Information Triage using Prospective Criteria

Sofus A. Macskassy[1], Haym Hirsh[1,2], Foster Provost[2], Ramesh Sankaranarayanan[2], and Vasant Dhar[2]

[1] Department of Computer Science, Rutgers University
110 Frelinghuysen Rd, Piscataway, NJ 08854-8019
{sofmac,hirsh}@cs.rutgers.edu
[2] Information Systems Department, NYU Stern School of Business
44 W. 4th St, New York, NY 10012
{fprovost,rsankara,vdhar}@stern.nyu.edu

Abstract: In many applications, large volumes of time-sensitive textual information require triage: rapid, approximate prioritization for subsequent action. In this paper, we explore the use of *prospective* indications of the importance of a time-sensitive document, for the purpose of producing better document filtering or ranking. By prospective, we mean importance that could be assessed by actions that occur in the future. For example, a news story may be assessed (retrospectively) as being important, based on events that occurred after the story appeared, such as a stock-price plummeting or the issuance of many follow-up stories. If a system could anticipate (prospectively) such occurrences, it could provide a timely indication of importance. Clearly, perfect prescience is impossible. However, sometimes there is sufficient correlation between the content of an information item and the events that occur subsequently. We describe a process for creating and evaluating approximate information-triage procedures that are based on prospective indications. Unlike many information-retrieval applications for which document labeling is a laborious, manual process, for many prospective criteria it is possible to build very large, labeled, training corpora automatically. Such corpora can be used to train text classification procedures that will predict the (prospective) importance of each document. This paper illustrates the process with two case studies, demonstrating the ability to predict whether the stock price of one or more companies mentioned in a news story will move significantly following the appearance of that story. We conclude by discussing that the comprehensibility of the learned classifiers can be critical to success.

1 Introduction

Professionals receive increasing amounts of information, some of which is time sensitive and is important for them to consider. The business news provides an interesting illustration: the job performance of financial analysts, attorneys, business-school professors, market makers, portfolio managers, reporters, and many others would benefit from timely attention to certain business news stories. Bloomberg, Reuters, Bridge, and several other companies have profited greatly selling a variety of instant-access, business information services. However, the volume of business news is so large that few professionals can pay attention to it all, let alone do so in a timely fashion. Business news, which is used in this paper, is just one example information source.

Information triage is the monitoring of one or more information sources to provide users with well-filtered, prioritized, and/or categorized information (cf., [15].) Our general information-triage framework consists of monitoring a potentially wide range of online information sources—such as news stories, stock data, weather reports, and other computer-based information feeds—and evaluating each item to assess its importance to a given user. Although information triage does not require multiple sources of information, we explicitly embrace such situations when they can create synergy and improve the information-triage process.

One of the key difficulties in building information-triage procedures is building models of importance that will be used to prioritize information. Ideally we would like to obtain from a user a direct statement of his or her interests. However, in many cases it is not clear that users can do so effectively. Instead information filtering and ranking procedures often rely on indirect statements of interest, such as user-provided, keyword-based profiles [10, 8], or samples of information items whose importance has been assessed by the user via relevance feedback methods [20, 22, 24]. We believe that such methods are crucial components of an effective information triage procedure, but we believe that there are other useful components as well. In this paper we concentrate on prospective indicators.

Often what makes information important is some subsequent occurrence that is directly or indirectly associated with the information. For example, consider the appearance of a news story about a publicly traded company, after which the company's stock value quickly plummets. The importance of the news story is based not solely on the story itself but also on the occurrence of the future event (observed in a separate information feed, in this case stock-market data). In many cases a user will be able to specify what future events would make an information item important—such as a substantial change in the value of a company.

A problem is that this importance criterion can not be evaluated directly at the time the information appears; its evaluation requires knowledge about the future. However, if there are patterns in the stories—for example, if many stories that are coupled with precipitous drops in a stock price have similar structure or content—we might be able to predict (approximately) that a story will be followed by an important event. We believe that even an approximate prediction can be quite useful for information triage.

