Entropy-based Sentence Selection with *Roget's Thesaurus*

Alistair Kennedy **Terry Copeck** **Diana Inkpen** **Stan Szpakowicz**

School of Information Technology and Engineering
University of Ottawa
800 King Edward Avenue
Ottawa, Ontario, Canada K1N 6N5
{akennedy, terry, diana, szpak}@site.uottawa.ca

Abstract

This year at the University of Ottawa we submitted two systems to the Guided Summarization challenge. In our submissions we tested how well an entropy-based measure of sentence selection worked against a baseline system. The entropy-based sentence selector showed improvement over the baseline: it increased the number of unique Summary Content Units selected, and reduced the number of redundant Summary Content Units.

Our Submission to AESOP consisted of a system that ranks a summary based on how many other peer summaries contained the same sentences it selected. Three versions of this system were submitted.

1 Introduction

This year a new component was added to the text summarization challenge called "guided summarization". In guided summarization, a summary must answer a fixed, previously known set of questions. In addition to guided summarization there was also an update summary component, as in previous years. For update summarization the document set is split into two – set A and set B – and two summaries must be generated. The summary for set A is generated normally, while the summary for set B is generated so that it contains only information not found in document set A[1].

[1] For a full description of the text summarization challenge see: www.nist.gov/tac/2010/Summarization/

In our system we did not aim to take full advantage of this new information provided for guided summarization. Instead, with our two runs we performed an experiment aimed at increasing the number of Summary Content Units (SCUs) and the readability of the summaries. Our system makes use of two resources: a SCU-labeled corpus – see Section 2 – and *Roget's Thesaurus* – see Section 3.

1.1 Guided Summarization

In guided summarization the kinds of summaries to be generated are divided into five categories: Accidents and Natural Disasters, Attacks, Health and Safety, Endangered Resources, and Trails and Investigations. Each of these five categories contains a number of aspects most of which pertain to *who*, *what*, *where*, *when, why* and *how* of a story.

Some training data was provided. Two sets of model summaries from two different queries were provided for each of the five categories. In these summaries sentence fragments were marked up to indicate which aspects they addressed. Original and update summaries were included.

Instead of using aspects, in previous years a topic statement and a number of questions about the topic were provided as a query. This year there is only the topic statement and an indication of which category – set of aspects – to build the summary around.

2 SCU-Labeled Corpus

As in previous years, the University of Ottawa made use of a SCU-labeled corpus. Pyramid Evaluation (Nenkova and Passonneau, 2004) is a manual evaluation method in which a set of manually constructed model summaries are generated by

hand and then annotated for relevant pieces of information; Summary Content Units (SCUs). Each SCU is weighted based on how many model summaries it appears in. The annotators then go through the peer summaries labeling SCUs wherever they are found. The quality of a peer summary is determined from the number and strength of the SCUs found in the summary.

This sort of evaluation gives us a set of fully annotated summaries. Since most systems at TAC use extractive summarization, it is possible to map these sentences back to the original corpus. Copeck et al. (2006) reported that 83% and 96% of sentences could be mapped back to the original corpus in 2005 and 2006 respectively. This process has been run on the 2005, 2006, 2007, 2007 update pilot, 2008 and 2009 data from TAC/DUC creating a partially labeled corpus of sentences with SCU information. There are three kinds of sentences in the corpus:

- Positive sentences – one or more SCUs
- Negative sentences – no SCUs
- Unlabelled sentences – not used in a summary

Each positive sentence has a weight corresponding to the sum of the weight of the SCUs it contains. This corpus is described in more detail in (Kennedy & Szpakowicz, 2010a).

3 Roget's Thesaurus

We make extensive use of *Roget's Thesaurus* in both our sentence ranking and sentence selection modules. Specifically we used a Java implementation of the 1911 version of *Roget's Thesaurus* called *Open Roget's*[2].

