# Context and Learning in Novelty Detection

**Barry Schiffman** and **Kathleen R. McKeown**
Department of Computer Science
Columbia University
New York, N.Y.
{bschiff,kathy}@cs.columbia.edu

## Abstract

We demonstrate the value of using context in a new-information detection system that achieved the highest precision scores at the Text Retrieval Conference's Novelty Track in 2004. In order to determine whether information within a sentence has been seen in material read previously, our system integrates information about the context of the sentence with novel words and named entities within the sentence, and uses a specialized learning algorithm to tune the system parameters.

## 1 Introduction

New-information detection addresses two important problems in a society awash in more digital information than people can exploit. A novelty detection system could help people who are tracking an event in the news, where numerous sources present similar material. It could also provide a way to organize summaries by focusing on the most recent information, much like an automated bulletin service.

We envision that many types of users would find such a system valuable. Certainly analysts, business people, and anyone interested in current events, would benefit from being able to track news stories automatically, without repetition. Different news organizations report on the same event, often working hard to make their reports look different from one another, whether or not they have new material to report. Our system would help readers to zero in on new information. In addition, a focus on new information provides a way of organizing a general summary.

Our approach is unique in representing and maintaining the focus in discourse. The idea stems from the fact that novelty often comes in bursts, which is not surprising since the articles are composed of some number of smaller, coherent segments. Each segment is started by some kind of introductory passage, and that is where we expect to find the *novel* words. Novel words are identified by comparing the current sentence's words against a table of all words seen in the inputs to that point. They let us know whether the entire segment is likely to contain more novel material. Subsequent passages are likely to continue the novel discussion whether or not they contain novel words. They may contain pronomial references or other anaphoric references to the novel entity. Our long-term goal is to integrate the approach described in this paper into our larger new-information detector, a system that performs a more complicated syntactic analysis of the input texts and employs machine learning to classify passages as new or old.

Meanwhile, we tested our focus-based approach at the Novelty Track at the Text Retrieval Conference(TREC) in 2004. The Novelty Tracks in 2003 and in 2004 were divided into four tasks; Task 1 and Task 3 incorporate retrieval, requiring submissions to locate the relevant sentences before filtering them for novelty. Tasks 2 and 4 are novelty detection alone, using the relevant sentences selected by humans as input. Since our interest is in nov-

elty detection, we chose to concentrate on Task 2[1] Our TREC submission was also designed to test a specialized learning mechanism we implemented to target either high precision or high recall.

In all, the problem of novelty detection is deceptively difficult. We were struck by the difficulty that all groups in the Novelty Track in 2002 and 2003 had in obtaining high precision scores. Submissions that classify a very large proportion of the input sentences as novel reached the highest F-measure scores by getting high recall scores, but failed to achieve any substantial compression of material for users. Given that our goal is to generate an update summary, we focused on improving precision and increasing compression, removing as many false positives as possible.

The next section discusses the Novelty Track and the approaches others have tried; Section 3 details our system, and Section 4 presents the experiments.

## 2 Novelty Track

Much of the work in new-information detection has been done for the TREC Novelty Track. The task is related to first story detection, which is defined on whole documents rather than on passages within documents. In Task 1 of the Novelty Track, a system is given about 25 documents on a topic and asked to find all sentences relevant to the topic. In Task 2, the inputs are the set of relevant sentences, so that the program does not see the entire documents. The program must scan the sentences in order and output all that contain new information, that is information not seen in the previous input sentences.

### 2.1 Related Work

At the recent TREC, Dublin City University did well by comparing the words in a sentence against the accumulated words in all previous sentences (Blott et al., 2004). Their runs varied the way in which the words were weighted with frequency and inverse document frequency. Like our system, theirs follows from the intuition that words that are new to a discussion are evidence of novelty. But our system distinguishes between several kinds of words, including common nouns, named persons, named organization, etc. Our system also incorporates a mechanism for looking at the context of the sentence.

