Relative Rank Statistics for Dialog Analysis

Juan M. Huerta
IBM T.J. Watson Research Center
1101 Kitchawan Road
Yorktown Heights, NY 10598
huerta@us.ibm.com

Abstract

We introduce the relative rank differential statistic which is a non-parametric approach to document and dialog analysis based on word frequency rank-statistics. We also present a simple method to establish semantic saliency in dialog, documents, and dialog segments using these word frequency rank statistics. Applications of our technique include the dynamic tracking of topic and semantic evolution in a dialog, topic detection, automatic generation of document tags, and new story or event detection in conversational speech and text. Our approach benefits from the robustness, simplicity and efficiency of non-parametric and rank based approaches and consistently outperformed term-frequency and TF-IDF cosine distance approaches in several experiments conducted.

1   Background

Existing research in dialog analysis has focused on several specific problems including dialog act detection (e.g., Byron and Heeman 1998), segmentation and chunking (e.g., Hearst 1993), topic detection (e.g., Zimmerman et al 2005), distillation and summarization (e.g., Mishne et al 2005) etc. The breath of this research reflects the increasing importance that dialog analysis has for multiple domains and applications. While historically, dialog analysis research has initially leveraged the corresponding techniques originally intended for textual document analysis, techniques tailored specifically for dialog processing eventually should be able to address the sparseness, noise, and time considerations intrinsic to dialog and conversations.

The approach proposed in this paper focuses on the relative change of rank ordering of words occurring in a conversation according to their frequencies. Our approach emphasizes relatively *improbable* terms by focusing on terms that are relatively unlikely to appear frequently and thus weighting their change in rank more once they are observed. Our technique achieves this in a non-parametric fashion without explicitly computing probabilities, without the assumption of an underlying distribution, and without the computation of likelihoods.

In general, non-parametric approaches to data analysis are well known and present several attractive characteristics (as a general reference see Hollander and Wolfe 1999). Non-parametric approaches require few assumptions about the data analyzed and can present computational advantages over parametric approaches especially when the underlying distributions of the data are not normal. In specific, our approach uses rank order statistics of word-feature frequencies to compute a relative rank-differential statistic.

This paper is organized as follows: in Section 2 we introduce and describe our basic approach (the relative rank differential RRD function and its sorted list). In Section 3 we address the temporal nature of dialogs and describe considerations to dynamically update the RRD statistics in an online fashion especially for the case of shifting temporal windows of analysis. In Section 4 we relate the RRD approach to relevant existing and previous dialog and text analysis approaches. In Section 5 we illustrate the usefulness of our metric by analyzing a set of conversations in various ways using the RRD. Specifically, in that section we will empirically demonstrate its robustness to noise and data sparseness compared to the popular term frequency and TF-IDF cosine distance approaches in

a dialog classification task. And finally, in Section 6, we present some concluding remarks and future directions

2 The Relative Rank Differential

Let $d^u = \{d_1^u, d_2^u, d_3^u ...\}$ denote the ranked dictionary of a language (i.e., the ordered list of words sorted in decreasing order of frequency). The superscript u denotes that this ranked list is based on the *universal* language. Specifically, the word d_i^u is the ith entry in d^u if its frequency of occurrence in the language denoted by $f(d_i^u)$ is larger than $f(d_j^u)$ for every j where $i < j$ (for notational simplicity we assume that no two words share the same frequency). In the case where want to relax this assumption we simply allow $i < j$ when $f(d_i^u) = f(d_j^u)$ as long as d_i^u precedes d_j^u lexicographically, or under any other desired precedence criteria. For d^u we assume that $f(d_i^u) > 0$ for every entry (i.e., each word has been observed at least once in the language).

Similarly, let now $d^S = \{d_1^S, d_2^S, d_3^S ...\}$ denote the corresponding ranked dictionary for a dialog, or dialog segment, S (ordered, as in the case of the language dictionary, in decreasing order of frequency)[1]. The superscript S denotes that this ranked list is based on the dialog S. The word d_i^S is the ith entry in d^s if its frequency of occurrence in the conversation segment S denoted by $f(d_i^S)$ is larger than $f(d_j^S)$ for every j where $i < j$. In this case we allow $f(d_i^S) \geq 0$ for every i so that the cardinality of d^u is the same as d^s.

