Mining user session data to facilitate user interaction with a customer service knowledge base in RightNow Web

Doug Warner, J. Neal Richter, Stephen D. Durbin & Bikramjit Banerjee
RightNow Technologies
77 Discovery Dr.
Bozeman, MT 59718-9300
doug, nealr, sdurbin, bbanerjee @rightnowtech.com

ABSTRACT

RightNow Web is an integrated software package for web-based customer service that has, at its core, a database of answers to frequently asked questions (FAQs). One major design goal is to facilitate end-user interaction with this dynamic document collection, i.e. make it as easy and efficient as possible for users to browse the collection and locate desired information. To this end, we perform several types of analysis on the session tracking database that records user navigation histories. First, using both explicit and implicit measures of user satisfaction, we infer a "solved count" representing the average utility of an FAQ. Second, using the user navigation patterns we construct a link matrix representing connections between FAQs. The technique of building up the link matrix and using it to advise users on related information amounts to a form of the "swarm intelligence" method of finding optimal paths. Both solved count and the link matrix are continuously updated as users interact with the site; furthermore, they are periodically "aged" to emphasize recent activity. The synergistic combination of these techniques allows users to learn from the database in a more effective manner, as evidenced by usage statistics.

Categories and Subject Descriptors

H.2.8 [**Database Applications**]: Data Mining; H.3.5 [**Online Information Services**]; I.2.11 [**Distributed Artificial Intelligence**]: Intelligent agents; I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems

General Terms

Human Factors, Algorithms

Keywords

User session, user data, self-help, customer service, multi-agent system, collaborative filtering, clustering

1. INTRODUCTION

The design of software applications that rely heavily on interaction with users is often handicapped by lack of knowledge about how users behave. This fact has stimulated the development of applications that monitor user activity and attempt to learn from this activity to improve this application's effectiveness. These types of applications can be divided into two classes: (1) applications used primarily by a single person, whose idiosyncrasies can be learned over time, and (2) applications that are used by many people. In the latter case, the design choice is whether to adapt to better fit a prototypical user, or to attempt to identify users as belonging to pre-defined or learned categories with corresponding behavior. Assuming the user can be categorized with reasonable accuracy, the last approach offers the possibility of the most adapted or personalized response, and hence should be the most effective. To achieve this, the record of user interactions, or clickstream, must be mined for information about classes of user behavior.

RightNow Web (RNW) is a complete product for online customer service. As such, it contains many features which are beyond the scope of this paper. Here we will not address the entire realm of administrative functions, including such things as problem ticket tracking, workflow routing, and individual and company based contract management. Instead, we will focus on the end-user interactions with a knowledge base in the form of FAQs. In RNW, FAQs contain a title, question description, and question answer. Each FAQ is tagged with products or categories assigned by an administrator to help end users identify FAQs specific to a particular product or type of issue. Depending on the configuration, there may also exist custom fields associated with each FAQ as defined by an administrator. In addition to businesses, educational and government entities use RNW as a means of providing information to the public; one example is the U. S. Social Security Administration help site (`http://ssa-custhelp.ssa.gov/cgi-bin/ssa`). Thus, the end-user activities of browsing and searching the knowledge base are similar to those in many database-related applications.

RNW tracks users' activities as they search the knowledge base for answers. However, users are anonymous until they specifically submit a request for help. Therefore we have data for an anonymous individual visiting the site, but no way to track a particular visitor on subsequent visits. This restriction forces a reconceptualization of several interesting problems. Most importantly, conventional collaborative filtering [2], which uses a database of user preferences to predict preferences of a new user, is inadequate to handle relatively short and anonymous visits, since there is a low probability of identifying common interests between users. Instead, we use a novel, multi-faceted approach that also uses concepts from work in swarm intelligence [1]. The latter technique relies on the collective effect of many "agents" traversing a network and interacting with that network to build up a structure representing paths that are optimal in some sense. In our case, the agents are the human users "foraging" for information in the database of FAQs, which develops a link structure representing the paths with highest utility. In this paper we describe the nature and construction of the link structure that we employ as well as a usefulness rating for individual FAQs that is reminiscent of a collaborative filtering approach.

