LING 573 - MultiDocument Summarization System

Audrey Holmes
University of Washington
auholmes@uw.edu

Eslam Elsawy
University of Washington
eslam@uw.edu

Masha Ivenskaya
University of Washington
marliven@uw.edu

Abstract

This paper presents a system of automatic multi-document extractive summarization of news articles. Our systems consists of several modules. The content selection module utilizes the LexRank algorithm, based on the similarity of sentences within the document corpus. The similarity measure used is TF-IDF cosine similarity. The reranking module creates the 100 word summary from the top-ranked sentences, taking into account the cosine similarity of a given sentence to the sentences already in the summary in order to avoid redundancy. The content realization module compresses the sentences using heuristic approaches before the sentences are added to the summary. Finally, he information ordering module reorders the sentences in the summary based on the Entity Grid approach. Our system is evaluated using the Rouge evaluation system and achieves Rouge-1 Recall score of 30.08, outperforming MEAD and LEAD baselines systems on TAC 2010 and 2011 summarization tasks.

1 Introduction

The goal of automatic multi-document summarization is to automatically produce a compressed version of the information presented by a group of documents on the same topic. In this paper our focus is on extractive summarization, meaning that the summary is directly produced from the existing sentences in the documents, and on non-query focused summarization, meaning that we treat the topic of a given group of documents as unspecified. In section 2 we present that overview of the architecture of our system. In section 3 we go over the methods - including data preprocessing, the methods of selecting the sentences for the summary, as well as of method of sentence compression and of ordering the sentences within the summary. In section 4 we present the results of evaluating our system using the ROUGE evaluation system.

2 System overview

Figure 1 shows the architecture of our summarization system. First, sentences from documents belonging to the same topic are extracted. Then, adjacency matrix is built with pairwise TF-IDF cosine similarities. After that, LexRank algorithm utilizes the similarity matrix to sort the sentences based on their importance. Sentences and LexRank scores are fed into the cosine similarity reranker module which picks the sentences for the final summary based on the score and on the similarity to sentences already in the summary. The Content Realizatioon module compresses these sentences before they are added in. Finally, the Reordering Module reorders the sentences within the summary to achieve better cohesion.

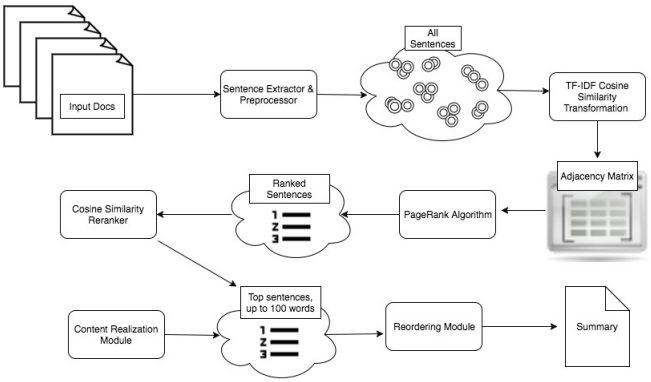

Figure 1: System architecture

3 Approach

3.1 Data

We use the TAC-2010 dataset that consists of the following:

- A set of n topics $T = \{T_1, T_2, ..., T_n\}$.
- For each topic T_i, a set of m documents $D_i = \{d_{i,1}, d_{i,2}, ..., d_{i,m}\}$.
- For each topic T_i a vocabulary V_i.

3.2 Content Selection

Sentence Extraction and Preprocessing. For each document, we extract the contents of each document from the XML file using Python's BeautifulSoup module. We then convert the text to lowercase and perform sentence tokenization.

Sentence Similarity. For each document $D_{i,j}$, we convert the sentences into vectors s of length $|V_i|$, where s_v is the number of times word v appears in the sentence. From these vectors, we create an N x $|V_i|$ matrix M_i, where N is the number of sentences within a topic. We then perform a tf-idf transformation on M_i. Finally, we transform M_i into a symmetric N x N matrix of pairwise cosine similarities.

LexRank. We binarize the entries of M_i such that values less than 0.15 are set to 0; all other values are set to 1. We then implement the Google PageRank (Page et al., 1999) (Erkan and Radev, 2004) to estimate sentence importance.