We propose having a user specify what would make an article important if we could perceive the future behavior of this or other relevant information feeds. We then *operationalize* [17] this importance criterion to be evaluable on a given story in the given information feed before we see the future. Key to our approach is the application of the user's specification to label historical documents by importance. These data then form a (perhaps very large) training corpus, to which inductive algorithms will be applied to build a text classifier. Although we believe this framework to be complementary to learning from labels elicited via relevance feedback (or other manually created labels), it has the advantage that the labeling of documents can be done automatically, and at a very large scale.

This paper describes a four-step process for creating and evaluating such operationalized approximations to a user's non-operational specification.

1. Elicit from the user and encode a specification of what future events would make a current piece of information interesting—for example, a news story would be interesting if, within the hour of the story being published, there is a significant/unusual move in the price of the stock of any company mentioned in the news story.
2. Use this specification to analyze information feeds received in the past to ascertain whether or not each item was interesting, thereby creating a set of data items labeled by whether or not each was interesting.
3. Apply inductive algorithms to these labeled data to form models that can estimate the extent to which an information item is interesting to a user directly from the item itself, without the need to look into the future.
4. Analyze the learned model to assess both whether it appears to be a plausible operationalization of the original criterion and whether it is something that appears trustworthy. If the "native" learned form is not easily interpretable, as is the case in

the two studies contained herein, then this may first require applying techniques for obtaining an understandable form of the operationalized criterion.

After providing further details of this process, the paper focuses on a case study involving two available information feeds, news stories and stock price data. (We had to remove the discussion of a second study on “Hot Story detection” due to space limitations. This discussion is available in an upcoming paper [14].) In the study, a news story is deemed important if the stock-market return of a company mentioned in the story is more than one standard deviation from its normal hourly return, in the hour following the appearance of the story.

2 Learning Operational Information Filters

We now describe in more detail the four-step process for performing one form of information triage: when an item is deemed interesting because something important subsequently happens—something that can be measured objectively (retrospectively) either in the given information feed or some coupled information feed. This process reifies and extends the process used previously by Fawcett and Provost [7] and the followup work by Lavrenko, et al. [13], where text documents (news stories) were labeled by referencing subsequent stock-market events.

2.1 Specifying a Non-Operational Criterion

Our first step is to acquire and encode the specification of how an item may be interesting based on possible events that may be subsequently observed. In general this can be a complex process. The non-operational criterion is non-operational only with respect to a world where no knowledge of the future is available. However, it needs to be fully operational when it does have access to the future, as is the case when it is being used to label data from the past. Thus, for example, saying that an article is interesting if it is followed by a substantial movement in a stock is rather high-level. If movement is defined in terms of what is typical it is necessary to compute what is typical, not to mention the need for a specification of how far a value must be from typical. The first step of our process requires that the criterion be stated in unambiguous detail, so it can be directly applied to data.

2.2 Generate and Label Data

The specification of a non-operational criterion will generally presume that a particular primary information feed is the focus of the criterion, and that the criterion will look into the future of either this feed or other coupled feeds in assessing the interestingness of items obtained on the primary feed. We thus need access to data from these feeds that have been seen in the past. Once available, we can use the non-operational criterion to label elements of this feed for use in the next step of our process. In some cases the criterion will focus solely on the future of the primary feed—such as if a story is interesting because a large number of follow-up stories are subsequently observed—or it may require access to one or more secondary feeds—such as stock price data. Further, often the criterion may require derived properties of the feed that are not immediately discernible directly from the feed. For example, if the feed is tick-level reporting of stock trades, but the criterion refers to the average normalized stock return over a one-hour period, some manipulation of the data will be necessary before the process can continue.

Once the data have been obtained and transformed into suitable form, they can be labeled using the interestingness criterion. This is performed in a fairly straight-forward fashion. For each information item in the data generated for the primary data feed, pretend that it has just appeared. The items that follow it chronologically represent the future that is about to follow the given item. Given access to the item, as well as the other information items that followed it (the item’s “future”), it becomes possible to use the user’s importance criterion to assess this item. The result is a corpus of information items from the past, each labeled by whether it is deemed important according to the user’s criterion.

