search

Maximizing Information Liquidity in Electronic Commerce

Vasant Dhar and Arun Sundararajan
Department of Information Systems
Stern School of Business
New York University
New York, New York 10012

***Abstract*- The emerging bottleneck in electronic commerce is that of converting vast amounts of customer behavior data into useful information. We view this as a problem of maximizing information liquidity – the rate at which organizations are able to transform the inherent information in a data set into an economically valuable action. We describe how to overcome this bottleneck, by presenting a model for maximizing information liquidity in electronic commerce. Our model is usable in a variety of situations. Specifically, when a large amount of transaction data already exists, the model is able to exploit this data to generate rules describing preferences that can be used to classify behaviors, and to subsequently map behaviors of non-customers into known ones. Alternatively, where the predominant data available are about behaviors, the model can be used to cluster these behaviors and combine the resulting clusters with available transaction data to generate rules describing preferences. In both cases, the central question addressed is "when do I have enough information to make a meaningful offer?" Acting too early can result in inappropriate offers, while acting too late can result in missed opportunities. Good information and timing are therefore critical; the model in this paper is a first step in this direction.**

I. INTRODUCTION

Electronic commerce has spawned a new form of interaction between a firm and it's consumers – one that is almost exclusively through a computer-based interface. It is unprecedented in terms of the amount of *accessible data* that is generated as a by-product of transaction or browsing. Walter Wriston, the former chairman of Citicorp made the observation over a decade ago that "information about money has become almost important as money itself." Today, it is fair to say that this comment applies not only to the financial industry, but also to the entire economy that is in the midst of the dramatic transformation by the Internet.

While most commerce sites log and store clickstream data that describes this electronic interaction, few have actually used this information in any meaningful way. The reasons for this are twofold. First, the concerns thus far have been on performance – the ability to serve up pages quickly. The second barrier has been the sheer volume of the data being generated, and the lack of a sound model for how to formulate the learning problem for leveraging this data. In terms of volume, for instance, of the leading servers of advertisements serve unto 300 million banners daily, which translates into about 100 gigabytes of daily data.

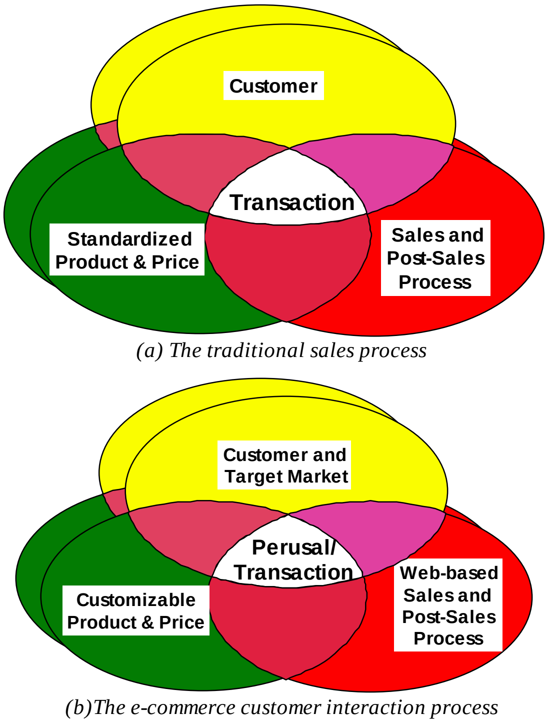
The bandwidth bottleneck is gradually becoming less of a problem. The emerging challenge, which we address in this paper, is how to overcome the *information bottleneck* – in other words, how to identify and extract *actionable information* from these large amounts of data [4]. We provide a formal model for learning from diverse sources of data that includes behaviors, transactions, and demographics. The model can help organizations determine *what data* to collect, how to *evaluate the economic value* of the data collected, and how to *exploit meaningful patterns* in this data. We pay particular attention to the speed of response, where the central question is "when do I have enough information about this individual to make a meaningful offer?" The cost of failure is high – a site that makes less-than-intelligent recommendations based on naïve patterns in clickstream data is likely to be perceived as unintelligent and invasive, and will consequently lose customers rapidly. The model we describe has been motivated by our collaboration with a number of e-commerce companies.

This remainder of the paper is organized as follows. In Section 2, we describe the typical data sources in e-commerce, and how they differ from those of traditional commerce. We also introduce the notion of "noise" in a data set (in particular, noise in behavior data), which motivates the use of *factors* as part of our model. In Section 3, we introduce the concept of *information liquidity* which we view as the ease with which an organization can transform data into usable knowledge. We use the concept of information liquidity as described informally by [11] to quantify the relative benefit that can accrue by using different (and possibly linked) groups of data sets. More specifically, we focus on one crucial dimension of information liquidity, having to do with information content, and provide a model and method for maximizing liquidity based on this dimension.

