the *correct predictions* category; otherwise, a value is incremented in the *incorrect predictions* category. The summary predictive accuracies in Fig. 11 reveal that the PTM model attains a higher predictive accuracy for many of the time periods, compared to the DT model.

To obtain a statistically valid conclusion on the predictive accuracies of the two models, we employ the Wilcoxon signed rank test which has been proposed in the data mining/machine learning literature as a suitable approach for comparing two models against multiple data sets [36]. The null hypothesis of the test is that the median difference between the two model accuracies is zero. The alternate hypothesis is that the accuracy of the DT model is less than that of the PTM model. Using a significance level of $\alpha = 0.05$, the null hypothesis (data in Fig. 11) is rejected with a p value of 0.0224, providing statistical evidence that the accuracy of the PTM algorithm exceeds that of the DT for the Cell Phone data set. We see that the predictive accuracy of both models diminishes over time with slightly above 50% in period 12. The PTM accuracy may be enhanced in future time periods by changing the k value of the k-ahead time predictions from 1 (in the cell phone model) to the specific future period of interest (1–12).

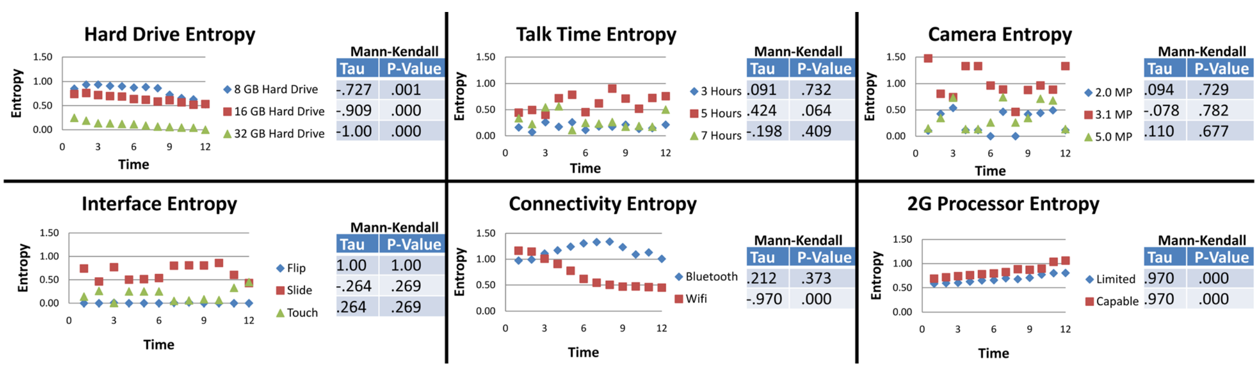


Fig. 10 Time Series Attribute Entropy values for irrelevance characterization

Table 2 Comparison of predictive accuracies between the PTM and DT models using time series data

Predictive model	Model validation characteristics						
	Data set	# Attributes	#Instances/Period	# Periods to Train	# Periods to Test	Higher % Accuracy	p-value
PTM	Car Evaluation	7	1728	24	12	x	0.00507
DT							
PTM	Cylinder Bands	10	540	36	24	x	0.00007
DT							
PTM	Automobile Brand	9	205	24	12	x	0.00008
DT							

Additional data sets from the UC Irvine machine learning repository were employed to further validate the two models. The UC Irvine machine learning repository is a collection of databases that have been used extensively in the machine learning community for empirical analysis and validation of data mining/machine learning algorithms [37]. To accommodate the time series nature of the proposed methodology, additional time series data for each UC Irvine data set were generated with varying data set conditions (attribute space, number of instances, number of time periods, etc.). The time series data sets were then tested against the two models for model accuracy, with the results presented in Table 2. The results from Table 2 emphasize the robustness of the proposed PTM algorithm in handling different types of time series data while still maintaining greater predictive accuracies, compared with the traditional decision tree model. Due to the variation in data set structure, size, etc., it is rare for an algorithm to outperform on every metric of performance [38]. Therefore, the proposed PTM model is well suited for data sets that exhibit monotonically increasing/decreasing or seasonal trends similar to the test data sets presented. In scenarios where no discernable trends exist in the data set, the PTM algorithm was found to perform comparable to traditional demand modeling techniques which should not be surprising, given the underlying formulation of the proposed PTM algorithm.