2.3 Applying Machine Learning

Once the data have been labeled, it is now possible to apply machine learning algorithms to them. Note that all knowledge of the future is embodied in the label associated with each item. The result of learning therefore looks at the item—with no information about the future—and makes a prediction about what the non-operational criterion will yield on that item. In other words, the result of learning represents an operationalized (albeit perhaps approximate) form of the importance criterion that can be used directly on items obtained from the information feed.

The selection of a learning method depends tremendously on the nature of the information feeds. If each item is a collection of numerical values (i.e., attribute/value data), learning methods suitable for such data would be used. In many cases—including those considered in the remainder of this paper—each information item is a text object, and thus text classification methods can be used to form the operationalized importance criterion. The accuracy of any such learning method will be impacted in no small part by the extent to which the contents of each information item provide clues to what the non-operational criterion may predict. Without at least some correlation of this sort the operationalization process should perform no better than random prediction. An assessment of the extent to which such correlations exist will usually take place at this stage.

This assessment is impacted by the fact that the data are temporal in nature. In particular, any estimates of the expected predictive accuracy of a learned model must be made on data that appeared later in time than the training data. Cross-validation methods are thus not appropriate for use in this context—evaluation must instead guarantee that all test examples appeared chronologically later than all training examples.

2.4 Analysis

Learning a model is only one part of the overall goal of using this framework. Also important is an analysis of the learned model, to gain insight into what actually was learned. This is important for two reasons. First and foremost, an analysis can be used to validate that the learned model actually has learned the criterion and does not reflect less-meaningful artifacts present in the data. Second, it can be used to gain insight into how the criterion works and can be used to *explain* what is happening in the model. The final step of our framework thus consists of analyzing the learned model, both with respect to how well it appears to match up with our intuitions about what the non-operational criterion was encoding, as well as simply with respect to whether it appears sufficiently plausible that the user would be willing to place some trust in it.

Performing such an analysis will depend substantially on the form of the learned model. A number of researchers have used machine-learning methods to extract interpretable models from difficult-to-understand models, such as complicated expert systems [4], neural networks [3], and ensemble classifiers [5]. Similarly, in our case studies we use the difficult-to-understand, word-based model to relabel the documents. This time, the labels correspond to the predictions of the word-based model. We then use the Ripper rule-learning system [1, 2] to learn, from these relabeled data, (more) interpretable approximations (explanations) of the word-based models. As we will see, these new models seem more interpretable, but still are not satisfactory to domain experts (and so we do a little more analysis).

3 Case Study: Stock Movement

In our second case study we consider a problem that correlates news stories with stock price data. We began with a problem that has been studied by others [7, 13], labeling a story interesting if the stock price of any company mentioned in this news story changes in a way prespecified as being interesting. Rather than inheriting from prior work a definition of an interesting change, to evaluate our four-step approach we "started from scratch", going to an expert on financial information systems to obtain his proposed non-operational criterion for this concept.

3.1 Specifying the Non-Operational Criterion

This case study goes beyond the first case study in one important respect, using a secondary information feed as the basis for assessing the interestingness of a news story. Unlike the previous case, the non-operational criterion can be stated more crisply. For each company's stock, we compute the mean and standard deviation of its one-hour return (relative change in price). We then label a story as "important" if the return of any stock mentioned in the story in the hour after the story appeared was more than one standard deviation from the norm. Note that this means that stories whose stock dropped as well as stories whose stock rose were included as being "important". We chose to combine the two cases to increase the amount of data for the minority class.[1]