Both the Dublin system and ours are preceded by the University of Iowa's approach at TREC 2003. It based novelty decisions on a straightforward count of new named entities and noun phrases in a sentence (Eichmann et al., 2003). In 2004, the Iowa system (Eichmann et al., 2004) tried several embellishments, one using synonyms in addition to the words for novelty comparisons, and one using word-sense disambiguation. These two runs were above average in F-measure and about average in precision.

The University of Massachusetts system (Abdul-Jaleel et al., 2004) mixed a vector-space model with cosine similarity and a count of previously unseen named entities. Their system resembled one of two baseline methods that we submitted without our focus feature. Their submission used a similarity threshold that was tuned experimentally, while ours was learned automatically. In earlier work with the TREC 2002 data, UMass (Allan et al., 2003) compared a number of sentence-based models ranging in complexity from a count of new words and cosine distance, to a variety of sophisticated models based on KL divergence with different smoothing strategies and a "core mixture model" that considered the distribution of the words in the sentence with the distributions in a topic model and a general English model.

A number of groups have experimented with matrix-based methods. In 2003, a group from the University of Maryland and the Center for Computing Sciences (Conroy et al., 2003) used three techniques that used QR decomposition and singular value decomposition. The University of Maryland, Baltimore County, worked with clustering algorithms and singular value decomposition in sentence-sentence similarity matrices (Kallurkar et al., 2003). In 2004, Conroy (Conroy, 2004) tested Maximal Marginal Relevance (Goldstein et al., 2000) as well as QR decomposition.

The information retrieval group at Tsinghua University used a pooling technique, grouping similar sentences into clusters in order to capture sentences that partially match two or more other sentences(Ru et al., 2004). They said they had found difficulties

[1]Task 4 was similar to Task 2, in that both have the human annotations as input. For Task 2, that's all participant get, but in Task 4, they also receive the novel sentences from the first five documents as input. We felt that we would learn as much from the one task as from both.

with sentence-by-sentence comparisons.

### 2.2 Precision

At all three Novelty Track evaluations, from 2002 to 2004, it is clear that high precision is much harder to obtain than high recall. Trivial baselines – such as accept all sentences as novel – have proven to be difficult to beat by very much. This one-line algorithm automatically obtains 100% recall and precision equal to the proportion of novel sentences in the input. In 2003, when 66% of the relevant sentences were novel, the mean precision score was 0.635[2] and the median was 0.7. In 2004, 41% of the relevant sentences were novel, and the average precision dropped to 0.46. The median precision was also 0.46. Meanwhile, average recall scores across all submissions actually rose to 0.861 in 2004, compared with 0.795 in 2003. In terms of a real world system, this means that as the number of target sentences shrank, the number of sentences in the average program output rose. Likewise, a trivial system could guarantee no errors by returning nothing, but this would have no value.

### 2.3 Sentences

Normally, in Information Retrieval tasks, stricter thresholds result in higher precision, and looser thresholds, higher recall. In that way, a system can target its results to a user's needs. But in new-information detection, this rule of thumb fails at some point as thresholds become stricter. Recall does fall, but precision does not rise. In other words, there seems to be a ceiling for precision.

Several participants noted that their simpler strategies produced the best results. For example, in 2003, the Chinese Academy of Sciences (Sun et al., 2003), noted that word overlap was surprisingly strong as a similarity measure. As we have seen above, the Iowa approach of counting nouns was incorporated by a few others for 2004, including us. This strategy compares words in a sentence against all previous seen words and thus, avoids computing pairwise similarity between all sentences. Almost all participants performed such pairwise comparisons of systems.

A sentence-by-sentence comparison is clearly not the optimal operation for establishing novelty. Sentences with a large amount of overlap can express much different thoughts. In the extreme, a single word change can reverse the meaning of two sentences: *accept* and *reject*. This phenomenon led the Tsinghua University group to remark, "many sentences with an overlap of nearly 1 are real novel ones." (Ru et al., 2004).

