我们从电影论坛上抽取了一些二货的对话，然后试图用这个来训练一个二货机器人

Extracting Chatbot Knowledge from Online Discussion Forums*

Jizhou Huang[1], Ming Zhou[2], Dan Yang[1]
[1]School of Software Engineering, Chongqing University, Chongqing, China, 400044
{jizhouhuang, dyang}@cqu.edu.cn
[2]Microsoft Research Asia, 5F Sigma Center, No.49 Zhichun Road, Haidian, Bejing, China, 100080
mingzhou@microsoft.com

Abstract

This paper presents a novel approach for extracting high-quality <thread-title, reply> pairs as chat knowledge from online discussion forums so as to efficiently support the construction of a chatbot for a certain domain. Given a forum, the high-quality <thread-title, reply> pairs are extracted using a cascaded framework. First, the replies logically relevant to the thread title of the root message are extracted with an SVM classifier from all the replies, based on correlations such as structure and content. Then, the extracted <thread-title, reply> pairs are ranked with a ranking SVM based on their content qualities. Finally, the Top-*N* <thread-title, reply> pairs are selected as chatbot knowledge. Results from experiments conducted within a movie forum show the proposed approach is effective.

1 Introduction

A chatbot is a conversational agent that interacts with users in a certain domain or on a certain topic with natural language sentences. Normally, a chatbot works by a user asking a question or making a comment, with the chatbot answering the question, or making a comment, or initiating a new topic. Many chatbots have been deployed on the Internet for the purpose of seeking information, site guidance, FAQ answering, and so on, in a strictly limited domain. Existing famous chatbot systems include ELIZA [Weizenbaum, 1966], PARRY [Colby, 1973] and ALICE.[1] Most existing chatbots consist of dialog management modules to control the conversation process and chatbot knowledge bases to response to user input. Typical implementation of chatbot knowledge bases contains a set of templates that match user inputs and generate responses. Templates currently used in chatbots, however, are hand coded. Therefore, the construction of chatbot knowledge bases is time consuming, and difficult to adapt to new domains.

An online discussion forum is a web community that allows people to discuss common topics, exchange ideas, and share information in a certain domain, such as sports, movies, and so on. Creating threads and posting replies are major user behaviors in forum discussions. Large repositories of archived threads and reply records in online discussion forums contain a great deal of human knowledge on many topics. In addition to rich information, the reply styles from authors are diverse. We believe that high-quality replies of a thread, if mined, could be of great value to the construction of a chatbot for certain domains.

In this paper, we propose a novel approach for extracting high-quality <thread-title, reply> pairs from online discussion forums to supplement chatbot knowledge base. Given a forum, the high-quality <thread-title, reply> pairs are extracted using a cascaded framework. First, the replies logically relevant to the thread title of the root message are extracted with an SVM classifier from all the replies, based on correlations such as structure and content. Then, the extracted <thread-title, reply> pairs are ranked with a ranking SVM based on their content qualities. Finally, the Top-*N* <thread-title, reply> pairs are selected as chatbot knowledge.

The rest of this paper is organized as follows. Important related work is introduced in Section 2. Section 3 outlines the characteristics of online discussion forums with the explanations of the challenges of extracting stable <thread-title, reply> pairs. Section 4 presents our proposed cascaded framework. Experimental results are reported in Section 5. Section 6 presents comparison of our approach with other related work. The conclusion and the future work are provided in Section 7.

2 Related Work

By "chatbot knowledge extraction" throughout this paper, we mean extracting the pairs of <input, response> from online resources.

Based on our study of the literature, there is no published work describing the use of online communities like forums for automatic chatbot knowledge acquisition. Existing work on automatic chatbot knowledge acquisition is mainly based on human annotated datasets, such as the work by Shawar and Atwell [2003] and Tarau and Figa [2004]. Their approaches are helpful to construct commonsense knowledge

* This work was finished while the first author was visiting Microsoft Research Asia during Feb.2005-Mar.2006 as a component of the project of AskBill Chatbot led by Dr. Ming Zhou.

[1] http://www.alicebot.org/

for chatbots, but are not capable of extracting knowledge for specific domains.

