Mining Experts in Technical Online Forums

Fernando Das Neves, Fernando Wasylyszyn

Snoop Consulting, Sarmiento 1320 Piso 8, Buenos Aires, Argentina
{fernando.dasneves, fernando.wasylyszyn}@snoopconsulting.com

Abstract. Many organizations use or host discussion lists, in the form of online forums and email lists. Analyzing the content of those discussion lists is an effective solution to the task of expert finding, since experts tend to participate often by giving advice, and receive the best feedback. We present a novel method to identify positive comments that helps to identify experts by combining author statistics with polarity mining. Our method is able to distinguish experts from flamers and other people that simply participates frequently in discussions. We demonstrate the validity of our approach by evaluating it with an online discussion forum in Spanish.

Keywords: expert finding, discussions, polarity mining, machine learning

1 Introduction

Any organization, be it a company, government or educational institution, faces the challenge of identifying their best people in any area of knowledge. Finding experts who have already dealt with similar problems and can validate the feasibility of an approach has often a profound effect on the outcome of a project, including the cost of carrying out the project. Organizations that can identify their experts can implement proper retainment procedures and increase their understanding of each member´s value in the context of knowledge management.

Expert finding is not a problem in stable and small organizations; experts are within reach and can be consulted. As the number of people in an organization grows and different working groups become disconnected, new members often do not personally meet the old experts or hear about them and their expertise, and old member do not know about what new knowledge the new members bring with them. The most direct method to identify experts, as carried out in traditional knowledge management initiatives like the ones described in [1][2], is to manually identify experts by analyzing content, interaction and peer recognition. This approach is resource intensive, needing people dedicated to compile (make explicit), filter, review, and disseminate valuable contributions from experts.

Given these shortcomings of manual expert identification, various automated approaches were devised to find experts by mining an organization´s corpora. Most of them aim at identifying experts by analyzing document authoring and the social network formed by communications among organization members, particularly via email. However, up to our knowledge there has not been any previous attempts before

this paper to improve expert finding by taking advantage of positive user feedback that is implicit and embedded in the text of the forum posts. User feedback in forums usually takes the form of messages of gratitude and reports of the result of following another user´s advice, and thus is ambiguous and not directly machine-readable. We developed an automatic method to identify and qualify that feedback. Furthermore, almost all previous work on opinion mining and expert finding has been carried out using test collections in English. The Spanish language has its own idiosyncrasies, particularly when it comes to non-formal speaking, which do no translate very well to text mining methods that assume a well-written, grammatically correct text. We built a forum test collection in Spanish for this purpose

This paper is organized as follows: Section 2 gives an overview of relevant previous work. Since there is no test collection in Spanish, section 3 describes how we built the Spanish forum test collection, and some collection statistics relevant to the problem addressed in this paper. Section 4 describes how we model expert finding. In Section 5 we present an experimental evaluation the validity of our method to identify relevant user feedback. Finally, in Section 6 we discuss our main findings.

2 Previous Work

Our work combines two main areas of research: Expert and Social Organization in Online Communities Finding using machine-learning and information retrieval (IR) methods, and Polarity Detection.

Applying machine learning and Information Retrieval for Expert Finding is an active area of research. One of the first systems that retrieved experts instead of just document results was P@noptic [3], employing a combination of IR and crawling techniques: employee pages were crawled, keeping those were *java* was mentioned more often, and from those pages, keywords and contact details were extracted, combining all pages from the same author. When a user issued a query, only those experts whose web page contents closely matched the query were returned, along with their contact details. More sophisticated approaches involving graph and content analysis took advantage of richer information contained in people´s interactions, fundamentally through emails in the Enron and W3C test collections. Both analytic graph analysis techniques like HITS [4] and PageRank [5], and descriptive techniques like [6] have been applied to identify those people that are authorities, that is, those that are the frequently looked after by other people when trying to solve a problem or having a question. Graph techniques only analyze relationship among people, ignoring content. Content Analysis combines queries, documents content and persons information to find knowledgeable people who, given the content of a set of documents authored by different people, and a query, are those most likely to have authored documents that are relevant to the query. Language Models are the preferred and most successful technique to compute these relationships, either by modeling experts as latent variables and documents as mixtures of different expert´s vocabularies [7], or by representing experts as mixtures of language models from associated documents [8]. Current question-answering portals like Vark [9] combine