Let $r_d(w)$ denote the rank of word w in the ranked dictionary d so that, for example, $r_{d^u}(d_i^u) = i$.

Based now on a dialog segment and a universal language, any given word w will be associated with a rank in d^u (the universal ranked dictionary) and a rank in d^s, the dialog segment ranked dictionary.

Let us define now for every word the relative rank differential (RRD) function or statistic[2] given by:

$$c_{d^s, d^u}(w) = \frac{\left| r_{d^u}(w) - r_{d^s}(w) \right|}{\left(r_{d^u}(w) \right)^\alpha}$$

The relative rank-differential is the ratio of the absolute difference (or change) in rank between the word's original position in the universal dictionary and the segment s. The exponent α in the denominator allows us to emphasize or deemphasize changes in terms according to their position or rank in the language (or universal) dictionary. Typically we will want to increase the denominator's value (i.e., deemphasize) for terms that have very low frequency (and their rank value in the universal dictionary is large) so that only relatively big changes in rank will result in substantial values of this function.

When alpha is zero, the RRD focuses on every word identically as we consider only the absolute change in rank. For alpha equal to 1.0 the relative change in rank gets scaled down linearly according to its rank, while for alphas larger than 1.0 the numerator will scale down or reduce to a larger extent the value of relative rank differential for words that have large rank value.

Based on each word's relative rank differential we can compute the ranked list of words sorted in decreasing order by their corresponding value of relative rank differential. Let this sorted list of words be denoted by $R(d^u, d^S) = \{w_1, w_2, ...\}$. So that $c(w_i)$ is larger than $c(w_j)$ [3] for every j where $i < j$.

We now provide some intuition on the ranked RRD lists and the RRD function. The ranked dictionary of a language contains information

---

[1] We only consider at this point the case in which both speakers' parts in a dialog interaction are considered jointly (i.e., single channel), however, our method can be easily extended to separate conversation channels. Also, for simplicity we consider at this point only words (or phrases) as features.

[2] For brevity, we refer to the Relative Rank Differential of a word given two utterances as a *statistic*. It is not, strictly speaking, a metric or a distance, but rather a function.

[3] For simplicity, c is written without subscripts when these are apparent from the context.

about the frequency of all words in a language (i.e., across the universe of conversations) while the segment counterpart pertains a single conversation or segment thereof. The relative rank differential tells us how different a word is ranked in a conversation segment from the universal language, but this difference is normalized by the universal rank of the word. Intuitively, and especially when alpha equals 1.0, the RRD denotes some sort of percent change in rank. This also means that this function is less sensitive to small changes in frequency in the case of frequent words and to small changes in rank in case of infrequent words.

Finally, the sorted list $R(d^u, d^S)$ contains in order of importance the most relatively salient terms of a dialog segment, as measured by relative changes or differences in rank.

3   Collecting Rank Statistics

We now discuss how to extend the metrics described in the previous section to consider finite-time sliding windows of analysis, that is, we describe how to update rank statistics, specifically the ranked lists and relative rank differential information for every feature in an on-line fashion. This is useful when tracking the evolution of single dialogs, when focusing the analysis to span shorter regions, as well as to supporting dynamic real-time analytics of large number of dialogs.

To approach this, we decompose the *word events* (words as they occur in time) into arriving and departing events. An arriving event at time *t* is a word that is covered by the analysis window at its specific time as the finite length window slides in time, and a departing word at time *t* is a feature that stops falling within the window of analysis. For simplicity, and without loss of generality, we now assume that we are performing the analysis in real time and that the sliding window of analysis spans from current time *t* back to *(t-T)*, where *T* is the length of the window of analysis.

An arriving word at current time *t* falls into our current window of analysis and thus needs to be processed. To account for these events efficiently, we need a new structure: the temporal event *FIFO* list (i.e., a queue where events get registered) that keeps track of events as they arrive in time. As an event (word w_t) arrives it is registered and processed as follows:

1. Find the corresponding identifier of w_t in the universal ranked dictionary and add it as d_i^u at the end of the temporal event list together with its time stamp.
2. The corresponding entry in d^s, the ranked segment dictionary, is located through an index list that maps $d_i^u \rightarrow d_k^s$ and the segment frequency associated is incremented $f(d_k^s) = f(d_k^s) + 1$
3. Verify if the rank of the feature needs to be updated in the segment rank list. In other words evaluate whether $f(d_{k-1}^s) > f(d_k^s)$ still holds true after the update. If this is not true then shift feature up in the rank list (to a higher rank) and shift down the predecessor feature in the rank list. In this single shift-up-down operation, update the index list and the value of *k*.
4. For every feature shifted down in 3 down re-compute the relative rank differential RRD function and verify if its position needs to be modified in $R(d^u, d^S)$ (a second index list is needed to compute this efficiently).
5. Repeat step 3 iteratively until feature is not able to push up any further in the ranked list.

