CHAPTER 32

TEXT SUMMARIZATION

EDUARD HOVY

Abstract

This chapter describes research and development on the automated creation of summaries of one or more texts. It presents an overview of the principal approaches in summarization, describes the design, implementation, and performance of various summarization systems, and reviews methods of evaluating summaries.

32.1 The Nature of Summaries

Early experimentation in the late 1950s and early 1960s suggested that text summarization by computer was feasible though not straightforward (Luhn 1959; Edmundson 1969). After a hiatus of some decades, progress in language processing, coupled with great increases of computer memory and speed, and the growing presence of on-line text—in corpora and especially on the web—renewed interest in automated text summarization.

Despite encouraging results, some fundamental questions remain unaddressed. For example, no one seems to know exactly what a **summary** is. In this chapter we use *summary* as a generic term and define it as follows:

(32.1) **Definition**: a summary is a text that is produced from one or more texts, that contains a significant portion of the information in the original text(s), and that is no longer than half of the original text(s).

'Text' here includes multimedia documents, on-line documents, hypertexts, etc. Of the many types of summary that have been identified (Spärck Jones 1999; Hovy and Lin 1999), **indicative** summaries (that provide an idea of what the text is about without giving any content) and **informative** ones (that do provide some shortened version of the content) are often referenced. **Extracts** are summaries created by reusing portions (words, sentences, etc.) of the input text verbatim, while **abstracts** are created by re-generating the extracted content.

Section 32.2 outlines the principal approaches to automated text summarization in general. Section 32.3 reviews particular techniques used in several summarization systems. Problems unique to multi-document summarization are discussed in section 32.4. Finally, although the evaluation of summaries (and of summarization) is not yet well understood, we review approaches to evaluation in section 32.5.

32.2 The Stages of Automated Text Summarization

Researchers in automated text summarization have identified three distinct stages (Spärck Jones 1999; Hovy and Lin 1999; Mani and Maybury 1999). Most systems today embody the first stage only.

The first stage, *topic identification*, produces the simplest type of summary. (We define **topic** as a particular subject that we write about or discuss.) Whatever criterion of importance is used, once the system has identified the most important unit(s) (words, sentences, paragraphs, etc.), it can either simply list them (thereby creating an extract) or display them diagrammatically (thereby creating a schematic summary). Typically, topic identification is achieved using several complementary techniques. We discuss topic identification in section 32.3.1.

In many genres, humans' summaries reflect their own *interpretation*: fusion of concepts, evaluation, and other processing. This stage generally occurs after topic identification. Since the result is something new, not explicitly contained in the input, this stage requires that the system have access to knowledge separate from the input. Given the difficulty of building domain knowledge, few existing systems perform interpretation, and no system includes more than a small domain model. We discuss interpretation in section 32.3.2.

The results of interpretation are usually unreadable abstract representations, and even extracts are seldom coherent, due to dangling references, omitted discourse linkages, and repeated or omitted material. Systems therefore include a stage of *summary generation* to produce human-readable text. In the case of extracts, generation may simply mean 'smoothing' the extracted pieces into a coherent, densely phrased, text. We discuss generation in section 32.3.3.

32.3 Review of Summarization Methods

32.3.1 Stage 1: Topic identification

To perform this stage, almost all systems employ several independent modules. Each module assigns a score to each unit of input (word, sentence, or longer passage); then a combination module combines the scores for each unit to assign a single integrated score to it; finally, the system returns the *n* highest-scoring units, according to the summary length requested by the user.

An open issue is the size of the unit of text that is scored for extraction. Most systems focus on one sentence at a time. However, Fukushima, Ehara, and Shirai (1999) show that extracting subsentence-size units produces shorter summaries with more information. On the other hand, Strzalkowski et al. (1999) show that including certain sentences immediately adjacent to important sentences increases coherence—fewer dangling pronoun references, etc.