Notably, there is some work on knowledge extraction from web online communities to support QA and summarization. Nishimura *et al*. [2005] develop a knowledge base for a QA system that answers type "how" questions. Shrestha and McKeown [2004] present a method to detect <question, answer> pairs in an email conversation for the task of email summarization. Zhou and Hovy [2005] describe a summarization system for technical chats and emails about Linux kernel. These researchers' approaches utilize the characteristics of their corpora and are best fit for their specific tasks, but they limit each of their corpora and tasks, so they cannot directly transform their methods to our chatbot knowledge extraction approach.

3 Our Approach

An online discussion forum is a type of online asynchronous communication system. A forum normally consists of several discussion sections. Each discussion section focuses on a specific discussion theme and includes many threads. People can initiate new discussions by creating threads, or ask (answer) questions by posting questions (replies) to an existing section. In a section, threads are listed in chronological order. Within a thread, information such as thread title, thread starter, and number of replies are presented. The thread title is the title of the root message posted by the thread starter to initiate discussion. One can access a thread from the thread list and see the replies listed in chronological order, with the information of the authors and posting times.

Compared with other types of web communities such as newsgroups, online discussion forums are better suited for chatbot knowledge extraction for the following reasons:

1. In a thread within a forum, the root message and its following up replies can be viewed as <input, response> pairs, with same structure of chat template of a chatbot.
2. There is popular, rich, and live information in an online discussion forum.
3. Diverse opinions and various expressions on a topic in an online discussion forum are useful to extract diverse <input, response> pairs for chatbots.

Due to technical limitations of current chatbots in handling dialogue management, we think that pairs of <input, response> for a chatbot should be context independent, which means that the understanding inputs and responses will not rely on the previous <input, response>.

However, because of the nature of a forum, it is difficult to extract high-quality <input, response> pairs that meet chatbot requirements:

1. Replies are often short, elliptical, and irregular, and full of spelling, usage, and grammar mistakes which results in noisy text.
2. Not all of replies are related to root messages.
3. A reply may be separated in time or place from the reply to which it responds, leading to a fragmented conversational structure. Thus, adjacent replies might be semantically unrelated.
4. There is no evidence to reveal who has replied to which reply unless the participants have quoted the entire entries or parts of a previously posted reply to preserve context [Eklundh, 1998].

To overcome these sorts of difficulties, lexical and structural information from different replies within threads are analyzed in our experiments, as well as user behaviors in discussions.

Therefore, to extract valid pairs of <input, response> from a forum, we first need to extract relevant replies to initial root messages. In this process, replies that are relevant to the previous replies rather than to the initial root message are ignored and the replies logically directly relevant to the thread title are extracted. The replies to the initial root message, in spite of being relevant, may have different qualities. To select high-quality replies, a ranking SVM is employed to rank the replies. Finally, the pairs of the title of the root message and the extracted Top-*N* replies are used as the chatbot knowledge.

4 Cascaded Hybrid Model

An input online discussion forum F contains discussion sections $s_1,s_2,...,s_k$. A section consists of T threads $t_1,t_2,...,t_u$. Each thread t is a sequence of replies $t=\{r_0,r_1,r_2,...,r_n\}$, where r_0 is the root message posted by the thread starter and r_i is the ith $(i \geq 1)$ reply. A reply r is posted by a participant p at a specific moment m with content c. A thread t can be modeled as a sequence of triplets:

$$t = \{r_0, r_1, r_2, ..., r_n\}$$
$$= \{(p_0, m_0, c_0), (p_1, m_1, c_1), (p_2, m_2, c_2), ..., (p_n, m_n, c_n)\}$$

We define an RR as a direct reply r_j $(j \geq 1)$ to the root message r_0 where r_j is not correlated with the other reply $r_{j'}$ $(j' \geq 1 \wedge j' \neq j)$ in the thread.

Therefore, chatbot knowledge (CK) can be viewed as the pairs of <input, response> that fulfill the following constraints:

$$CK = \{(input, response)\}$$
$$= \{(thread\text{-}title, high\text{-}quality\ RR)\}$$

A thread title is used to model the user input of a chatbot and RRs of this thread are used to model the chatbot responses. The high-quality pairs of <thread-title, RR> will be selected as chatbot knowledge.

