NLP Systems and Applications: Automatic Summarization

Claude Zhang Julia McAnallen Genevieve Peaslee Zoe Winkworth
University of Washington, Seattle
{youyunzh, jmcanal, genevp, zoew2}@uw.edu

Abstract

This paper describes the design and implementation of three related multidocument summary generator systems: a baseline lead sentence system, a system modeled on MEAD (Radev et al., 2001); and MELDA, an expansion of MEAD that incorporates enhancements using LDA topic modeling (Blei et al., 2003). When run on evaluation data (newswire document sets originally used in the 2011 TAC summarization shared task), LEAD has a ROUGE-2 F1 score of 0.30071; the final versions of MEAD and MELDA had 0.28449 and 0.27300 ROUGE-2 F1 scores, respectively.

1 Introduction

We present a multi-document summarization system with three different content selection strategies: lead sentence, MEAD score-based (Radev et al., 2001; Radev et al., 2002; Radev et al., 2004), and MEAD with LDA topic modeling enhancements to content selection and information ordering. We compare ROUGE scores on output summaries produced by different parameter combinations and discuss the implications of these results on future work.

2 System Overview

Our three related systems are illustrated in Figure 1 and each carries out the three main subtasks of summarization: content selection, information ordering, and content realization.

2.1 Pre-processing

We use NLTK to segment documents into sentences and sentences into words. NLTK's sentence segmentation is not perfect so we check for "sentences" that are all stopwords or punctuation and exclude those from the content selection process, though this still leaves some incomplete or ungrammatical sentences in the pool of candidates.

In an attempt to more accurately measure sentence similarity, we tried lemmatizing non-named entity tokens and replacing all co-referring spans of text with one unique mention per referent (helping sentences with similar meanings to look more similar on the surface) but this added a significant amount of processing time and actually gave lower ROUGE scores, so the final system does none of this pre-processing.

2.2 Summary Generator Module

The summary generator module of our system handles tokenization and initiates the three core steps of the summarization task, described below, for each cluster of documents in the input.

2.3 Content Selector Module

The content selector module selects sentences that are most salient to the topic from the set of topic documents, as evaluated by one of three selection strategies.

2.4 Information Ordering Module

The Lead system orders content chronologically and the MEAD system orders by descending MEAD score. MELDA has a more sophisticated information ordering strategy that specifically addresses inter-sentence cohesion, which we call the cohesion gradient.

2.5 Content Realization

The content realization module for the Lead and MEAD systems outputs the selected sentences in their original form. The MELDA system performs a series of compressions on sentences before adding them to the summary.