The performance of topic identification modules is usually measured using Recall and Precision scores (see section 32.5 and Chapter 22). Given an input text, a human's extract, and a system's extract, these scores quantify how closely the system's extract corresponds to the human's. For each unit, we let *correct* = the number of sentences extracted by the system and the human; *wrong* = the number of sentences extracted by the system but not by the human; and *missed* = the number of sentences extracted by the human but not by the system. Then

(32.2) $\mathit{Precision} = \mathit{correct} \,/\, (\mathit{correct} + \mathit{wrong})$

(32.3) $\mathit{Recall} = \mathit{correct} \,/\, (\mathit{correct} + \mathit{missed})$

so that Precision reflects how many of the system's extracted sentences were good, and Recall reflects how many good sentences the system missed.

Positional criteria. Thanks to regularities in the text structure of many genres, certain locations of the text (headings, titles, first paragraphs, etc.) tend to contain important information. The simple method of taking the lead (first paragraph) as summary often outperforms other methods, especially with newspaper articles

(Brandow, Mitze, and Rau 1995). Some variation of the **position method** appears in Baxendale (1958); Edmundson (1969); Donlan (1980); Kupiec, Pedersen, and Chen (1995); Teufel and Moens (1997); Strzalkowski et al. (1999); Kupiec et al. and Teufel and Moens both list this as the single best method, scoring around 33 per cent, for news, scientific, and technical articles.

In order to automatically determine the best positions, and to quantify their utility, Lin and Hovy (1997) define the genre- and domain-oriented Optimum Position Policy (OPP) as a ranked list of sentence positions that on average produce the highest yields for extracts, and describe an automated procedure to create OPPs given texts and extracts.

Cue phrase indicator criteria. Since in some genres certain words and phrases ('significant', 'in this paper we show') explicitly signal importance, sentences containing them should be extracted. Teufel and Moens (1997) report 54 per cent joint recall and precision, using a manually built list of 1,423 **cue phrases** in a genre of scientific texts. Each cue phrase has a (positive or negative) 'goodness score', also assigned manually. In Teufel and Moens (1999) they expand their method to argue that rather than single sentences, these cue phrases signal the nature of the multi-sentence rhetorical blocks of text in which they occur (such as Purpose/Problem, Background, Solution/ Method, Conclusion/Claim).

Word and phrase frequency criteria. Luhn (1959) used Zipf's Law of word distribution (a few words occur very often, fewer words occur somewhat often, and many words occur infrequently) to develop the following extraction criterion: if a text contains some words unusually frequently, then sentences containing these words are probably important.

The systems of Luhn (1959), Edmundson (1969), Kupiec, Pedersen, and Chen (1995), Teufel and Moens (1999), Hovy and Lin (1999), and others employ various frequency measures, and report performance of between 15 per cent and 35 per cent recall and precision (using word frequency alone). But both Kupiec et al. and Teufel and Moens show that word frequency in combination with other measures is not always better. Witbrock and Mittal (1999) compute a statistical model describing the likelihood that each individual word in the text will appear in the summary, in the context of certain features (part-of-speech tag, word length, neighbouring words, average sentence length, etc.). The generality of this method (also across languages) makes it attractive for further study.

Query and title overlap criteria. A simple but useful method is to score each sentence by the number of desirable words it contains. Desirable words are, for example, those contained in the text's title or headings (Kupiec, Pedersen, and Chen 1995; Teufel and Moens 1997; Hovy and Lin 1999), or in the user's query, for a **query-based summary** (Buckley and Cardie 1997; Strzalkowski et al. 1999; Hovy and Lin 1999). The query method is a direct descendant of IR techniques (see Chapter 29).

Cohesive or lexical connectedness criteria. Words can be connected in various ways, including repetition, coreference, synonymy, and semantic association as expressed in thesauri. Sentences and paragraphs can then be scored based on the degree of connectedness of their words; more-connected sentences are assumed to be more important. This method yields performances ranging from 30 per cent (using a very strict measure of connectedness) to over 60 per cent, with Buckley and Cardie's use of sophisticated IR technology and Barzilay and Elhadad's lexical chains (Salton et al. 1997; Mitra, Singhal, and Buckley 1997; Mani and Bloedorn 1997; Buckley and Cardie 1997; Barzilay and Elhadad 1999). Mani and Bloedorn represent the text as a graph in which words are nodes and arcs represent adjacency, coreference, and lexical similarity.