Roget's Thesaurus is a hierarchical thesaurus nine levels deep; words always appear at the bottom. The names of the levels in the hierarchy from top to bottom are:

- Class
- Section
- Sub-Section
- Head Group
- Head
- Part of Speech (POS)
- Paragraph
- Semicolon Group
- Words/Phrases

There are 8 Classes at the top of the structure. *Roget's* contains approximately 1000 Heads, which are considered to be the central concepts in the *Thesaurus*. Another interesting feature is how the division by part of speech appears in the middle of the structure as opposed to at the very top as in resources like WordNet (Fellbaum, 1998). Words always appear at the bottom of the structure and each word can appear multiple times in the *Thesaurus*.

3.1 *SemDist*

One of the most useful components of *Roget's Thesaurus* is its ability to measure semantic distance between pairs of words (Jarmasz & Szpakowicz, 2004). This application, known as *SemDist*, is used in our sentence-ranking system. *SemDist* assigns a score to two words based on how close they appear to each other in the *Thesaurus*. A score of 16 is given if they are in the same Semicolon Group, 14 for Paragraph, …, 2 if they are in the same Class and 0 if they are in different classes or one or both words are not found.

4 Our System

This year the University of Ottawa submitted two systems to TAC for the text summarization challenge. The focus of our two systems was to build a system that has improved unique SCU count while reducing the redundant SCU count. We hoped that such a system would increase readability, as reducing redundancy should make the summaries more readable.

Our system is divided into three main components. The first one ranks sentences based on relatedness to a query. The second part is basic query expansion. The third part is a system for measuring the entropy of a summary, used to estimate the amount of unique information it contains. We give summaries a final score based on a balance of the scores assigned by the sentence ranker and those of the uniqueness measure. We also have separate modules to generate update summaries. After the summaries are generated an anaphora resolution system is used to resolve some pronouns in the summaries.

4.1 Sentence Ranking

For sentence Ranking we used a system proposed and evaluated in Kennedy & Szpakowicz (2010b). This system makes use of a semantic distance function based on the 1911 version of *Roget's The-*

[2] rogets.site.uottawa.ca

saurus (Kennedy & Szpakowicz 2008). This same sentence ranking system was employed in the 2009 TAC competition (Copeck et al., 2009).

The sentence ranker ranks the distance between every word $q_1 \ldots q_n$ in a query Q and every word $w_1 \ldots w_m$ in a sentence S. A score x_i is calculated for every query word q_i.

$$x_i = \max semDist(w_j, q_i) : \forall w_j \in S$$

A single score for a sentence *score(S)* is generated by taking the sum of the scores $x_1 .. x_n$ for S.

$$score(S) = \sum_{i=0}^{n} x_i$$

Score(S), we will refer to as a "sentence score" can then be used to rank sentences in the order of their relevance to the query. An entire summary *SUM* can be assigned a score, *score(SUM)*, which is the sum of the sentence scores $score(S_k)$ for each sentence S_k in *SUM*.

A few other heuristic criteria were taken into account for sentence ranking. We excluded sentences containing quotes and those with five or fewer words.

4.2 Query Expansion

Rather than attempt to answer all the different aspects for each summary we used the training data to perform query expansion. Training data was provided with model summaries of both the original and update summaries for two different queries. These summaries were marked up indicating which sentence fragments address which aspects of the summary.

We select all words that appear in the same aspect in at least one summary from both of the two summary sets provided in the training data. Only words that appeared in both summary sets were used to eliminate words specific to a particular query/topic. We then measured the pointwise mutual information (PMI) of that word and the aspect it appears in and selected all those words with a PMI score of greater than approximately 2.0. For some aspects no words had a PMI score high enough, so no query terms were generated. These new words were then added to the query – a simple topic statement – in order to improve our sentence ranking system.