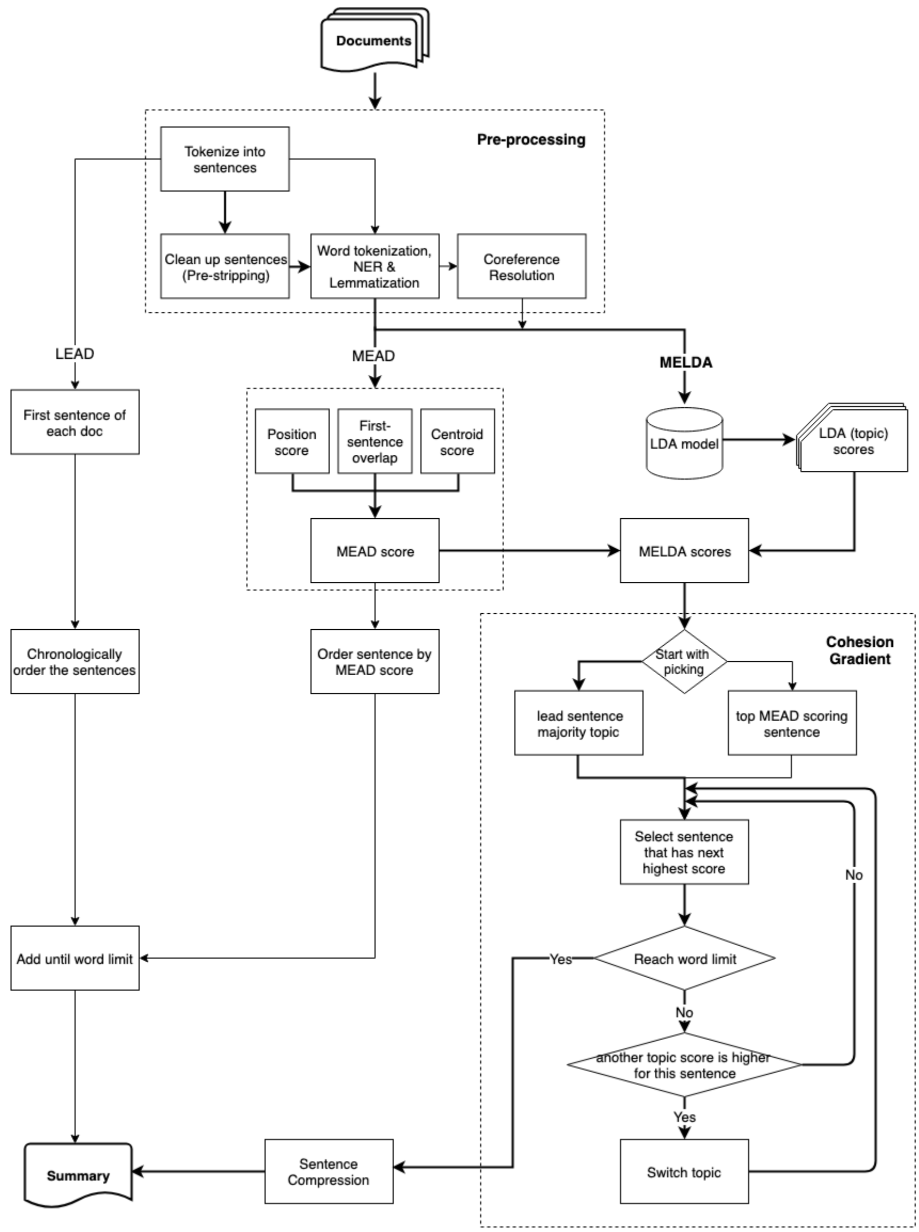

Figure 1: System Architecture

3 Approach

For each of our systems, we implement the three core subtasks of the summarization task as follows.

3.1 Lead Sentence System

Our lead sentence system generates a summary by selecting from only the first sentence in every document.

3.1.1 Content Selection

To select the content eligible to be included in the summary, we simply select the first sentence in each document included in the topic.

3.1.2 Information Ordering

Sentences selected by the content selector are ordered chronologically by document date and then by article ID.

3.1.3 Content Realization

The final output contains sentences as they appear in the original documents, without editing. Sentences are added to the summary one by one, most recent first, until the addition of the next sentence would make the total word count of the summary greater than 100. Only full sentences are added to the summary, but sentences that are over 100 words are skipped to prevent empty outputs.

3.2 MEAD-based System

Our MEAD system selects content by calculating MEAD scores for every sentence. The MEAD score is composed of the sentence position score, the first sentence overlap, the centroid score, and a redundancy penalty.

3.2.1 Content Selection

The MEAD score calculation has four components. A centroid score, a positional score, and a first sentence-overlap score are optionally weighted and then summed to get a preliminary score for each sentence. The fourth component is a redundancy penalty applied to all remaining sentences each time an individual sentence is added to the summary.

Centroid Score A centroid vector is computed for each topic by multiplying count and IDF, following Radev (2004). Count values are the average counts of words in a given topic, which is then multiplied by IDF values calculated from an external corpus. Both the Reuters and Brown Corpora from NLTK were tested as external corpora (Bird et al., 2009). Brown resulted in marginally better final results, and was chosen as the base corpora for the MEAD submission.

Next, a predefined threshold value is applied to the cluster centroid; words with centroid values below the threshold are set to zero. We developed threshold settings based on the top quartile, mean, and bottom quartile of word centroid values. (Radev (2004) did not provide guidance on selecting a threshold value.)

A centroid score for each sentence is then calculated by summing the centroid value for each word in the sentence after the threshold has been applied:

$$C_s = \sum_w C_{w,s}$$

where C_s is the centroid score of sentence s, w ranges over all the words in the sentence, and $C_{w,s}$ is the centroid value for each word w in sentence s.

