Appeared in: Proceedings of the Conference of the Canadian Society for Computational Studies of Intelligence, Lecture Notes in Computer Science, Vol 1822, Springer, pp. 164-175.

Interact: A Staged Approach to Customer Service Automation

Yannick Lallement[1] and Mark S. Fox[1,2]

[1] Novator Systems, 444 Yonge street, Toronto, ON M4Y 1B5, Canada
yannick@novator.com
[2] Dept. of Mechanical and Industrial Engineering, University of Toronto, 4 Taddle Creek Road, Toronto, ON M5S 3G9, Canada
msf@novator.com

Abstract. Electronic commerce websites often have trouble keeping up with the large amount of customer-service related email they receive. One way to alleviate the problem is to automate responding to that email as much as possible. Many customer messages are in essence frequently asked questions, for which it is easy to provide a reply. This paper explores a staged approach to message understanding: an incoming message is first classified in a specific category. If the category of the message corresponds to a specific frequently asked question, the answer is provided to the customer. If the category corresponds to a more complex question, a finer understanding of the message is attempted. Messages are categorized by a combination of Bayes classifier and regular expressions, that significantly improves performance compared to a simple Bayes classifier. A first version of the system is installed on the FTD website (Florist Transworld Delivery). It can classify more than half of the customer messages, with 2.3% error; three quarters of the categorized messages are frequently asked questions, and receive an automatic response.

1 Introduction

Recent studies by several media research companies have underlined the poor performance of many electronic commerce websites in terms of customer service. A study of 125 "top websites" by Jupiter Communications [2] indicates that 42% either never responded to their customers' needs, or took more than 5 days to respond, or do not offer email options to their customers. In another study of the 100 largest companies websites, Brightware [1] found that only 15% answered a simple email query ("what is your headquarters address?") within three hours; 36% could not be emailed from their website; 10% never answered.

Both studies indicate than about 50% of the websites fail to provide satisfying customer service. The most likely reason for this high failure rate is that many customer service departments are not ready to deal with unexpectedly high quantities of customer email. As the number of internet users and on-line buyers continues to grow, e-commerce companies have to take action to solve this customer service problem.

H. Hamilton and Q. Yang (Eds.): Canadian AI 2000, LNAI 1822, pp. 164–175, 2000.

Currently, there are several types of tools available to help companies deal with customer service interaction. Some tools make it possible for a customer to talk live to a customer service representative over the web; other tools help a pool of representatives deal with incoming email (by offering a centralized queue, pre-defined responses for frequently asked questions, and monitoring facilities). These tools are mostly ports to the internet of phone-based customer service technologies. While they may be useful in improving the customer service efficiency for the company, and experience for the customer, they fail to take into account a major aspect of the internet: the possibility of automation.

Because customer input over the internet comes as text instead of voice, the interaction between customer service and customer can be partially automated. Automation will help the retail company decrease its customer service costs, and will also improve customer experience by providing immediate response. The response will be immediate even in burst demand situations, such as the demand for flowers prior to mothers day or toys prior to Christmas. Because response will be provided by a program, it will also always be consistent: customers asking identical questions will receive the same reply. A well designed system will give only relevant feedback (when the customer input is understood with a high enough degree of certainty) and refer to a human representative in case of uncertainty.

In this article, we present the first version of the automated customer service software Interact. Section 2 addresses the architecture of the software; section 3 presents the text classification technology; section 4 discusses the implementation of Interact on the Florist Transworld Delivery e-commerce website, *www.ftd.com*.

2 The Interact Staged Approach

2.1 Different Types of Customer Messages

Incoming customer service messages can be divided into two types: messages that can be answered by a pre-defined reply (called type I), and messages that need a specific answer (called type II).

Examples of type I messages are *frequently asked questions* like the ones found on the large number of FAQ-lists available on the internet: for example "Do you deliver on Sundays?", or "What is your return policy?". Comments (e.g. "Your site is great") or advice (e.g. "You should have more choice") are also type I messages. These messages form a large portion of the incoming customer email. For example, over 75% of the order-form suggestions on the FTD website are of type I (measured on 6000 messages). Automating only the answering of type I messages would therefore be a major benefit.

Type II messages need to be answered with a specific answer, depending on each message. Examples of such messages may be "I forgot my password, please send it back to me", or "Give me my AAA 10% discount", or "Is that shirt available in blue?".