The process for dealing with departing events is quite similar to the arriving process just described. Of course, as the analysis window slides in time, it is necessary to keep track of the temporal event FIFO list to make sure that the events at the top are removed as soon as they fall out of the analysis window. The process is then:

1. The departing event is identified and its corresponding identifier in the universal ranked dictionary d_i^u is removed from the top of the temporal event list.
2. Its location in d^s the ranked segment dictionary is located through the index list. The corresponding segment frequency associated is decreased as follows: $f(d_k^s) = f(d_k^s) - 1$.
3. Verify if the rank of the feature needs to be updated in the segment rank list. In other

words evaluate if $f(d_{k+1}^s) < f(d_k^s)$ still holds true after the update. If not shift feature down in rank (to a lower rank, denoting less frequent occurrence) and shift the successor feature up in the rank list. In this single shift up-down operation, update the index list and the value of k.

4. For every feature shifted up in step 3 recompute the relative rank differential and verify if its location needs to be modified in $R(d^u, d^S)$

5. Repeat step 3 until the feature is not able to shift down any further in the ranked list.

The procedures just described are efficiently implementable as they simply identify entries in rank lists through index lists, update values by incrementing and decrementing variables, and performed some localized and limited re-sorting. Additionally, simple operations like adding data at the end and removing data at the beginning of the FIFO list are needed making it altogether computationally inexpensive.

4 Related Techniques

Our work relates to several existing techniques as follows. Many techniques of text and dialog analysis utilize a word frequency vector based approach (e.g., Chu-Carroll et al 1999) in which lexical features counts (term frequencies) are used to populate the vector. Sometimes the term frequency is normalized by document size and weighted by the inverse document frequency (TF-IDF). The TF-IDF and TF metrics are the base of other approaches like discriminative classification (Kuo and Lee 2003; and Li and Huerta 2004), Text Tilling or topic chains (Hearst 1993; Zechner 2001), and latent semantic indexing (Landauer et al 1998). Ultimately, these types of approaches are the foundation of complex classification and document understanding systems which use these features together with possibly more sophisticated classification algorithms (e.g., D'Avanzo et al 2007).

When using TF and TF-IDF approaches, it is important to notice that by normalizing the term frequency by the document length, TF-based approaches are effectively equivalent to estimation of a multinomial distribution. The variance of the estimate will be larger as the number of observations decreases. Recently, approaches that explicitly es-

tablish this parametric assumption and perform parameter inference have been presented in (Blei et al 2003). This work is an example of the potential complexity associated when performing parameter inference.

The area of adaptation of frequency parameters for ASR, specifically the work of (Church 2000), is relevant to our work in the sense that both approaches emphasize the importance of and present a method to update the lexical or semantic feature statistics on-line.

In the area of non-parametric processing of dialog and text, the work of (Huffaker et al 2006), is very close to the work in this paper as it deals with non-parametric statistics of the word frequencies (rank of occurrences) and uses the Spearman's Correlation Coefficient. Our work differs from this approach in two ways: first, the Relative Rank Differential tells us about the relative change in rank (while SCC focuses in the absolute change) and secondly, from the ranked RDD list, we can identify the saliency of each term (as opposed to simply computing the overall similarity between two passages).

5 Experiments

In order to illustrate the application of the RRD statistic, we conducted two sets of experiments based on conversations recorded in a large customer contact center for an American car manufacturer. In the first group of experiments we took a corpus of 258 hand transcribed dialogs and conducted classification experiments using the basic RRD statistic as feature. We compared its performance against term frequency and TF-IDF based cosine distance approaches. The second set of experiments is based on ASR transcribed speech and for this we used a second corpus consisting of a set of 44 conversations spanning over 3 hours of conversational speech.