Positional Score The positional score P_s for each sentence is the sentence's position in the document (s; 1st = 0, 2nd = 1, etc.) scaled by the distance from the beginning of the document:

$$P_s = \frac{n - s}{n}$$

where P_s is the positional score of sentence s and n is the number of sentences in the document. Note that our calculation diverges from the equation provided in Radev (2004) in two ways. First, they add one to the numerator under the assumption that the first sentence has a position score of 1; however, we used Python file IO functions that start numbering at 0. Second, the positional score is often scaled against a maximum centroid score, which we did not use for this baseline.

First Sentence Overlap Score Overlap with first sentence is calculated using cosine similarity to compare each sentence with the first sentence in the document containing it.

$$F_s = \frac{S_0 \cdot S_s}{||S_0|| \times ||S_s||}$$

where F_s is the first sentence overlap score of sentence s and S_s is the TF*IDF-weighted vector representation of sentence s. Note that this differs from the implementation of Radev (2001) who use the inner product of the TF*IDF-weighted vector representations of a given sentence and the first sentence.

MEAD Score The scores for centroid, position and first sentence overlap are summed for all sentences in the document.

$$score = w_c C_s + w_p P_s + w_f F_s$$

where w_c, w_p, and w_f are optional weights applied to the scores. Note that we also depart from Radev, et al. (2001) here since they normalize all three features in the range 0 - 1 while we leave them as is.

Redundancy Penalty The redundancy penalty is calculated each time a sentence is added to the summary to prevent redundant sentences from being included. Each sentence is compared to the last sentence added to the summary, and the penalty is computed by dividing the number of overlapping tokens by the number of tokens in the sentence pair and doubling the result. This penalty is then subtracted from all the sentence scores.

$$R_s = 2 \times \frac{S_s * S_l}{cnt(S_s + S_l)}$$

where R_s is the redundancy penalty for sentence s and S_s is the TF*IDF-weighted vector representation of sentence s and S_l is the TF*IDF-weighted vector representation of the sentence most recently added to the summary.

3.2.2 Information Ordering

Sentences selected by the content selector are ordered by descending MEAD score. After each sentence is added to the summary, scores are recalculated by subtracting the redundancy penalty and all sentences are re-ordered by the new scores before the next summary sentence is added.

3.2.3 Content Realization

The final output contains sentences as they appear in the original documents without editing. As in the lead sentence implementation, sentences are added to the summary one by one until the addition of the next highest-scoring sentence would push the total word count of the summary over 100 words. Only full sentences are added to the summary, but sentences that are over 100 words are ignored, to prevent empty summaries.

3.3 MELDA System

3.3.1 Content Selection

The foundation of MELDA content selection is the approach described for MEAD in §3.2.1. MELDA additionally uses LDA topic modeling (Blei et al., 2003) to choose the top sentences according to both their MEAD scores and their LDA topic scores. Therefore, top MEAD scores that are also highly representative of a topic are more likely to be selected than high scores that are not highly representative of a topic.

Latent Dirichlet Allocation The LDA method is used to build statistical models that classify text in a document into abstract "topics". To avoid redundancy, we will use the term "topics" for the LDA extracted topics and "cluster" for the document themes.

In this system we use the gensim package to build an LDA model over the text of all the documents in a cluster and get the LDA topic scores for each sentence, which represent the probability that a sentence belongs to each of the abstract topics. Considering that the length of the summary is 100 words and the average sentence length is around 20, for now we set the number of topics to three by default. We then tried different settings for other parameters; the results are in §4.

To integrate the LDA and MEAD score, we simply add the MEAD score for a sentence to each LDA topic score to compute MELDA scores (resulting in as many MELDA scores per sentence as there are topics) and select content based on the result. We use a hyper-parameter n (default is 5) to determine the number of sentences per topic to include, and select the top n sentences per topic with the highest MELDA scores as candidate sentences.

3.3.2 Information Ordering

Information ordering in MELDA is more sophisticated than the MEAD approach of ordering the top scoring sentences by score. We instead leveraged LDA topic scores to improve coherence between sentences.

Cohesion Gradient The LDA scores calculated for content selection were used to improve information ordering in the system. In particular LDA topic scores for the top sentences were used to order the sentences according to a cohesion gradient, which aimed to improve summary coherence.

Before applying the cohesion gradient, the content selection process pre-selects a fixed number of sentences to be ordered. Then, the cohesion gradient function is applied as follows: first, the most prevalent topic among the lead sentences across the entire document cluster is chosen. Then the sentence in the pre-selected set with the highest topic score for that topic is chosen as the first sentence of the summary. The sentence with the next highest topic score for the topic is chosen next, and this process is repeated until another topic takes over as the highest valued topic for a sentence. Each time this happens, the topic switches.