Other Similarity Measures. In addition to tf-idf cosine similarity, we experimented with using Doc2Vec to measure sentence similarity. We trained a doc2vec model on the full Reuters corpus and then used this model to calculate similarity scores between sentences. These scores were then binarized and ran through the PageRank algorithm. We experimented with different binarization thresholds, as well as with removing stopwords from the sentences before measuring similarity. We also experimented with manually computing sentence vectors using pretrained GloVe embeddings. However the Rouge scores obtained through these approaches were lower than the results obtained using tf-idf cosine similarity.

3.3 Improving Content Selection

To improve upon our baseline metrics, we explored several different pre-processing options. First, we removed sentences with with fewer than 35 characters, as these seemed to be sentence fragments or short quotations. We also removed certain types of metadata within the sentences and tried lemmatization and stemming. Finally, we improved upon the *tf-idf* approach.

Metadata Removal. We used regular expressions to remove the sentences that contained phone numbers or websites. We also removed extra location phrases such as **CHICAGO, Illinois (AP) –**.

We also tried removing ages and acronyms in parentheses such as Anna**, 36,** and Papua New Guinea **(PNG)**. However, these changes did not improve performance.

Lemmatization and Stemming. Python's NTLK toolkit has several options for lemmatization and stemming. First we tried the WordNet lemmatizer. However, this option did not seem to have very good coverage, as there were many words in our corpus that did not get lemmatized.

The NTLK package also implements a Porter stemmer and a Snowball stemmer. The Porter stemmer lowered ROUGE scores. We think this is because the stemmer was too aggressive, often removing too much of a word and thus losing important information. The Snowball stemmer, on the other hand, was less aggressive, and provided a slight improvement in ROUGE scores.

Vectorization. We improved upon our baseline *tf-idf* vectorizer by using a binary term frequency for the numerator (keeping the inverse document frequency denominator the same).

We also experimented with using unigrams, bigrams, and trigrams. We were somewhat surprised to find the best results with unigrams. Our hypothesis is that bigrams and trigrams might be more helpful with a larger corpus.

3.4 Information Ordering

For our baseline approach, we ordered the sentences by descending PageRank score and cut off each summary at the maximum word limit. We also implemented a cosine similarity reranker (Radev et al.,), skipping sentences that are too similar to sentences already in the summary.

Table 1: ROUGE-1 evaluation results for different baseline system configurations

Similarity Measure	Information Ordering	Precision	Recall	F-score
TF-IDF Cosine Similarity	Dummy	.29	.24	.26
Doc2Vec	Dummy	.18	.23	.20
TF-IDF Cosine Similarity	Cosine Reranker Thr=0.2	**.3**	**.26**	**.28**
Doc2Vec	Cosine Reranker Thr=0.2	.20	.22	.21

Table 2: ROUGE-L evaluation scores for baseline

ROUGE-L	Precision	Recall	F-Score
ROUGE-1	0.3	0.26	0.28
ROUGE-2	0.08	0.07	0.08
ROUGE-3	0.03	0.02	0.03
ROUGE-4	0.01	0.01	0.01

Figure 2 shows that the best achieved results happen at cosine similarity threshold = 0.2

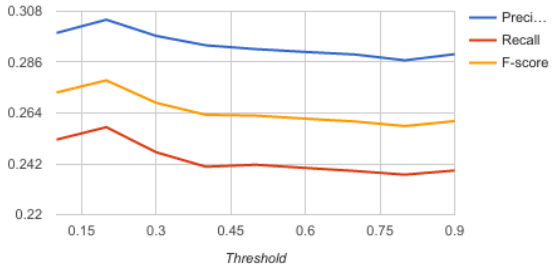

Figure 2: Precision, Recall and F-measure at different reranker cosine similarity thresholds

We also implemented a module that reorders the sentences that have been selected for the summary. We experimented with several approaches to reordering, summarized below. Reordering of the sentences had little or no effect on the Rouge scores, so we discuss the advantages and disadvantages of each approach based on qualitative analysis of the output summaries.