2. METHODS

2.1 Background

In RNW, the user interface from the end-user perspective has three main routes through which one may discover an answer. The default path contains a list of all publicly accessible FAQs on the site, the second path is a search on the list of publicly accessible FAQs, and the third path is a form with which to submit an unanswered question. The first two paths allow an end-user to find an answer unassisted, while the final path usually receives a human response. The effectiveness of the application is determined by the number of visitors who succeed in finding their answer through one of the first two methods without resorting to asking a customer service worker directly via the third method. Since users of the internet expect to be able to rapidly find an answer, several techniques to make the site adaptive are applied to the first two paths of the user interface to increase the likelihood of finding an answer in the smallest amount of time.

2.2 Determining FAQ Importance

The first route most end-users visit is the list of FAQs, as shown in Figure 1. A static list of FAQs would not fare well in the highly dynamic world of internet customer service. First, the contents of the FAQ list is continually changing. Companies add new products, change existing products, and drop support for products, sometimes on a daily basis. The list of FAQs must keep current with the company's product line. In addition, customer focus changes over time. As new products are introduced the customer demand for information on the new products will increase, and the demand for information on the older products will decline. In an attempt to present the FAQs in an order that keeps the FAQs in highest demand at the most visible point in the list, the ordering within the list must continually change.

RNW handles the changing customer emphasis on individual FAQs through both an explicit and an implicit method. Both methods manipulate a counter associated with each FAQ. The explicit adjustment comes from a questionnaire displayed with each FAQ which asks for a rating of helpfulness for the FAQ, as seen in the "How well did this answer your question" area in the middle of Figure 2. These ratings could be anything from a two-way choice of helpful versus unhelpful up to a five-way rating of the degree of usefulness. If the visitor rates an FAQ as unhelpful, the overall ranking for the FAQ drops. If, instead, the visitor rates the FAQ as helpful, the ranking for the FAQ increases. Since the list of FAQs is displayed sorted in descending order of usefulness based in part on explicit rankings, those FAQs with higher ratings can be thought to float to the top of the list, while those with lower ratings will sink lower in the list, as seen in Figure 1.

Unfortunately, only a small percentage of visitors actually explicitly rate FAQ usefulness. A random sampling of sites suggests that between 0.1% and 10% of visitors actually rate an FAQ. Implicit feedback was a method devised to augment the low response rate of visitors on the explicit feedback measure. With an implicit measure there is a guarantee of full participation at the potential cost of accuracy of the information. Thus, implicit feedback measures are always counted with less weight than explicit measures. The philosophy behind the implicit measures of usefulness is loosely based on a swarm intelligence or ant colony [1] approach. With our implicit measures we turn the tables from the standard approach with these techniques where each member of the swarm is a computer function searching for a local optimum. We use human visitors as the swarm members instead. To this end we must assume, instead of relying on explicit algorithmic instructions in the computer swarm model, that each human is searching the information space in a greedy, locally optimal way. For human visitors to an FAQ server, this means that we first assume each person comes to the site with a specific question to which they are attempting to find an answer. We further assume that once at the site the search approach they use is directed as opposed to random. In this case, a directed search means that the visitor will choose only those items which, based on the available information, appear to answer their question. Through the implicit measure, we increment our usefulness counter for each item an end-user visits. The last FAQ visited in a session increments the counter slightly more than the other FAQs visited in that session, under the assumption that the session ended with the successful discovery of the answer to the visitor's question. Thus, each visited FAQ in a session is rated higher than before the session began, and the terminal FAQ is rated higher than the non-terminal FAQs in that session.

The implicit and explicit rankings are tracked through the same mechanism, but the explicit rankings are given significantly more weight in the calculation. Implicit rankings are based on assumptions of usefulness, so they are not considered as reliable as the explicit rankings. Through the combination of the implicit and explicit manipulations to the usefulness counter, we can place an order on the list of FAQs that directly corresponds to the relative usefulness of each FAQ.

2.3 Links

At the same time as the implicit feedback is computed, links between incidents are also generated using the session history information. These links are generated by our human visitors, which we assume are operating in a non-random manner. Thus, a link is created between two FAQs or a usage counter of an existing link is incremented when an end-user visits two FAQs in sequence in the same session. Each link contains at least four characteristics, a "from" location, a "to" location, a usefulness counter, and a link type. The link type is not necessary in conceptualization, but is