On the other hand, it's not hard to find cases where realizations of equivalent statements take many different surface forms – with different choices of words and different syntactic structures. The data in the Novelty Task is drawn from three news services and clustered into fairly cohesive sets. The news writers consciously try to avoid echoing each other, and over time, echoing themselves. Sentences such as these have low word overlap, but are not novel. For this reason, we turned to a strategy of classifying each sentence $S_i$ against the cumulative background of all the words in all preceding sentences $S_{1...i-1}$.

## 3 System

The system described in this paper was built with the Novelty Track in mind. The goal was to look at ways to consider longer spans of text than a sentence, and to avoid sentence by sentence comparisons.

In the Novelty track, the relevant sentences are presented in natural order, i.e. by the date of the document they came from, and then by their location in the document. Our program:

- For each relevant sentence, our program calculates a sum of novel terms, which are terms that have not been previously seen. The terms are weighted according to their category, like person, location, common noun or verb. The weights are learned automatically.

- For the entire set, the program maintains a focus variable, which indicates whether the previous sentence is novel or old. Thresholds determine whether to continue or shift the focus. These are also learned automatically.

All input documents are fed in parallel into a named-entity recognizer, which marks persons, or-

[2] One group appeared to have submitted a large number of irrelevant sentences in its submission, since it obtained relatively high recall scores, but very low precision scores, causing the average to drop below 0.66. The average precision of all other groups is about 0.7.

ganizations, locations, part-of-speech tags for common nouns, and into a finite-state parser, which is used only to identify sentences beginning with subject pronouns. The output from the two preprocessing modules are merged and sent to the classifier.

The classifier reads a configuration file that contains a set of weights to apply to different classes of words that have not been previously seen.

For each sentence, the system adds up the amount of novelty from the weighted terms in a sentence and compares that to a learned threshold; it classifies the sentence as novel if it exceeds the threshold. It also stores the classification in a focus variable. If the novelty threshold is not met, the system performs a series of tests described below, and possibly classifies some sentences with few content words as novel, depending on the status of the focus variable. We are trying to cover all cases of changes in focus, and to test these in the order that allows the system to make the decision it can be most confident about first. Thus, when we find a named entity new to the discussion, we can be pretty sure that we have found a novel sentence. We can classify that sentence as new without regard to what preceded it. But, when we find a sentence devoid of high-content words, like "She said the idea sounded good," the system uses the classification of the previous sentence. If the antecedants to *she* or *idea* are novel, then this sentence must also be novel. The series of learned thresholds are imposed in a cascade to maximize the number of correct decisions over the training cases, in hopes the values will also cover unseen cases.

Thus, the classifier puts each sentence through the tests below, using the learned thresholds and weights described in Section 3.1. If any test succeeds, the system goes on to the next sentence.

1. *If there is a sufficient concentration of novel words, classify the sentence as novel* A sufficient concentration occurs when the sum of the weights of the novel content words (including named entities) exceeds a threshold, $T_{novel}$. If the previous focus was old, this indicates the focus has shifted to a novel segment.

2. *If there is a lack of novel words, classify the sentence as old* This is computed by comparing the sum of the weights of the already-seen content words to a separate threshold, $T_{old}$. If the previous focus was novel, this means the focus has shifted to an old segment.

3. For any remaining sentences, the classification is based on context:

   (a) *If the sentence does not have a sufficient number of content words, use the classification in the focus variable* This adds the sums of both new and old content words and compares that to a threshold, $T_{keep}$.

   (b) *If the first noun phrase is a third person personal pronoun, use the classification in the focus variable* Pronouns are known to signal that the same focus continues (Grosz and Sidner, 1986).

   (c) *If the sentence has not met any of the above tests but has a minimum number of content words, shift the focus* If all tests above fail and there are a minimum number of content words, with a sum of $T_{shift}$ shift the focus.

4. *Default* This rarely occurs but the default is to continue the focus, whether novel or old.