Subsequent to describing information liquidity, we present our model in Section 4. The model is a first step towards prescribing what clickstream data to collect and how to optimally identify factors that that will leverage its economic value maximally. We conclude by arguing that, over time, organizations will move towards *adaptive web sites* that have the intelligence to use previously identified customer interaction patterns to tailor their subsequent design and interaction, in order to make relevant, precise and timely offers to consumers.

II. DATA SOURCES, NOISE AND FACTORS

Figure 1a shows the "traditional" sales process, where the *transaction* brings together information about three entities: the product, the salesperson or sales channel, and the customer. The product is typically a "standard" product with a pre-specified price, sold through a specific salesperson or channel. Figure 1b shows the same process in e-commerce. One of the most striking differences is that the existence of perusal or web site *interaction* data, which typically dwarfs the amount of transaction data. Also, it deals with the *target markets*, instead of merely customers, thereby shifting the focus solely from existing customers towards customer acquisition. Again, this set can be much larger than the set of



(a) The traditional sales process

(b)The e-commerce customer interaction process

Fig 1: Transaction information: traditional commerce versus e-commerce

customers. Finally, the product is much more easily customizable, and the sales and post-sales process is embedded in the web site.

The relevant sources of information available to an e-commerce firm are summarized in Figure 2. These consist of three datasets. The first is *demographic* data, which has been shown to be a reasonable predictor of certain types of preferences and attitudes. This dataset changes slowly for obvious reasons. The second is *transactional* data, which is also a rich source of customer preferences, probably at a more detailed level than demographic data. This data changes quite rapidly, depending on the product or context. The third set is *interaction* data, which is a potentially rich source of indications of interest. This data is the most voluminous. However, this data is also probably the most transient, making it difficult to extract meaningful information from it.

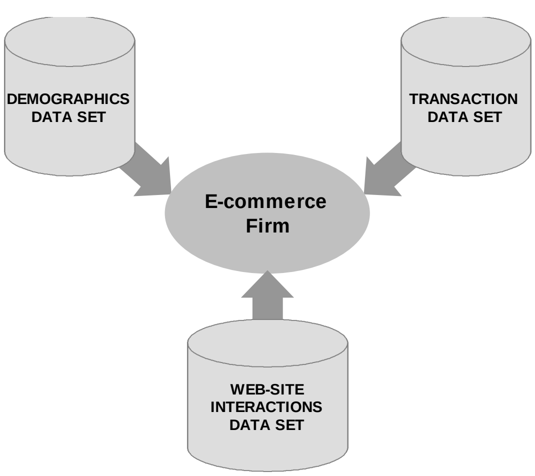


Fig. 2: Sources of customer information for e -commerce firms

Another problem with interaction data is that it is noisy – in other words, along with the meaningful patterns of interaction, there is also a substantial amount of variation in the information, the causes of which have no apparent logic or immediate economic value. For instance, a customer may stop a sequence of clicks on a web site because their doorbell rings, or because their ISP logs them off; similarly, a college student may traverse a set of shopping links as part of a school assignment. In general, a substantial amount of web browsing is done ‘ on a whim’ , rather than due to serious interest on the part of the consumer. Of course, in theory, every click, however irrelevant, tells the firm something about the relevant consumer – but it is safe to say that in practice, a *complete* description of each consumer’ s preferences, besides being way more complex than is possible to determine, is impossible to construct from the limited data that a firm collects from web browsing and transaction patterns.

Given these limitations on the model of consumer preferences one can hope to construct, these whimsical and unintended browsing data points constitute what we call *noise* in the customer interaction data. It is similar to noise in a financial time series; the variation that is not caused by any of the factors that are theorized to determine how a stock price moves. It is also analogous to noise in a communications channel – where the true signal – or in our problem, the true utility function of the customer – is distorted by imperfections in the channel – in our problem, a sequence of links that is not exactly what the customer would like, a sudden external distraction, an infrequent technological difficulty unrelated to the consumer’ s interests that causes a shift in web-browsing, or any one of a number of other reasons.

Transaction data is not impervious to noise either, although it is not as noisy as behavioral data. This is because transactions usually have some degree of involvement and commitment from the user, making them a truer indicator of customer preferences than behavior data. However, individual preferences can change over time, and perhaps more importantly, so can the base set of items purchased. For this reason, it is more robust to deal with abstractions of goods and customers instead of trying to develop *individualized* models of preference or behavior. We refer to these abstractions as *factors* that are intended to account for the meaningful variations, or patterns, in the behavior of similar groups of customers.