both people interaction, probabilistic topic and user modeling to decide who is in the set of people likely to be most qualified to answer a question.
Polarity Detection, the other research area this project is closely related, was developed in the context of automatically identifying positive or negative comments that people add to blogs, news and reviews. The term *polarity* here is used to refer to the automatic classification (*detection*) of an opinion into one of two very different groups: positive/negative, pro/against, etc. Methods range from totally supervised training sets based on words list with known valence [10], to weakly supervised methods based on seed terms [11], to approaches that also include the strength of the opinion, like [13]. These methods perform very well when many training examples from a single domain (i.e. movies or cell phones) are available, and the object or subject being qualified by the opinion is clear and present in the training set. As we will see later, our approach is different from traditional opinion mining in that the opinions we classify have very few words, and they do not qualify a subject or object, but they qualify another persons´s opinions.

3 Building a Test Collection

Almost all of the expert finding research evaluation has been performed using either the Enron or the W3C test collections. Unfortunately these collections have the following drawback for our purposes:

- Theses collections are not discussion forums, they are email collections. The main difference is that in emails, person A looks for person B´s advice in a closed conversation. In a forum, person A looks for advice, and can get help from anyone from person B to Z, all with very different knowledge levels.
- Furthermore, in an email collection it is very clear that person B received a specific message from person A. In a forum thread, this is not that clear; a post does not have a clear receiver, since it is sent to the thread, not to a particular message in a thread. However, that post is a reply to some other post. Fortunately, because of the serial structure of discussion threads the related post is very frequently the previous post or the post before that.
- Neither Enron nor W3C collections contain Spanish text. This was important to our work since we were interested in finding experts in Spanish forums. Even more, Spanish used in online forums is different from formal Spanish, since is usually riddled with typos, missing punctuation marks, and made-up words.
- Experts in online forums are “willing to offer advice and assistance, presumably driven by a mixture of motivations including altruism, a wish to be seen as an expert, and the thanks and positive feedback contributed by the people they have helped.” [14].

We harvested a collection by scrapping an online web forum in Spanish about the Linux operating system. This collection is one of the so-called “expert forums”; where users come with questions, hoping to be helped by a more knowledgeable user. This forum is frequented mostly by Argentinean users, and it contains 1629 threads totaling 11287 posts from 1314 unique users. The language used in the thread is

colloquial and full of idioms. Figures 1 and 2 summarize the characteristics of the forum in the form distribution of posts per thread and author frequency per thread:

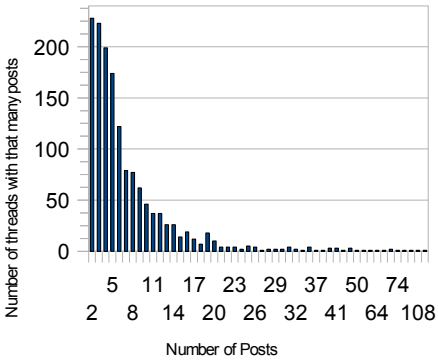

Figure 1: Number of posts per thread versus Number of threads with that number of posts.

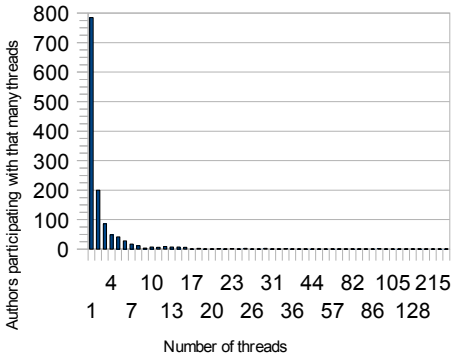

Figure 2: Number of threads and author participation (not counting first post).

These figures display the usual long-tailed participation pattern: A few users that participate often in many conversations, and most threads contain a low number of posts. This collection has also some features that makes it ideal for a test collection: each thread has a tag assigned by the thread´s author, identifying its purpose. To create a test set we will concentrate on those threads tagged as "CONSULTA" or "PREGUNTA", since they are guaranteed to be questions. In this collection 1036 of the 1629 threads contains at least one additional post from the same user that started the thread, and 432 of those threads are marked as having a "CONSULTA" or "PREGUNTA".