A high-quality pair of <thread-title, RR> for the chatbot should meet the following requirements:

1. The thread-title is meaningful and popular.
2. The RR provides descriptive, informative and trustworthy content to the root message.
3. The RR has high readability, neatly short and concise expressive style, clear structure.
4. The RR is attractive and can capture chatter's interest.
5. Both thread-title and RR should have NO intemperate sentiment, no obscene words and exclusive personal information.
6. Both thread-title and RR should have proper length.

In this paper, identifying the qualified thread-title is not our focus. Instead, we focus on selecting qualified RR. Figure 1 illustrates the structure of the cascaded model. The

first pass (on the left-hand side) applies an SVM classifier to the candidate *RR* to identify the *RR* of a thread. Then the second pass (in the middle) filters out the *RR* that contains intemperate sentiment, obscene words and personal information with a predefined keyword list. The *RR* which is longer than a predefined length is also filtered out. Finally the *RR* ranking module (on the right-hand side) is used to extract the descriptive, informative and trustworthy replies to the root message.

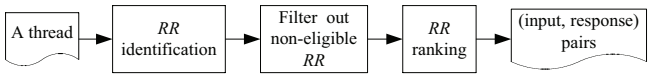

Figure 1. Structure of Cascaded Model.

4.1 *RR* Identification

The task of *RR* identification can be viewed as a binary classification problem of distinguishing *RR* from *non-RR*. Our approach is to assign a candidate reply $r_i (i \geq 1)$ an appropriate class y (+1 if it is an *RR*, -1 or not). Here Support Vector Machines (SVMs) is selected as the classification model because of its robustness to over-fitting and high performance [Sebastiani, 2002].

SVMlight [Joachims, 1999] is used as the SVM toolkit for training and testing. Table 1 lists the feature set to identify *RR* for a pair of <thread-title, reply>.

1	**Structural features**	
	1-1	Does this reply quote root message?
	1-2	Does this reply quote other replies?
	1-3	Is this reply posted by the thread starter?
	1-4	# of replies between same author's previous and current reply
2	**Content features**	
	2-1	# of words
	2-2	# of content words of this reply
	2-3	# of overlapping words between thread-title and reply
	2-4	# of overlapping content words between thread-title and reply
	2-5	Ratio of overlapping words
	2-6	Ratio of overlapping content words between thread-title and reply
	2-7	# of domain words of this reply
	2-8	Does this reply contain other participants' registered nicknames in forum?

Table 1. Features for *RR* Classifier.

In our research, both structural and content features are selected. In structural features, quotation maintains context coherence and indicates the relevance between the current reply and the quoted root message or reply, as discussed in [Eklundh and Macdonald, 1994; Eklundh, 1998]. Two quotation features (feature 1-1 and feature 1-2) are employed in our classifier. Feature 1-1 indicates that the current reply quoting the root message is relevant to the root message. On the contrary, feature 1-2 indicates the current reply might be irrelevant to the root message because it quotes other replies. We use features 1-3 and 1-4 based on the observation of behaviors of posting replies in forums. The thread starter, when participants reply to the starter's thread, usually adds new comments to the replies. Therefore, the added replies gradually diverge from the original root message. If a participant wants to supplement or clarify his previous reply, he can add a new reply. Therefore, the participant's new reply is often the supporting reason or argument to his previous reply if they are close to each other.

Content features include the features about the number of words and the number of content words in the current reply, the overlapping words and content words between the root message and the current reply. In our work, words that do not appear in the stop word list[2] are considered as content words. Feature 2-7 estimates the specialization of the current reply by the number of domain specific terms. To simplify the identification of domain specific terms, we simply extract words as domain specific words if they do not appear in a commonly used lexicon (consists of 73,555 English words). Feature 2-8 estimates a reply's pertinence to other replies, because some participants might insert the registered nicknames of other participants and sometimes add clue words such as "P.S." to explicitly correlate their replies with certain participants.