Factors have been used to great advantage in the investment community for similar reasons, of noise and simplicity. For example, in constructing a portfolio, managers must take into account the covariance between instruments in the portfolio in order to make the optimal asset allocation decision. If General Motors and Ford are perfectly correlated, it makes little sense to allocate capital to both stocks. But this is rarely the case; so individual covariances between pairs of stocks would have to be computed. However, for a universe of, say, 5000 stocks, this would involve computing a 5000 by 5000 covariance matrix. This requires a very large number of data points, and (perhaps more importantly) such a matrix tends to be highly unstable due to the inherent noise in the price movement of financial instruments. The solution to this problem was provided by *arbitrage pricing theory* [10] where the basic idea is to group securities by factors such as industry group, yield, volatility, and so on. For example, securities in the same industry group would have similar behavior in response to news about that industry group. Similarly, securities with a yield (i.e. dividends) would tend to be affected similarly by news on interest rates. It also turns out that the covariance matrix between factors is relatively stable, implying that it deals better with filtering out the inherent noise in the price movements of individual securities.

Our motivation for using factors is similar. We wish to develop a parsimonious and understandable theory of the domain, by overcoming the limitations imposed by the inherent noise in the web-interaction data.

III. Information Liquidity

When a potential customer is browsing in a physical store, it is usually difficult for a salesperson to determine the customer's intent and preferences from the customer's actions The information available from the customer, his perusal patterns, are difficult to interpret, and attempts to extract more information might be viewed by the customer as invasive.

In contrast, consider a direct sales process, where a salesperson is keenly aware of, and actively leverages the information provided by his one-to-one interaction with customers. This is because the salesperson has the ability to rapidly capture the *relevant* information from the interaction, and also has a *framework* which he or she can immediately *evaluate and use* this information, possibly by relating the information from the interaction with information about past transaction success or failure. For instance, if a customer wants more detail after hearing about the new features of a product, the salesperson might adopt an aggressive strategy to close the sale. On the other hand, if the customer appears disinterested, the salesperson might offer aggressive discounts, or switch to another offer.

In the second example, the interaction is one-to-one, and the salesperson has the intelligence to channel the interaction in a useful way based on indications of interest from the potential customer. In the first case, the seller finds it hard to interpret the movements of the potential customer through physical space, and finds it difficult to channel the interaction is a useful way.

Essentially, what the salesperson is doing in the second example is making his information *liquid*, that is, translating information from one form into another with *minimum loss.* We use the term liquidity in the same sense as in financial markets, namely, how easily one can transform one asset into another, more desirable one without loss of value. A loss in value typically arises because a dealer charges a spread, or because the asset is volatile, which makes it hard to predict the value at which the exchange will occur. It also occurs when the *rate* of asset transformation is restricted, either by the absence of sufficient buyers and sellers, or by accessibility constraints, in which case the time required to transform the asset is indicative of the difficulty of transforming it into the more desirable asset.

We define information liquidity as a measure of the rate at which one can transform the inherent information in a data set into an economically valuable action. Our model of the information liquidity of a group of data sets is determined by three dimensions:

1) Connectivity: This measures how rapidly the information in a group of data set can be collected. It is influenced by two independent factors:

(a) Access: How rapidly can data items from possibly diverse and remote data sets be accessed and linked?

(b) Representation: How are data items in the set represented? For the same data, a flat file would offer the least connectivity, while a relational table would offer superior connectivity.

2) Comprehensiveness: This measures the depth and breadth of information than can be extracted from the data. It is influenced by the following two factors:

a) Size: How large is each data set? One can infer more from a set of a few thousand customer transactions than one can from a dozen transactions, since the former allows one to identify patterns in buying behavior. However, if the data set in unduly large, the timeliness of the inference is reduced, making it less liquid.

b) Span: How many attributes does each record in the data set contain? A sizeable data set on the milk buying patterns of consumers, or the cookie-buying patterns of consumers, may not have much liquidity in isolation, however, a combination of the two may generate useful patterns.

3) Content: This dimension measures the inherent ability to inform us about a consumer's preferences in general. It is determined by two aspects of the data:

a) Density: How much useful information is there, on the average, per data element?

b) Interconnectedness: How closely connected are the different data items to other data items, and how closely connected are different data records [1]?

Content refers to the intrinsic value of the data that is inherent in its attributes or the relationships among the attributes. For instance, a history of book buying patterns is intrinsically (and paradoxically) more liquid than a history of milk-buying patterns, simply because, in general, the books one reads say a lot more about one's preferences than the milk one drinks, and one's purchases of books are more closely tied to other choices one makes than one's purchases of milk. With respect to the relationships among the attributes, if the data is inherently noisy, that is, the relationships are difficult to find, its liquidity is low, in the sense that more effort must be expended to extract its inherent information content. In this case, more intelligence is required from a system in order to make the information more liquid.