The purpose of selecting only words that appear in both summary sets is so that they will be context-independent. To confirm that we were only selecting words that were general and not specific to one of the two queries we manually evaluated each word that was ranked with PMI. Only in a few cases did we remove any words. Around 100 query expansion words were discovered with anywhere from 14 to 33 words added for each category.

4.3 Measuring Uniqueness in Summaries

The next module in our text summarization system is one for measuring unique information in summaries. As stated earlier, the aim of our experiment is to select sentences that increase the unique SCU count while reducing the redundant SCU count of our summaries. For this we first define a baseline system and then a second system that should increase the unique SCU count.

The baseline system is fairly straightforward. We select the top 10 ranked sentences for each query and arrange them in every possible order creating a large set of summaries. We then select the summary *SUM* that has the highest score *score(SUM)* as defined in Section 4.1. This baseline attempts to maximize the total amount of SCUs a summary, but it does not actually take into account repeated SCUs that may appear in two or more sentences.

The aim of our second system is to assemble a group of sentences that maximizes the amount of unique information. We use an entropy-based measure to do this. We hypothesize that a summary that discusses as varied a set of topics as possible is likely to contain more unique information and have as less redundancy. In order to determine how many topics a summary discusses, we turn to *Roget's Thesaurus*. We map words to concepts in *Roget's Thesaurus* and then measure the entropy of those concepts. A concept in *Roget's* could be any of Class, Section, …, Semicolon Group, however we will focus on the Heads within *Roget's*.

In (Kennedy & Szpakowicz, 2008) a system for sentence representation is used, where each word is mapped to concepts in *Roget's*. If a word w has n senses $w_1 \ldots w_n$ in *Roget's*, then each of these senses is assigned a score of $1/n$. All concepts in *Roget's* have their score increased by $1/n$ for every instance of w contained within. This is repeated for every word in the sentence – stop words removed. This way a score could be given to each Class, Section, ..., Semicolon Group and its entropy measured. We

only mapped words to the Head, as this was found to be the best for representing sentences (Kennedy & Szpakowicz, 2008). Open Roget's contains 1044 heads $h_1 \ldots h_{1044}$ each of which receives a weight $weight(h_i)$ by this method. Although Kennedy & Szpakowicz, (2008) intended this to be used for representing single sentences, it can easily be used to represent entire summaries. Once a summary is represented in such a way, it is possible to measure the entropy of that summary. The probability of a given head, $p(h_i)$ is defined as the weight of that head normalized by the sum of the weight of all heads:

$$p(h_i) = weight(h_i) \Big/ \sum_{j=0}^{1044} weight(h_j)$$

Using this we can then calculate the entropy of an entire summary.

$$H(SUM) = -\sum_{i=0}^{n} p(h_i) \log_2 p(h_i)$$

$H(SUM)$ is taken to be a measure of uniqueness in our summary, while *score(SUM)* is a measure of how much information – SCUs – is in a summary.

4.4 Parameters

We now have two measures that we want to balance: the uniqueness, and a score derived from our sentence ranker. To do this we generated all possible summaries made from the top 10 sentences and then find the summary with the highest entropy, max$H$ and highest sentence score max*Score*. We use these scores to normalize the entropy and content scores for each summary and take a weighted average using the following equation:

$$finalScore(SUM) = \alpha \frac{score(SUM)}{\max Score} + (1-\alpha) \frac{H(SUM)}{\max H} \quad (1)$$

We now need a way to find a value for α. To do this we ran an experiment with document sets A from the 2008 and 2009 SCU labeled corpus data – discussed in Section 2 – in order to find an optimal parameter. We attempted to measure α=0, 0.2, 0.4, 0.6, 0.8, 1.0 and then used the SCU-labeled corpus to select the value of α which gave us the highest SCU score.

Although our aim is to both increase unique SCUs and decrease redundant SCUs, when tuning we must pick one measure to maximize. We suppose that maximizing the average weighted unique SCU score over all the summaries from the 2008 and 2009 TAC data will approximate this. As the 2008 and 2009 challenge was for update summaries, we only used set A in this evaluation.