3.2 Generating and Labeling Data

For this case study, we consider news stories from a set of public newswires (including Business Wire, Canada NewsWire, CCN Disclosure, Internet Wire, PR Newswire, PrimeZone, and Reuters) as well as a stock price news feed. Each story averages roughly 400–500 words, and each has been analyzed to extract its complete date stamp as well as the stock symbols of all the companies mentioned in that news story. For the purpose of our experiments, we used a more manageable-sized subset of these data, limited to 50,158 stories that appeared between January 5, 1999 and September 14, 1999, where stories with incomplete time stamps (both date and time of day), duplicate stories, and stories that mentioned more than eight companies (typical for stories that discuss the market in general rather than a particular stock or segment) removed. We further take only those that appeared during normal trading hours (excluding those appearing in the final hour of trading, for consistency in evaluation), leaving 39,167 stories. Thus we remove data that are guaranteed to have no change in stock-price values since normal trading has ceased.[2]

The second source of information is trade-level data for over 8000 publically traded companies on the NYSE, AMEX and NASDAQ exchanges. We have data collected since January, 1993, and we use this full data set to calculate the one-hour mean and standard deviation for each stock. Due to the enormous amount of data, we aggregate into 5-minute intervals—for each stock we maintain its price at both the start and end of the interval, as well as its trading volume during that interval. At this point we can apply the non-operational criterion to the data obtained between January 5, 1999 and September 14, 1999 (the dates of the selected stories), resulting in 3608 of the stories being labeled as important.

3.3 Applying Machine Learning

Given data labeled with our non-operational criterion, we can then proceed to the learning step. To evaluate how well learning performs we run our learning methods on a per-day basis. For each day we use as training data all stories that appeared before it, skipping all data that appeared earlier in the same day or in the immediately preceding day. (The reason for imposing a gap was to minimize the risk that learning will perform well due

[1] We did, in fact, run experiments for predicting directional importance separately, with similar results as shown here.

[2] Note that much news that is important to the market is released "after the bell."

to occasional duplicate stories, something that is rarely the case for stories that are more than a day apart.)[3]

The criterion we use labels only a small number of stories as important. If testing begins too early in the historical data feed, there is the chance that there may be few or no relevant examples of the minority class to learn from. In order for the learner to have sufficient training data, our evaluations thus begin at the chronological date where at least half of the "important" stories will be in the training set. This meant that only 20,249 instances were being tested with 1461 of those being "important".

To evaluate the ability of a learning method to form the approximate operationalization of the importance criterion we present our results using ROC curves. ROC analysis is an evaluation technique used in signal detection theory, which has seen increasing use for other types of diagnostic, machine-learning, and information-retrieval systems [25, 19, 18]. ROC graphs plot false-positive rates on the x-axis, and true-positive rates on the y-axis. ROC curves are generated in a similar fashion to precision/recall curves, by varying a threshold across the output range of a scoring model, and observing the corresponding classification performances. Although ROC curves are isomorphic to precision/recall curves, they have the added benefits that they are insensitive to changes in marginal class distribution, and that the area under the ROC curve has a well-defined statistical meaning [9].

Although we used a range of standard text categorization algorithms, all performed roughly comparably. Due to their relatively quick run times we therefore only report on results using the Naive Bayes [6] and TFIDF [23, 21] classification methods.[4] Naive Bayes estimates the *a posteriori* probability that an example belongs to a class given the observed feature values of the example, assuming the independence of the features given the class label. The class with the maximum *a posteriori* probability is assigned to the example. The TFIDF classifier [21, 12, 24] is based on Rocchio's [20] relevance feedback algorithm. A prototype vector is formed for each class from the positive and negative examples of that class. To classify a new document d, the cosines of the prototype vectors with the corresponding document vector are calculated for each class. The scores are normalized to sum to one, and d is assigned the score it is given by the class with the greater score. The resulting ROC curves are shown in Figure 1. It shows that, regardless of what ultimately is the appropriate trade-off between false positives and false negatives, it appears that there is sufficient information in the two information sources to be able to predict considerably better than random.