4.2 *RR* Ranking

Further, after the *RRs* have been identified, non-eligible *RRs* are filtered out with a keyword list with 33 obscenities, 62 personal information terms (terms beginning with "my", such as my wife, my child) and 17 forum specific terms (such as Tomatometer, Rotten Tomato, etc.). Replies with more than *N* words are eliminated because people may become bored in chatbot scenarios if the response is too long. In our experiments, *N* is set as 50 based on our observation.[3]

We analyzed the resulting *RRs* set of 4.1. For some *RRs*, there is certain noise left from the previous pass, while for other *RRs*, there are too many *RRs* with varied qualities. Therefore, the task of *RR* ranking is to select the high-quality *RRs*. The ranking SVM [Joachims, 2002] is employed to train the ranking function using the feature set in Table 2.

The number of being quoted of a reply is selected as a feature (feature 1-1) because a reply is likely to be widely quoted within a thread as it is popular or the subject of debate. In other words, the more times a reply is quoted, the higher quality it may have. This motivates us to extract the quoted number of all the other replies posted by an author within a thread (feature 2-9) and throughout the forum (feature 2-10).

We also take "author reputation" into account when assessing the quality of a reply. The motivation is that if an author has a good reputation, his reply is more likely to be reliable. We use the author behavior related features to assess his "reputation." An earlier work investigates the relationship between a reader's selection of a reply and the author of this reply, and found that some of the features raised from authors' behavior over time, correlate to how

[2] http://dvl.dtic.mil/stop_list.html

[3] 50 is the average length of 1,200 chatbot responses which preferred by three chatters through sample experiments.

likely a reader is to choose to read a reply from an author [Fiore *et al.*, 2002]. Features 2-1 to 2-7 are author behavior related features in the forum. Feature 2-8 models how many people have chosen to read the threads or replies of an author in the forum by using the measurement of the influence of participants. This is described in detail in [Matsumura *et al.*, 2002].

1	**Feature of the number of being quoted**
1-1	# of quotations of this reply within the current thread
2	**Features from the author of a reply**
2-1	# of threads the author starts in the forum
2-2	# of replies the author posts to others' threads in the forum
2-3	The average length of the author's replies in the forum
2-4	The longevity of participation
2-5	# of the author's threads that get no replies in the forum
2-6	# of replies the author's threads get in the forum
2-7	# of threads the author is involved in the forum
2-8	The author's total influence in the forum
2-9	# of quotations of the replies that are posted by the author in current thread
2-10	# of quotations of all the replies that are posted by the author in the forum

Table 2. Features for *RR* Ranking.

5 Experimental Results

5.1 Data for Experiments

In our experiments, the *Rotten Tomatoes forum*[4] is used as test data. It is one of the most popular online discussion forums for movies and video games. The *Rotten Tomatoes forum* discussion archive is selected because each thread and its replies are posted by movie fans, amateur and professional filmmakers, film critics, moviegoers, or movie producers. This makes the threads and replies more heterogeneous, diverse, and informative.

For research purposes, the discussion records are collected by crawling the *Rotten Tomatoes Forum* over the time period from November 11, 1999 to June 15, 2005. The downloaded collection contains 1,767,083 replies from 65,420 threads posted by 12,973 distinctive participants, so there are, on average, 27.0 replies per thread, 136.2 replies per participant, and 5.0 threads per participant. The number of thread titles in question form is 16,306 (24.93%) and in statement form is 49,114 (75.07%). We use part of these discussion records in our experiments.

5.2 *RR* Identification

To build the training and testing dataset, we randomly selected and manually tagged 53 threads from the *Rotten Tomatoes* movie forum, in which the number of replies was between 10 (min) and 125 (max). There were 3,065 replies in 53 threads, i.e., 57.83 replies per thread on average. Three human experts were hired to manually identify the relevance of the replies to the thread-title in each thread. Experts annotated each reply with one of the three labels: a) *RR*, b) *non-RR* and c) *Unsure*. Replies that received two or three *RR* labels were regarded as *RR*, replies with two or three *non-RR* labels were regarded as *non-RR*. All the others were regarded as *Unsure*.

