Evaluating a summarizer for legal text with a large text collection

Frank Schilder & Hugo Molina-Salgado
R&D
Thomson Legal & Regulatory
610 Opperman Drive
Eagan, MN 55123, USA
<FirstName.LastName>@Thomson.com

Abstract

This paper describes a novel approach to summarizing legal text (i.e. case law) and shows how two automatic evaluation methods (i.e. ROUGE and Gestalt pattern matching) and a (semi-)human-based evaluation can be used for evaluating summarizers using a large legal corpus.

1 Introduction

Summarizing legal text faces different challenges than summarizing news messages. Foremost, legal text has a different text structure than news messages which are often written in the inverse pyramid style. Consequently, recent approaches to summarizing case law documents focus on categorizing sentences according to so-called argumentative roles (Hachey and Grover, 2005). However, this requires extensive linguistic analysis and the automatic categorization of each sentence according to its role.

We take a different route by leveraging the repetition of (legal) phrases in the text. Similar to recent graph-based approaches to summarization, we propose an approach that generates a graph-representation of the text solely based on a similarity function between sentences. The similarity function as well as the voting algorithm on the derived graph representation is different from other graph-based approaches (e.g. LexRank) and shows good results compared with other systems based on term frequency, Maximum Marginal Relevance (MMR) (Carbonell and Goldstein, 1998) or centroids (Radev et al., 2004).

In addition to proposing a new approach to summarizing legal text, we show that an automatic evaluation method used mainly for news (i.e. ROUGE) can also be used for legal text. Moreover, we propose a new automatic evaluation approach that relies on a large corpus of text that comes with annotated summaries and links to relevant text segments. For the evaluation, we generated over 4000 summaries.[1]

2 SummaryFinder: summarization of legal text

For legal text, we hypothesized that some paragraphs summarize the entire text or at least parts of the text. In order to find such paragraphs, we implemented a system that computes inter-paragraph similarity scores and selects the best match for every paragraph. The system acts like a voting system where each paragraph casts a vote for another paragraph (its best match). The top paragraphs with the most votes were selected as the summary. The vote casting can be seen as a similarity function based on phrase similarity. Phrase similarity is computed by looking for phrases that co-occur in two paragraphs. The longer the matched phrase is the higher the score. The logic behind this ranking is that a writer often copies and pastes phrases from the main text into a summary and re-phrases the text only minimally.

2.1 General approach

Scoring Each paragraph (p_i) is compared with every other paragraph ($p_j, i \neq j$) and a score is computed for each pair (p_i, p_j). To generate this score, the text in each paragraph is stemmed and stop words are discarded, then a word by word compar-

[1] The current DUC competition relies, for example, only on 50 summaries for their evaluation.

ison is performed between each pair of paragraphs counting the number of words that match. During this matching process, if a sequence of consecutive words is found in both paragraphs, the individual word counts are accumulated on position as in a prefix summation: for example, consider two paragraphs with the following non-consecutive words (r,s,t,u and v) in them:

$$.....r....s....t..u.........v....$$

$$..r....s....t....u...v....$$

For this pair the score would be 5, because five non-consecutive matches were found. Now consider the case where the words are consecutive:

$$.....rstuv....$$

$$.....rstuv....$$

Here for the r position the count would be 1 (r : 1), then when the next word also matches the count for s would be 2 and so on, for t : 3, u : 4 and v : 5 because we have 5 matches in a row. Finally, the score for the paragraph's match would be $1 + 2 + 3 + 4 + 5 = 15$. Therefore, for a matching sequence seq of n consecutive non-stop words, the score grows by $\sum_{i=1}^{n} i$.

Voting When the scoring process is finished, every paragraph p_i, $1 \leq i \leq m$ casts a vote, for paragraph p_j, with $1 \leq j \leq m$ and $i \neq j$, whose matching score (s_{ij}) was the highest among all paragraphs in the text (i.e. $s_{ij} = max(s_{i1}, s_{i2} \dots s_{im})$).