In the first set of experiments we intend to illustrate two things: first the usefulness of RRD as a feature in terms of representational accuracy and second, its robustness to noise and data sparseness compared to other popular features. In the second set of experiments we illustrate the versatility and potential of our technique to be applied in dialog-oriented analysis.

5.1 RRD for Dialog matching

For this set of experiments we used a corpus of 258 hand transcribed conversations. Each dialog was treated like a single document. Using the set of dialogs we constructed different query vectors and affected these queries using various noise conditions, and then we utilized these vectors to perform a simple document query classification experiment. We measured the cosine distance between the noisy query vector and the document vector of each document in this corpus. A noisy query is constructing by adding zero mean additive gaussian noise to the query vector with amplitude proportional to the value of a parameter N and with floor value of zero to avoid negative valued features. We allow, in these experiments, for counts to have non-integer values; as the dialog becomes larger, the Gaussian assumption holds true due to the Central Limit Theorem, independently of the actual underlying distribution of the noise source. This distortion is intended to mimic the variation between two similar dialogs (or utterances) that are essentially similar, except for a additive zero mean random changes. A good statistic should be able to show robustness to these types of distortions. A correct match is counted when the closest match for each query is the generating document.

	N=0.0	N=.05	N=0.1	N=0.2	N=0.4
TF-cosine	99.6	98.0	84.9	60.0	32.5
TF-IDF cosine	99.6	99.6	97.3	88.0	67.4
RRD-dot	99.6	99.6	97.6	91.8	70.9

Table 1. Query match accuracy for 3 features under several query noise conditions.

Table 1 shows the percent correct matches for the TF, TF-IDF and Relative Rank Differential features, under various levels of query noise. As we can see, in clean conditions the accuracy of the 3 features is quite high but as the noise conditions increase the accuracy of the 3 techniques decreases substantially. However, the TF feature is much more sensitive to noise than the other two techniques. We can see that our technique is better than both TF and TF-IDF in noisy conditions.

We also conducted experiments to test the comparative robustness or the RRD feature to query data sparseness. To measure this, we evaluated the accuracy in query-document match when using a random *subset* of the document as query. Figure 1 show the results of this experiment using the RRD feature, the Term Frequency, and the TF-IDF feature vectors. We can see that with as little as 5% of the document size as query, the RRD achieves close to 90% accuracy while the TF-IDF feature needs up to 20% to achieve the same performance, and the TF counts only need close to 70%.

These results empirically demonstrate that RRD statistics are more robust to noise and to term coverage sparseness than TF and TF-IDF.
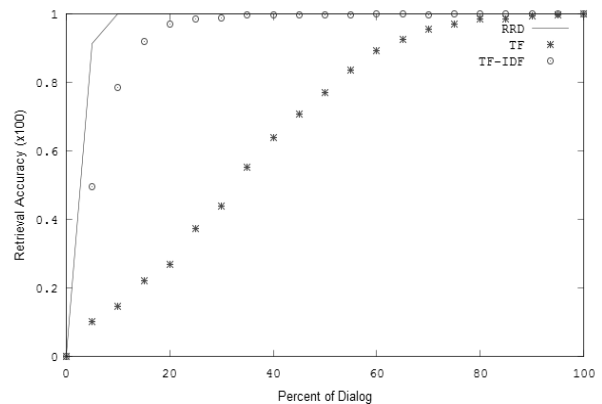

Figure1. Query match accuracy for 3 feature types under various query data sparseness conditions

5.2 ASR Based experiments

For the experiments of this section we used 44 dialogs. Manual transcriptions for these 44 conversations were obtained in order to evaluate the speech recognition accuracy. While we could have used the manual transcripts to perform the analysis, the results reported here are based on the recognition output. The reason for using ASR transcripts as opposed to human transcription is that we wanted to evaluate how useful our approach would be in a real ASR based solution dealing with large amounts of noisy data at this level of ASR error.

Each dialog was recorded in two separate channels (one for the agent and one for the customer) and automatically transcribed separately using a large vocabulary two-stage automatic speech recognizer system. In the first stage, a speaker independent recognition pass is performed after which the resulting hypothesis is used to compensate and adapt feature and models. Using the adapted feature and models the second stage recognition is performed. After recognition, the single best hypothesis with

time stamps for the agent and customer are weaved back together.