The optimal value for α will depend very much on the length of the summary being generated, how many sentences are considered from the ranked list as well as which sentence ranker is chosen. We found that α=0.4 gave the optimal value for our sentence ranker using the top 10 sentences to generate summaries of 100 words or less. Note that when α=1 we are actually generating the baseline system's summaries. The SCU count, score and number of repeat SCUs across all summaries from set A of the 2008 and 2009 data are shown in Table 1. These results show a 4% increase in the unique SCU count, a 7% increase in unique SCU score and a 6% reduction in redundant SCUs.

	Baseline (α=1)	Entropy (α=0.4)
SCU count	314	327
SCU score	752	803
Repeat SCUs	66	62

Table 1: SCU results on tuning data.

Every possible summary made from the top 10 ranked sentences is generated and the summary with the highest *finalScore* is selected. Typically this method will produce between 50 and 200 candidate summaries.

4.5 Update Summarization

This same process can be applied to the task of generating update summaries. To establish this, we perform an experiment where a second update summary is generated that maximizes Equation (1) across two summaries instead of just one. The summary for document set A is generated as described in this paper and the best one is selected.

To generate update summaries the set of all summaries generated from the top 10 ranked sentences of document set B are created, but before they are evaluated for a final score – as described in Section 4.4 – they are concatenated onto the summary for document set A and then together they are evaluated. This should ideally create an update summary that contains as little information found in summary A as possible. The value α=0.4 is once again used in this part.

In our two runs at TAC this system for updating summaries is compared against a simple system which generates summaries the same way the baseline for document set A will be generated. Thus, we have one run – a baseline – where update summaries are generated as normal summaries and a second run where the entropy measure is used to both maximize information in the summary and to ensure no repetition with the summary for document set A.

4.6 Results

The evaluation showed that our entropy-enhanced system made a small improvement over the baseline. Modified Pyramid score increased from 0.210 to 0.223. In terms of linguistic quality and responsiveness the improvement was quite insubstantial improving from 3.01 to 3.02 and from 2.10 to 2.11 respectively.

Of more interest are the average SCU count and the count of redundant SCUs. Table 2 shows counts of the unique and repeated SCUs for all the data as well as specific scores for data sets A and B. What we have found is that the number of unique SCUs has increased while the number of repeated SCUs has decreased for both data sets. The effects in terms of reducing redundant SCUs was most noticeable on document set B, however both document sets saw an increase in their unique SCU count.

Measure	Baseline	Entropy
Unique SCUs – total	2.74	3.05
Repeat SCUs – total	0.33	0.27
Unique SCUs – set A	3.65	4.17
Repeat SCUs – set A	0.41	0.39
Unique SCUs – set B	1.83	1.98
Repeat SCUs – set B	0.24	0.13

Table 2: Unique and repeated SCU counts

Although the improvements in terms of overall responsiveness and readability were very small, we have found that our entropy-based method of sentence selection can increase the number of unique SCUs while simultaneously reducing the number of redundant SCUs.

5 AESOP

TAC 2010 once again set its participant the task of rating the summaries that peers had submitted to this year's Guided track. We once again used measures that assess each summary in terms of the others submitted on the topic. This approach has no practical application, as it is quite unlikely that a rating set of similar summaries would be found in any real world setting. Our assessments do however provide data about the set of summaries, which may be of interest to fellow researchers, and are presented on that basis.

Our single submission to the initial run of AESOP in 2009 treated all summaries, both peer and model, on the same basis. This proved to be an error, one that affected our results sufficiently greatly to suggest an overall negative correlation across the subject set, despite our rating of individual peer summaries to be generally in line with the TAC responsiveness metric (Copeck et al., 2009).