The votes for each paragraph (p_i) are stored in a vector R whose elements r_i contains the number of votes p_i received from the other paragraphs in the text. With this information we create a list of pairs (r_i, i), which is sorted in descending order according to the value of r_i and produces a ranking that contains the most important (most voted for) paragraphs at the top.

2.2 Static vs. dynamic

We developed two versions of the SummaryFinder:

static Given a document, the SummaryFinder produces an ordered list of paragraphs according to their importance. The first one or two paragraphs in this list should provide a good description of what the case is about.

dynamic Given a list of paragraphs and sentences from the original document, this version of the SummaryFinder extracts the sentence that is most similar to the query. The scoring is same as the scoring used for the static summarization, but without the voting.

3 Experiment setup

We were able to use a large collection of cases called *FSupp* collection (Turtle, 1994) containing short summaries of the points of laws discussed in the respective cases. These summaries, or head notes, are written by editorial staff and capture the different points of law in a case. Head notes are then classified to a topical hierarchy called the key number system. Each category or key number has a string associated with it (e.g. *Requisites and Validity of Contract Ratification of voidable contract*). In addition, editors provide a link to a paragraph in the text where the respective point of law is being discussed.

We carried out the following experiment in query-based summarization: the key number text was taken as a query in order to simulate a task-based summarization. The head note was the model summary the output from the systems was compared to.

For the experiments, we ran the following summarizer programs with about 5 queries per case: (a) SummaryFinder (b) OTS: a simple term-based open source summarizer,[2] (c) Lemur: an open source indexer that does Maximum Marginal Relevance (MMR),[3] (d) MEAD: a multi-document summarization system developed by the University of Michigan that also allows for query-based single-document summarization,[4] (e) Extractor: a commercial system developed by Peter Turney.[5]

In addition, we constructed with three baselines: (a) first n words. (b) last n words, and (c) first $n/2$ words and last $n/2$ words. The first baseline is a very strong baseline for news messages, since news

[2] `http://libots.sourceforge.net`

[3] `http://www.lemurproject.org/doxygen/lemur/html/MMRSummApp.html`

[4] `http://www.summarization.com/mead/`

[5] `http://www.extractor.com/`. We obtained a restricted evaluation copy. The version we tried requires to press a button every time you run the program. Hence we were only able to run the program for about 100 cases.

messages are written according to the inverted pyramid scheme.

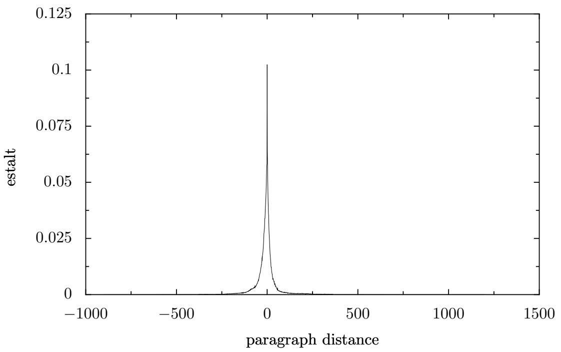

Figure 1: Micro-averaged similarity-scores for linked paragraph (all documents)

4 Evaluation methods

To automatically evaluate the summaries, we used ROUGE[6] and a string matching algorithm that is based on the longest common subsequence and called Gestalt pattern matching or Ratcliff/Obershelp pattern recognition (Ratcliff and Metzener, 1988).

For the human-based evaluation, we leveraged the information given by the editors in form of the head notes and the key number link. Using this editorial link from the headnote to the case, we hypothesized that a good summarization system would on average produce more sentences that are within the linked paragraph. In order to back up our assumption that these linked paragraphs do indeed contain information on the point of law indicated by the key number, we carried out the following experiment. We measured the Gestalt pattern matching score for the head note and all paragraphs in the text. Figure 1 is a plot of these scores as a function of the distance between the link and the paragraphs. As indicated in Figure 1, we can conclude that the linked paragraph has indeed a high degree of overlap with the head note text.