The overall Word Error Rate is about 24% and varies significantly between the set of agents and the set of customers (the set of agents being more accurate).

The universal dictionary we used consists exclusively of the words occurring in the corpus which total 2046 unique words. Call length ranged from just less than 1 minute to more than 20 minutes with most of the calls lasting between 2 and 3 minutes. The corpus consists of close to 30k tokens and does not distinguish between agent channel and customer channel. A universal dictionary of ranked words is built from the set of dialogs and each dialog is treated as a segment.

Dialog Tagging and Topic Saliency

In this analysis we look at complete dialogs. A useful application of the methods we describe in this work is to identify and separate calls that are interesting from non-interesting calls[4], furthermore, one could also be interested in singling out which specific terms make this dialog salient. An application of this approach is the automatic generation of tags (e.g., social-network style of document tagging). In our approach, we will identify calls whose top entries in their sorted relative rank differential lists are above a certain threshold and deem these calls as semantically salient.

We now describe in detail how an interesting call can be distinguished from a non-interesting call using the relative rank differential statistic.

Figure 2 below shows the ranked dictionary $d^S = \{d_1^S, d_2^S, d_3^S ...\}$ (i.e., the universal rank id's as a function of their observed ranks) and Figure 3 shows the plot of the sorted relative rank differential list $R(d^u, d^S)$ for when the segment corresponds to an interesting call (as defined above).

The chosen call, specifically shows as topic AIRBAG deployment in the context of a car accident.

Specifically, Figure 2 shows the corresponding rank in the universal ranked dictionary versus the rank in the dialog or segment. We can see that the

---

[4] For the purpose of this work, we simply define as an *interesting call* a call that deals with an infrequent or rare topic which influences the distribution of keywords and key-phrases. Examples of calls in our domain meeting this criterion are calls dealing with accidents and airbags.

right-most part of the plot is largely monotonic, meaning that most entries of lesser frequency occur in the same ranked order both in the universal and the specific dialog (including zero times for the segment), while a subset across the whole range of the universal dictionary were substantially relocated up in the rank (i.e., occurred more frequently in the dialog than in the language). If the plot was a single straight line each word would have the same rank both in the language and in the dialog.

We argue that while the terms of interest lie in that subset of interest in the graph (the terms whose rank increased substantially), not all of those words are equally interesting or important and rather than simply looking at absolute changes in rank we focus on the relative-rank differential RRD metric. Thus, Figure 3 shows the sorted values of the relative rank differential list (with $\alpha = 1.3$). The top entries and their rank in the universal dictionary (in parentheses) are: AIRBAGS (253), AS (55), FRONT (321), DEPLOY (369), SIDE (279), ACCIDENT (687). As we can see, the top entries are distributed across a broad range of ranks in the universal dictionary and relate to the topic of the conversation, which from the top ranked entries are evidently the deployment of front and side airbags during an accident, and thus, for this call were able to identify its semantic saliency from the corpus of conversations.

Other interesting or salient calls also showed a similar this profile in the RRD curve.

The question now is what the behavior of our approach for uninteresting calls is. We repeated the procedure above for a call which we deemed semantically un-interesting (i.e., dealing with a common topic like call transfer and other routine procedures). Figure 4 shows the sorted relative rank differential values and, especially when compared with Figure 2, we see a large monotonic component on the higher ranked terms and not so marked discontinuities in the low and mid-range part of the curve.

We computed the relative rank differential RRD metric for each feature similarly as with the interesting call, and ranked the words based on these values. The distribution of the ranked values is shown in Figure 5. The resulting words with top values are CLEAR (1113), INFORMATION (122) BUYING (1941), and CLEARLY (1910). From these words we cannot really tell what is the specific topic of the conversation is as easily as with