2.2 Categories of Messages

Incoming messages can be grouped by categories corresponding to the topics they deal with; for example messages like "great site", "I like your site", "keep it up" will be grouped in the category *compliment*. Other examples of categories could be *have more choice, password problem, discount request*. Each application of Interact to a specific website will have a specific set of categories, although evidence suggests that a large number of categories (e.g. the ones mentionned above) are common to many electronic commerce applications. We are currently working on a customer-service message ontology, that will help us define categories for new applications.

Some of these categories correspond to type I messages (e.g. *compliment, have more choice, lower your prices...*) and others correspond to type II messages (e.g. *password problem, discount request, question about product...*).

2.3 The Interact Architecture

When a new message comes in, Interact's first operation is to *classify* the message into a specific category. Knowing the category of a message provides only a crude understanding of the message, but in many cases this is enough to answer the message: if the message falls into a type I category, then Interact simply sends the corresponding pre-defined reply back to the customer.

If the message falls into a type II category, it is necessary to *extract more information* from the message before being able to answer it meaningfully. For example, if a message falling in the *discount request* category says "please give me the 10% AAA discount, my membership number is XXX", the system needs to extract the type of discount ("AAA"), the value ("10%") and the membership number ("XXX"). When these data are retrieved, the system can look up databases and business rules, decide whether the discount is applicable or not, and use a reply template to compose the appropriate answer.

If the information is not present in the message, or if it can't be retrieved, then the system needs to collect that information from the customer. For example, if the incoming message says "please give me the 10% AAA discount", the system needs to answer "Please provide us with your AAA membership number" to the customer; in brief, the system must start a conversation with the customer.

These three possibilities (message is of type I; message is of type II and all necessary information is present; message is of type II and some necessary information is missing) dictate the Interact architecture shown on figure 1. The three message processing modules (classification, information extraction and conversation management) deal with messages of growing complexity; each one is called only if the previous one could not permit the generation of a meaningful answer to the message.

In the current version of Interact, only the classification module is implemented; therefore only type I messages receive a reply. Type II and unknown messages are forwarded to a customer service representative (type II messages can be routed to specific representatives according to their category). In the next section, we examine the classification module in more details.

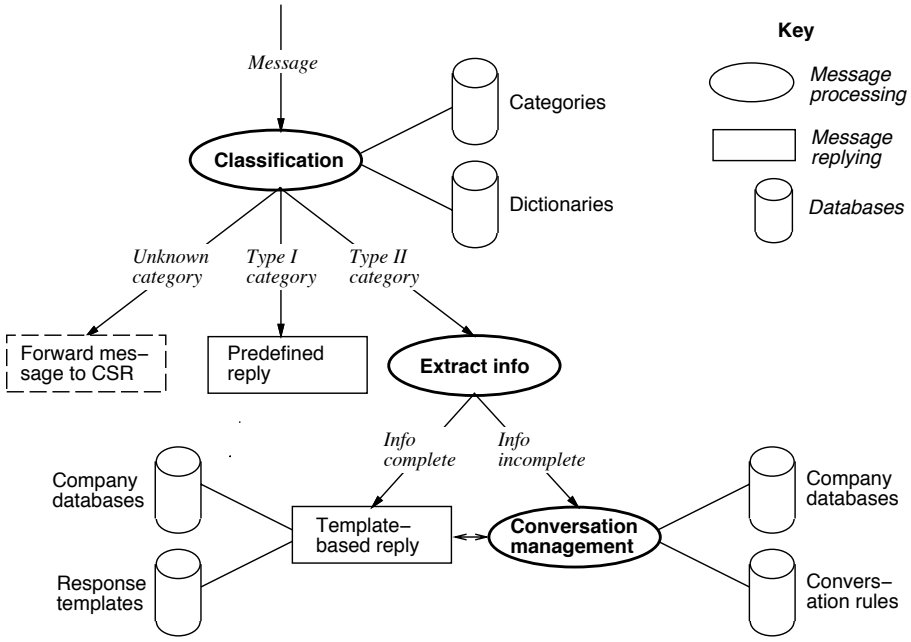

Fig. 1. Automated customer service architecture

3 Classification Technology

Interact's classification technology is based on another duality observed in the message categories; some categories, such as *compliment*, can be expressed by a wide variety of words and phrases. On the contrary, other categories, such as *discount request*, or *send catalog*, concern precise topics and often contain a few specific words. This duality prompted the two-level classifier architecture we have defined, that combines two technologies: naive Bayes classification [6] and regular expressions.