4 Modeling Expert Finding

It is quite common in an online discussion forum that those participants with a doubt or problem start a new discussion thread with a question, waiting for other participants to reply with help or an answer to the question. Those posts in a thread that are useful are frequently followed by a post by the same participant who started the thread, in which he or she thanks the user who replied for his or her answer. When the author of a thread thanks for an answer, its content is usually more than courtesy, and indicates a positive outcome, since an incorrect or useless answer is almost never acknowledged: it is easier to ignore it than to reply with a complain and risk an unnecessary dispute with the answer´s author.

Since being an expert implies knowledge in a certain topic, we do not assume that experts are all-knowing, even within their area of expertise. Instead, we aim at finding experts within particular subsets of the forum, focused on particular topics.

We model the process of identifying experts by combining many user participation statistics and user feedback. We define experts as *"answer-persons with positive feedback given by other forum participants"***.** An answer-person is, generally

speaking, someone that participates frequently answering questions, does not engage in chatting, and that seems to know about the subject matter regarding the questions that he or she answers. Being an answer persons and receiving positive feedback are two related indicators of the presence of an expert, but one does not completely depend on the other: flamers (those who like to start arguments just for the sake of arguing), or very opinionated people can sometimes be confused as experts. We calculate and combine answer person statistics with a positive feedback score to assign each author an expert score that separates them from other answer persons.

4.1 Finding Expert Candidates

To score a forum participant on his or her expertise, based on past studies [11][12], we evaluated the impact of the following user statistics:
-The number of threads in which he or she participated but did not initiate,
-The number of threads with he initiated (first post in the thread),
-Number of posts per thread by the author.
-Average umber of threads in which he or she participated but did not initiate,
-Average number of threads with he initiated,
-Average number of posts per thread by the author.

From past studies [11][12] it is known that answer persons participate frequently in the forums, when participation is measured in number of threads where they posted. Answer persons, however, do not usually engage in discussions. We do not count as threads where the user participated in any thread in which the user has more than 3 posts. We do so in order to exclude from the answer persons group those who participate in a thread for the sake of arguing, and thus has a high number of posts per thread in order to keep the discussion going. We do count those threads towards threads initiated by the user.
We evaluated the importance of the above indicators in detecting an answer person, and we found *number of threads initiated*, *number of discussion threads where the author participated without having initiated*, and *number of threads in which an author participated* to be the most significant predictors of a particular user being an answer person.

In order to separate experts from flamers and opinionated people, we look for positive feedback in threads that start with questions. If we can identify those posts that contain thank-you notes and positive feedback to an answer, then we can use that feedback to gain evidence to separate experts from people willing to answer, but unable to provide useful answers, since those posts that are useful answers are more likely to have positive feedback. We address the identification of these posts as a classification problem, involving the following sets:

- *T* is the training set, derived from the posts in discussion threads from a forum. In this set any post indicating positive acknowledgment or gratitude in a thread has been tagged as “positive feedback” by an expert;
- *Q* is a subset of *T* containing all the threads that start with any author posing a question;

- A is the subset of T in which the author of a thread (the user who posed the question) added a positive acknowledgment or gratitude post to the thread, after another author offered a solution or answer in that thread.
- X' is the set of discussion threads where we want to find out if they contain positive feedback, thus serving as evidence of a possible expert: namely, the person who added the helpful answer to the thread.

Here we have a training set T made of triplets (x_i,q_i,a_i), where $x_i \subset \Re^n$ represents the thread content, $q_i \in \{0,1\}$ represents whether the thread starts with a question, and $a_i \in \{0,1\}$ represents whether the thread contains positive feedback added by the same user that started the thread. Our objective is to classify the vectors $x_i \in X'$ into two classes: $a=1$ y $a=0$.

We make one simplification to the classification problem by adding the rule $q=0 \Rightarrow a=0$ (if there is no question, then there is no positive feedback). Although this is not always true, from our observation useful help appears far more often in question threads than in survey or comment threads, where due to the subject and intent of the thread, feedback can be simply comments or biased opinions. The simplification above has the effect of concentrating the classifier on threads where user feedback is more likely to be associated to helpful answers, the kind of answers derived from knowledge that we want to detect.

We approach the problem of assigning values from a to $x_i \in X'$ as a classification problem. The straight approach of training with (x_i,q_i,a_i) to predict q and a is not the best choice, since as we said earlier we only want to classify with $a=1$ or $a=0$ those vectors where $q=1$. It is simpler to create a classification rule $q=0 \Rightarrow a=0$, and train the classifier with $(x_i,q=1,a_i)$ than training with (x_i,q_i,a_i) and risk misclassifying some vector with $(x_i,q=0,a_i)$.