the interesting call. More importantly, we can now compare Figures 3 and 5 and see that the highest relative rank differential value of the top entry in Figure 3 (larger than 10) is significantly larger than the largest relative rank differential value in Figure 5 (just above 7) reflecting the fact that the relative rank differential metric could be a useful parameter in evaluating semantic saliency of a segment using a static threshold. As an interesting point, conceivably the highly ranked features based on RRD could reflect language utilization idiosyncrasies.
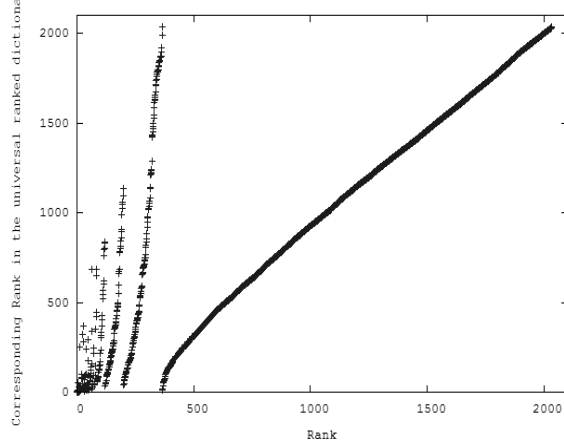
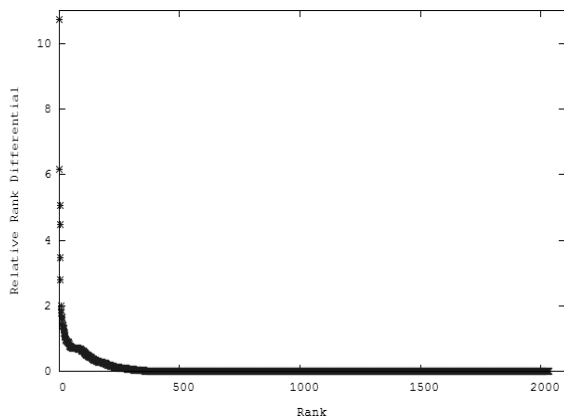**Figure 2.** Ranked dictionary entry vs Universal Rank for a salient call

Figure 3. Sorted relative rank differential values of $R(d^u, d^S)$ for a semantically salient call.
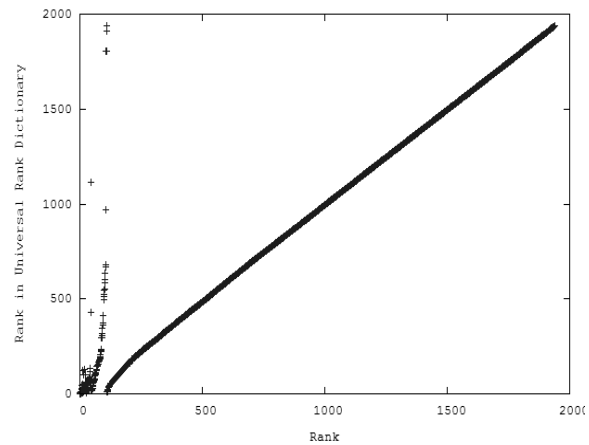

Figure 4. Ranked dictionary entry vs Universal Rank for a non-salient call
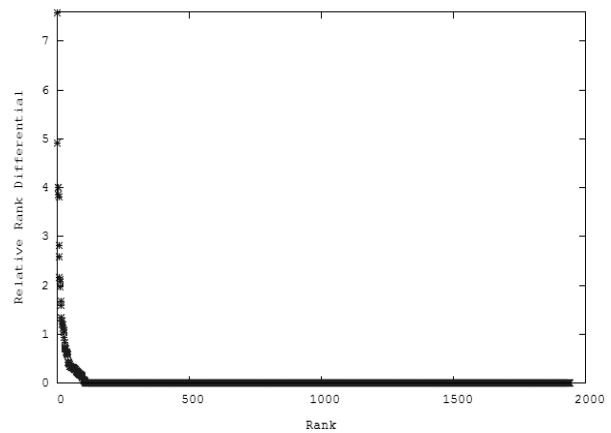

Figure 5. Sorted relative rank differential values of $R(d^u, d^S)$ for a non-interesting (semantically non-salient) call.

6   Conclusions

In this paper we presented a novel non parametric rank-statistics based method for the semantic analysis of conversations and dialogs. Our method is implementable in segment-based or dialog-based modalities, as well as in batch form and in on-line or dynamic form. Applications of our method include topic detection, event tracking, story/topic monitoring, new-event detection, summarization, information filtering, etc. Because our approach is based on non-parametric statistics it has favorable intrinsic benefits, like making no assumptions about the underlying data, which makes it suitable for the use of both lexical semantic features as well as classifier-based semantic features. Furthermore, our approach could, in the future, benefit from

classical non-parametric approaches like block-treatment analysis etc.