We tried two different approaches for choosing the topic to start with. Our default method is to choose the highest MELDA score, but we also experimented with choosing the most common topic among all the first sentences of documents in the cluster, shown in Table 1 as MELDA-FS.

In this way, the information in the summary is ordered so that sentences more gradually transition between topics, hence are more cohesive. In reality, with short summaries that are rarely more than three sentences, we expect a maximum of two topic shifts per summary.

The goal is to order information first by most salient topic overall among lead sentences, then gradually shift topics from sentence to sentence to avoid jarring topic changes, as well as "flip-flopping" back and forth between the same topics.

3.3.3 Content Realization

Further improving on the MEAD system, MELDA performs sentence compression on the selected summary sentences to remove unnecessary tokens and phrases, allowing more contentful text into the summary and, in theory, improving readability.

Sentence Compression We used the Spacy package to obtain POS tags and dependency parses for each sentence and used that information to identify potential tokens to remove from a sentence. Following CLASSY, we remove adverbs, initial conjugations, and any extraneous words such as bylines (Conroy et al., 2006). We also use the dependency parse to remove attributional phrases (e.g. "police said"), appositives, parentheticals and temporal modifiers, and any prepositional phrase containing a number.

Sentences are added to the summary one by one until the addition of the next highest-scoring sentence would push the total word count of the summary over 100 words. Only full sentences are added to the summary, but sentences that are over 100 words are ignored, to prevent empty summaries.

We also experimented with adding extra sentences to the summary (MELDA-EXTRA in Table 1). In order to do this, we add sentences to the summary according to the information ordering strategy described above, but if the next sentence would result in a summary over 100 words, it is skipped and the next sentence is considered. This approach allows us to fit as many candidate sentences into the summary as possible, while still getting some of the benefits of the information ordering strategy.

4 Results

4.1 ROUGE Scores

The final results for our two baseline systems Lead and MEAD; our MELDA system for the previous deliverable; and our latest system on both development and evaluation sets are given in Table 2.

ROUGE results for various configurations of MELDA are in Table 1. For all MELDA configurations, the underlying MEAD parameters, unless specified otherwise, are: Brown corpus (NLTK version) for IDF calculations, MEAD score weights of 1-1-1, three LDA topics, and five sentences chosen per topic. The default strategy for choosing the first topic to use for the cohesion gradient information ordering is the highest MELDA score. The default configuration also does not use NER or lemmatization and uses the NLTK word tokenization method.

The baseline MELDA score with this configuration is compared first with the addition of preprocessing. Despite our expectations, it is clear that the addition of NER and lemmatization using the Spacy tokenization functionality actually lowers our ROUGE scores; therefore, the rest of the results shown do not include pre-processing.

We also compared our default strategy for selecting the initial topic for the cohesion gradient with our alternate strategy of majority vote using the first sentences of all documents. This led to an increase in all scores, so we adopted this as our approach for all other configurations described in Table 1.

The other configuration parameter we explored was the inclusion of extra sentences in the summary. When the next sentence to add, as determined by MELDA score and cohesion gradient, is too long to fit under the token limit, rather than stopping we move on to the next best candidate and continue adding as many sentences as possible forcing the length of the summary to be as close to 100 tokens as possible. It is clear from the results shown in Table 1 that, as one might expect, this gives a significant boost to recall values, but lowers precision. However, since it boosted our F1 score for ROUGE-1, this configuration is the one we report in the final numbers for MELDA on the development and evaluation sets for this deliverable in Table 2.

We also experimented with different LDA parameters ($\langle k \rangle$-$\langle n \rangle$ in the configuration identifiers). Our default parameters are 3 topics with 5 sentences per topic as described above. To understand the impact of these parameters, we tried to run with many topics but only 1 sentence per topic, essentially testing whether LDA on its own would be able to identify an appropriate breadth of salient topics. This lowered our scores, indicating that not all the topics identified by LDA contain salient sentences.