3.1 Naive Bayes Classification

Naive Bayes classification considered one of the most efficient means of text classification (see [4], p. 180). Indeed, it proved to be the best for our application compared to other techniques we experimented with, for example ensemble of oblique decision trees [5, 3] and least squares fit mapping [7]. This section briefly introduces the naive Bayes classifier and the threshold computation mechanism we added to it.

Naive Bayes Algorithm A naive Bayes classifier determines the category c of a document composed of n words $w_1, w_2 ... w_n$ (in no particular order) according to the following equation:

$$c = \underset{c_j \epsilon C}{argmax}\, P(c_j) \prod_i P(w_i|c_j) \quad (1)$$

where C is the set of categories, $P(c_j)$ is the probability of the category c_j and $P(w_i|c_j)$ the probability that the word w_i appears in documents of category c_j. These values can be estimated from a training set composed of a series of documents categorized by a human. When a message is classified, the Bayes classifier returns both the category and the probability that the message belongs to that category (see [4] for more details).

Threshold Learning A crucial aspect of our application is that the rate of false positives (messages classified in the wrong category) must be very low, even if that means fewer messages will be classified at all; giving the customer irrelevant feedback is worse than giving no feedback at all. In conventional text classification terms [8], *precision* is more important for us than *recall*.

In order to enforce a specific maximum rate of false positives, a threshold can be set so that the Bayes classifier's output is accepted only if its probability is high enough.

The simplest way to compute the threshold is as follows: to begin, the maximum rate of admissible false positives is chosen by the administrator of Interact (e.g., 3%). Next, the threshold is set to zero (meaning the classification proposed by the classifier will always be accepted). Then the classifier is tested on a test corpus, and the rate of false positives on this corpus is computed. If this rate is too high, the value of the threshold is raised by a fixed, small increment, and the test is run again. The threshold is set to the desired value when the rate of false positives produced by the classifier on the test set is below the maximum rate.

This threshold computation method has one drawback: it is efficient only if the messages are equally distributed among the categories. The probability that an example is in a specific category is limited by the frequency of that category in the training set (see Equation 1). Therefore, if a message belongs to a rare category, the probability associated to its classification will be low, and possibly lower than the global threshold computed by the simple method. To alleviate this problem, we refined the threshold computation method by defining one threshold per category rather than a unique threshold; the multiple threshold computation algorithm is similar to the simple one, except that the threshold *for a specific category* is raised as long as too many false positive are produced *in this specific category*. The improvement obtained with this method compared to the simple method is illustrated in section 4.

3.2 Regular Expressions

Regular expressions can detect specific patterns in a sentence; for example, the expression `send[^\.]*catalog` will match messages containing the word "send", followed by the word "catalog" but with no period between both (i.e. "send" and

"catalog" are in the same sentence). Unlike simple keywords, regular expressions can take into account word order, word combinations, synonyms, punctuation, etc. By examining a set of messages of a given category and looking for common patterns, the administrator can design one or more regular expressions for that category.

Regular expressions are well adapted for specific messages that tend to be expressed in a limited number of ways; for example customers asking to be sent the catalog of the company. The administrator must make sure, during the regular expression development process, that the proposed expressions do not generate many false positives. This is achieved by defining very specific regular expressions, like the one shown above, that are not likely to match unwanted messages. This would be much more difficult to achieve using only keywords. The improvement obtained by adding regular expressions to the naked Bayes classifier is discussed in section 4.

3.3 Interact's Classifier

Interact's classifier is depicted in figure 2. An incoming message is first classified by the Bayes classifier. If no category is recognized, the message is classified by the regular expressions. We chose this architecture because regular expressions do not have the threshold "safety mechanism", so we trust the Bayes classifier more, and because the Bayes classifier can identify a larger number of messages than regular expressions. Performance of the classifier on a specific application is given in the next section.

