# The University of Michigan in Novelty 2004

**Güneş Erkan**
Department of EECS
University of Michigan
gerkan@umich.edu

## Abstract

This year we participated in the Novelty track. To find the relevant sentences, we combine sentence salience features that are inherited from text summarization domain with other heuristic features based on topic statements. We propose a novel method to extract the new sentences based on the graph-based ranking of the similarity relation between the sentences.

## 1. Overview

The University of Michigan participated in all four tasks of the TREC 2004 Novelty track.

To find the relevant sentences in Tasks 1 and 3, we experimented with more than ten features. The system was trained with all possible subsets of these features on the Novelty 2003 data using different learning algorithms to be explained below. All of the features were integrated into the MEAD[1] text summarization system (Radev, Blair-Goldensohn, & Zhang, 2001). The following is a brief description of the features we used in the actual submissions, which gave us the best results on the training data:

- **Centroid**: The centroid score that is a measure of how close is the sentence to the centroid pseudo-sentence of the entire cluster. This is a measure of sentence salience which is proven to be successful in multi-document summarization domain (Radev, Jing, & Budzikowska, 2000).
- **LexRank**: The LexRank score (Erkan & Radev, 2004) is a measure of sentence salience based on the eigenvector centrality of the graph-based representation of the sentences in a cluster. We will give a brief explanation of how to compute LexRank in Section 2 and Section 3.
- **Length**: The number of words in the sentence.
- **QueryTitleCosine**: The cosine similarity between the "title" field of the topic statement and the sentence weighted by the word idf's. Formally, the cosine between two sentences $x$ and $y$ is defined by

$$\text{idf-modified-cosine}(x,y) = \frac{\sum_{w \in x,y} \text{tf}_{w,x}\text{tf}_{w,y}(\text{idf}_w)^2}{\sqrt{\sum_{x_i \in x}(\text{tf}_{x_i,x}\text{idf}_{x_i})^2} \times \sqrt{\sum_{y_i \in y}(\text{tf}_{y_i,y}\text{idf}_{y_i})^2}}$$

  where $\text{tf}_{w,s}$ is the number of occurrences of the word $w$ in the sentence $s$.
- **QueryDescriptionCosine**: The idf-modified-cosine similarity between "description" field of the topic and the sentence.
- **QueryTitleWordOverlap**: The similarity between the "title" field of the topic statement and the sentence based on simple word overlap.
- **QueryDescriptionWordOverlap**: The similarity between the "description" field of the topic statement and the sentence based on simple word overlap.

1. http://www.summarization.com

- **FQueryDescriptionWordOverlap**: Same as QueryDescriptionWordOverlap, but after expanding the "description" field with the relevant sentences in the first 5 documents of the cluster. This feature is used only in Task 3.

First three features above do not use the topic statements at all while the others use only one field of the topic statement. We never used the "narrative" field of the topic statement since it gave us no improvement in our experiments on last year's data.

To find the new sentences, we used two approaches. In Task 1, we implemented a very simple idea based on the cosine (dis)similarity between a sentence and the sentences that precede that sentence. In other tasks, we developed a novel method that makes use of the LexRank feature, which is explained in Section 3.

## 2. Task 1

To find the relevant sentences in Task 1, we used several sets of features for each run. We used Maxent 2.1.0 maximum entropy tool[2] for training our system on the 2003 data. Since all of the features are real-valued, we discretized the features using the entropy-based discretization algorithm proposed in (Fayyad & Irani, 1993).

The LexRank feature we used in our fourth submission is a measure of sentence salience originally proposed for multi-document summarization (Erkan & Radev, 2004). To compute LexRank, we construct an undirected graph where each node is a sentence. We define an edge between any two sentences if the cosine similarity between the sentences is above a pre-defined threshold. Edge weights are normalized so that the sum of the outgoing edges of a node is always equal to 1. Since the matrix representation of the graph is stochastic, we can consider this graph as a discrete Markov chain. The LexRank score of a sentence is the corresponding value in the stationary distribution of this Markov chain. Intuitively, the LexRank score of a sentence will be high if it is similar to many other sentences and the sentences that it is similar to have also high LexRank scores. We set the cosine similarity threshold to 0.1 while constructing the similarity graph to compute LexRank scores in Task 1.