In order to increase the liquidity of one's data sets, the first step is to ensure that *both* connectivity (reach) and comprehensiveness (range) are high. It is little use having data at your fingertips unless it is comprehensive and digestible, and similarly, it is useless if you have comprehensive data that is not accessible to decision makers. Both situations are highly prevalent. For example, credit card issuers have gobs of data about their customers, but often have trouble getting it in a meaningful form. Similarly, Internet service providers have massive amounts of data on the browsing patterns of people, but have very little other information about these people. Last, but not least, it is of

[1] One might argue that this is dependent on the firm's customer base, rather than of the data then firm has. While this observation is valid, there are different types of data about a fixed customer base, which inform one about them to different degrees.

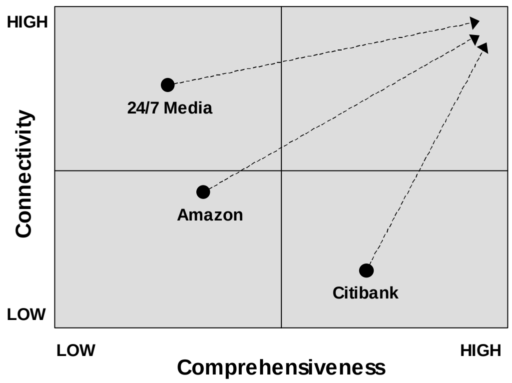


Fig. 3: A projection of the information liquidity space

little economic value to make data accessible and comprehensive, unless the data has inherent information content, or high density; section 4 deals with this dimension in more detail. Meanwhile, Figure 3 shows how three organizations – Amazon, 24/7 Media, and Citigroup fall on the accessibility versus comprehensiveness grid.

Increasing accessibility is largely a technological issue. Universally accessible, high bandwidth TCP/IP-networks can increase liquidity by addressing connectivity issues, while the multidimensional indexing done in hypercube representations, such as those found in OLAP systems, increase liquidity by improving data representation.

Comprehensiveness is a more subtle issue. Figure 4 shows how liquidity is related to size and span, the two relevant measures, and is based on a simple economic model of the value of customer information. As the size of a data set increases, the liquidity increases up to a point, since the firm's breadth of knowledge about its target customers increases. After a point, however, the additional value of the information is dominated by the illiquidity arising from the processing overhead. Span, on the other hand, does not induce the same type of processing overhead concerns. Increasing the span of a data item, in a reasonably sized data set, informs the firm better about each individual customer, thereby increasing the depth of customer knowledge. Its contribution to increasing liquidity diminishes after a point, however

IV. A Model of Liquidity and Behavior Patterns

Intuitively, in order to understand the density of a data set, and to maximize the liquidity of the data set, we need to isolate the factors from each data set, identify which ones are interesting, and associate them with an optimal action or set of actions. We formalize and elaborate on these ideas in this section. We present a model of the e-commerce firm, from the perspective of its customers, products and data. We use this model to describe different approaches to identifying factors of interest, or interesting patterns of customer interaction.

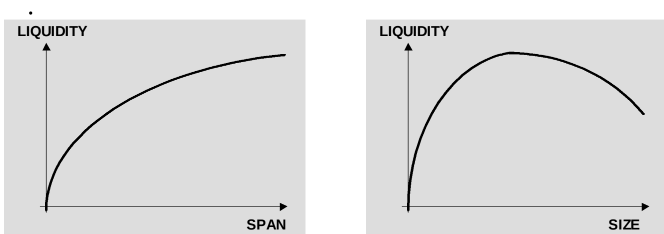


Figure 4: The impact of comprehensiveness on information liquidity

The objective of the model and its associated approaches is *twofold*. It simultaneously *informs* the firm about the inherent density of its data (which can allow it to set customer interaction objectives, or seek out new data sets), and *increases* the liquidity of the data sets up to a point. The more suited an approach is to a specific e-commerce firm, the higher the resulting liquidity of the data.

A. Elements of the Model

We model the ecommerce firm as consisting of the following primitives:

- A set **P** of possible customer profiles. A typical element of **P** is $\boldsymbol{p} = (p_1, p_2, \ldots, p_M)$
- A set $\mathbf{O} = \{o_1, o_2, \ldots, o_N\}$ of offers.
- A set $\mathbf{H} = \{h_1, h_2, \ldots, h_T\}$ of hyperlinks.
- A set $\mathbf{C} = \{c_1, c_2, \ldots, c_K\}$ of target customers.