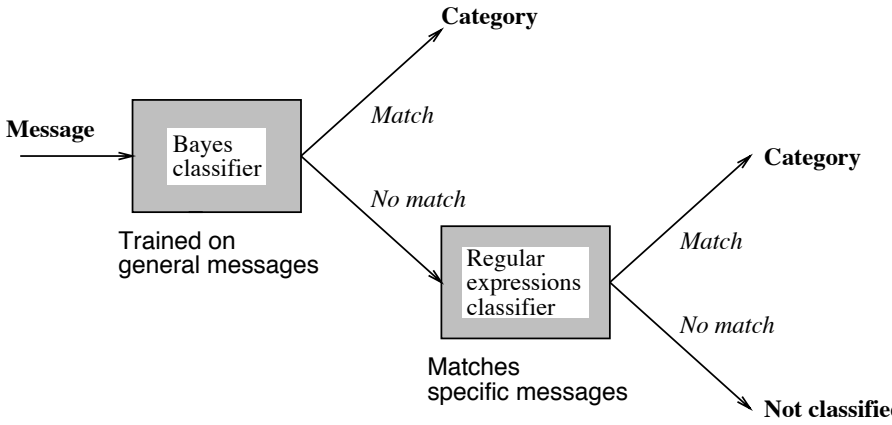

Fig. 2. Interact classifier

4 Interact on the FTD Website

4.1 Context

A first version of Interact, that handles only type I messages, is installed on the *www.ftd.com* website (Florist Transworld Delivery). On the FTD website, customers can leave a message (in the form of free text) in the suggestion field on the order form (see Figure 3).

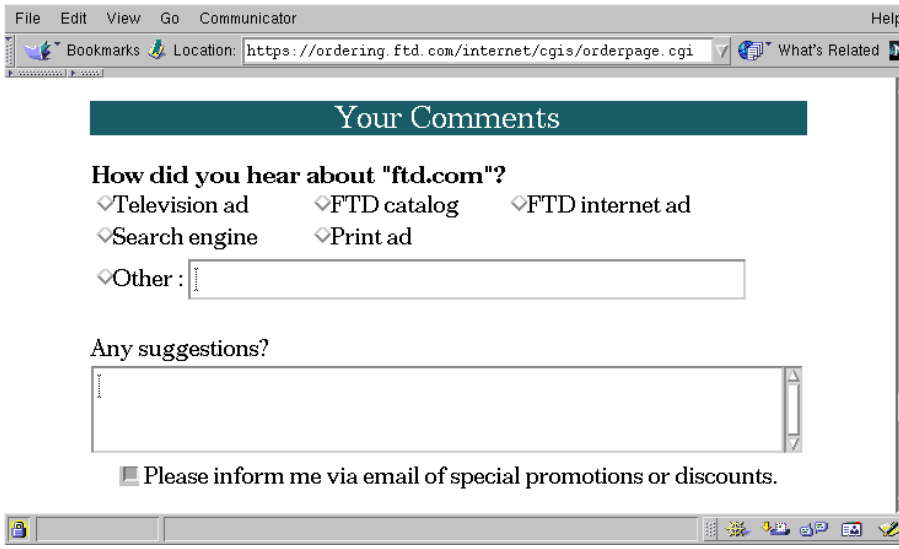

Fig. 3. Portion of the FTD order form

4.2 Categories of Messages

Interact currently classifies messages in one of 31 different categories. Approximately 75% of the classified messages fall into a type I category; Interact answers them by writing a pre-defined message on the order confirmation form, shown to the customer after the order has been recorded. Table 1 provides some examples of type I categories, customer messages and the corresponding pre-defined replies.

4.3 Classification Performance

In this specific application, every customer is prompted for suggestions, and many customers explicitly say they do not have any. These "negative" messages don't need a reply, and are easy to classify. On the opposite, we wish to give

Table 1. Examples of type I categories, with actual customer comments and pre-defined replies on the FTD website. Underlined text denotes a web link.