We examined the 2003 Novelty Track data and found that more than half the novel sentences appear in sequences of consecutive sentences (See Table 1). This circumstance creates an opportunity to make principled classifications on some sentences that have few, if any, clearly novel words, but continue a new segment. The use of a focus variable handles these cases.

### 3.1 Learning

In all, the system uses 11 real-valued parameters, weights and thresholds, and we wanted to learn optimal values for these. In particular, we wanted to be able to target either high recall or high precision, As we noted above, precision was much more difficult, and for a summarization task, much more important.

To learn the optimal values for the parameters, we opted to use an ad hoc algorithm. The main advantage in doing so was when considering instance $i$, the program can reference the classifications made for instance $i-1$, $i-2$, and possibly all the way back to instance 1, because the classification for instance $i$ partly depends on the classification of pre-

| Length of Run | Count |
|---|---|
| 1 | 1338 |
| 2 | 421 |
| 3 | 132 |
| 4 | 72 |
| 5 | 43 |
| 6 | 22 |
| 7 | 11 |
| 8 | 2 |
| 9 | 3 |
| 10 | 3 |
| 11 | 2 |
| 12 | 2 |
| 15 | 2 |
| 17 | 1 |

Table 1: Novelty often comes in bursts. This table shows that 1,338 of the novel sentences in the 2003 evaluation were singletons, and not a part of a run of novel sentences. Meanwhile, 1,526 of the sentences were part of runs of 2, 3 or 4 sentences.

vious instances. Not only do many standard supervised learning methods assume conditional independence, but they also do provide access to the online classifications during learning. We decided to construct a randomized hill-climbing. The learner is structured like a neural net, but the weight adjustments are chosen at random as they are in genetic algorithms (See Figure 1). The evaluation, or fitness function, is the Novelty Track score itself, and the training data was the 2003 Novelty Track data.

Changes to the hypothesis are selected at random and evaluated. If the change does not hurt results, it is accepted. Otherwise the program backtracks and chooses another weight to update. At first, we required the new configuration to produce a score greater than the previous one before we accepted it. But we altered this to accept configurations that produce scores equal to the previous one. The choice of which weight to update is made at random, in an effort to avoid local minima in the search space, but with an important restriction: the previous $n$ choices are kept in a history list, which is checked to avoid re-use. This list is updated at each iteration. The configurations usually converge well within 100 iterations.

1. Initialize weights, history
   Weights take random values
2. Run the system using current weight set
3. If current score $>=$ previous best
   Update previous best
4. Otherwise
   Undo move
5. Update history
6. Choose next weight to change
7. Go to step 2

Figure 1: The learning algorithm uses a randomized hill climbing approach with backtracking

### 3.2 Bias Adjustment

In training on the 2003 data, the biggest problem was to find a way to deal with the large percentage of novel sentences. About 65% of the instances are positive, so that a random system achieves a relatively high F-measure by increasing the number of sentences it calls novel – until recall reaches 1.0. Another strategy would be to choose only the sentences in the first document, achieving a high precision – more than 90% of the relevant sentences in the first document for each topic were called novel.

In the Novelty Track the F-measure was set to give equal weight to precision and recall, but we wanted to be able to coax the learner to give greater weight to either precision or by adjusting the F-measure computation:

$$F = \frac{1}{\frac{\beta}{prec} + \frac{(1-\beta)}{recall}}$$

$\beta$ is a number between 0 and 1. The closer it gets to 1, the more the formula favors precision.

We chose whether to emphasize precision or recall by altering the value of $\beta$. At the most extreme, we set $\beta$ at 0.9 for the largest emphasis on precision. When emphasizing recall, we left $\beta$ at 0.5.

The design was motivated by the need to explore the problem more fully and inform the algorithm for deciding novelty as much as to find optimal parameters for the values. Thus, we wanted to be able to record all the steps the learner made through the search space, and to save the intermediate states. At times, the learner would settle into a configuration

that produced a trivial solution, and we could choose one of the intermediate configurations that produced a more reasonable score.