Chronological Reordering In this approach, we ordered the sentences of the summary based on the publication date and time of the original documents. In those cases when two sentences came from the same document, or when the date/time of publications coincided, sentences were ordered in order of appearance in the documents(s).
This ordering seemed to do well for topics about events that unfold, and get reported on, over time, such as natural disasters, for example:

"the papua new guinea (png) defense force, the police and health services are on standby to help the victims of a tsunami that wiped out several villages, killing scores of people, on png's remote north-west coast friday night. igara said reports so far indicated that a community school, government station, catholic mission station and the nimas village in the sissano area west of aitape had been completely destroyed, where 30 people were dead. the death toll in papua new guinea's (png) tsunami disaster has climbed to 599 and is expected to rise, a png disaster control officer said sunday."

It was problematic for many other topics, as the following summary illustrates:

"for example, new roads will be banned in national forests around the park, servheen said. fish and wildlife service is poised to remove the park's renowned bears from the endangered species list. federal wildlife officials estimate that more than 600 grizzly bears live in the region surrounding yellowstone in idaho, montana and wyoming. grizzly bears in and around yellowstone national park should be removed from the endangered species list after 30 years of federal protection, the u.s. department of interior said tuesday. the only other large population of grizzlies in the united states is in and around glacier national park."

Cohesion Reordering In this approach, we first created all permutations of the sentences within the summary. For each permutation, we calculated a cohesion score as a sum of the cosine similarity scores of adjacent sentences. We then picked the order with the highest cohesion score.
This approach worked well for certain topics, especially those that concern a general issue that doesn't have a strong chronological component:

"in the united states, 21 percent of known species are threatened or extinct. the survey, published online by the journal science, studied the 5,743 known amphibian species and found that at least 1,856 of them face extinction, more than 100 species may already be extinct, and 43 percent are in a population decline many for unknown reasons. the researchers called for efforts to protect the habitat of amphibians and to reproduce the threatened species in captivity. habitat decline, from deforestation to water pollution and wetlands destruction, threatens them because the animals live both on land and in water."

However, it was often problematic for topics where new information was reported over time:

"burke was in the family's boulder home when 6-year-old jonbenet was found beaten and strangled dec. 26, 1996. hunter took the jonbenet case to the grand jury shortly after a former boulder police detective on the case and three former friends of the ramseys publicly demanded that colorado's governor, roy romer, replace hunter on the case with a special prosecutor. although the police chief and district attorney both have said that the ramseys fall under "the umbrella of suspicion," they have not formally named any suspects. police say her parents, john and patsy ramsey, remain under suspicion. "

Entity Grid Reordering

We implemented the apporache described in (Barzilay and Lapata, 2008).

For training we used the training dataset in AQUAINT, For each topic we pick two documents, and for each document we consider the original ordering of sentences and build the feature vector for it as a positive (i.e. good cohesion) training sample, and then randomly order the sentences and build the feature vector as a negative (i.e. bad cohesion) training sample. Our training dataset has 174 samples. For co-reference we built noun clusters based on lexical match, we used Stanford entity recognizer (Manning et al., 2014) for detecting named entities. For detecting grammatical roles we used Stanford dependency parser (Klein and Manning, 2003), we detect four roles, (S) subject, (O) object, (X) other, (-) missing. We don't distinguish between focused and not focused entity clusters based on frequency.

At run time, we use cosine based similarity approach to pick an initial ordering for the candidate summary, then we consider the original ordering plus 20 different random ordering and pick the summary which achieves the best coherence score. To determine the coherence score we used the model built during the training stage and built a KNN classifier with K=11. We used sklearn KNN classifier which can predict the probability that the test sample belongs to the class of high coherent training samples, then we pick the ordering which achieves the highest probability.

Table 3 shows a sample output of this module, the initial ordering of sentences was 1, 2, 3, 4 which achieved coherent score of 0.55 after running this module the best coherent ordering was 2, 1, 4, 3 which achieved coherent score of 0.73

3.5 Content Realization

After content selection and information ordering, we focused on sentence compression to allow more information into our summaries. We also made some changes to improve readability.

The first step for improving readability to was to reinstate proper capitalization which had been removed during content selection.