Therefore we divide the problem of classifying the test X' in two steps: A classifier $C_1:x_i' \rightarrow q'$ (q' is the value of Q predicted by C_1) that acts as a filter, and another classifier $C_2:x_i' \rightarrow a'$ that is applied only to those vectors for which C_1 predicted $q'=1$. If C_1 predicts $q'=0$, then we assign $a=0$ to x_i, if $q'=1$ then C_2 gives the final predicted value $a'=1$ or $a'=0$.

It is important to note that although C_2 will only be applied to those vectors where C_1 predicts $q'=1$, C_2 will be trained with the whole set (x_i,a_i). The rationale for this is that the whole set (x_i,a_i) contains both positive and negative examples of the relationship between x_i and a_i. Also, q_i and a_i are not completely correlated, and that for a certain x_i there may be examples $(x_i,a_i=0)$ and $(x_i,a_i=1)$.

The combined classifier approach does not a-priori limit the accuracy of the final result, even if one of the classifiers does not achieve 100% accuracy. Perfect accuracy occurs when the classifier $C_1:x' \rightarrow q'$ does not produce false negatives (even if it does produce some false positives), and $C_2:x' \rightarrow a'$ does not produce any classification errors. In other words, if:

- C_1 does not produce false negatives, correctly predicting $q=1$ to all vectors in x' that are really $q'=1$, but it does have a certain false positive rate where it predicts $q'=1$ when it should really be $q'=0$, and if
- C_2 is a perfect classifier,

then the combined classifier $C_2|C_1$ will have accuracy=1. The classifier C_2 will only classify the true positives (TP) and false positives (FP) of C_1; the true negatives (TN)

and false negatives (FN) of C_1 are assigned $a'=0$ without applying C_2, since in all these later cases we assume to have predicted $q'=0$.
Accuracy of the combined classifier $C_2|C_1$ is calculated as follows:

$$accuracy(C_2|C_1)=\frac{TP(C_2)+TN(C_2)+TN(C_1)}{TN(C_1)+TP(C_1)+FN(C_1)+FP(C_1)}$$

since all true positives of C_1 are classified by C_2, and some may be misclassified.

The whole process of identifying experts then is now like this:
1. A list of answer persons is generated from the whole collection and a specific thread set, along with their statistics.
2. Authors with positive feedback are identified. These are authors that posted replies to one or more threads that start with a question, and where another author recognized those posts as useful by giving positive feedback.
3. A combined score is calculated from 1 and 2.

Sections 4.2 and 4.3 describe in detail the implementation of the C_1 and C_2 classifiers. Section 4.4 describes the combined expert score.

4.2 Classifier C_1: Detecting Questions

Identifying when a paragraph is a question in Spanish is not easy. Questions in Spanish may not need to include an interrogation mark (e.g. "quisiera saber como llegar a Junín"), and even when they should end with an interrogation mark, the question mark is often missing when written in an online forum (e.g "estoy perdido, como hago para que mi auto no se detenga"). Complicating things further, adverbs like "donde", "como", "cuando" are written without accents and as often used in questions as they are used in the middle of sentences ("¿como hago?" vs. "hago como siempre"). These adverbs are typed without accents in the colloquial language used in online forums, making both cases syntactically indistinguishable.
Therefore question marks and the presence of adverbs are not enough to detect a question in a paragraph, so we took a supervised approach to build a classifier that detects questions.
We start with a set of sentences that are marked as questions, and another set of sentences that are not questions but include adverbs often used in questions, which are: "cómo", "cuándo", "cuánto", "qué", "donde", and "por qué". We call these adverbs *question indicators*. For each sentence containing a question indicator, a feature vector is generated with each word up to 3 words to the left and up to 2 words to the right of the question indicator, regardless of whether the question indicator is typed with or without accents. Each feature in the feature vector is a pair (word, offset in words from question indicator). In case a word is close to more than one question indicator, we only include it in the vector corresponding to the closest question indicator. Thus, one sentence may become one or more feature vectors.
We end up with a set of vector features, some coming from questions, some not. Sentences that end with a question mark are classified as questions. Sentences without

a question mark but where the text contains a question indicator, are classified by unweighted majority voting using k-NN with k=3 and cosine similarity.