### 3.3 Vector-Space Module

In addition to the system which integrates novel word features with focus tracking, we also implemented a vector-space approach as a baseline – the *Cosine* run. We tested the vector-space system alone to contrast it with the focus system, but we also tested a version which integrated the vector-space system with the focus system.

Our vector-space module assigns all non-stopwords a value of 1, and uses the cosine distance metric to compute similarity.

$$Cos(u, v) = \frac{u \cdot v}{\|u\| \cdot \|v\|}$$

and

$$Novel(s_i) \begin{cases} True & \text{if} \quad Cos(s_i, s_j) < T, \\ & \text{for} \quad j = 1 \ldots i-1 \\ False & \text{otherwise} \end{cases}$$

As each sentence is scanned, its similarity is computed with all previous sentences and the maximum similarity is compared to a threshold $T$. If that maximum exceeds $T$, it is considered novel. We chose the value of $T$ after trials on the 2003 Novelty Track data. It was set at 0.385, resulting in a balanced system that matched the results of one of the strongest performers at the TREC evaluations that year.

On the 2003 data, when we set $T$ at .9, we found that we had a precision of .71 and a recall of 0.98, indicating that about 6% of the sentences were quite similar to some preceding sentence (See Figure 2). After that, each point of precision was very costly in terms of recall. Our experience was mirrored by the participants at TREC 2003 and again at TREC 2004.

We considered this vector-space model to be our baseline. We also tried it in combination with the *Recall* run explained above. Because both the *Recall* and *Cosine* runs produced a relatively large output and because they used different methods, we thought the intersection would result in higher precision, though with some loss of recall.

In practice, the range of recall was much greater than precision. Judging from the experiences of the participants at TREC and our own exploratory experiments, it was difficult to push precision above 0.80 with the TREC 2003 data, and above 0.50 with the TREC 2004 data.

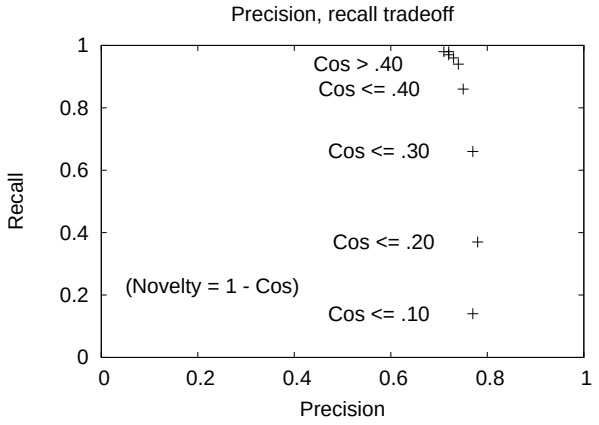


Figure 2: The precision and recall scores of a vector-space model with cosine similarity at different thresholds, on the TREC 2003 data. Making the test for novelty stricter fails to improve precision but has a drastic effect on recall.

## 4 Experiments

### 4.1 Results from TREC 2004

Our results are encouraging, especially since the configurations that were oriented toward higher precision, indeed, achieved the best precision scores in the evaluation, with our best precision run about 20% higher in precision than the best of all the runs by other groups (See Figure 3.) Meanwhile, our recall-oriented run was one of eight runs that were in a virtual tie for achieving the top f-measure. These eight runs were within 0.01 of one another in the measure.