Next, we removed sentence-initial adverbs and conjunctions. For example:

But the new research, led by Baker, suggests that heat-tolerant algae may move in to replace strains lost in bleaching events.

was replaced with

The new research, led by Baker, suggests that heat-tolerant algae may move in to replace strains lost in bleaching events.

Next we removed all parenthetical expressions. For example:

Australian Prime Minister John Howard said Wednesday that his country would provide 1 billion Australian dollars (about 764 million US dollars) in loans and grants to assist Indonesia in its rebuilding after the Dec. 26 earthquake-tsunami disaster.

Table 3: Sample summary

Id	Sentence
1	the announcement comes just two weeks after merck pulled its painkiller, vioxx, which is in the same class of drugs as bextra, from the market because a study showed that the risk of heart attacks doubled for patients who had taken vioxx for 18 months or longer.
2	vioxx was approved after trials held under the auspices of the food and drug administration showed it to be effective (which it was).
3	gilmartin was clear that the trial should be halted and that the drug might have to be taken off the market.
4	the drug was not pulled at that point.

was replaced with

Australian Prime Minister John Howard said Wednesday that his country would provide 1 billion Australian dollars in loans and grants to assist Indonesia in its rebuilding after the Dec. 26 earthquake-tsunami disaster.

Finally, we removed ages. For example:

A judge heard motions Friday from lawyers for the two brothers, Deepak Kalpoe, 21, and Satish Kalpoe, 18, and a Dutch youth, Joran Van Der Sloot, 17.

was replaced with

A judge heard motions Friday from lawyers for the two brothers, Deepak Kalpoe and Satish Kalpoe, and a Dutch youth, Joran Van Der Sloot.

We experimented with removing attribution phrases such as *, he said.* and *, according to*, but this created many readability issues, as the regular expressions we used were not robust enough to remove the entire phrase and left hanging words.

We also experimented with removing noun appositives, but this was problematic due to incorrect parses.

4 Results

Table 1 summarizes our results using different similarity measures and reordering algorithms. The best achieved results occurred when using TF-IDF cosine similarity and cosine sentences reranker with threshold = 0.2.

Table 2 shows best achieved ROUGE-L (Lin,) evaluation scores for our baseline system.

Table 4: ROUGE-L evaluation scores for system with improved content selection and information ordering in D3

ROUGE-L	**Precision**	**Recall**	**F-Score**
ROUGE-1	0.31155	0.27056	0.28879
ROUGE-2	0.08850	0.07684	0.08199
ROUGE-3	0.03017	0.02596	0.02780
ROUGE-4	0.00861	0.00739	0.00792

Table 5: ROUGE-L evaluation scores for final system on devtest dataset

ROUGE-L	**Precision**	**Recall**	**F-Score**
ROUGE-1	0.30584	0.26828	0.28516
ROUGE-2	0.08641	0.07581	0.08056
ROUGE-3	0.02749	0.02396	0.02553
ROUGE-4	0.00679	0.00591	0.00630

Table 4 shows the achieved ROUGE-L evaluation scores using improved content selection and information ordering in D3.

Table 5 shows the ROUGE-L scores for our final system on the devtest dataset.

Table 6 shows the ROUGE-L scores for our final system on the evaltest dataset.

Figure 3 shows the progress achieved on the system across different phases of the project.

Finally, we achieved an average readability score of 3.524 (out of 5). This score was manually assessed.

Table 6: ROUGE-L evaluation scores for final system on evaltest dataset

ROUGE-L	Precision	Recall	F-Score
ROUGE-1	0.33312	0.30800	0.31933
ROUGE-2	0.10115	0.09428	0.09739
ROUGE-3	0.03810	0.03601	0.03695
ROUGE-4	0.01773	0.01696	0.01730

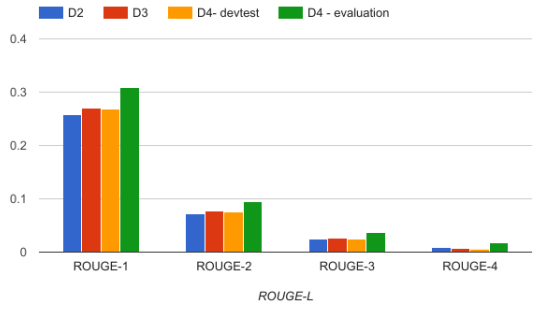

Figure 3: Rouge-L scores across different phases of the project

5 Discussion

In this section, we provide error analysis and assessment of each component in the system