4.3 Classifier C_2: Detecting Positive Feedback

For those discussion threads where the first paragraph is classified as containing a question, we try to detect if the thread contains positive feedback. We only look for positive feedback from the same user who started the thread, since that user is the one that is most likely to reply with positive feedback by having tried the proposed solution. Positive feedback for a post in a forum, however, is indirect. We define a post to have positive feedback in a forum thread if one of the two posts that immediately follows the candidate post have a positive tone and looks like it is reporting success, or looks like a gratitude post. Examples of posts with a positive tone are "funcionó, muchas gracias!", and "¡sos lo maximo!".

A typical approach to polarity detection based on recognizing keywords with a positive valence (like "excelente") or negative valence (like "malísimo") is impractical in this context since:

- It does not generalize well: the set of words that are used to denote positive or negative connotations vary from forum to forum; e.g. "chico" has a positive valence for cell phones, but it usually has a negative valence for cars.
- It excludes localisms: In online forums, many expressions used to show approval are highly local, like for example "es bárbaro!", which from all Spanish-speaking countries, it only means "excelent!" in Argentina.

Our hypothesis is that there is a subset of adjectives that is common to all Spanish speakers and that have a definite valence, and that other adjectives that are strongly associated to those adjectives share those adjective´s valence.

We use a semi supervised approach where we start with a small list of *seed adjectives* with known valence common to all Spanish speakers. Our list contains 11 seed adjectives with positive valence and 10 with negative valences. Positive adjectives include "excelente", "buenisimo", "hermoso/a", buen/a/o when preceded by "es", "sos", "son", lo or muy", "gracias", etc. Negative examples include "pésimo", "malísimo", "erróneo", "malo", etc. These adjectives are chosen because they are very common and have the same meaning across Spanish variants.

For each sentence in the training set, we use a Part-of-Speech (PoS) tagger to identify all adjectives. This PoS tagger is able to parse and analyze sentences in colloquial Spanish, with defective grammar and verbs with Argentinean stemming (like "*contás*" instead of "*cuentas*"). All adjectives that within 15 words from a seed adjective are detected, and for each occurrence of an adjective, we tentatively associate the valence of the closest seed to that adjective, unless there are words like "*no*" "*ni*" and "*pero*" nearby and in between both adjectives, in which case the tentative polarity is set to be the opposite of the seed.

After analyzing all posts, a final polarity is assigned to each adjective as follows:

$$polarity(adj_i) = \sum_{p_j \in paragraphs} \frac{1}{distance(adj_i, closest\ seed \in p_j)} \times polarity(closest\ seed)$$

since this score does not take on account the frequency of words, rare adjectives may end up with a polarity that does not correspond to its true meaning. Those adjectives are filtered by excluding adjectives that appear as often in the first post as in the last post in the same thread, when both post are authored by the same user, on the assumption that those adjectives do not indicate polarity, since first posts are questions, not opinions, and thus they have no polarity. Polarity of a post is then the sum of the polarity of all adjective contained in the post.

The words "*gracias*" is a special case. It can be used both as an expression (e.g. "*gracias a dios*"), or as a display of gratitude (e.g. "*muchas gracias*"). We separate these cases with a simple heuristic rule, based on the observation of people's writing style in forums: When a post is short, and include "*gracias*", it is usually an acknowledgment, either positive ("*muchas gracias!*") or negative ("*no funcionó pero gracias igual*"); when "*gracias*" appears in the text of a long post and not in the last line, then it is usually part of an expression.

We hypothesized that most of the short posts that included the word "*gracias*" were positive, like "*Gracias Por La Respuesta*", but there were some negative too, like "*Gracias pero paso. No me gusta para nada*". Based on analysis of short posts with positive and negative posts that include that word, we developed the following rule:
If a post contains 4 sentences or less, and no sentence is longer than 20 words, and any of the sentences include the word "gracias", then we mark the post as positive feedback, unless "*gracias*" is nearby "*pero*", "*gracias*" or "*de todas maneras*", in which case we assign a non-positive polarity to the post.