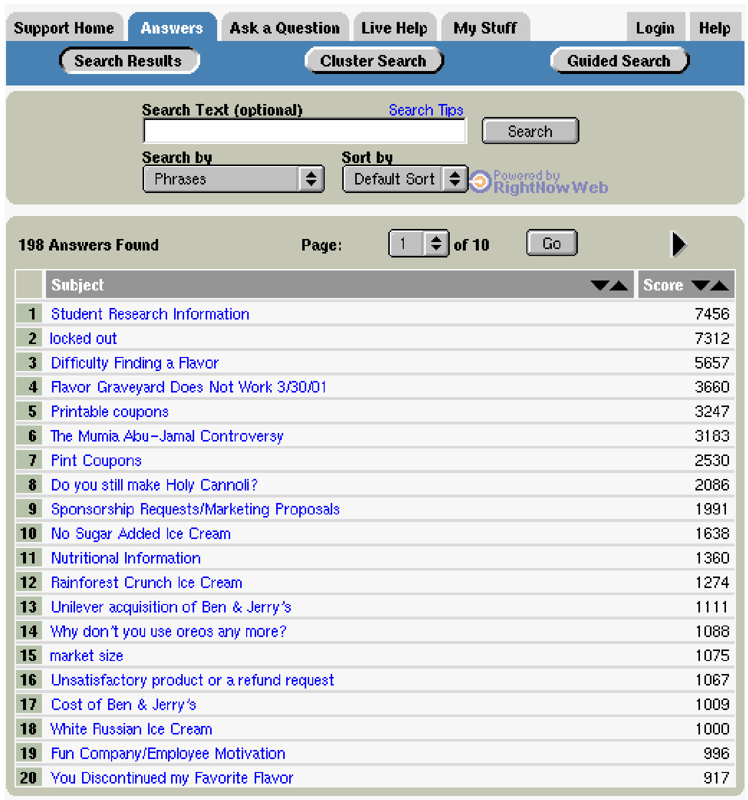

Figure 1: Screen capture of a RNW end–user page displaying the usage sorted list of FAQs.

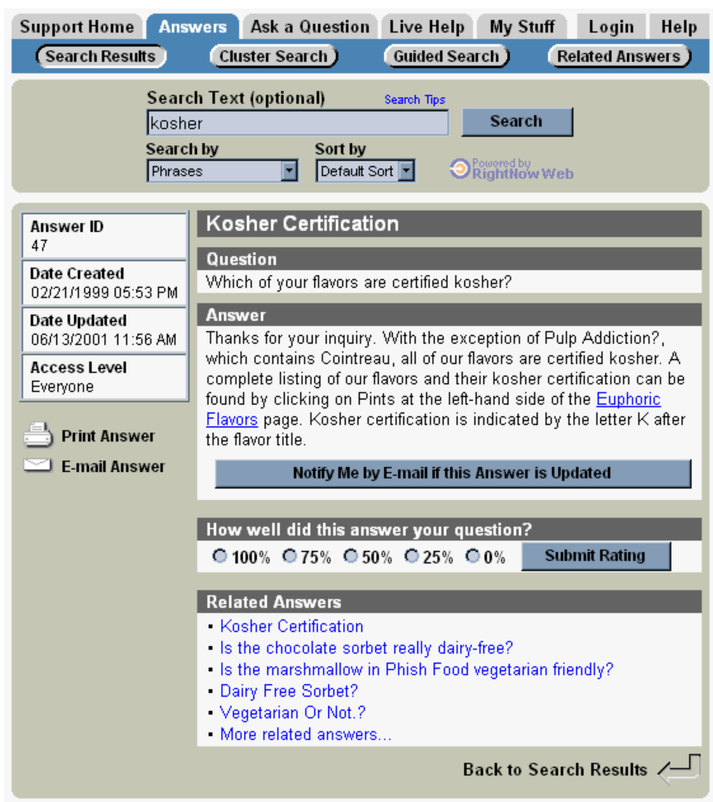

Figure 2: Screen capture of a RNW end–user page displaying the explicit solved count rating form and the user related document suggestions.

used within RNW to allow identification of special relationships between FAQs, such as similar product or category. In all cases of end-user traversals of at least two FAQs, at least one link is generated and additional links identifying special relations may also be created.

The links are used to suggest FAQs related to the last FAQ a visitor selected. Thus, an end result similar to entire session collaborative filtering is achieved without the analysis required to compare sessions between users or against user prototypes. Instead, small windows of the session, specifically the transition from one FAQ to another, are used to reflect data similarity instead of user similarity. In fact, the relationships are built with respect to the information content of the FAQs as interpreted by humans. Humans with similar interests are likely to interpret each FAQ in a similar way, resulting in the high accuracy of this technique, as seen in the "Related Answers" section in the lower part of Figure 2.