6 Conclusion and Path Forward

The major contribution of this research is to propose a machine learning model that captures emerging customer preference trends within the market space. Using time series customer preference data, we employ a time series exponential smoothing technique that is then used to forecast future attribute trend patterns and generate a demand model that reflects emerging product preferences over time. The Mann Kendall statistical trend detection technique is then used to test for attribute trends over time. An attribute irrelevance characterization technique is also introduced to serve as a guide for design engineers trying to determine how the classified attributes are deemed irrelevant by the predictive model. The insights gained from the preference trend mining model will enable engineers to anticipate future product designs by more adequately satisfying customer needs. Future work in customer preference trend mining will include expanding the current approach to handle the continuous attribute and class domain.

Acknowledgment

The work presented in this paper is supported by Sandia National Labs, SURGE and the National Science Foundation under Award No. CMMI-0726934. Any opinions, findings and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of Sandia National Labs, SURGE or the National Science Foundation. The authors would like to acknowledge Yohannes Kifle for the programming aspects and software development of this work.

Nomenclature

PTM = preference trend mining
DT = decision tree
OA = obsolete attribute classification
SA = standard attribute classification
NS = nonstandard attribute classification
Tj = subset of the training data T that contains one of the mutually exclusive outcomes of an attribute
t = A given instance in time

References

[1] Pullmana, M., Mooreb, W., and Wardellb, D., 2002, "A Comparison of Quality Function Deployment and Conjoint Analysis in New Product Design," J. Prod. Innovation Manage., **19**(1), pp. 354–364.
[2] Green, P., Carroll, J., and Goldberg, S., 1981, "A General Approach to Product Design Optimization via Conjoint Analysis," J. Marketing, **45**, pp. 17–37.
[3] Tucker, C., and Kim, H., 2009, "Data-Driven Decision Tree Classification for Product Portfolio Design Optimization," J. Comput. Inf. Sci. Eng., **9**(4), 041004.
[4] Tucker, C. S., and Kim, H. M., 2008, "Optimal Product Portfolio Formulation by Merging Predictive Data Mining With Multilevel Optimization," Trans. ASME J. Mech. Des., **130**, pp. 991–1000.
[5] Tucker, C., Hoyle, C.., Kim, H., and Chen, W., 2009, "A Comparative Study of Data-Intensive Demand Modeling Techniques in Relation to Product Design and Development," Proceedings of the ASME Design Engineering Technical Conferences, San Diego, CA, DETC2009-87049.
[6] Wassenaar, H., Chen, W., Cheng, J., and Sudjianto, A., 2005, "Enhancing Discrete Choice Demand Modeling for Decision-Based Design," ASME J. Mech.Des., **127**(4), pp. 514–523.
[7] Michalek, J. J., Feinberg, F., and Papalambros, P., 2005, "Linking Marketing and Engineering Product Design Decisions via Analytical Target Cascading," J. Prod. Innovation Manage.: Spec. Issue Des. Market. New Product Dev., **22**, pp. 42–62.
[8] Lewis, K., Chen, W., and Schmidt, L., eds., 2006, *Decision Making in Engineering Design*, ASME Press, New York.
[9] Wassenaar, H. J., Chen, W., Cheng, J., and Sudjianto, A., 2005, "Enhancing Discrete Choice Demand Modeling for Decision-Based Design," Trans. ASME J. Mech. Des., **127**, pp. 514–523.
[10] Frischknecht, B., Whitefoot, K., and Papalambros, P. Y., 2010, "On the Suitability of Econometric Demand Models in Design for Market Systems," ASME J. Mech. Des., **132**(12), 121007.
[11] Wassenaar, H. J., and Chen, W., 2003, "An Approach to Decision-Based Design With Discrete Choice Analysis for Demand Modeling," Trans. ASME J. Mech. Des., **125**, pp. 490–497.
[12] Greene, W., and Hensher, D., 2003, "A Latent Class Model for Discrete Choice Analysis: Contrasts With Mixed Logit," Transp. Res., **37**, pp. 681–698.
[13] Shannon, C. E., 2001, "A Mathematical Theory of Communication," SIGMOBILE Mob. Comput. Commun. Rev., **5**(1), pp. 3–55.
[14] Quinlan, J., 1986, "Induction of Decision Trees," Mach. Learn., **1**(1), pp. 81–106.
[15] Mela, C., Gupta, S., and Lehmann, D., 1997, "The Longterm Impact of Promotion and Advertising on Consumer Brand Choice," J. Market. Res., **34**(2), pp. 248–261.
[16] Jedidi, K., Mela, C., and Gupta, S., 1999, "Managing Advertising and Promotion for Long-Run Profitability," Market. Sci., **18**(1), pp. 1–22.
[17] Seetharaman, P. B., 2004, "Modeling Multiple Sources of State Dependence in Random Utility Models: A Distributed Lag Approach," Market. Sci., **23**(2), pp. 263–271.
[18] Lachaab, M., Ansari, A., Jedidi, K., and Trabelsi, A., 2006, "Modeling Preference Evolution in Discrete Choice Models: A bayesian State-Space Approach," Quant. Market. Econ., **4**(1), pp. 57–81.
[19] Böttcher, M., Spott, M., and Kruse, R., 2008, "Predicting Future Decision Trees From Evolving Data," *Proceedings of ICDM '08*, pp. 33–42.
[20] Günther, C. W., Rinderle, S. B., Reichert, M. U., and van der Aalst, W. M. P., 2006, "Change Mining in Adaptive Process Management Systems," *(CoopIS'06) Volume 4275 of Lecture Notes in Computer Science*, Montpellier, France; Springer-Verlag, Berlin/Heidelberg/New York, pp. 309–326.
[21] Li, P., Hu, X., and Wu, X., 2008, "Mining Concept-Drifting Data Streams With Multiple Semi-Random Decision Trees," *in ADMA '08: Proceedings of the 4th international conference on Advanced Data Mining and Applications*, Springer-Verlag, Berlin/Heidelberg, pp. 733–740.
[22] Wacker, J. G., and Trelevan, M., 1986, "Component Part Standardization: An Analysis of Commonality Sources and Indices," J. Oper. Manage., **6**(2), pp. 219–244.