Conversely, we also tried the minimum number of topics LDA allows, with a large number of sentences selected per topic. This raised our scores above the values for the default parameters. With this configuration and using cohesion gradient as

	R1-R	R1-P	R1-F1	R2-R	R2-P	R2-F1
MELDA	0.20306	0.24274	0.21999	0.04725	0.05689	0.05131
MELDA-PRE	0.19585	0.23607	0.21290	0.04277	0.05212	0.04677
MELDA-FS	0.20869	0.25054	0.22683	0.04733	0.05764	0.05175
MELDA-EXTRA	**0.22234**	0.23709	**0.22891**	**0.05073**	0.05397	0.05216
MELDA-5-1	0.19933	0.24632	0.21858	0.04547	0.05598	0.04976
MELDA-2-10	0.20763	**0.25293**	0.22661	0.04899	**0.06057**	**0.05377**
MELDA-2-10-EXTRA	0.22055	0.23317	0.22607	0.04968	0.05189	0.05061
MELDA-110.5	0.19661	0.24514	0.21706	0.04605	0.05804	0.05105

Table 1: Experimental MELDA Runs

	R1-R	R1-P	R1-F1	R2-R	R2-P	R2-F1
DEVELOPMENT SET (D3)						
Lead	0.18494	0.23168	0.20364	0.04753	0.06031	0.05247
MEAD	**0.19726**	**0.23328**	**0.21247**	**0.04378**	0.05176	**0.04717**
MELDA	0.16827	0.21598	0.18818	0.03968	**0.05182**	0.04474
DEVELOPMENT SET (D4)						
Lead	0.18999	0.23108	0.20654	0.04856	0.05862	0.05245
MEAD	0.19842	**0.25310**	0.21961	0.04816	**0.06131**	**0.05314**
MELDA	**0.22234**	0.23709	**0.22891**	**0.05073**	0.05397	0.05216
EVALUATION SET						
Lead	**0.29149**	**0.31217**	**0.30071**	**0.08923**	**0.09534**	**0.09192**
MEAD	0.28589	0.28410	0.28449	0.07912	0.07874	0.07878
MELDA	0.26877	0.27904	0.27300	0.07350	0.07580	0.07440

Table 2: Final ROUGE Results

our information ordering strategy, it is unlikely that more than one LDA topic is being included in our summaries. This indicates that one LDA topic contains more salient information than the other, so choosing more sentences from this topic results in a better summary.

However, forcing extra sentences into the summaries with this configuration did not boost the scores more than the default LDA parameters with extra sentences. It might be the case that with two LDA topics, one topic represents more salient sentences than the other, so forcing sentences into the summary from the less salient topic reduces scores, whereas with three topics, the extra sentences being forced into the summary might still be more salient.

We also noticed that there were still a large number of sentences included in the output summaries that were lead sentences. To ensure that our algorithm was not weighting lead sentences too greatly, we ran MELDA with a down-weighted first sentence score for MEAD, but since this reduced our ROUGE scores, we are confident that our weights are appropriate.

For reference, we have included the ROUGE-2 scores from the TAC 2010 and 2011 guided summarization tasks in Table 3. We can see that MELDA performs better than both of the baselines in TAC 2010 and the LEAD baseline in TAC 2011.

5 Discussion

5.1 System Strengths and Weaknesses

Beyond ROUGE scores, MELDA produced summaries that are quite readable and coherent in some cases, but that are less clear in other cases.

MELDA is overall effective in the domain of content selection; it selects topical and relevant sentences for the output summaries. In the MELDA summary in (1) (with the MELDA-EXTRA configuration), sentences (a) - (d) are, in general, examples of topical and informative sentence selection. (See §5.2 below for error analysis addressing issues such as the incomplete removal of the attributive modifier in sentence (b).)

That said, on occasion the sentence compression in the content realization stage removes important parts of the sentence, which leads to out-

Rouge2-F1	
TAC 2010 guided summarization task	
LEAD baseline	0.05376
MEAD baseline	0.05927
Best official	0.09574
TAC 2011 guided summarization task	
LEAD baseline	0.06410
MEAD baseline	0.08682
Best official	0.1344

Table 3: TAC Reference Scores from 2010 and 2011

	D1105A Plane crash Indonesia	D1106A Tuna overfishing	D1107B China food safety	D1108B Cyclone Sidr	D1109B Dimona attack	D1110B Sichuan earthquake	mean
Team 2	2.7	4.7	4.0	4.3	2.0	2.7	**3.389**
Topic mean all teams	3.2	3.0	3.0	2.9	2.9	3.4	3.2
Topic stddev all teams	0.7	0.8	0.8	0.94	0.6	0.7	