Discourse structure criteria. A sophisticated variant of connectedness involves producing the underlying discourse structure of the text and scoring sentences by their discourse centrality, as shown in Marcu (1997, 1998). Using a GSAT-like algorithm to learn the optimal combination of scores from centrality, several of the above-mentioned measures, and scores based on the shape and content of the discourse tree, Marcu's (1998) system does almost as well as people for *Scientific American* texts.

Combination of various module scores. In all cases, researchers have found that no single method of scoring performs as well as humans do to create extracts. However, since different methods rely on different kinds of evidence, combining them improves scores significantly. Various methods of automatically finding a combination function have been tried; all seem to work, and there is no obvious best strategy.

In their landmark work, Kupiec, Pedersen, and Chen (1995) train a Bayesian classifier (see Chapter 19) by computing the probability that any sentence will be included in a summary, given the features paragraph position, cue phrase indicators, word frequency, upper-case words, and sentence length (since short sentences are generally not included in summaries). They find that, individually, the paragraph position feature gives 33 per cent precision, the cue phrase indicators 29 per cent (but when joined with the former, the two together give 42 per cent), and so on, with individual scores decreasing to 20 per cent and the combined five-feature score totalling 42 per cent.

Also using a Bayesian classifier, Aone et al. (1999) find that even within the single genre, different newspapers require different features to achieve the same performance.

Using SUMMARIST, Lin (1999) compares eighteen different features, a naive combination of them, and an optimal combination obtained using the machine learning algorithm C4.5 (Quinlan 1986). These features include most of the above mentioned, as well as features signalling the presence in each sentence of proper names, dates, quantities, pronouns, and quotes. The performances of the individual methods and the naive and learned combination functions are graphed in Fig. 32.1, showing extract length against f-score (joint recall and precision). As expected, the top scorer is the learned combination function. The second-best score is achieved by query term overlap (though in other topics the query method did not do as well). The third best score (up to the 20 per cent length) is achieved equally by word frequency, the lead method, and the naive combination function. The curves in general indicate that to be most

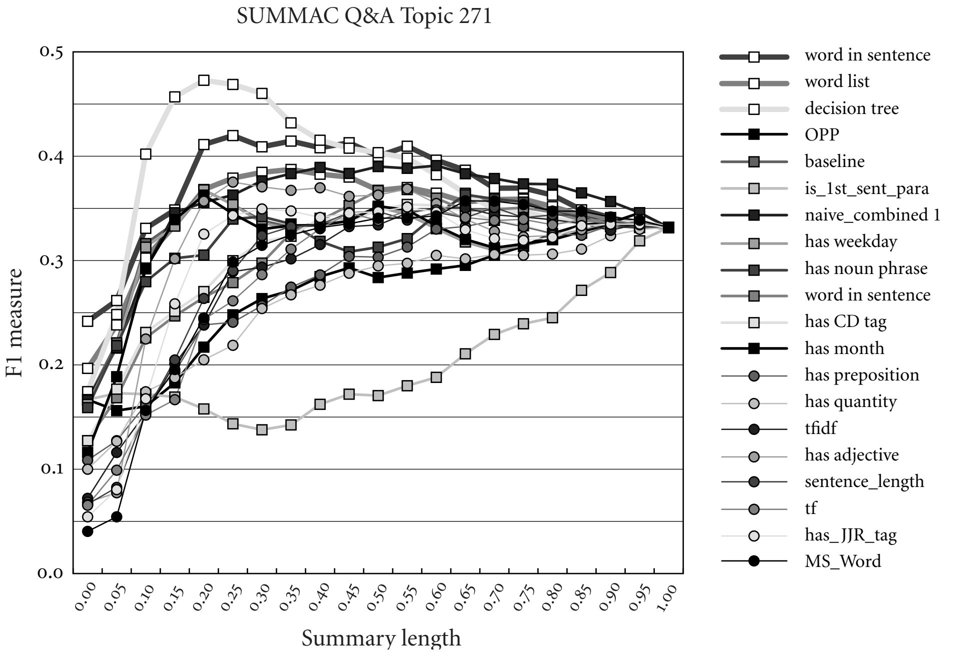

Fig. 32.1 Summary length vs. f-score for individual and combined methods of scoring sentences in SUMMARIST

useful, summaries should not be longer than about 35 per cent and not shorter than about 15 per cent; no 5 per cent summary achieved an f-score of over 0.25.