The problem was due to including the four model summaries with the 55 automatic ones rated for each topic. The metric employed was intended to answer the question, *how well suited to be in an automatic summary are the sentences that peers select most often?* Since previous analysis of DUC and TAC results has established that the preponderance of participants produce summaries automatically by extracting likely sentences from the topic document collection (and, we presume, continue to do so in 2010), this is a fair question to pose—for automatic summaries. Model summaries though are composed by human authors from their understanding of the topic document collection, and are very unlikely to reproduce verbatim any sentence appearing in source text.

Since AESOP 2010 asks for two summary ratings, one with (All Peers) and one without (No Models) considering model summaries, we requested and received permission to submit only a No Models run[3] this year.

In addition to repeating last year's measure, we made a second submission using a near neighbour metric in which the model summaries serve as a benchmark against which peer automatic summaries were measured. The issue in question is *whether an automatic summary's ranking tends to correlate with its use of concepts appearing in one or more of the human-written model summaries*, where content phrases are taken as proxies for con-

[3] More exactly, to satisfy the automated submission script we submitted runs with both designations, but the All Peers one had dummy data, a fact NIST staff were aware of.

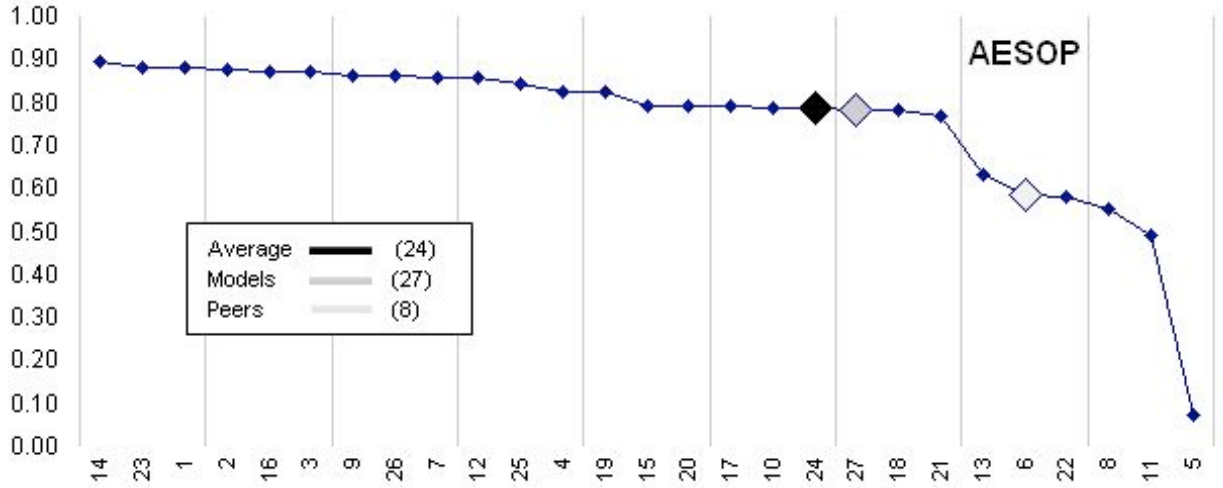

Figure 1: TAC 2010 Rating Measure Correlation

cepts. This was done in the following way. For each topic, each model summary was broken into its sentences. Each sentence was broken into its content phrases using a stop list of closed-class words. The stop list of 175 words is considerably smaller than that used elsewhere in our system, where a capability to distinguish sequences of words as being composed of adjoining phrases was sought. The list of content phrases produced was stored in a hash table whose values accumulated the count of the number of uses of the phrase in the four model summaries provided for each topic.

Peers' automatic summaries were then processed against this hash. Broken into sentences in the same way as the model summaries, each received a rating, which accumulated the values of the model summary hash phrases appearing in its sentences. To reward breadth only the first occurrence of each phrase was counted.

A third submission averaged a summary's ratings on peer summary sentence centrality and model summary concept coverage to address the possibility that *each of the two measures was effective in certain situations, ineffective in others*. Were this true, an average rating should outperform each constituent.