Table 4: Peer evaluations of MELDA summarization system

put summaries with little relevant content remaining. In (1), sentence (e) demonstrates this issue. The sentence in the summary output is not very informative and contributes little to the summary. Before compression removed the temporal modifier, sentence (e) was more informative: "Thailand's largest coral clean-up operation intensified Saturday, with hundreds of divers rushing to clear tonnes of debris polluting world-class dive sites in the Andaman Sea before April monsoons." This problem could perhaps be mitigated by performing sentence compression *before*, instead of after, the content selection phase.

In the case of information ordering, the cohesion gradient method was effective in some instances, producing summaries with sentences that gradually shifted topics. In example (1) we see relatively smooth shifting of topics from one sentence to the next. But, in other cases, summaries contain redundancies, or the summaries are too short, e.g, two sentences in length, and do not incorporate all of the cluster topics, thus leaving out information central to the news topic. This is discussed in more detail for example (4) in §5.2. Additional experiments with information ordering, including testing a wider variety of LDA topics, could lead to a configuration with improved intersentential cohesion.

1. (a) Long - term environmental lessons must be drawn from Asia's tsunami disaster experts say.
   (b) The Thai government has launched a project for volunteer divers to help revive coral reefs damaged in the tsunami last month, the Thai News Agency.
   (c) More than 58 percent of world's coral reefs are endangered because of pollution, over - harvesting of reef fish and other human factors.
   (d) Damage from the Indian Ocean tsunami could have been reduced if coastal areas had maintained their protective shields of mangrove swamps and coral reefs.
   (e) Thailand's largest coral clean - up operation intensified Saturday.

MELDA and MEAD often have overlapping elements in their summary outputs, which is expected given that MELDA is an extension of MEAD. Example (2) is the same summary for MEAD as (1) for MELDA (both are for the document cluster D1041). Both systems chose similar sentences, but the first sentence in the MELDA summary is arguably more suitable than the lead sentence chosen by MEAD.

2. (a) The Thai government has launched a project for volunteer divers to help revive coral reefs damaged in the tsunami last month, reported the Thai News Agency on Friday.
   (b) Long-term environmental lessons must be drawn from Asia's tsunami disaster, especially the consequences of ripping out mangroves and destroying coral reefs that help protect coasts from sea and storms, experts say.
   (c) More than 58 percent of the world's coral reefs are endangered because of pollution, the over-harvesting of reef fish and other human factors, the World Conservation Union (IUCN) said Friday.

5.2 Error Analysis

The errors we see frequently in MELDA output summaries are incomplete or otherwise ungrammatical sentences; illegal punctuation; missing clauses; confusing references to people, events, and times; and redundancy.

Ungrammatical sentences are sometimes a result of sentence compression ("Hackers overwhelmed three." and "Three Americans were.") and other times a product of errors in NLTK's sentence segmentation ("Gov."). Improving our compression rules in the first case and checking dependency parses or even simply word counts of each sentence in the second would help filter out these non-sentences.

Example (3) shows a summary with several of the other types of errors:

3. (a) Rabei Ousmane Sayed Ahmed is considered by Spanish prosecutors to be one of four masterminds of the deadly attacks.
   (b) Youssef Belhadj is believed to be " Abou Dujanah who appeared on a video found after the blasts claiming responsibility.
   (c) Two members of Basque separatist group arrested while transporting half a tonne of explosives to Madrid Monday received jail sentences.
   (d) There is no material evidence accused, his lawyer told AFP.
   (e) Rabei Ousmane Sayed Ahmed rejected all charges against him.
   (f) A Moroccan suspected refused to answer questions from a prosecutor Friday, as the trial in the case went.
   (g) The trial opened Thursday.

In (3), unpaired quotation marks (sentence (b)), pronouns with no clear antecedents (sentence (d), (e)), missing nouns in noun phrases (sentence (f)), and unresolved temporal and other references (sentence (g)) hurt the cohesion and coherence of this summary. Errors like these could be improved by implementing coreference resolution and improving sentence compression.