We demonstrated that our approach is as effective in query classification as TF and TF-IDF in low noise and no noise (i.e., distortion) conditions, and consistently better than those techniques in noisy conditions. We also found RRD to be more robust to query data sparseness than TF and TF-IDF. These results provide a motivation to combine our statistic with other techniques like topic chains, textilling, latent semantic indexing, and discriminant classification approaches; specifically RRD could replace TF and TF-IDF based features.

Future work could focus on applying ranking statistics to techniques for mining and tracking temporal and time-changing parameters in conjunction with techniques like (Agrawal and Srikant 1995; Pratt 2001; Last et al 2001).

Another area of possible future work is the detection and separation of multiple underlying trends in dialogs. Our approach is also suited for the analysis of large streams of real time conversations, and this is a very important area of focus as presently more and more conversational data gets generated through channels like chat, mobile telephony, VoIP etc.

References

Agrawal R. and Srikant R. 1995. *Mining Sequential Patterns*. In Proc. of the 11th Int'l Conference on Data Engineering, Taipei, Taiwan.

Berger, A. L., Pietra, V. J., and Pietra, S. A. 1996. *A maximum entropy approach to natural language processing*. Comp. Linguist. 22, 1

Blei D., Ng A., and Jordan M. 2003. *Latent Dirichlet allocation*. J. of Machine Learning Research

Byron, D. K. and Heeman, P. A. 1998. *Discourse Marker Use in Task-Oriented Spoken Dialog.* TR664, University of Rochester.

Chu-Carroll, J, and Carpenter R. 1999. *Vector-Based Natural Language Call Routing*. Journal of Computational Linguistics, 25(30), pp. 361-388

Church, K. 2000. *Empirical estimates of adaptation: The chance of two Noriega 's is closer to p/2 than p2*. In Coling

D'Avanzo E., Elia A., Kuflik T., Vietri S. 2007. *University of Salerno, LAKE System at DUC 2007,* Proc. Document Understanding Conference

Hearst, M. 1993. *TextTiling: A Quantitative Approach to Discourse Segmentation*, Technical Report UCB:S2K-93-24, Berkeley, CA

Hollander & Wolfe 1999. *Nonparametric Statistical Methods,* Second Edition, John Wiley and Sons

Huffaker, D., Jorgensen, J., Iacobelli, F., Tepper, P., & Cassell, J. 2006. *Computational Measures for Language Similarity across Time in Online Communities.* Workshop on ACTS at HLT-NAACL, New York City, NY.

Klinkenberg R. and Renz I. 1998. *Adaptive information filtering: Learning in the presence of concept drifts*. In Learning for Text Categorization, Menlo Park

Kuo H.-K.J. and Lee C. H. *Discriminative training of natural language call routers,* IEEE Transactions on Speech and Audio Processing, Volume 11, Issue 1, Jan 2003 Page(s): 24 - 35.

Landauer T., Foltz P. W., and Laham D. *Introduction to Latent Semantic Analysis* Discourse Processes 25, 1998.

Last M., Klein Y., and Kandel A., *Knowledge Discovery in Time Series Databases* IEEE Trans. on Systems, Man, and Cybernetics 31B(2001).

Li X. and Huerta J.M., *Discriminative Training of Compound word based Multinomial Classifiers for Speech Routing* Proc. ICSLP 2004

Mishne, G., Carmel, D., Hoory, R., Roytman, A., and Soffer, A. 2005. *Automatic analysis of call-center conversations*. In *Proc. of the 14th ACM international Conference on information and Knowledge.*

Pratt K. B. *Locating patterns in discrete time series.* Master's thesis, Computer Science and Engineering, University of South Florida, 2001.

Stanley K. O. *Learning concept drift with a committee of decision trees*. Comp. Science Dept., University of Texas-Austin. TR AI-03-302, 2003.

Zechner K. *Automatic Summarization of Spoken Dialogues in Unrestricted Domains*. PhD thesis, LTI, CMU, 2001

Zimmermann M., Liu Y., E. Shriberg, and A. Stolcke, *Toward Joint Segmentation and Classification of Dialog Acts in Multiparty Meetings* MLMI workshop, 2005