Our five submitted runs were:

**Prec1** aimed at moderately high precision, with reasonable recall.

**Prec2** aimed at high precision, with little attention to recall.

**Recall** weighted precision and recall equally.

**Cosine** a baseline of a standard vector-space model with a cosine similarity metric.

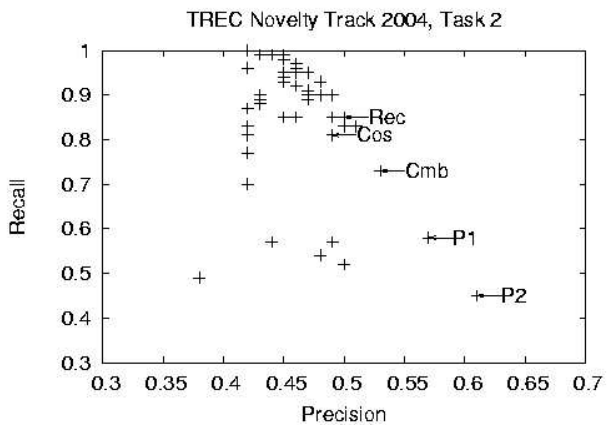


Figure 3: The graph shows all 54 submission in Task 2 for the Novelty Track, with our five submissions labeled. Our precision-oriented runs were well ahead of all others in precision, while our recall-oriented run was in a large group that reached about 0.5 precision with relatively high recall.

**Combo** a composite submission using the intersection of Recall and Cosine.

Table 2 shows the numbers of our performance of our five submissions. *Prec1* had an F-score close to the average of 0.577 for all systems, while *Prec2* was 50% ahead of random selection in accuracy. Both our *Combo* system and our baseline *Cosine* were above average in F-measure. Our emphasis on precision is justified in a number of ways, although the official yardstick was the F-measure.

An analysis of the system's behavior under the different parameters showed that the precision-oriented runs, in particular *Prec1*, valued verbs and common nouns more than named entities in deciding novelty. The precision-oriented runs also benefited more from the focus variable, with their scores about 5% higher in terms of F-measure than they were without it. The pronoun test, however, was rarely used, firing less than 1% of the time.

We note that we are developing novelty detection for summarization, where compression of the report is valuable. Table 2 shows the lengths of our returns. It is impossible to compare these precisely with other systems, because the averages given by NIST are averages of the scores for each of the 50 sets, and we do not have the breakdown of the numbers by set for any submissions but our own. However, we can estimate the size of the other output by considering average precision and recall as if they were computed over the total number of sentences in all 50 sets. This computation shows an average output for all participants of about 6,500 sentences and a median of 6,981 – out of a total of 8,343 sentences. However, this total includes some amount of header material, not only the headline, but the document ID and other identifiers, the date and some shorthand messages from the wire services to its clients. In addition, a number of the sets had near perfect duplicate articles. This is in sharp contrast with typical summaries. At the 2004 Document Understanding Conference, the typical input cluster contained more than 4,000 words, and the task required that this be reduced to 100 words. We contend there is little value in a system that does no more than weed out very few sentences, even though they might have achieved high F-measures.

Second, our experience, and the results of other groups, shows that high precision is harder than high recall. In all three years of the Novelty Track, precision scores tended to hover in a narrow band just above what one would get by mechanically labeling all sentences as *novel*.

## 5 Conclusion

The success of our use of context in the TREC Novelty Track led us to incorporate the idea into a larger system. This system identifies *clauses* within sentences that express new information and tries to identify semantic equivalents. It is being developed as part of a multi-document summarizer that produces topical updates for users.

In addition, the work here suggests three directions for future work:

- Adapt the features used here to some of the newer probabilistic formalisms, like conditional random fields.
- Try full segmentation of the input documents rather than treat the sentences as a sequence.
- Try to identify all nominal references to canonical forms.

Still, with this experimental system, we obtained the the top precision scores in the Novelty Track,

| Run-Id | Precision | Recall | F-meas | Output length |
|---|---|---|---|---|
| Prec1 | 0.57 | 0.58 | 0.562 | 3276 |
| Prec2 | 0.61 | 0.45 | 0.506 | 2372 |
| Recall | 0.51 | 0.82 | 0.611 | 5603 |
| Cosine | 0.49 | 0.81 | 0.599 | 5537 |
| Combo | 0.53 | 0.73 | 0.598 | 4578 |
| Choose All | 0.41 | 1.000 | 0.581 | 8343 |
| Average All Runs | 0.46 | 0.86 | 0.577 | 6500 |

Table 2: Comparison of results of our five runs, compared to a random selection of sentences, and the overall average F-scores by all 55 submissions.

and we obtained the program settings to do this automatically. High precision is, nonetheless, very difficult to obtain, and every point in precisions costs too much in recall. Further exploration is needed to determine whether linguistic knowledge will help, and whether state-of-the-art tools are powerful enough to improve performance.