Example (4) contains a summary with the redundancy error:

4. (a) A major manufacturer of dog and cat food sold under Wal - Mart, Safeway, Kroger and other store brands recalled 60 million containers of wet pet food Friday after reports of kidney failure and deaths.
   (b) They expected the death toll to rise from pet food contamination that has prompted a recall of 60 million cans and pouches of " cuts - and - gravy" meals.
   (c) Like many retail chains, PetSmart pulled several varieties of wet dog and cat foods made by Menu Foods that were recalled and have been linked by the U.S. Food and Drug Administration to the deaths.

In (4), the second and third sentences refer to events mentioned in the first sentence as if for the first time, and all three sentence in the summary say very similar things about the event.

5.3 Human Evaluation

The results of the human evaluation of our eval summaries are given in Table 4. Six summaries were evaluated by three individuals. Three of our summaries outperform other systems and three of our summaries score lower than other systems. Overall, on a scale of 1-5, our human evaluation average score is 3.389, which is slightly higher than the class average of 3.2.

The highest scoring MELDA summary, with an average human evaluation score of 4.7, is shown in

(5). It is for eval document cluster D1106A about tuna overfishing.

5. (a) The European Union and Japan agreed to slash their tuna quotas by more than 20 percent in an effort to prevent the popular fish being hunted to extinction.
   (b) 300 representatives from five international conservation bodies and more than 60 countries and regions gathered in the Japanese port city of Kobe.
   (c) Five international tuna conservation bodies, known as Regional Fisheries Management Organizations adopted a historical joint action plan, pledging to strengthen efforts to combat tuna overfishing and ensure sustainable use of the resources.

The high score is most likely attributed to the summary containing full coherent sentences, salient information, cohesive transitions from one sentence to the next, and minimal overlap in sentence topics. However, one reviewer noted that this summary is "awkward mostly due to repetitiveness" and scored the summary 4/5. The other reviewers scored the summary 5/5.

MELDA's lowest scoring summary, with an average human evaluation score of 2.0, is shown in (6). The topic is the Dimona attack (eval document cluster D1109B).

6. (a) A spokesman for al - Aqsa Martyrs Brigades, denied the reports that the two suicide bombers attacking Israel entered Israel via Egypt's Sinai Peninsula the border fence with Gaza had been blew holes.
   (b) A Palestinian suicide bomber denoted in a commercial center in southern Israeli city of Dimona, a year, killing an Israeli woman and wounding eight others.
   (c) Responsibility.
   (d) With emotions running after a suicide bombing in southern Israeli town of Dimona, Israeli Prime Minister Ehud Olmert Monday vowed at a Kadima Party meeting to fight terror ,.

In this case, a number of factors likely contribute to the lower score. The most prominent is the effect of sentence compression leading to ungrammatical or hard-to-interpret sentences (sentences (a), (b)). Sentence (c) of the summary is just one word, "Responsibility," which is also the result of over-compression. These omissions detract from the readability and cohesion of the summary. Lastly, the punctuation is incorrect in a few places, e.g. the end of sentence (d).

One of the reviewers commented on this summary, saying, "that this is about a suicide bombing in Israel is about the only information that can be recovered with confidence." We agree that this is not the best performing example output from our system, and it highlights a number of areas that could be improved in MELDA that we discussed above.

6 Conclusion

We developed two baseline systems: a lead-sentence system and a system based on the original MEAD model (Radev et al., 2001), as well as an expanded system called MELDA, which adds LDA topic modeling to the MEAD score calculation process. LDA is used in content selection by incorporating topic scores into overall scores for sentences, and again in information ordering to improve cohesiveness. The outputs of the MELDA system have comparable ROUGE scores to those from MEAD with some variation depending on the dataset.

References

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python*. O'Reilly Media.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

John M Conroy, Judith D Schlesinger, Dianne P Oleary, and Jade Goldstein. 2006. Back to basics: Classy 2006. In *Proceedings of DUC*, volume 6, page 150.

Dragomir R Radev, Sasha Blair-Goldensohn, and Zhu Zhang. 2001. Experiments in single and multidocument summarization using mead. In *First document understanding conference*, page 1À8. Citeseer.

Dragomir Radev, Adam Winkel, and Michael Topper. 2002. Multi document centroid-based text summarization. In *ACL 2002*. Citeseer.

Dragomir R Radev, Hongyan Jing, Małgorzata Styś, and Daniel Tam. 2004. Centroid-based summarization of multiple documents. *Information Processing & Management*, 40(6):919–938.