please confirm delivery	• Send back confirmation of delivery and time of delivery on email. • Can someone please send me an e-mail confirming reciept of this order.....Thanks • e-mail verification of my order would be appreciated.
	We appreciate your comments. At this time, we do not automatically confirm delivery of orders. If you have concerns about your order, please complete our Order Inquiries Form.
lower prices	• Lower the price of Roses. They are good but not that good. • I think you could offer lower prices for local deliveries. • Pretty expensive-lower your prices.
	We appreciate your comments. FTD makes every effort to keep our online prices competitive and to offer fresh beautiful flowers at market value. In a recent review, we found that our prices for the same and similar products were less than or equal to those of our major competitors on the Web.
message box is too short	• Increase the number of characters you can type for the message...it is way too short! • Enable messages of greater than 150 characters. • The message box doesn't allow for enough words. You need more room for a 'personalize' message.
	We appreciate your comments. Our florists' gift cards are not much larger than a standard business card. Your message will be hand-written on this card. For this reason, we need to limit the number of characters in your message. The next time you visit our site, you may try our Quotable Sentiments(sm) library for a message that will easily fit on the gift card.

a reply to every positive message (actual suggestion or comment). We give the performance measurements of Interact both on the whole set of messages, and on the set of positive messages only.

The Bayes classifier was trained on 6000 customer messages received consecutively and classified by a human. Multiple thresholds were then computed to limit the false positive rate to 3%. The classifier uses 25 regular expressions that were hand-crafted by looking for patterns in the same set of 6000 messages. The performance figures were measured on a test corpus of 795 messages classified by a human; all the messages were previously unseen by the Bayes classifier, and the messages were not used to help craft the regular expressions. Out of those 795 messages, 600 were positive.

Interact classified 63% of the whole set of messages in one of 31 different categories with a false positive rate of 2%. Interact classified 51.3% of the positive messages in 30 different categories (the previous one except the *no comment* category) with a false positive rate of 2.3%. Figure 4 shows the following figures for the all-messages and positive messages only cases:

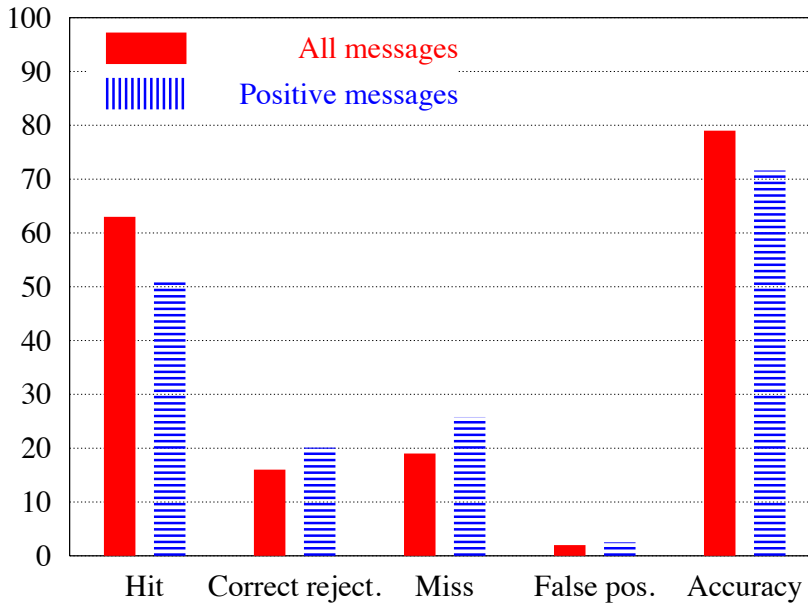

Fig. 4. Performance (in percent) of the Interact classifier

- Hits: percentage of messages that were classified in the same category by the human and the classifier.
- Correct rejections: percentage of messages that were labelled as "not classified" by the human and the classifier.

- Misses: percentage of messages that were classified in a category by the human, but labelled "not classified" by the classifier.
- False positives: percentage of messages that were classified in two different categories by the human and the classifier.
- Accuracy: hits + correct rejections: percentage of messages on which the classifier and the human agreed.

4.4 Comparison with Partial Classifier Performances

To illustrate the interest of adding the multiple threshold mechanism and the regular expressions to the basic Bayes classifier, table 2 gives the performance of the classifier in four different cases:

- Bayes classifier only, with one global threshold (basic Bayes classifier, B1T)
- Bayes classifier only, with multiple thresholds (BMT)
- Bayes classifier plus regular expressions, one global threshold (B1T+R)
- Bayes classifier plus regular expressions, multiple thresholds (Interact classifier, same as section 4.3, BMT+R)

Table 2. Performance of different versions of the classifier, with one or multiple thresholds (B1T and BMT), with or without regular expressions (R).