After the labeling process, we found out that 1,719 replies (56.08%) were *RR*, 1,336 replies (43.59%) were *non-RR*, 10 (0.33%) were *Unsure*. We then removed 10 unsure replies and 60 replies with no words. We randomly selected 35 threads for training (including 1,954 replies) and 18 threads for testing (including 1,041 replies). Our baseline system used the number of replies between the root message and the responding reply [Zhou and Hovy, 2005] as the feature to classify *RRs*.

Table 3 provides the performance using SVM with the feature set described in Table 1.

Feature set	Precision	Recall	F-score
Baseline	73.24%	66.86%	69.90%
Structural	89.47%	92.29%	90.86%
Content	71.80%	85.86%	78.20%
All	90.48%	92.29%	91.38%

Table 3. *RR* Identification Result.

With only the structural features, the precision, recall and f-score reached 89.47%, 92.29%, and 90.86%. Content features, when used alone, the precision, recall and f-score are low. But after adding content features to structural features, the precision improved by 1.01% while recall stayed the same. This indicates that content features help to improve precision.

Root message
Title: *Recommend Some Westerns For Me?*
Description: *And none of that John Wayne sh*t.*

1. *The Wild Bunch It's kickass is what it is.*
2. *Once Upon a Time in the West*
3. *Does Dances With Wolves count as a western? Doesn't matter, I'd still recommend it.*
4. *White Comanche This masterpiece stars ……*
5. *Here's some I'm sure nobody else ……*
6. *for Dances with Wolves.*
7. *: understands he's a minority here: ……*
8. *Open Range is really good. Regardless ……*
9. *One of the best films I've ever seen.*
10. *The Good the Bad and the Ugly ……*

Figure 2. A Sample of *RRs*.

Figure 2 presents some identified *RRs* listed in chronological order for the root message with the title, "*Recommend Some Westerns For Me?*" and description for the title, "*And none of that John Wayne sh*t.*".

5.3 Extract High-quality *RR*

To train the ranking SVM model, an annotated dataset was required. After the non-eligible *RRs* were filtered out from

[4] http://www.rottentomatoes.com/vine/

the identified *RRs*, three annotators labeled all of the remaining *RRs* with three different quality ratings. The ratings and their descriptions are listed in Table 4.

Rating	Description
Fascinating	This reply is informative and interesting, and it is suitable for a chatbot
Acceptable	The reply is just so-so but tolerable
Unsuitable	This reply is bad and not suitable for a chatbot

Table 4. *RR* Rating Labels.

After the labeling process, there were 568 (71.81%) *fascinating RRs*, 48 (6.07%) *acceptable RRs*, and 175 (22.12%) *unsuitable RRs* in the 791 *RRs* of the 35 training threads. And in the 511 *RRs* of the 18 test threads, there were 369 (72.21%) *fascinating RRs*, 25 (4.89%) *acceptable RRs*, and 117 (22.90%) *unsuitable RRs*.

We used mean average precision (MAP) as the metric to evaluate *RR* ranking. MAP is defined as the mean of average precision over a set of queries and average precision ($AvgP_i$) for a query q_i is defined as:

$$AvgP_i = \sum_{j=1}^{M} \frac{p(j) * pos(j)}{number\ of\ \ positive\ instances}$$

where j is the rank, M is the number of instances retrieved, $pos(j)$ is a binary function to indicate whether the instance in the rank j is positive (relevant), and $p(j)$ is the precision at the given cut-off rank j.

The baseline ranked the *RRs* of each thread by their chronological order. Our ranking function with the feature set in Table 2 achieved high performance (MAP score is 86.50%) compared with the baseline (MAP score is 82.33%). We also tried content features such as the cosine similarity between an *RR* and the root message, and found that they could not help to improve the ranking performance. The MAP score was reduced to 85.23% when we added the cosine similarity feature to our feature set.

5.4 Chat Knowledge Extraction with Proper *N* Setting

The chat knowledge extraction task requires that the extracted *RRs* should have high quality and high precision. After we got the ranked *RRs* of each thread, the Top-*N RRs* were selected as chatbot responses. The baseline system just selected Top-*N RRs* ranked in chronological order. Figure 3 shows the comparison of the performances of our approach and the baseline system at different settings of *N*.