Whether this prediction is good enough depends, of course, on how it will be used. Different users have different spans of attention and different needs. The ROC curves show that if the stories were to be ranked solely by this single estimation of importance (which more generally would be a component of a greater definition of interestingness), the top of the ranking would be substantially denser with important stories than would the bottom of the ranking. Any user restricted to examining only a subset of the stories would examine considerably more important stories. To be specific, each day there are on average about 266 stories (using our corpus, which contains a subset of all the business news), and about 19 of them will be important by the current definition. Without ranking, if a user selected (randomly) 14 stories, 1 would be important (or, roughly 7.22% of the stories are important). With ranking, if a user selected the top 14 stories, 2 would be important: an increase in precision of 100%. As the user goes further down

[3] We did in fact do some experiments without imposing the one-day gap, and observed little effect on the performance of the learned model.

[4] We used the versions of these learners found in the publicly available Rainbow package [16].

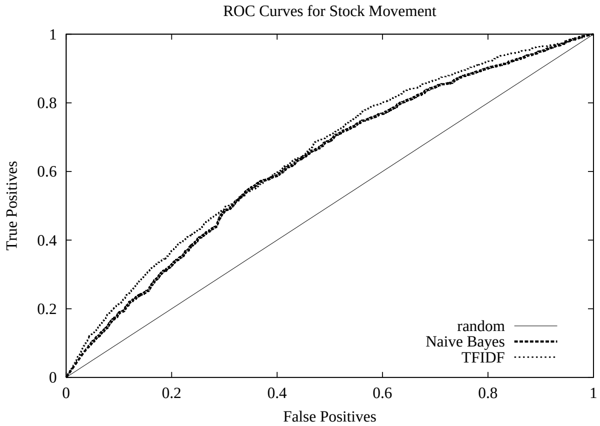

Fig. 1. ROC Curve for stock movement

the list, this increase changes. Without ranking, if a user selected (randomly) 100 stories, 7 would be important. With ranking, if the user selected the top 41 stories, 7 would be important (17% precision!), about 150% improvement in precision.

Of course, some users may have to read all the news stories (often subject to other filtering criteria). It seems initially that such users would not benefit from such triage. However, this conclusion ignores the issue of timeliness. At any point, a user will have a queue of news stories pending examination. A triage system would maintain a priority queue of news stories, and even users who eventually must read all the stories may benefit in terms of timeliness of information: important stories are more likely to be inserted higher in the priority queue.

3.4 Analysis

As mentioned previously, it is also important to understand the result of the learning process. The original criterion is specified with respect to future, as yet unseen information, but its learned form only refers to information present in the given information item. It is important for a user to have confidence that the operationalized criterion matches—even if only in part—the intentions of the original non-operational criterion.

If the learning methods generated interpretable results, it may be possible to inspect the results directly to understand what aspects of an information item are correlated with the non-operationalized criterion. However, there is no guarantee that such methods will actually be used in practice, for example if the learning method that yields interpretable results runs slowly. Our experiments represent such a case, where we use relatively fast methods that combine scores on words in a holistic fashion, making it difficult to interpret how they behave.

To understand the results of the operationalization process better we approximate the learned classifier using a learning method whose output is more understandable. We step through a collection of data on a day by day basis, as was described in the previous section. Each day's data are labeled by the results of learning from the earlier days' data. As a result, on a day-by-day basis, we have the "compiled wisdom" of the learned model, as seen in how it labels the data to which it is applied. That labeled data can then be used as input to a learner that will give more interpretable results.

To demonstrate this approach to analysis we used it to understand the results of the Naive Bayes classifier. This was done using four steps:

1. For each day we used Naive Bayes to label that day's data using earlier data, doing so for the entire data set.

· statements months share
· president business financial
· announced differ board
· announced president officer senior served
· statements cash gain announces
· announced statements differ press division
· statements development
· announced statements executive terms
· contact president headquartered corporation
· today officer president management business
· announced president officer chairman
· today services management companys financial corporation
· contact president made results
· contact president investors vice recent
· contact board cash
· contact board directors security
· today products international manufacturing
· services management nyse trading
· contact release differ performance
· prnewswire share alerts
· announced president officer countries
· announced release terms international
· services management board industry senior
· announced president technology investment key
· contact informed
· sales net cost
· company actual reuters provided
· announced president technology corp line
· contact industry wire
· services management president forward
· today services acquisition acquired
· services management serving investment
· services today announces board
· president officer directors
· today business statements changes