4.4 Ranking Experts by combining data

User are ranked according to a score that include both participation statistics, and the positive feedback the author has received. This rank orders users by their presumed expertise; top users in this rank are considered experts.
For all the forum member who are classified in either of the two answer persons sets, we compute a participation score for each forum member as follows:

$$answerscore(author_i) = \log(TP(author_i)) \times PQ(author_i) / AP(author_i)$$

where:

$TP(author_i)$ = Σ threads in which $author_i$ participated + threads in which $author_i$ has positive feedback + threads iniciated by $author_i$.

$PQ(author_i)$ = Σ threads in which $author_i$ participated + threads in which $author_i$ has positive feedback / Σ threads iniciated by $author_i$.

$AP(author_i)$ = Σ posts by $author_i$ / Σ threads with posts by $author_i$.

The *TP* score is a measure of total author participation, dampened by log to scale the differences; while an author with 32 posts is much better than an author with only 16 posts, an author with 90 posts is not twice as good as an author with 180 posts; both are very frequent forum participants.

The *PQ* quotient is a measure of the willingness of an author to participate in other people threads versus starting their own; those people that are very knowledgeable are much more likely to answer than to ask, and thus they get a higher *PQ* score.
The *AP* ratio penalizes authors that engage in chatting and arguing, but only for those authors that are already in the answer persons set.
We use the answer score to impose an order within the set of authors in a set. Since this score can be calculated for any discussion thread subset, we can take any specialized thread set, filtered by user-assigned tags or by searching, keep the most relevant threads, and within those threads, report the experts with the higher score.

5 Experimental Evaluation

From the harvested forum we picked a dataset made of 721 threads with at least 3 posts, 615 of those contained questions, and 106 that did not. We used the first posts from threads marked as "CONSULTA" or "PREGUNTA" as test set for questions, and the rest for no-questions. The dataset is biased towards questions since from observation in pilot tests, and as can be seen in table 1, we realized that C_1 rarely confuses a no-question for a question (81% recall for no-questions). A balanced dataset would increase the number of no-questions producing artificially high results for C_1 that would not represent its real performance.

	Predicted Question	Predicted No-Question
True Question	**462**	153
True No-Question	20	**86**
Total	282	239

Table 1. Confusion Matrix for Classifier C_1

C_1 had an accuracy of 76% at predicting if the first post of a thread contains a question. Of the 482 thread that C_1 predicted to start with a question, 68 of those do not contain another post by the same user that started the thread. The others (414) contain at least one post by the same user that started the thread, sometimes positive, sometimes not; as we mentioned before, besides positive feedback, a post may be a comment, opinion, or clarification.

Only FP and TP of C_1 were used to evaluate C_2 (the rationale from this was explained in section 3). An expert read every thread and marked the last post from the user who started the thread (when it exists) as containing positive feedback or not. The result of applying C_2 to predict that the last post in a thread that is known to start with a question (68 posts that do not contain another post by the same user) contains positive feedback after an answer, is as follows:

	Predicted Positive	Predicted No-Positive
True Positive Feedback	**107**	44
True No-Positive Feedback	22	**241**
Total	129	285

Table 2. Confusion Matrix for Classifier C_2.

Therefore when C_2 is applied to the TP and FP of C_1 we have 84% accuracy. When C_2 is applied after C_1 to the whole test set to predict that the last post in a thread, written by the same user that started the thread **with a question,** contains positive feedback, we achieved 70% accuracy, counting those 68 posts that do not contain another post by the same user towards TN of C_1. 70% accuracy is remarkable for detecting information otherwise ignored in a noisy dataset, which is very valuable since expert evidence is cumulative: an expert may have positive feedback in many posts.

5.1 Expert Ranking Precision

Finally, we evaluated the score's precision at ranking participants as experts, since when searching for experts, precision is more important than recall: users search for a small set of experts whose answers they can read and trust, rather than a vast group of users, a few of which can be trusted. Here we describe an experiment we carried out to evaluate the expert-scoring algorithm.

The collection we harvested does not include explicit feedback in the form of a score given by users to other users and their questions and answers, so we gathered 10 judges with expertise in the Linux operating system to get that feedback. Since a forum participant may be an expert in one topic but a beginner in another, we first needed to select topics. We selected 10 popular topics from the forum (e.g. “router”, “firewall”), and we searched for threads that contained those topics and that started the thread with a question. Whenever possible, judges excluded those questions that were not really *about* the topic. The result is a group of 10 sets of discussion threads, each set about a different topic. We applied the algorithm to score all participants in all topic sets, and we selected the top 5 experts (as ranked by the algorithm) to be evaluated by human judges which are themselves experts in the collection's subject matter. Each of the 10 judges evaluated 5 forum experts in 1 topic. The task given to the judges was to read each thread, evaluate each of the 5 potential experts answers in each thread, and assign one of two labels to each potential expert: a judge could label a potential expert as *Expert* or *Not-an-Expert*. Ordering of topics and threads was randomized for each judge. Since what means to be an expert may vary from person to person, judges were given the following definition: "An expert is a person that proved to master a certain topic by giving or pointing to useful answers, that in almost all cases solved a problem posed by other users".