We implemented a very simple idea for finding the new sentences. For each run, we considered the set of relevant sentences we found as the correct set. Then for each "relevant" sentence, we computed the cosine similarity between the sentence and all of the sentences that precede this sentence. We marked a sentence as "new" if the maximum cosine value found for that sentence exceeded a predefined threshold.

Table 1 shows the features and the novelty cosine threshold used in each run. We also include the average scores for all of our submissions in the table. The motivation behind using our topic-independent features Centroid and LexRank was to eliminate the sentences in the irrelevant documents that were introduced in this year's document sets. Centroid and LexRank try to extract the most salient information in a document cluster so that irrelevant sentences in a noisy cluster are usually eliminated provided that the noise in the cluster is small. However, we found out that the number of irrelevant documents in some clusters are so large (even exceeding the number of relevant documents in some cases) that they change and dominate the general topic of the cluster. Since we did not make any other effort to eliminate the irrelevant documents in Task 1, our results were severely affected by this phenomenon.

| Run No. | Relevance Features / Novelty Threshold | Relevant sentences | | | New sentences | | |
|---|---|---|---|---|---|---|---|
| | | Prec. | Recall | F | Prec. | Recall | F |
| 1 | Centroid,Length,QueryTitleCosine / 0.7 | 0.30 | 0.83 | 0.408 | 0.14 | 0.78 | 0.219 |
| 2 | Centroid,Length,QueryTitleCosine / 0.9 | 0.30 | 0.83 | 0.408 | 0.14 | 0.80 | 0.216 |
| 3 | Centroid,QueryDescriptionWordOverlap / 0.7 | 0.29 | 0.86 | 0.399 | 0.13 | 0.80 | 0.210 |
| 4 | LexRank,QueryDescriptionWordOverlap / 0.7 | 0.27 | 0.84 | 0.374 | 0.12 | 0.79 | 0.200 |
| 5 | Length,QueryTitleWordOverlap / 0.7 | 0.27 | 0.89 | 0.386 | 0.13 | 0.85 | 0.208 |

Table 1: Average precision, recall and F scores for Task 1.

2. `http://maxent.sourceforge.net`

## 3. Task 2

To find the new sentences, we used LexRank to model the dissimilarity of a sentence to the former sentences. We formed the cosine similarity graph of the sentences like we did in the summarization or the relevant sentences domain with the difference that a sentence was allowed to "vote" for only the sentences that appeared later in the cluster. Figure 1 shows a sample similarity graph constructed for detecting new sentences. Sentence 1 in the figure is similar to sentences 2 and 4, sentence 2 is similar to 3, etc.

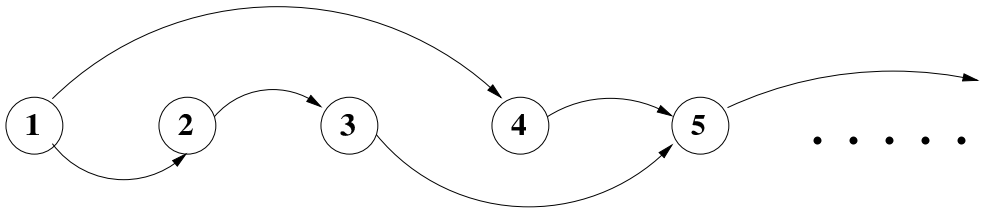


Figure 1: A sample cosine similarity graph for detecting new sentences.

To assess the new information in each sentence, we ran LexRank on this directed graph. Ideally, the sentences with *low* LexRank scores should contain new information since they have no or few incoming arcs, i.e. they are dissimilar to almost all of the former sentences. Note that the first sentence cannot have any incoming arcs, thus it will always have a low LexRank value.

We used a simple decision list algorithm to predict the novelty of each sentence based solely on its LexRank value. The system was trained on Novelty 2003 data. While constructing the similarity graphs, we used a different cosine threshold in each submission to define the similarity between two sentences. Unlike in the summarization domain, higher thresholds are better in this case since we try to model the "information subsumption". Low cosine similarities (values between 0 and 0.5) say that two sentences share some common information, but are not enough to conclude that one sentence subsumes the information in the other. Table 2 shows the cosine threshold used and the average scores we got in each submission.

| Run | Cosine | New sentences | | |
|---|---|---|---|---|
| No. | Threshold | Prec. | Recall | F |
| 1 | 0.6 | 0.45 | 0.85 | 0.551 |
| 2 | 0.7 | 0.45 | 0.93 | 0.594 |
| 3 | 0.9 | 0.43 | 0.90 | 0.553 |
| 4 | 0.5 | 0.42 | 0.70 | 0.492 |
| 5 | 0.8 | 0.43 | 0.88 | 0.554 |

Table 2: Average precision, recall and F scores for Task 2.