As in 2009, generic and update topic sets were not distinguished.

5.1 AESOP Results

Figure 1 shows the results. NIST provided Pearson, Spearman and Kendall correlations between each participant's AESOP rating of each main and update summary and its pyramid and responsiveness ratings established in the Guided summarization track. These two manual measures can be taken as TAC's gold standard for summary assessment. The correlation values appearing in Figure 1 consolidate all three types of correlation measurement for both kinds of summary for both Guided measures. The resulting single value has as its main advertisement the benefit of being all-inclusive—no data is left unconsidered.

The most striking feature of Figure 1 is the high degree of correlation achieved by 21 of the 27 automatic AESOP ratings, all of which exhibit a correlation with the Guided track rating of 0.77 or better. Our measures of summary quality, while meaningful in isolation, fall below the midpoint of the set of participants. In particular peer sentence selection ($\rho = 0.55$) is a poor indicator of summary quality. Model summary concept (content phrase) coverage has a respectable correlation of 0.78, while the average of the two schemes is highest ($\rho = 0.79$).

6 Conclusion

Overall our experiments this year are more of a refinement of the work performed last year (Copeck et al., 2009) than new approaches. We conducted a more thorough evaluation of our Entropy base method of sentence selection and adapted our existing system for Guided Summarization. For AESOP we provided a more in-depth evaluation of our system. Although the system in itself probably cannot be used for evaluating summaries in real world situations it is an interesting

baseline against which other summary evaluation systems should be compared.

Acknowledgments

Partial support for this work comes from the Natural Sciences and Engineering Research Council of Canada (NSERC). We would like to thank Anna Kazantseva for her help in running anaphora resolution software.

References

Nenkova, A., Passonneau, R.J. 2004. “Evaluating content selection in summarization: The pyramid method.” In: HLT-NAACL. (2004) 145–152.

Terry Copeck, Diana Inkpen, Anna Kazantseva, Alistair Kennedy, Darren Kipp, Vivi Nastase and Stan Szpakowicz. 2006. “Leveraging DUC”. In Proceedings of the Workshop on Automatic Summarization (DUC 2006), HLT/NAACL-2006.

Alistair Kennedy and Stan Szpakowicz. 2010a. “Towards a Gold Standard for Extractive Text Summarization”. In Advances in Artificial Intelligence, 23rd Canadian Conference on Artificial Intelligence, Canadian AI 2010, Ottawa, Canada, May 31 – June 2. 2010, 51-62.

Christine Fellbaum. 1998. “WordNet: An electronic lexical database”. MIT Press.

M. Jarmasz and S. Szpakowicz. 2004. “*Roget’s Thesaurus* and Semantic Similarity”. In N. Nicolov, K. Bontcheva, G. Angelova, and R. Mitkov, editors, Recent Advances in Natural Language Processing III: Selected Papers from RANLP 2003, Current Issues in Linguistic Theory, volume 260, pages 111–120.

Alistair Kennedy and Stan Szpakowicz. 2010b. “Evaluation of a Sentence Ranker for Text Summarization Based on Roget’s Thesaurus”. In the Proceedings of the 13th International Conference on Text, Speech and Dialogue, TSD 2010, Brno, Czech Republic. Lecture Notes in Computer Science 6231, 101-108.

Alistair Kennedy and Stan Szpakowicz. 2008. "Evaluating Roget's Thesauri". In Proceedings of ACL-08: HLT, Columbus, Ohio, USA, Association for Computational Linguistics, 416-424.

Terry Copeck, Alistair Kennedy, Martin Scaiano, Diana Inkpen and Stan Szpakowicz. 2009. “Summarizaing with Roget’s and with FrameNet”. In, Proceedings of the Workshop on Automatic Summarization (TAC 2009), Gaithersburg, Maryland, USA, 2009.