[23] Wu, T., Chen, Y., and Han, J., 2007, "Association mining in large databases: A re-examination of its measures," *Proceedings of the 11th European conference on Principles and Practice of Knowledge Discovery in Databases*, PKDD 2007, Springer-Verlag, Berlin/Heidelberg, pp. 621–628.
[24] Geng, L., and Hamilton, H., 2006, "Interestingness Measures for Data Mining: A Survey," ACM Comput. Surv., **38**(3), p. 9.
[25] Chatfield, C., 1978, "The Holt-Winters Forecasting Procedure," J. R. Stat. Soc. Ser. C, Appl. Stat., **27**(3), pp. 264–279.
[26] Grubb, H., and Mason, A., 2001, "Long Lead-Time Forecasting of Uk Air Passengers by Holt-Winters Methods With Damped Trend," Int. J. Forecast., **17**(1), pp. 71–82.
[27] Yue, S., Pilon, P., and Cavadias, G., 2002, "Power of the Mannkendall and Spearman's Rho Tests for Detecting Monotonic Trends in Hydrological Series," J. Hydrol., **259**(1–4), pp. 254–271.
[28] Cleveland, R. B., Cleveland, W. S., Mcrae, J. E., and Terpenning, I., 1990, "Stl: A Seasonal-Trend Decomposition Procedure Based on Loess," J. Off. Stat., **6**(1), pp. 3–73.
[29] Smith, B., Williams, B., and Oswald, R., 2002, "Comparison of Parametric and Nonparametric Models for Traffic Flow Forecasting," Transp. Res., Part C: Emerg. Technol., **10**(4), pp. 303–321.
[30] John, G., Kohavi, R., and Pfleger, K., 1994, "Irrelevant Features and the Subset Selection Problem," *International Conference on Machine Learning*, pp. 121–129.
[31] Zhao, Z., and Liu, H., 2009, "Searching for Interacting Features in Subset Selection," Intell. Data Anal., **13**(2), pp. 207–228.
[32] Mann, H. B., 1945, "Nonparametric Tests Against Trend," Econometrica, **13**(3), pp. 245–259.
[33] Kendall, M., and Gibbons, J. D., 1990, "Rank Correlation Methods," *A Charles Griffin Title*, 5th ed., Oxford University Press, New York.
[34] Hirsch, R., and Slack, J., 1984, "A Nonparametric Trend Test for Seasonal Data With Serial Dependence," Water Resour. Res., **20**(6), pp. 727–732.
[35] Witten, I. H., and Frank, E., 2002, "Data mining: Practical Machine Learning Tools and Techniques With Java Implementations," SIGMOD Rec., **31**(1), pp. 76–77.
[36] Demšar, J., 2006 "Statistical Comparisons of Classifiers Over Multiple Data Sets," J. Mach. Learn. Res., **7**(1), pp. 1–30.
[37] Frank, A., and Asuncion, A., 2010, "UCI Machine Learning Repository," University of California, Irvine, CA. http://archive.ics.uci.edu/ml.
[38] Domingos, P., and Pazzani, M., "Beyond Independence: Conditions for the Optimiality of the Simple Bayesian Classifier," in *Machine Learning*, pp. 105–112.