With this model one can easily see how a human visitor acting in a directed manner will identify related FAQ documents simply by moving from one FAQ to another. Subsequent visitors with similar questions can then use those suggestions of FAQ relatedness to guide their non-random traversal of the available information. With the usefulness counter on each link, all links from a given location can be ranked by their relative usefulness to other visitors, increasing the likelihood of each visitor finding the most relevant information.

A good example of human–generated relationships can be gleaned from the U. S. Social Security Administration site. As this paper is being written, the following FAQs are displayed as the top five items related to the FAQ entitled, "What are the disability requirements for an adult?": (1) "What kind of disability benefits does Social Security pay?" (2) "Where can I get a list of disabling impairments for Social Security Disability?" (3) "What is the difference between Social Security disability and SSI?" (4) "How long does it take to get notified of a decision about disability benefits?" (5) "How much can I earn and still receive Disability benefits?"

Perhaps the success of our modified collaborative filtering approach relies on a rather homogeneous set of visitors interacting with a more limited information source than is considered for the traditional implementations. Specifically, the U. S. Social Security site has only 575 visible items at the time of this writing. Yet this relatively small number of FAQs is accessed by millions of people. Those millions of people are visiting the site because of a specific interest in U. S. Social Security issues, not any of the millions of other topics available on the World Wide Web. Thus, the authors feel that the change of focus from the human–centric approach of traditional collaborative filtering to the data–centric approach described above has benefits in these, and many other, situations.

2.4 Data Aging

The ant colony optimization model contains a central idea that paths between points can evaporate. This pheromonal model ensures that paths must be frequently reinforced to remain relevant [1]. We have added this concept to our usefulness counters on both FAQs and links. This process, which we call "data aging," is done to enforce a preference for recently learned relationships and to keep the document usefulness rankings current with respect to recent user traffic patterns.

Conceptually the data aging process is rather simple. First, each time an end-user visits an FAQ, an access timestamp is updated on that FAQ. Periodically the access timestamps on the entire set of FAQs are analyzed, and those FAQs that have not had recent visitation have their usefulness rankings decreased. Both the amount of decrease and the frequency of aging are configurable depending on the particular demands of the site.

When an unreinforced usefulness ranking drops as a result of the data aging process, the FAQ will slide down the list of information presented to the user. Once an FAQ consistently drops below the first few pages of information, the reduction in visitation will result in a drop in the frequency of link reinforcement.

Data aging is an important consideration for keeping information current in real-world systems. For example, Pitney Bowes is a large supplier of postage metering equipment. Recently, when the U. S. postage rates increased, the RNW installation of Pitney Bowes had a sudden spike of questions related to this change. As one might guess, those FAQs that had previously appeared at the top of the usefulness rankings likely had no relevance to the sudden flurry of questions about the rate increase. If previously top-ranked items are no longer visited in favor of the newly preferred information, the new information will reach the top of the usefulness listing sooner if the old information is aged. In customer service applications, the faster information is provided to customers, the lower the cost of support becomes.

The system can also be configured to present the list of linked FAQs sorted by link strength instead of by solved count. As in the previous configuration, when a link is reinforced less frequently than other links it slides down the list of information presented. As the FAQ moves down the list it will be visited less frequently resulting in the usefulness ranking being reinforced less frequently.

It is obvious that there is a strong coupling between the reinforcement of the links and the reinforcement of the usefulness rankings. This coupling implies a linear relationship between the strength of the usefulness ranking and the total strengths of the links. If the number of links associated with a given FAQ is high, the effect of a particular link becoming weaker or disappearing altogether will have a low effect on the usefulness ranking of that document.

For documents with a low number of links the aging process can result in suppressed usefulness rankings and vice versa depending on the configuration of the interface. Intuitively, these types of documents are examples of isolated information. An RNW feature not covered in this paper is that, if desired, the aging process can be configured to change the status of isolated documents and signal administrators.

Maintainers of the knowledge base need to be aware to watch for these types of status changes and take appropriate action to either remove the document, alter it to be more related to stronger documents, or add new information similar to the given document in an attempt to build a group of related information.