32.3.2 Stage 2: Interpretation or topic fusion

As described in section 32.2, the stage of *interpretation* is what distinguishes extract-type summarization systems from abstract-type systems. During interpretation, the topics identified as important are fused, represented in new terms, and expressed using a new formulation, using concepts or words not found in the original text.

No system can perform interpretation without prior knowledge about the domain; by definition, it must interpret the input in terms of something extraneous to the text. But acquiring enough (and deep enough) prior domain knowledge is so difficult that summarizers to date have only attempted it in a small way.

At first glance, the template representations used in information extraction (Chapter 30), or other interpretative structures in terms of which to represent stories for summarization, hold some promise (DeJong 1978; Lehnert 1981; Rau and Jacobs 1991). But the difficulty of building such structures and filling them makes large-scale summarization impractical at present.

Taking a more formal approach, Hahn and Reimer (1999) develop operators that condense knowledge representation structures in a terminological logic through conceptual abstraction (Chapter 5). To date, no parser has been built to produce the knowledge structures from text, and no generator to produce language from the results.

Taking a leaf from IR, Hovy and Lin (1999) use topic signatures—sets of words and relative strengths of association, each set related to a single headword—to perform topic fusion. By automatically constructing these signatures (using 30,000 texts from the *Wall Street Journal* and *TF*IDF* to identify for each topic the set of words most relevant to it) they overcome the knowledge paucity problem. They use these topic signatures both during topic identification (to score sentences by signature overlap) and during topic interpretation (to substitute the signature head for the sentence(s) containing enough of its words). The effectiveness of signatures to perform interpretation has not yet been shown.

Interpretation remains blocked by the problem of domain knowledge acquisition. Before summarization systems can produce abstracts, this problem will have to be solved.

32.3.3 Stage 3: Summary generation

The third major stage of summarization is generation. When the summary content has been created through abstracting and/or information extraction, it exists within the computer in internal notation, and thus requires the techniques of natural language generation, namely text planning, sentence (micro-)planning, and sentence realization. For more on this topic see Chapter 15.

However, as mentioned in section 32.2, extract summaries require no generation stage. In this case, though, various dysfluencies tend to result when sentences (or other extracted units) are simply extracted and printed—whether they are printed in order of importance score or in text order. A process of 'smoothing' can be used to identify and repair typical dysfluencies, as first proposed in Hirst et al. (1997). The most typical dysfluencies that arise include repetition of clauses or NPs (where the repair is to aggregate the material into a conjunction), repetition of named entities (where the repair is to pronominalize), and inclusion of less important material such as parentheticals and discourse markers (where the repair is to eliminate them). In the context of summarization, Mani, Gates, and Bloedorn (1999) describe a summary revision program that takes as input simple extracts and produces shorter and more readable summaries.

Text compression is another promising approach. Knight and Marcu's (2000) prize-winning paper describes using the EM algorithm to train a system to compress the syntactic parse tree of a sentence in order to produce a single, shorter, one, with the idea of eventually shortening two sentences into one, three into two (or one), and so on. Banko, Mittal, and Witbrock (2000) train statistical models to create headlines for texts by extracting individual words and ordering them appropriately.

Jing and McKeown (1999) make the extract-summary point from the generation perspective. They argue that summaries are often constructed from the source document by a process of cut and paste—fragments of document sentences are combined into summary sentences—and hence that a summarizer need only identify the major fragments of sentences to include and then weave them together grammatically. To prove this claim, they train a hidden Markov model to identify where in the document each (fragment of each) summary sentence resides. Testing with 300 human-written abstracts of newspaper articles, Jing and McKeown determine that only 19 per cent of summary sentences do not have matching sentences in the document.

In an extreme case of cut and paste, Witbrock and Mittal (1999; see section 32.3.1) extract a set of words from the input document and then order the words into sentences using a bigram language model.