	B1T	BMT	B1T+R	BMT+R
Hits	41.1	46.3	48.6	51.3
Correct rejections	20.3	20.3	20.1	20.1
Misses	36.3	31.1	29.1	26.4
False positives	2.1	2.1	2.3	2.3
Accuracy	61.4	66.6	68.7	71.4

In each case, the Bayes classifier was trained with the same 6000 messages as in section 4.3, and, when used, the regular expressions were the same 25 as in section 4.3. In each case, the performance figures were computed on the same set of 600 positive messages as in section 4.3.

Table 2 shows that the two modifications: multiple thresholds rather than single thresholds (BMT), and addition of regular expressions (B1T+R) improve the hit rate of the basic Bayes classifier (B1T). The combination of multiple thresholds and regular expressions (BMT+R) also improves the hit rate of each of them (BMT and B1T+R) separately.

In the Interact classifier case (BMT+R), the Bayes classifier is responsible for 90% of the hits and 92% of the false positives, and the regular expressions are responsible for the remaining 10% of the hits (5 percentage points) and 8% of the false positives (0.2 percentage points). This shows the validity of our classifier compared to a simple Bayes classifier: the adjunction of regular expressions to the Bayes classifier significantly improves the hit rate, while degrading the false positive rate only slightly.

5 Conclusion and Work in Progress

The Interact system illustrates the benefits of a staged approach to natural language processing. A relatively straightforward technique to set up, classification, often gives enough information on messages to let the system answer them. If more information is needed, other natural language understanding techniques can be applied.

The installation of the first version of Interact on FTD offers three important benefits:

- Providing customers with feedback and answering their concern immediately when possible enhances their shopping experience and demonstrates their value to FTD.
- Interact diminishes the load of the customer service representatives who take care of the order-form suggestions by a factor of about two; this is especially important in burst-demand situations, to help the company keep a good responsiveness.
- Interact keeps statistics on the number of suggestions in each category and their evolution, providing FTD with valuable customer feedback in a summarized and easy to understand form.

A web-based interface has been designed that lets the administrator define regular expressions, build test and training corpus, train and test the system, and define the categories and the automated responses.

We are currently working on the information extraction and conversation management modules of Interact, to enable it to handle messages that cannot be answered by a pre-defined reply. We are also continuing work on the classifier, since this module is at the root of the system; we are in particular exploring methods to help the administrator define the regular expressions, or to automatically build them.

Our longer term plans include applying artificial intelligence technologies to more aspects of electronic commerce. Future versions of Interact will be pervasive throughout a website, providing each customer with a personalized interaction based on the customer profile, his or her previous interactions with the company, and the business process with which he or she is currently involved in.

Acknowledgements

This project is supported in part by the National Research Council (NRC), Industrial Research Assistance Program (IRAP), contract number 357643.

References

[1] Brightware. Survey finds only 15% of top u.s. firms answer e-mails in 3 hours. Press release available at www.brightware.com /news/press/1999_1_14_F100_Survey.html, January 1999.

[2] Jupiter Communications. 42 percent of web sites fail at customer service. Press release available at www.jup.com/jupiter/press/releases/1998/1109a.html, November 1998.

[3] Y. Lallement. A hierarchical ensemble of decision trees applied to classifying data from a psychological experiment. In *Eleventh International FLAIRS conference*, Sanibel Island, Florida, 1998. AAAI Press.

[4] T. Mitchell. *Machine Learning*. McGraw Hill, 1997.

[5] S. K. Murthy, S. Kasif, and S. Salzberg. A system for induction of oblique decision trees. *Journal of artificial intelligence research*, 2:1–32, 1994.

[6] K. Nigam, A. McCallum, S. Thrun, and T. M. Mitchell. Learning to classify text from labeled and unlabeled documents. In *AAAI/IAAI*, 1998.

[7] Y. Yang and C. G. Chute. An application of least squares fit mapping to text information retrieval. In *Proceedings of the 16th annual international ACM SIGIR conference on research and development in information retrieval*, pages 281–290, New York, 1993. ACM.

[8] Y. Yang and C. G. Chute. An example-based mapping method for text categorization and retrieval. *ACM transactions on information systems*, 12(3):252–277, 1994.