Figure 4 shows the Top-*N* (*N*=6, *N* can be adjusted to get proper equilibrium between quantity and quality of *RRs* when extracting chatbot knowledge) *RRs* after ranking the *RRs* in Figure 2. As an instance, we uniformly extracted Top-6 high-quality *RRs* from each thread. Altogether 108 <thread-title, reply> pairs were generated from 18 threads. Among these extracted pairs, there were 97 fascinating pairs and 11 wrong pairs, which showed that 89.81% of the extracted chatbot knowledge was correct.

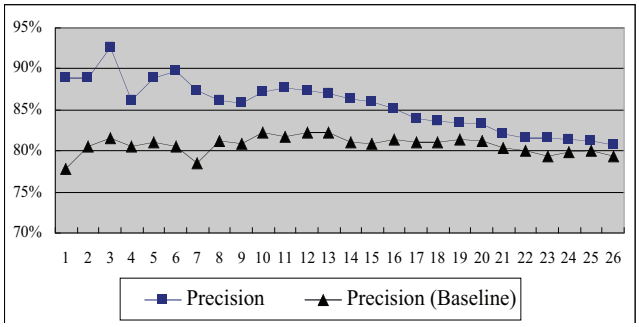

Figure 3. Precision at Different *N*.

Input: *Recommend Some Westerns For Me?*
Chatbot responses:
6. for Dances with Wolves.
11. Young Guns! & Young Guns 2!
2. Once Upon a Time in the West
9. One of the best films I've ever seen.
27. I second the dollars trilogy and also Big Hand ……
18. Classic Anthony Mann Westerns: The Man from Laramie (1955) ……

Figure 4. Top-6 *RRs*.

6 Comparison with Related Work

Previous works have utilized different datasets for knowledge acquisition for different applications. Shrestha and McKeown [2004] use an email corpus. Zhou and Hovy [2005] use Internet Relay Chat and use clustering to model multiple sub-topics within a chat log. Our work is the first to explore using the online discussion forums to extract chatbot knowledge. Since the discussions in a forum are presented in an organized fashion within each thread in which users tend to respond to and comment on specific topics, we only need to identify the *RRs* for each thread. Hence, the clustering becomes unnecessary. Furthermore, a thread can be viewed as <input, response> pairs, with the same structure of chat template of a chatbot, making a forum better suited for the chatbot knowledge extraction task.

The use of thread title as input means that we must identify relevant replies to the root message (*RRs*), much like finding adjacent pairs (APs) in [Zhou and Hovy, 2005] but for the root message. They utilize AP to identify initiating and responding correspondence in a chat log since there are multiple sub-topics within a chat log, while we use *RR* to identify relevant response to the thread-title. Similarly, we apply an SVM classifier to identify *RRs* but use more effective structural features. Furthermore, we select high-quality *RRs* with a ranking function.

Xi *et al.* [2004] use a ranking function to select the most relevant messages to user queries in newsgroup searches, and in which the author feature is proved not effective. In our work, the author feature also proves not effective in identifying relevant replies but it is proved effective in selecting high-quality *RRs* in *RR* ranking. This is because irrelevant replies are removed in the first pass, making author features more salient in the remaining *RRs*. This also indi-

cates that the cascaded framework outperforms the flat model by optimally employing different features at different passes.

7 Conclusions and Future Work

We have presented an effective approach to extract <thread-title, reply> pairs as knowledge of a chatbot for a new domain. Our contribution can be summarized as follows:

1. Perhaps for the first time, our work proposes using online discussion forums to extract chatbot knowledge.
2. A cascaded framework is designed to extract the high-quality <thread-title, reply> pairs as chabot knowledge from forums. It can optimally use different features in different passes, making the extracted chatbot knowledge of higher quality.
3. We show through experiments that structural features are the most effective features in identifying *RR* and author features are the most effective features in identifying high-quality *RR*.

Compared with manual knowledge construction methods, our approach is more efficient in building a specific domain chatbot. In our experiment with a movie forum domain, 11,147 <thread-title, reply> pairs were extracted from 2,000 threads within two minutes. It is simply not feasible to have human experts encode a knowledge base of such size.