An example of a profile is $\boldsymbol{p}$=(*male, 39, professor, 10012*). In this case, **P** has four dimensions: gender, age, profession and zip code. Offers correspond to items that a firm may wish to present to a customer, and an offer can include product attributes and price. For example, the hardcover edition of *The Tin Drum* for $19.95 including shipping, and a 160-by-120 advertisement for a Cadillac, are both examples of offers. Associating a price with an offer enables us to treat offers as generally as possible; however, it rules out the use of our model for determining, for instance, dynamic pricing rules.

Hyperlinks form the basic building blocks of the electronic interface that the customer sees. It may turn out that a firm does not index each individual hyperlink on its page, but chooses to group them together as 'weather-related hyperlinks', 'chat-related hyperlinks', 'product recommendation hyperlinks' and so on.

A customer c_i comprises some suitable key, such as customer ID, and perhaps, non-generalizable profile information, such as name, apartment number and so on. Based on these sets, we define the following:

U is the set of all complete, transitive preference orderings on subsets of **O**. For any element $u \in \mathbf{U}$, $o_i \succ_u o_j$ means that under preference ordering u, o_i is preferred to o_j.

The set of profile groupings $\mathbf{2^P}$ is the power set of **P** – the set of all subsets of **P**. We denote a typical element of $\mathbf{2^P}$ as *{p}*.

B, the set of behaviors, is the set of all frequency distributions over **H**. We denote a typical element of **B** as $b=\{p_b(h), N(h)\}$

Completeness and transitivity ensure that the preference orderings are *rational*; while studies in psychology (e.g. the framing experiments of Kahneman and Tversky) have indicated that consumer preferences are not necessarily transitive, these are the basic assumptions made about preferences in all economic models. One can think of u as simply being an ordered sequence of elements from **O**, each of which appears exactly once in the sequence. This allows for a preference ordering that simply specifies a single offer –

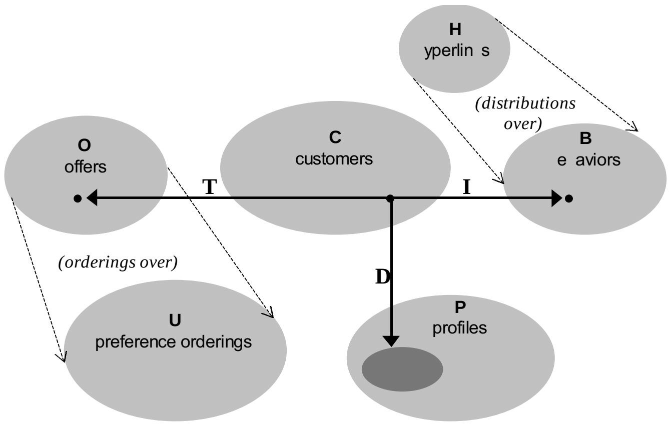


Fig. 5: Transactions (T), demographics (D) and patterns of interaction (I)

the interpretation here is that, under this ordering, the only offer that is of any value is the offer specified [2].

An example of a profile grouping *{p}* is ‘males between the ages of 25 and 45 who live in Manhattan’. The notion of ordering in a behavior can preserve the potentially valuable information that may be contained in the *sequence* with which a customer clicks on hyperlinks. In our first pass at operationalizing the model, we ignore sequence, and treat a behavior as a discrete distribution over hyperlinks.

Each customer c_i has a true preference ordering u and a true profile p. Were the firm to know the true preference ordering of the customer, they would know exactly what offer to make the customer. The following mappings specify and extend the three data sets of Figure 1 in the context of our model, and are shown in Figure 5 .

A transaction data set **T: C O** is a mapping from the set of customers **C** to the set of offers **O**. A transaction data element therefore consists of a pair *(c, o)*.

A demographic data set **D: C** $\mathbf{2^P}$ is a mapping from the set of customers **C** to the set of groups of profiles $\mathbf{2^P}$. A demographic data element therefore consists of a pair *(c, {p})*

A pattern-of-interactions data set **I: C B** is a mapping from the set of customers **C** to the set of behaviors **B**. A pattern-of-interaction data element therefore consists of a pair *(c, {h})*.

While transactions and demographic data sets are familiar, patterns of interaction are unique to e-commerce. How should the behaviors be represented? It may be useful to merge all the behaviors of each customer into a single distribution; on the other hand, it may help to maintain each ‘session’ as a separate behavior. There are trade-offs in each approach. In the context of our model, we favor the former, since it is simpler, and probably less susceptible to noise. Therefore, the mapping **I** associates each customer with a maximum of one behavior; this behavior is generated by finding the frequency distribution of the hyperlinks in all the sequence of clicks the user has generated, and then normalizing it by the total number of clicks, to generate a size and a probability mass function.