In this system, groups of similar FAQs will tend to form "cliques" where all members of the group have strong links to most of the other members. These groups tend to be very stable while isolated FAQs tend to have trouble stay-

Table 1: Saving figures (varied units) for various customer companies

Company	Savings
Ben and Jerry's Ice Cream	99.5%
ClickRebates.com	> 90%
Remington Arms	90%
Schwinn	70%
Xerox	> 50%
Air Reserve Personnel Center	50% ˜$200K/mo
California Chamber of Commerce	50%
Quip!	45%
Bissell, Inc.	30%
Commission Junction	30%
Sanyo Fisher Service Corporation	30%
University of South Florida	20%
Big Planet	10-20% ˜$100K/mo
Social Security Administration	> ˜$1.2M/mo
Pitney Bowes	> $100K/mo
Mindspring	7,500 hours/mo

ing stable with respect to the average usefulness and link strengths. Of course this is not always the case; an isolated document can remain popular as a result of its inherent utility to users. In general, the system performs best when all documents have a number of very similar documents to facilitate the establishment of stable groups.

2.5 Jump Starting the Knowledge base

As currently described, the system suffers from a bootstrapping problem. The process of end-user generated links and usefulness rankings is slow to initialize and reach a representative state. We have addressed this issue by using a statistical text clustering algorithm highly tied to the details of this system which outputs a links structure which attempts to give the system a good initial condition.

This issue of bootstrapping also exists for newly added groups of documents, such as when a new product line is added. The clustering algorithm can be run periodically after the aging process is completed. This gives new FAQs a starting condition in proportion to the current state of the existing knowledge base structures.

Using the clustering algorithm also has an additional benefit. The typical user interface setup shows only the titles of the FAQs. As a result, end-users select and implicitly reinforce FAQs based only on their view of the usefulness of the title. Users that explicitly rate the document's usefulness can correct this, but in practice most do not. This creates the need for documents to have a very good summary title. Alternatively, the interface is configurable to display more than just the FAQ titles, allowing end-users to provide more accurate link information. Since the clustering algorithm operates on the entire set of words in each document, it will create links between documents even if their titles are not necessarily similar.

3. RESULTS

Based on self–reported case studies from within one to six months of initial implementation, users of RNW have experienced between a 10% and 99.5% reduction in customer support load (See Table 1). A few of the companies report

Table 2: Growth and pre-growth saving figures for various customer companies

Company	Company Growth	Overall Savings
Military.com	300%	10%
ScholarOne	100%	17%
Specialized	10%	10%
Air Canada	dramatic	60%

a dollar savings or a number of hours saved, instead of a reduction rate. In some notable cases RNW customers reported an estimated savings of more than $1.2 million per month as a result of reduced phone support volume. A recent study by Forrester Research [5] claims an industry wide average of $33 per phone support call, so any reduction in phone support volume can reduce costs quickly. Table 2 shows self–reports of overall savings in spite of an increase in customer base over the same time period. Businesses closely track the number of support requests as well as the average cost to respond to each request as part and parcel of their operating expenses. Thus, while these numbers are self–reports, they are likely to be fairly accurate.

These numbers refer only to the percentage of end-users who find their answer within the entire FAQ list. The comparison group are, effectively, the same end-users before there existed a dynamic FAQ list. Unfortunately this data does not break down the usefulness of the individual approaches outlined above. However, from these numbers it is apparent that the suite of data mining and knowledge discovery approaches as described certainly aids a large percentage of the general public to find solutions to their problems faster than if the suite were unavailable.

A more complete analysis performed by Doculabs, Inc. [8] on 3.7 million service requests to 202 companies in the first quarter of 2001 shows an 86.9% self–service rate with RNW. According to Doculabs, that equates to a savings of more than $100 million quarterly. The breakdown of self-service by industry is shown in Table 3.

4. DISCUSSION

4.1 Current And Future Improvements

Our objective in using a links matrix and solved counts was to meet the user's demands as quickly as possible. However, from the discussions in the previous sections, we know that these rely heavily on the actual usage patterns. Explicit solved count techniques require user feedback, which, when added to the user–dependence of the links matrix, could bias these metrics undesirably. The key to achieving user–independence would be to "understand" the similarities of text FAQs using relatively simple knowledge of natural language. Grouping similar FAQs using clustering techniques would not only enhance user interaction, but also provide a semantic tool to automate the process of answering user queries.