5.1 Content selection

The LexRank approach proved to be very successful for our tasks - even prior to implementing the information ordering and content realization modules, our system was already achieving relatively high Rouge scores. The biggest boost to the content selection module came from using tf-idf cosine similarity in place of regular cosine similarity. Also, using cosine similarity to prevent sentences that are too similar to the sentences already in the summary helped to prevent redundancy and to create a summary that contains a more diverse set of sentences.

We were surprised that more sophisticated measures of similarity, such as using word embedding vectors to compute similarity between sentences, did not produce better results. Our analysis showed that sentences that receive high similarity scores by this measures are often not in fact closely related. A possible hypothesis is that the contextual similarity captures through these approaches may not the right match for the document summarization task - i.e. if we are measuring how connected a sentence is to other sentences in the topic, we are interested in actual overlap of entities and concepts, not broad semantic relatedness. One of the experiments performed on the content selection component is to remove all sentences that don't have any named entities, and trying to analyze the effect of that on rouge scores and readability of the output summary. Figures 4 and 5 shows that rouge scores didn't significantly change after applying named entities restriction, however since we are using entity grid approach for information ordering, then having a summary that is rich in named entities will make the entity grid less sparse, and therefore we will have more entity transitions and this will positively affect the ordering accuracy and hence the readability of the summary.

5.2 Information ordering

The system uses cosine similarity as an initial ordering technique, however the results of this approach are sensitive to the threshold value. Figure 6 shows that the threshold at which best rouge scores are achieved on the devtest dataset is different from the best threshold for the evaluation dataset. To overcome this problem the system can automatically try different thresholds on a new dataset and pick the threshold at which best results are achieved. Without entity grid module the system runs in around 5 minutes, so having around 5 or 6 runs with different thresholds is feasible in a reasonable time.

In order to pick the best information ordering strategy, we tried to combine different ordering modules and pick the one that achieves the best readability, we tried 4 combination, (1) cosine similarity and entity grid ordering, (2) cosine similarity and chronological expert ordering, (3) cosine similarity and cohesion expert ordering and (4) only using cosine similarity. Table 7 shows a sample summary. On that summary the combination of cosine similarity and entity grid ordering was the only ordering technique to choose sentence id = 4 as the start of the summary, the ordering was as following (4, 2, 1, 3, 5), and hence this technique was chosen for our system.

5.3 Content realization

Our approach to content realization was to compress the sentences using heuristic methods, in order to potentially fit more information into the summary and thus improve the Rouge scores, while maintaining the readability of the summary.

We avoided large-scale sweeping changes, su
as removing entire syntactic constituents, as the
proved to result in poor summary readability. Sti
the small changes that we made to the origin
sentences, mostly using regular expressions, act
ally slightly decreased our Rouge scores, possib
due to decreasing the n-gram overlap between o
summaries and gold-standard summaries.

6 Conclusion

In this paper, we demonstrated a multi-docume
summarization system that extracts a comprehe
sive 100 words summary from a collection of doc-
uments talking about the same topic. The system
uses lex-rank algorithm for content selection, and
uses cosine similarity and entity grid approaches
for information ordering. Also the system utilizes
several sentence compression techniques for co
tent realization. The results shows that the sy
tem performance outperforms MEAD and LEA
baseline systems on TAC 2010 and 2011 guid
summarization tasks.

References

Regina Barzilay and Mirella Lapata. 2008. Modeli
local coherence: An entity-based approach. *Comp
tational Linguistics*, 34(1):1–34.

Günes Erkan and Dragomir R Radev. 2004. Lexrar
Graph-based lexical centrality as salience in te
summarization. *Journal of Artificial Intelligence
Research*, 22:457–479.