[2] An alternate formulation is to make the preference orderings over the entire set, as is done in traditional models of utility; this poses possible implementation problems, however.

B. Characterizing factors

Customer groups of interest could be, for instance, the set of all customers who have bought televisions, or the set of all customers between the ages of 15 and 21, or the set of all customers who click on weather-related links 70% of the time. Clearly, any set of offers, behaviors or groups of profiles could constitute an interesting way of segmenting customers. Also, the objective of identifying factors is to associate them with offers, in order to make relevant offers to consumers. With this objective in mind, we define factors and rules as follows:

A factor f is a union of a set of subsets of **O**, **B** and $\mathbf{2^P}$, which defines a corresponding subset of **C**, through the mappings **T**, **D** and **I**. The set of all possible factors is therefore $\mathbf{2^O}$ $\mathbf{2^B}$ $\mathbf{2^P}$, with a little abuse of notation – we assume that if one is considering a factor with more than one group of profiles, we merge these groups into one larger group, which is an element of $\mathbf{2^P}$. We denote the set of relevant factors as **F** $\mathbf{2^O}$ $\mathbf{2^B}$ $\mathbf{2^P}$; **F** is defined as the set of factors that are part of the domain of a rule database **R**.

A rule database **R: F U** is a mapping from the set of relevant factors to the set of preference orderings **U**. A rule is therefore a pair *(f, u)*.

Each rule specifies a factor, which is a group of profiles, offers and/or behaviors, and a consequent preference ordering u. The preference orderings are generated by creating simple (single offer) rules, and then grouping the identical factors and ordering their associated offers by confidence, support, or some combination. For instance, mining a set of supermarket transactions may have yielded the two rules “If the person is male, under 25, lives in Manhattan, and has bought beer, he is likely to buy pretzels”, and “If the person is male, under 25, lives in Manhattan, and has bought beer, he is likely to buy diapers”. The former rule has a confidence of 85% and the latter a confidence of 80%. The factor f is ‘male, age 0 to 25, zip code 10001 through 10999, has bought beer’ and the preference ordering is *{pretzels, diapers}*. The preference ordering in this case is constructed based on the fact that the confidence of the rule associated with pretzels is higher. The true preferences of the individual customers who are male, under 25 live in Manhattan, and have bought beer, however, still remain uncertain.

Rules can be generated using a variety of algorithms, such as entropy reduction, as described in [1], [9],or genetic search [7]. In this way, we can identify the ‘relevant’ factors as well as the salient rules in the same step.

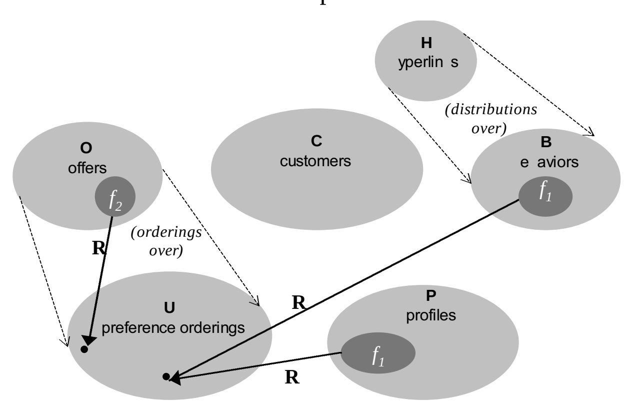


Fig. 6: Factors (f_1, f_2) and rules (R)

C. *Grouping behaviors through entropy minimization*

As stated earlier, we summarize behaviors using two measures: the size of *b*, which is the number of elements in *b*, and the distribution of *b*, which is the relative frequency of appearance of elements of H in *b*. Let *n(h)* is the number of times the element *h* **H** appears in *b*. The two measures are computed from web log files as follows:

$$\text{The } size \text{ of } b,\ N(b) = \sum_{h\ b} n(h)$$

$$\text{The } distribution \text{ of } b,\ p_b(h) = \frac{n(h)}{N(b)}$$

Apart from enabling us to better group behaviors, these measures are useful and succinct summaries of these behaviors; a representation that helps to increase the liquidity of **I**.

Our objective eventually is to capture the information contained in a customer's behavior; a natural measure of this information is the *entropy* of the distribution of the behavior. The entropy of the behavior *b* is:

$$E(b) = \sum_{h\ b} p_b(h) \log p_b(h)$$

The lower the entropy of a behavior, the more relevant (or informative) the behavior is likely to be, so long as the size of the behavior is sufficient. This may seem somewhat in contrast to the standard interpretation of entropy, where in the higher the entropy, the more the information required to code the distribution. A little thought, however, reveals that both notions are consistent.