Table 1. Ripper rules for stock movement

2. We extract the top 250 stemmed words from all the data using standard entropy-based measures.
3. For each news story, we remove words whose stem is not present in the top 250 words.
4. The resulting labeled data were then given to Ripper [1, 2], a learning system that forms rules, a representation that is perceived by many as being more understandable. Ripper was run with a Loss-Ratio of 0.5 in order to form more rules.

Table 1 shows the 35 rules generated by Ripper. Although it is satisfying that many of the rules appear to be plausible expressions of circumstances that could lead to stock price movements, in this domain we have additional resources we can use to understand these rules.

In particular, we would like to understand if there is a more general phenomenon underlying the words in these rules. To answer this question we build off a taxonomy from the accounting literature that labels each story with one or more from a list of 12 categories [11]. They include:

Percent	Category	Percent	Category
21%	E: Earnings announcements	11%	F: Forecast
17%	D: Dividend announcements	10%	C: Capital/ownership changes
15%	P: Product related	9%	M: Management related
11%	S: Asset changes		

(The other five categories label 2% of the stories or less.)[5]

In order to use this taxonomy we focused on two prototypical rules that appeared to have some significance in terms of the words within them. In each case we hand-labeled each story that the rule matched with all of the categories that appeared to apply to it.

In the case of the first rule we selected, announced president officer senior served ⟶ interesting, our hypothesis was that this rule indicates a management change, and indeed, 92% of the matched stories were management related—mostly management changes, although a few concerned matters such as managers receiving awards.

[5] Coverage figures reflect stories from their 1987 study, and will likely differ for our own story corpus. However, it is the categories that we focus on here and not on these 1987 figures.

In the case of our second rule, announced president technology investment key ⟶ interesting, our conjecture was that stories that match it concern some important technology-related announcement, such as a new product or a joint venture with a technology company. Indeed, 95% of the stories are technology related, although this may not be too surprising given the prevalence of information technology and biotechnology in the market place. More telling, 79% of the stories are product-related and/or asset related (joint venture, merger, etc.). (Although the stories for the first rule were typically unambiguous in labeling, these typically could be labeled either as asset related or product related—for example, joint ventures to produce technology products, acquisitions to get technology products, etc.) On the other hand, none of the other categories covered more than 17% of the stories.

These results suggest that if we could learn to recognize the categories already identified in the accounting literature include these in our learning process we may be able to further improve our results. Our ongoing work explores our ability to directly label stories with categories, as well as continuing our labeling process, examining the distribution of categories for the stories covered by each rule. We are hopeful that this exercise will lead towards a tentative theory of the relative importance of various classes of stories.

4 Final Remarks

This paper introduced a four-step process for identifying information items that may be important based on their correlation with the occurrence of subsequent events. The paper further presented two case studies of this approach concerning news stories— recognizing “hot stories” that have many similar stories following them, and recognizing stories that mention a stock that will have a significant movement in value.

While the first steps of our process are fairly well understood, we have only started on the final step of the analysis to get a better understanding of the resulting models. While the analysis presented in this paper presents us with a good set of human-understandable rules that give us a sense of plausibility for the learned models, it still leaves something to be desired with respect to actually being able to *explain* and understand the final model or gaining any insight into what makes the criterion work. We are currently working on more elaborate techniques to discern the underlying rules and correlations, to get a better understanding of the domain and criteria presented in this paper.

Acknowledgments

We thank Tom Fawcett for all his help, with conceptualization and with procuring and dealing with the data. We thank Stephen Ryan and Gideon Saar for helping us to begin to understand the effects of firm-specific news on market performance, and for directing us to the literature. We are grateful to IBM for a Faculty Partnership Award. Ted Stohr suggested that we look at ”hot” stories, as a measure of importance.