For our (semi-)human-based evaluation called LinkPara, we matched the sentences that were extracted by the summarizers with the sentences from this paragraph and counted the words. The higher the number of words the higher the score for the summarization system. Note that this scoring is different from ROUGE-1 which counts uni-grams between the model summaries and the system's summary, because we require a sentence match first.

5 Evaluation results

The evaluation was carried out for 50-word-long summaries which is the average length of a headnote. Before describing the results in more detail, we need to point out some possible caveats with using ROUGE for the legal domain and the type of data we applied it to. ROUGE produces more reliable results if more than one model summary is used. The *FSupp* collection, however, contains only head notes, which are written by the same person for that case. We still think that we can use ROUGE, because (a) legal summaries are much more standardized summaries than summaries for news stories and (b) we ran the evaluation on a much larger data set (ca. 800 cases).

The results for all three metrics showed that the dynamic SummaryFinder outperformed the other systems except for Lemur, as measured by ROUGE-2 (cf. Figure 2). Lemur obtained high scores with the exception of the Gestalt pattern score (cf. Figure 3).

The various MEAD systems[7] received rather average scores for all three metrics. OTS system shows higher scores for Gestalt and LinkPara, but not for ROUGE-2.

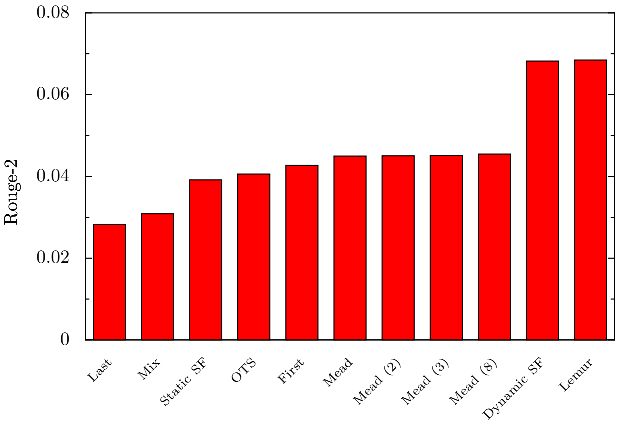

Figure 2: ROUGE-2 scores

[6] http://www.isi.edu/∼cyl/ROUGE/

[7] We ran the MEAD system with different weights for the query-based information (w=1/2/3/8). There were no significant changes in the results observable.