We also characterize the distance between two behaviors in terms of the similarity between their distributions. Specifically, the distance *(b₁,b₂)* of behavior b_2 from behavior b_1 is:

$$(b_1, b_2) = \sum_{h\ (b_1\ \ b_2)} p_{b_1}(h) \log \frac{p_{b_1}(h)}{p_{b_2}(h)} \quad if\ b_1\ \ b_2\ \ ;$$

$$(b_1, b_2) = \quad if\ b_1\ \ b_2 = \ .$$

(.) is not a conventional measure of distance – it is not symmetric, and does not satisfy the triangle inequality. It is similar to *relative entropy*, but ignores the probability mass on elements of **H** that are in b_2 but not b_1. One can describe distribution distance measures of the same form that use functions other than the logarithm, and we are investigating the feasibility and optimality of other functional forms, in the specific context of describing similarity between customer behavior profiles. See, for instance, [2], [3], and [5].

D. *Identifying relevant behaviors*

We prescribe two strategies for identifying factors of relevance: *top-down* and *bottom-up*. The methods differ in their starting set of interest – whether it is the transaction/demographic data set, or the behavior data set. A broad description of these strategies is provided here – more detailed algorithms are available in [5].

1) The top-down strategy: The top-down strategy starts with the non-behavioral data (i.e., **T** and **D**), identifies relevant factors and rules, and then maps these factors onto the set of behaviors. First, rules are generated from **T** and **D** Subsequently, the non-behavioral factors (sets of offers and profiles) in each of these rules are mapped to sets of behaviors, or behavioral factors. Note that there is a set of (possibly diverse) behaviors corresponding to each non-behavioral factor. What this says is that a group of customers that have similar transactions/demographics may exhibit a number of behaviors. These behaviors are then clustered into more homogeneous sets. This also makes it easier to match an *unknown* incoming behavior with a known factor, by matching it with an aggregate measure, such as the composite distribution of the factor.

The top-down approach will work best in an environment where there are significant transaction and demographic data sets available. One is also likely to favor this approach initially, if there is no significant hyperlink metadata available, i.e., if there is no method of a priori classification of hyperlinks, making the set **H** very large.

2) The bottom-up strategy: The bottom-up strategy starts with the behavioral data (**B**), identifies behavior types directly from customer behaviors, maps these into transactions, and then discovers rules and factors based on these types. First, behavior types, or clusters of similar behaviors, are created. Behaviors are distributions over hyperlinks, and without clustering them into types, one is either faced with the task of dealing with a very large number of behaviors, or a very large, sparsely populated set of fields, one corresponding to each hyperlink. We believe it is more meaningful to represent behavior in terms of the aggregate set of hyperlinks, and not separate out the individual values of its constituent hyperlinks. This pre-processing step is necessary in order to capture the information that is aggregated in these behaviors, while reducing the number of different attribute values on which the generated rules are based on. The customers are mapped to their respective behavior types, and then, the behavior types are grouped into larger clusters, to create behavioral factors and the corresponding rules. It is likely that a large number of behavior types contain noisy behaviors, and are essentially uninformative. These types are excluded, and only the 'interesting' types are grouped into factors, depending on the associated customer transactions.

The bottom-up approach does not use demographic data. It is possible that the factors and the associated rules could be refined using this data set. However, this data is frequently unavailable to pure e-commerce firms, especially those who rely on their own transaction data, rather than buying data sets from a vendor. Besides, the recent privacy furor created by DoubleClick's use of Abacus' data is likely to discourage some e-commerce firms from using personally identifiable and other demographic data. Our approach is therefore ideally suited to these firms, who also benefit from the rich set of behavior data they have, relative to a more established company with a less central e-commerce focus.

V. Conclusions

Currently Websites are designed on a relatively ad hoc basis. There are some generally accepted ideas about style, but it is possible to organize content in a very large number of ways. How do we know that an existing interface is a good one, or that there aren't better alternatives? We usually don't. Running any business is a knowledge building exercise, involving learning by doing and on feedback. In electronic commerce, the knowledge building and feedback cycle becomes compressed. With every click, knowledge, and hence the potential to act, exists. There are therefore two basic questions that can be addressed on a real-time basis:

When do you know enough to act, i.e. to make an offer? This could be viewed not only as acting on knowledge, but also as initiating an experiment.

When do you know that you should change the current interface with existing or potential customers? This could be viewed as evaluating the results from experiments, making adaptations, and setting up the next set of experiments.

Our model for maximizing information liquidity is well suited as a model for designing and building adaptive web sites, which leverage the behavior factors generated to adapt to customer behavior. Over and above transaction and demographic data-based customization, which are currently used to some extent, we foresee two levels at which these web sites will become adaptive over time.