References

1. W. W. Cohen. Fast effective rule induction. In *Proceedings of the Twelfth International Conference on Machine Learning*, Lake Tahoe, California, 1995.
2. W. W. Cohen. Learning trees and rules with set–valued features. In *AAAI96*, 1996.
3. M. W. Craven and J. W. Shavlik. Extracting tree-structured representations of trained networks. In *Advances in Neural Information Processing Systems*, pages 24–30, 1996.
4. A. Danyluk and F. Provost. Small disjuncts in action: Learning to diagnose errors in the telephone network local loop. In *Proceedings of the Tenth International Conference on Machine Learning*, pages 81–88, San Mateo CA, 1993. Morgan Kaufamnn.

5. P. Domingos. Knowledge acquisition from examples via multiple models. In *Proceedings of the Fourteenth International Conference on Machine Learning*, pages 98–106, 1997.
6. P. Domingos and M. Pazzani. Beyond independence: Conditions for the optimality of simple bayesian classifier. In *Proceedings of the 13th International Conference on Machine Learning*, pages 105–112, 1996.
7. T. Fawcett and F. Provost. Activity monitoring: Noticing interesting changes in behavior. In *Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining*, 1999.
8. P. W. Foltz and S. T. Dumais. Personalized information delivery: an analysis of information filtering methods. *Communications of the ACM*, 35(12):51–60, Dec. 1992.
9. D. J. Hand. *Construction and Assessment of Classification Rules*. Chichester:John Wiley and Sons, 1997.
10. E. M. Houseman and D. E. Kaskela. State of the art of selective dissemination of information. *IEEE Transactions on Engineering Writing and Speech*, 13(2):78–83, 1970.
11. R. B. T. II, C. Olsen, and J. R. Dietrich. Attributes of news about firms: An analysis of firm-specific new reported in the *wall street journal index*. *Journal of Accounting Research*, 25(2), 1987.
12. T. Joachims. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning*, 1997.
13. V. Lavrenko, M. Schmill, D. Lawrie, P. Ogilvie, D. Jensen, and J. Allan. Language models for financial news recommendation. In *Proceedings of the Ninth International Conference on Information and Knowledge Management*, pages 389–396, 2000.
14. S. A. Macskassy, H. Hirsh, F. Provost, R. Sankaranarayanan, and V. Dhar. Intelligent information triage. In *The 24th Annoual Internationl Conference on Research and Development in Information Retrieval (SIGIR)*, New Orleans, LA, September 2001. To Appear.
15. C. Marshall and F. Shipman. Spatial hypertext and the practice of information triage. In *Proceedings of the '97 ACM Conference on Hypertext*, pages 124–133, Apr 1997.
16. A. K. McCallum. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. `http://www.cs.cmu.edu/~mccallum/bow`, 1996.
17. D. J. Mostow. Machine transformation of advice into a heuristic search procedure. In *Machine Learning: An Artificial Intelligence Approach*, pages 367–403. Morgan Kaufmann, 1983.
18. K.-B. Ng and P. Kantor. Predicting the effectiveness of naive data fusion on the basis of system characteristics. *Journal of American Society for Information Science*, 51(13):1177–1189, 2000.
19. F. Provost and T. Fawcett. Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions. In *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, pages 445–453, 1997.
20. J. Rocchio. Relevance feedback in information retrieval. In Salton, editor, *The SMART Retrieval System: Experiments in Automatic Document Processing*, chapter 14, pages 313–323. Prentice–Hall, 1971.
21. G. Salton and C. Buckley. Term weighting approaches in automatic text retrieval. Technical report, Department of Computer Science, Cornell University, 1987.
22. G. Salton and C. Buckley. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41:288–297, 1990.
23. G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., 1983.
24. R. Schapire, Y. Singer, and A. Singhal. Boosting and rocchio applied to text filtering. In *Proceedings of ACM SIGIR*, pages 215–223, 1998.
25. J. Swets. Measuring the accuracy of diagnostic systems. *Science*, 240:1285–1293, 1988.