Beyond new-information detection, the idea of tracking context with a surface means like the focus variable is worth exploring in other tasks, including summarization and question-answering.

## References


Nasreen Abdul-Jaleel, James Allan, W. Bruce Croft, Fernando Diaz, Leah Larkey, Xiaoyan Li, Donald Metzler, Mark D. Smucker, Trevor Strohman, Howard Turtle, and Courtney Wade. 2004. Umass at trec 2004: Notebook. In *The Thirteenth Text Retrieval Conference (TREC 2004) Notebook*.

James Allan, Courtney Wade, and Alvaro Bolivar. 2003. Retrieval and novelty detection at the sentence level. In *Proceedings of the ACM SIGIR conference on research and development in information retrieval*.

Stephen Blott, Oisin Boydell, Fabrice Camous, Paul Ferguson, Georgina Gaughan, Cathal Gurrin, Noel Murphy, Noel O'Connor, Alan F. Smeaton, Barry Smyth, and Peter Wilkins. 2004. Experiments in terabyte searching, genomic retrieval and novelty detection for trec-2004. In *The Thirteenth Text Retrieval Conference (TREC 2004) Notebook*.

John M Conroy, Daniel M. Dunlavy, and Dianne P. O'Leary. 2003. From trec to duc to trec again. In *TREC Notebook Proceedings*.

John M. Conroy. 2004. A hidden markov model for trec's novelty task. In *The Thirteenth Text Retrieval Conference (TREC 2004) Notebook*.

David Eichmann, Padmini Srinivasan, Marc Light, Hudong Wang, Xin Ying Qiu, Robert J. Arens, and Aditya Sehgal. 2003. Experiments in novelty, genes and questions at the university of iowa. In *TREC Notebook Proceedings*.

David Eichmann, Yi Zhang, Shannon Bradshaw, Xin Ying Qiu, Li Zhou, Padmini Srinivasan, Aditya Kumar Sehgal, and Hudon Wong. 2004. Novelty, question answering and genomics: The university of iowa response. In *The Thirteenth Text Retrieval Conference (TREC 2004) Notebook*.

Jade Goldstein, Vibhu Mittal, Jaime Carbonell, and Mark Kantrowitz. 2000. Multi-document summarization by sentence extraction. In *Proceedings of ANLP/NAACL-2000 Workshop on Automatic Summarization*.

Barbara J. Grosz and Candace L. Sidner. 1986. Attention, intention, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

Srikanth Kallurkar, Yongmei Shi, R. Scott Cost, Charles Nicholas, Akshay Java, Christopher James, Sowjanya Rajavaram, Vishal Shanbhag, Sachin Bhatkar, and Drew Ogle. 2003. Umbc at trec 12. In *TREC Notebook Proceedings*.

R. Ohgaya, A. Shimmura, and T. Takagi. 2003. Meiji university web and novelty track experiments at trec 2003. In *TREC Notebook Proceedings*.

Liyun Ru, Le Zhao, Min Zhang, and Shaoping Ma. 2004. Improved feature selection and redundancy computing – thuir at trec 2004 novelty track. In *The Thirteenth Text Retrieval Conference (TREC 2004) Notebook*.

Ian Soboroff. 2004. Draft overview of the trec 2004 novelty track. In *The Thirteenth Text Retrieval Conference (TREC 2004) Notebook*.

Jian Sun, Wenfeng pan, Huaping Zhang, Zhe Yang, Bin Wang, Gang Zhang, and Xueqi Cheng. 2003. Trec-2003 novelty and web track at ict. In *TREC Notebook Proceedings*.