Dan Klein and Christopher D Manning. 2003. Ac-
curate unlexicalized parsing. In *Proceedings of the
41st Annual Meeting on Association for Computa-
tional Linguistics-Volume 1*, pages 423–430. Ass
ciation for Computational Linguistics.

Chin-Yew Lin. Rouge: A package for automatic ev
uation of summaries.

Christopher D Manning, Mihai Surdeanu, John Bau
Jenny Rose Finkel, Steven Bethard, and David M
Closky. 2014. The stanford corenlp natural la
guage processing toolkit. In *ACL (System Demo
strations)*, pages 55–60.

Lawrence Page, Sergey Brin, Rajeev Motwani, a
Terry Winograd. 1999. The pagerank citation ran
ing: Bringing order to the web. Technical repo
Stanford InfoLab.

Dragomir R Radev, Timothy Allison, Sasha Bla..
Goldensohn, John Blitzer, Arda Celebi, Stanko
Dimitrov, Elliott Drabek, Ali Hakim, Wai Lam,
Danyu Liu, et al. Mead-a platform for multidocu-
ment multilingual text summarization.

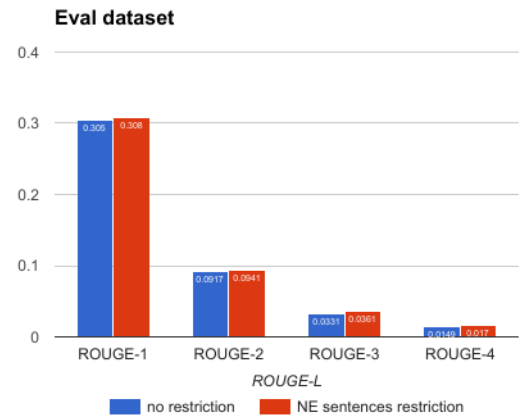

Figure 4: Effect of applying NE restriction on evaluation dataset

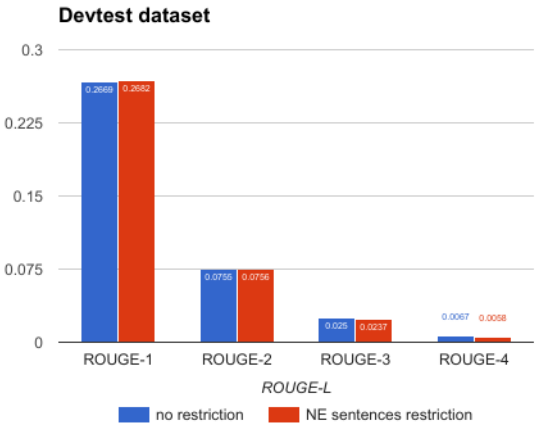

Figure 5: Effect of applying NE restriction on devtest dataset

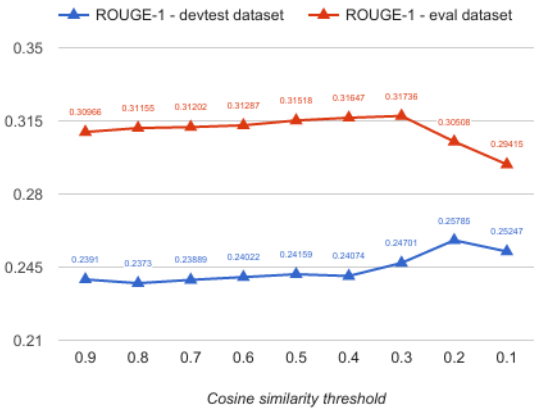

Figure 6: Average Rouge-1 scores at different cosine similarity thresholds

Table 7: Sample summary

Id	Sentence
1	The area is close to Bangladesh's border with the Indian state of West Bengal, which is also expecting highly destructive winds and torrential rains.
2	A powerful cyclone with strong winds started pounding on Bangladesh's south and southwestern coast from Thursday evening.
3	Officials in both Bangladesh and India have been racing to evacuate hundreds of thousands of people from the area over the past 48 hours.
4	Cyclone Sidr, described as the worst storm in years to hit low-lying and disaster-prone Bangladesh, crashed into the southwestern coast Thursday night before sweeping north over the capital Dhaka.
5	Six water treatment systems were mobilized.