From this experiment we gathered 50 data points, each point being the label given by a judge to a potential expert in a particular topic. From these, 41 were determined to be expert by the judges, with X=4.1 $\sigma_{\bar{X}}$=0.87, giving 82% accuracy (4 out of 5) within the top-5 experts for each topic. From these data we have a strong indication that, given a topic-focused set, the scores given by the algorithm have high precision within the top five values, thus proving that the expert-ranking algorithm is useful at detecting experts.

6 Conclusions and Future Work

In this paper we presented a novel technique to enhance expert detection by combining descriptive statistics with machine-learning based techniques to detect those forum participants in so-called “expert forums”, that are likely to have answered questions and had positive feedback to their answers. We achieved 70% accuracy in predicting if the content of a post acknowledges useful help by another user, even without clear question or polarity indications, including misspellings and missing punctuation marks, and 82% accuracy at ranking experts within the top 5 results. This knowledge would otherwise be ignored, and now can be used to detect experts with higher confidence. This approach, however, is likely to be not as successful in forums dominated by opinions instead of questions and answers.
Future work includes combining this technique with forums containing explicit user feedback to posts in the form of user-assigned scores, since both approaches can be combined to have a better understanding of expertise of the user base.

References

1. Collison, C. and Geoff Parcell, G. Learning To Fly: Practical lessons from one of the World's Leading Knowledge Companies (2nd Ed.). J Wiley / Capstone (2005).
2. Earl,M. Knowledge Management Strategies: toward a taxonomy. Journal of Management Information Systems 18(1), pp. 215—223 (2001).
3. Craswell, N., Hawking, D., Vercoustre A., and P. Wilkins. P@noptic expert: Searching for Experts not just for Documents. Ausweb, pp. 21—25 (2001).
4. D’Amore, R. Expertise Community Detection. Proc. ACM SIGIR 2004, pp. 498—499 (2004).
5. Zhang, J, Ackerman, MS , Adamic L. Expertise Networks in Online Communities: Structure and Algorithms. Proc. of WWW 2007, pp. 221—230 (2007).
6. Welser, H. , Gleave, E., and Smith, Marc A. Visualizing the Signatures of Social Roles in Online Discussion Groups. Journal of Social Structure, 8(2), 2007.
7. Serdyukov, P., Hiemstra, D. Modeling Documents as Mixtures of Persons for Expert Finding. Proc. of 30th European Conference on Information Retrieval, pp. 309—320 (2008).
8. Petkova D., Croft B.. Hierarchical language models for expert finding in enterprise corpora. Proc. of 18th IEEE Intl. Conf. on Tools with Artificial Intelligence, pp. 599—608 (2006).
9. Horowitz, D., Kamvar, S. The Anatomy of a Large-Scale Social Search Engine. Proc. of WWW 2010, pp. 431—440 (2010).
10. Pang, B., Lee,L. Vaithyanathan, S. Thumbs up? Sentiment Classification using Machine Learning Techniques. Proc. of Conference on Empirical Methods in Natural Language Processing, pp. 79—86 (2002).
11. Turney, P. Thumbs up or Thumbs down? Semantic Orientation applied to Unsupervised Classification of Reviews. Proc. of the 40th Annual Meeting of the Association for Computational Linguistics, pp. 417—424 (2002).
12. Nam, K., Ackerman, M., Adamic, L. Questions in, Knowledge iN? A Study of Naver’s Question Answering Community. Proc. of ACM CHI 2009, pp. 779—788 (2009).
13. Wilson, T., Wiebe, J., Hwa, R. 2004. Just how mad are you? Finding strong and weak opinion clauses. Proc. of AAAI-2004, pp. 761—769, (2004).
14. Marwick, A. Knowledge Management Technology. IBM Systems Journal, 40 (4), pp .814—830 (2001).