As future work, we plan to improve the qualities of the extracted *RRs*. The method of selecting valid thread titles and extracting completed sentences from the extracted *RRs* is an area for exploration. In addition, we are also interested in extracting questions from threads so that <question, reply> pairs can be used to support QA style chat.

We currently feed the extracted <thread-title, reply> directly into the chatbot knowledge base. But there is much room to improve quality in the future. For example, we can generalize the chat templates by clustering similar topics and grouping similar replies, and improve coherence among the consecutive chat replies by understanding the styles of replies.

Acknowledgements

The authors are grateful to Dr. Cheng Niu, Zhihao Li for their valuable suggestions on the draft of this paper. We also thank Dwight for his assistance to polish the English. We wish to thank Litian Tao, Hao Su and Shiqi Zhao for their assistance to annotate the experimental data.

References

[Colby, 1973] K. M. Colby. Simulation of Belief systems. In *Schank and Colby (Eds.) Computer Models of Thought and Language,* pp.251-286, 1973.

[Eklundh and Macdonald, 1994] K. S. Eklundh and C. Macdonald. The Use of Quoting to Preserve Context in Electronic Mail Dialogues. In *IEEE Transactions on Professional Communication,* 37(4):197-202, 1994.

[Eklundh, 1998] K. S. Eklundh. To quote or not to quote: setting the context for computer-mediated dialogues. In *S. Herring (Ed.), Computer-Mediated Conversation.* Cresskill, NJ:Hampton Press, 1998.

[Fiore *et al.*, 2002] A. T. Fiore, S. Leetiernan and M. A. Smith. Observed Behavior and Perceived Value of Authors in Usenet Newsgroups: Bridging the Gap. In *Proceedings of the CHI 2002 Conference on Human Factors in Computing Systems,* pp.323-330, 2002.

[Joachims, 1999] T. Joachims. Making large-scale SVM learning practical. *Advances in Kernel Methods - Support Vector Learning,* MIT-Press, 1999.

[Joachims, 2002] T. Joachims. Optimizing Search Engines Using Clickthrough Data. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD),* pp.133-142, 2002.

[Matsumura *et al.*, 2002] N. Matsumura, Y. Ohsawa and M. Ishizuka. Profiling of Participants in Online-Community. *Chance Discovery Workshop on the Seventh Pacific Rim International Conference on Artificial Intelligence (PRICAI),* pp.45-50, 2002.

[Nishimura *et al.*, 2005] R. Nishimura, Y. Watanabe and Y. Okada. A Question Answer System Based on Confirmed Knowledge Developed by Using Mails Posted to a Mailing List. In *Proceedings of the IJCNLP 2005,* pp.31-36, 2005.

[Sebastiani, 2002] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys,* 34(1):1-47, 2002.

[Shawar and Atwell, 2003] B. A. Shawar and E. Atwell. Machine Learning from dialogue corpora to generate chatbots. In *Expert Update journal,* 6(3):25-29, 2003.

[Shrestha and McKeown, 2004] L. Shrestha and K. McKeown. Detection of question-answer pairs in email conversations. In *Proceedings of Coling 2004,* pp.889-895, 2004.

[Tarau and Figa, 2004] P. Tarau and E. Figa. Knowledge-based conversational Agents and Virtual Story-telling. In *Proceedings 2004 ACM Symposium on Applied Computing,* 1:39-44, 2004.

[Weizenbaum, 1966] J. Weizenbaum. ELIZA - A Computer Program for the Study of Natural Language Communication between Man and Machine. *Communications of the ACM,* 9(1):36-45, 1966.

[Xi *et al.*, 2004] W. Xi, J. Lind and E. Brill. Learning Effective Ranking Functions for Newsgroup Search. In *Proceedings of SIGIR 2004,* pp.394-401, 2004.

[Zhou and Hovy, 2005] L. Zhou and E. Hovy. Digesting Virtual "Geek" Culture: The Summarization of Technical Internet Relay Chats. In *Proceedings of ACL 2005,* pp.298-305, 2005.