## 4. Task 3

In Task 3, we tried to eliminate the irrelevant documents first, and then perform relevant sentence extraction. To eliminate the irrelevant documents, we implemented two different algorithms, both of which made use of the relevant sentences in the first 5 documents that were provided for Task 3. In the first method, we computed the word overlap between each document and the relevant sentences in the first 5 documents, then took 25 documents that had the highest overlap score. In the second method, we computed LexRank values for the sentences in each document by appending the relevant sentences to the document. We took 25 documents that had the highest average LexRank score. These document filtering methods performed badly so that we got lower scores compared to Task 1.

To find the new sentences, we used the algorithm in Task 2. Since our starting point was the set of relevant sentences we extracted, the scores for new sentences were affected by the low accuracy for the relevant sentences. Table 3 shows the features we used for extracting relevant sentences, the cosine threshold used in building the directed graph for detecting new sentences, and the scores for each submission.

| Run No. | Relevance Features / Novelty Threshold | Relevant sentences | | | New sentences | | |
|---|---|---|---|---|---|---|---|
| | | Prec. | Recall | F | Prec. | Recall | F |
| 1 | Length,FQueryDescriptionWordOverlap / 0.7 | 0.36 | 0.50 | 0.373 | 0.14 | 0.42 | 0.182 |
| 2 | Length,QueryDescriptionCosine / 0.6 | 0.32 | 0.60 | 0.358 | 0.14 | 0.51 | 0.185 |
| 3 | Length,FQueryDescriptionWordOverlap/ 0.5 | 0.36 | 0.50 | 0.373 | 0.14 | 0.41 | 0.182 |
| 4 | Length,FQueryDescriptionWordOverlap/ 0.6 | 0.36 | 0.47 | 0.369 | 0.14 | 0.38 | 0.181 |
| 5 | Length,QueryDescriptionCosine / 0.6 | 0.32 | 0.59 | 0.367 | 0.14 | 0.52 | 0.193 |

Table 3: Average precision, recall and F scores for Task 3.

## 5. Task 4

For Task 4, we applied the same methods we used in Task 2. The only difference was that we used the relevant sentences in the first 5 documents provided for Task 4 as our training data, instead of using the 2003 data as we did in Task 2. However, this did not give us any improvement over Task 2 as shown in Table 4.

| Run No. | Cosine Threshold | New sentences | | |
|---|---|---|---|---|
| | | Prec. | Recall | F |
| 1 | 0.5 | 0.39 | 0.89 | 0.513 |
| 2 | 0.6 | 0.38 | 0.92 | 0.519 |
| 3 | 0.8 | 0.39 | 0.92 | 0.521 |
| 4 | 0.4 | 0.39 | 0.93 | 0.525 |

Table 4: Average precision, recall and F scores for Task 4.

## 6. Conclusion

Although we performed significantly better than last year, the results are not still satisfactory. This year, we proposed a novel method, LexRank, for detecting new sentences and got our most promising results in Task 2. Considering the simplistic approaches we have followed, this gives us a motivation for future improvements by integrating more advanced methods into our graph-based model. Our long-term benefit from studying novelty detection is to understand how the information in a set documents is organized across sentences and use this knowledge in other natural language processing problems, especially in text summarization and question answering.

## References

Erkan, G., & Radev, D. R. (2004). Lexpagerank: Prestige in multi-document text summarization. In Lin, D., & Wu, D. (Eds.), *Proceedings of EMNLP 2004*, pp. 365–371 Barcelona, Spain. Association for Computational Linguistics.

Fayyad, U. M., & Irani, K. B. (1993). Multi-interval discretization of continuous-valued attributes for classification learning. In *IJCAI-93*.

Radev, D., Blair-Goldensohn, S., & Zhang, Z. (2001). Experiments in single and multi-document summarization using MEAD. In *First Document Understanding Conference* New Orleans, LA.

Radev, D. R., Jing, H., & Budzikowska, M. (2000). Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In *ANLP/NAACL Workshop on Summarization* Seattle, WA.