Real-time offer adaptation: This will happen every time a potentially new customer visits. When this new customer arrives at the web site and begins browsing, essentially, the firm is receiving a signal, which consists of a stream of clicks, from the customer. After every click, the firm can match the current behavior to the behavior factors that have already been generated through the learning exercise. As soon as there is a strong-enough match with a factor, the highest ranked offer in the rule associated with the factor is made to the customer. If this offer is ignored, then the next highest ranked offer is made, or a new factor is identified, depending on whether the new link traversed has altered the behavior substantially.

It is likely that this method can be enhanced by adaptive design of the web pages themselves. For instance, if the e-commerce firm has a partial match between a new behavior and a particular, well-known, behavior type, the interface could be adapted to reflect this match, for example, by altering the placement of hyperlinks in the web page according to the latter behavior. This is more compelling if one considers the fact that the web page itself may be the firm's offer, such as in a service organization that offers technical support.

Real-time factor refinement: If the firm adopts real-time offer adaptation, they can also choose to refine the preference orderings associated with their factors, depending on the outcomes of their offers to customers. Note that this does not involve high computational overhead, since the actual factors and rules are not being regenerated. On the other hand, there is simply an (Bayesian) update of the confidence level associated with the relevant offer in the relevant rule, and a consequent reordering of the preference orderings.

In general, the trade-off faced by an E-commerce firm is one of accuracy for timeliness. The longer the wait until the offer is generated, the more likely that the consumer will leave the site without seeing the offer. However, the sooner the type is decided and the offer made, the greater the chance of an inaccurately targeted offer. In practice it is possible to set these levels depending on the costs and benefits associated with opportunity costs and misclassification costs as outlined in [8] Our notion of maximizing information liquidity can improve the estimation and minimization of both these sets of costs, in this context.

To summarize, the information liquidity bottleneck represents one of the major problems that organizations must deal with in the space of electronic commerce. As the connectivity bottleneck is addressed, the content bottleneck will become the major hurdle to be overcome. We have been working with several key players in this area, ranging from established organizations with large amounts of historical information about large numbers of customers, to organizations that serve advertisements on the Web who have virtually no transaction data, but gobs of behavior data. The former type of organization is choosing the top-down approach, hoping to better exploit their existing transaction data before committing the resources to analyze behavior data. Their business focus is to first service existing customers better, and to exhaust business value propositions in this arena. The target variable is typically revenue per customer. The latter type of organization, in contrast, is focused on creating interesting patterns of interaction data, where the dependent variable is not revenue, but something like the "interest" in different types of offers, manifested in terms of variables such as click rates. Such data can be used to measure and improve the success of ad campaigns, or the relative stickiness of different web sites.

Over the longer run, both types of organizations must become more adaptive in their sales and post-sales process. As interaction volume increases over the Internet, it will become imperative for web sites to be more intelligent in terms of recognizing patterns of interaction, anticipating user needs, making offers, and adapting on the basis of outcomes. The model we have presented is a first step in this direction.

ACKNOWLEDGMENT

We thank Foster Provost and the participants of the IS department seminar at NYU for useful feedback and suggestions.

REFERENCES

[1] Breiman, L., Friedman, J., Olshen, R., and Stone, C., *Classification and Regression Trees*, Wadsworth, Monterey, CA, 1984.

[2] Cover, T. M. and Thomas, J. A. *Elements of Information Theory*. John Wiley and Sons, 1991.

[3] Csiszar, I. Information type measures of difference in probability distributions and indirect observations. *Studia Scientarium Mathematica Hungaria*, 2: 299-318, 1967.

[4] Dhar, V. and Stein, R. *Seven Methods for Transforming Corporate Data into Business Intelligence*. Prentice Hall, 1997.

[5] Dhar, V. and Sundararajan, A. Customer Interaction Patterns in Electronic Commerce: Maximizing Information Liquidity for Adaptive Decision Making. Working Paper IS99-017, Stern School of Business, New York University

[6] Kullback, S. and Leibler, R. A. On information and sufficiency. *Annals of Mathematical Statistics*, 22: 79-86, 1951.

[7] Packard, N., A Genetic Learning Algorithm, Tech Report, University of Illinois at Urbana Champaign, 1989.

[8] Provost, F., and Fawcett, T., Robust Classification for Imprecise Environments, Technical Report, Bell Atlantic, 500 Westchester Avenue, White Plains, NY, 1999.

[9] Quinlan, J., *Machine Learning and ID3*, Morgan Kauffman, Los Altos, 1992.

[10] Rosenberg, B., The arbitrage theory of capital asset pricing, *Journal of Economic Theory*, 13, 341-360, 1976.

[11] Teflian, M., Information Liquidity, *Time Zero White Paper*, 1999