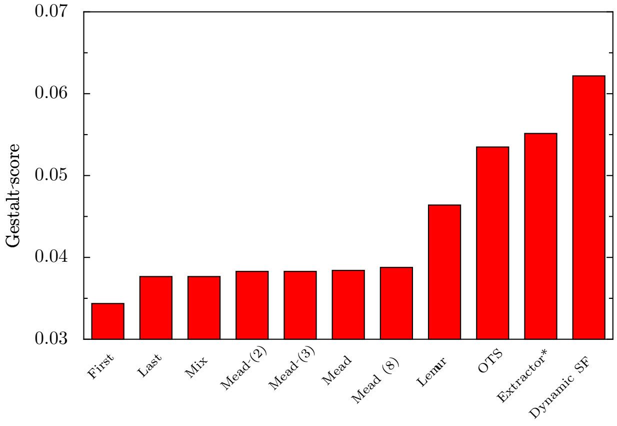

Figure 3: Gestalt pattern matching scores

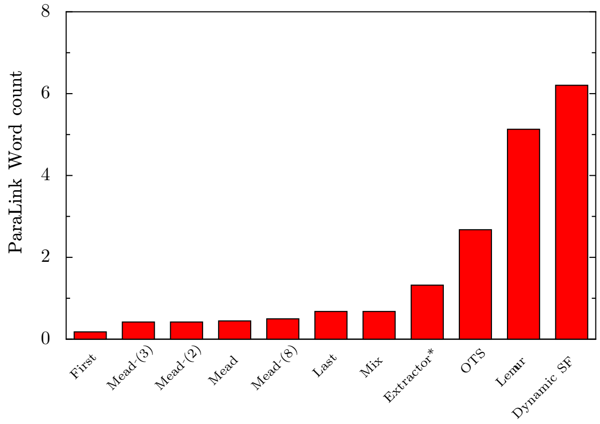

Figure 4: LinkPara scores: micro-averaged number of words in matched sentences

For the data and results we collected,[8] we computed the Pearson coefficients to show whether automatic evaluation methods actually correlate to the (semi-)human-based evaluation. Table 1 shows the results of all possible automatic/(semi-)human-based combinations.

	LinkPara	Gestalt
ROUGE-2	0.90 (0.00631)	0.65 (0.11235)
ROUGE-SU4	0.93 (0.00274)	0.69 (0.08647)
Gestalt	0.85 (0.01461)	1

Table 1: The Pearson coefficients between the automatic and the (semi-)human-based metrics

The scores for ROUGE-2-LinkPara and ROUGE-SU4-LinkPara are highly significant ($p < 0.01$), hence we can conclude a correlation between the (semi-)human-based evaluation and ROUGE. The Pearson coefficient for Gestalt-LinkPara is not highly significant but still high ($p < 0.02$).

Unfortunately, the two automatic measures ROUGE and Gestalt do not correlate to each other. One reason for this could be the relatively small number of systems. In order to determine whether the Gestalt pattern matching algorithm is generally a good evaluation method, we applied this method also to last year's DUC data, but we found no correlation between the human-based metric of responsiveness and the Gestalt method.

These findings could be explained by the different writing styles one finds in news summaries and legal head notes. The former contains more paraphrases and rewriting of the original text, whereas the latter often copies sentences or clauses from the case. Consequently, the Gestalt pattern matching approach may be sufficient for an evaluation of legal data, but not for news data.

6 Conclusions

We showed that a summarization system for legal text that relied on the repetitiveness of legal text outperforms current state-of-the-art summarization systems. We tested our approach with two automatic methods (i.e. ROUGE and a string comparison algorithm) as well as a (semi-)human-based evaluation method. We were able to show that the automatic methods correlate highly to the (semi-)human-based evaluation method but not to each other. Moreover, we found that the string comparison algorithm worked well for legal data but not for news data we took from DUC.

References

Jaime G. Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *SIGIR '98: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 24-28 1998, Melbourne, Australia*, pages 335–336. ACM.

Ben Hachey and Claire Grover. 2005. Automatic legal text summarisation: Experiments with summary structuring. In *Proceedings of the 10th International Conference on Artificial Intelligence and Law (ICAIL 2005)*, Bologna, Italy.

[8] We did not include the results for the Extractor summarizer for the correlation computation.

Dragomir Radev, Timothy Allison, Sasha Blair-Goldensohn, John Blitzer, Arda Çelebi, Stanko Dimitrov, Elliott Drabek, Ali Hakim, Wai Lam, Danyu Liu, Jahna Otterbacher, Hong Qi, Horacio Saggion, Simone Teufel, Michael Topper, Adam Winkel, and Zhu Zhang. 2004. MEAD - a platform for multidocument multilingual text summarization. In *LREC 2004*, Lisbon, Portugal, May.

John W. Ratcliff and David Metzener. 1988. Pattern matching: The Gestalt Approach. *Dr. Dobb's Journal*, 7:46.

Howard R. Turtle. 1994. Natural language vs. boolean query evaluation: A comparison of retrieval performance. In W. Bruce Croft and C. J. van Rijsbergen, editors, *Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Dublin, Ireland, 3-6 July 1994 (Special Issue of the SIGIR Forum)*, pages 212–220. ACM/Springer.