In the release of RNW currently under development, we have made many modifications which are the topic of another paper. First, we incorporate a heavily modified variation on the incremental and hierarchical clustering technique, BIRCH [9]. This algorithm allows us to organize the FAQs in a tree-structure where each internal node of the tree stands as a representative summary of the FAQs in the

Table 3: Self–service by industry, over 200 companies analyzed, 3.7 million customer visits

Industry	Visitation	Self–service
General Equipment	342,728	98.79%
Manufacturing	22,784	97.85%
Education	8,400	96.23%
Entertainment/Media	113,047	95.91%
Financial Services	40,574	95.14%
Contract Manufacturers	77,838	94.60%
Utility/Energy	19,035	94.11%
ISP/Hosting	147,671	94.06%
IT Solutions Providers	53,804	93.91%
Computer Software	449,402	93.90%
Dot Coms	267,346	92.40%
Medical Products	17,892	91.89%
Professional Services	24,862	91.38%
Insurance	40,921	91.36%
Automotive	3,801	90.19%
Retail/Catalog	44,145	86.07%
Consumer Products	1,044,199	84.46%
Computer Hardware	101,209	84.43%
Government	108,955	84.08%
Travel/Hospitality	27,099	82.99%
Association/Nonprofit	14,620	81.04%
Telecommunications	809,320	75.02%
Overall Total	**3,779,652**	**86.90%**

subtree below. This tree will be used to guide the user to the desired FAQs, and given the usually shallow BIRCH trees, this should be a matter of a few clicks. We have tried various distance measures to compute the similarities of FAQs and found the quality of results depends on the distance algorithm. The metrics proposed in [9] take into account the intra-cluster coalescence, but ignore reducing inter-cluster coupling. The cluster-utility measure proposed in [7] is more complete in that sense, and is also easily integrable in the main BIRCH framework. BIRCH also offers iterative methods to reduce data-order-dependence.

Natural language word–scoring, stemming, and part–of–speech–tagging [3] as well as disambiguation techniques similar to [4] serve as advanced tools for better match–criteria. We discovered an immense increase in both overall speed of clustering and accuracy of resulting clusters when these approaches are used to identify cluster features. Part of our modifications to the BIRCH approach is the integration of a simple rule-learner for context-sensitive text categorization building upon the above natural language techniques. Similar to Cohen's classifier approach [6], we have created classifier to reflect the hierarchical structure prescribed by BIRCH, with a redistribution of documents at various levels in the tree. This hierarchical classifier thus effectively identifies documents that can act as representatives of all those documents that reside in the same subtree.

As an extension to the techniques covered in this paper, the authors are also refining an approach to identifying emotional content in questions submitted to the site. While this approach does not aid the end-user in finding answers, this knowledge discovery technique aids the customer support worker to prioritize their communications and identify those that need immediate attention or referral as well as tailor their responses to best suit the situation. Previously we relied on word–scoring and word–stemming for this, but we have recently added the part–of–speech–tagging techniques to our repertoire. This technique complements the other knowledge discovery and data mining approaches discussed in this paper to make for a more complete software application.

5. CONCLUSION

Various knowledge discovery techniques applied to user sessions have proven themselves very useful in RightNow Web, an online customer service tool used by approximately 1,200 companies, schools, and government organizations. With the knowledge discovery techniques in RightNow Web, human support load is significantly reduced by allowing end-users to easily find answers to their questions. Several new techniques are identified for future versions of the product, hopefully allowing even more time to spend providing a personal touch for those customers requiring more involved solutions to their problems.

6. REFERENCES

[1] E. Bonabeau and G. Théraulaz. Swarm Smarts. *Scientific American*, pages 73 – 79, March 2000.

[2] J. S. Breese, D. Heckerman, and C. Kadie. Empirical Analysis of Predictive Algorithms for Collaborative Filtering. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, 1998.

[3] E. Brill. Some Advance in Transformation Based Part of Speech Tagging. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, 1994.

[4] E. Brill and P. Resnik. A rule-based approach to prepositional phrase attachment disambiguation. In *Proceedings of the 15th International Conference on Computational Linguistics*, Kyoto, Japan, 1994.

[5] B. Chatham, D. M. Cooperstein, and A. A. Reinhard. Managing customer profitability, 2000.

[6] W. W. Cohen. Fast effective rule induction. In *Proceedings of the Twelfth International Conference on Machine Learning*, California, 1995.

[7] D. Fisher. Iterative optimization and simplification of hierarchical clusterings. In *Technical report, TR CS-95-01*. Vanderbilt University, 1995.

[8] J. Watson, G. Donnelly, and J. Shehab. The Self–Service Index Report: Why Web–Based Self–Service is the ROI Sweet–Spot of CRM. In *http://www.doculabs.com/*. Doculabs, Inc., 2001.

[9] T. Zhang, R. Ramakrishnan, and M. Livny. BIRCH: An Efficient Data Clustering Method for Very Large Databases. In *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*, pages 103 – 114, Montreal, Canada, 1996.