32.4 Multi-Document Summarization

Summarizing a single text is difficult enough. But summarizing a collection of thematically related documents poses several additional challenges. In order to avoid repetitions, one has to identify and locate thematic overlaps. One also has to decide what to include of the remainder, to deal with potential inconsistencies between documents, and, when necessary, to arrange events from various sources along a single timeline. For these reasons, **multi-document summarization** is much less developed than its single-document cousin.

Various methods have been proposed to identify cross-document overlaps. SUMMONS (Radev 1999), a system that covers most aspects of multi-document summarization, takes an information extraction approach. Assuming that all input documents are parsed into templates (whose standardization makes comparison easier), SUMMONS clusters the templates according to their contents, and then applies rules to extract items of major import. In contrast, Barzilay, McKeown, and Elhadad (1999) parse each sentence into a syntactic dependency structure (a simple parse tree) using a robust parser and then match trees across documents, using paraphrase rules that alter the trees as needed.

To determine what additional material should be included, Carbonell, Geng, and Goldstein (1997) first identify the units most relevant to the user's query, using methods described in section 32.3.1, and then estimate the 'marginal relevance' of all remaining units using a measure called Maximum Marginal Relevance (MMR).

SUMMONS deals with cross-document overlaps and inconsistencies using a series of rules to order templates as the story unfolds, identify information updates (e.g. increasing death tolls), identify cross-template inconsistencies (decreasing death tolls), and finally produce appropriate phrases or data structures for the language generator.

Multi-document summarization poses interesting challenges beyond single documents (Goldstein et al. 2000; Fukumoto and Suzuki 2000; Kubota Ando et al. 2000). An important study (Marcu and Gerber 2001) show that for the newspaper article genre, even some very simple procedures provide essentially perfect results. For example, taking the first two or three paragraphs of the most recent text of a series of texts about an event provides a summary equally coherent and complete as that produced by human abstracters. Obviously, this cannot be true of more complex types of summary, such as biographies of people or descriptions of objects. Further research is required on all aspects of multi-document summarization before it can become a practical reality.

32.5 Evaluating Summaries

How can you evaluate the quality of a summary? The growing body of literature on this interesting question suggests that summaries are so task and genre specific and so user oriented that no single measurement covers all cases. In section 32.5.1 we describe a few evaluation studies and in section 32.5.2 we develop some theoretical background.

32.5.1 Previous evaluation studies

As discussed in Chapter 22, many NLP evaluators distinguish between black-box and glass-box evaluation. Taking a similar approach for summarization systems, Spärck Jones and Galliers (1996) define **intrinsic** evaluations as measuring output quality (only) and **extrinsic** as measuring user assistance in task performance (see also Chapter 22).

Most existing evaluations of summarization systems are intrinsic. Typically, the evaluators create a set of ideal summaries, one for each test text, and then compare the summarizer's output to it, measuring content overlap (often by sentence or phrase recall and precision, but sometimes by simple word overlap). Since there is no 'correct' summary, some evaluators use more than one ideal per test text, and average the score of the system across the set of ideals. Comparing system output to some ideal was performed by, for example, Edmundson (1969); Paice (1990); Ono, Sumita, and Miike (1994); Kupiec, Pedersen, and Chen (1995); Marcu (1997); Salton et al. (1997). To simplify evaluating extracts, Marcu (1999) and Goldstein et al. (1999) independently developed an automated method to create extracts corresponding to abstracts.

A second intrinsic method is to have evaluators rate systems' summaries according to some scale (readability; informativeness; fluency; coverage); see Brandow, Mitze, and Rau (1995) for one of the larger studies.

Extrinsic evaluation is easy to motivate; the major problem is to ensure that the metric applied correlates well with task performance efficiency. Examples of extrinsic evaluation can be found in Morris, Kasper, and Adams (1992) for GMAT testing, Miike et al. (1994) for news analysis, and Mani and Bloedorn (1997) for information retrieval.

The largest extrinsic evaluation to date is the TIPSTER-SUMMAC study (Firmin Hand and Sundheim 1998; Firmin and Chrzanowski 1999), involving some eighteen systems (research and commercial), in three tests. In the Categorization task testers classified a set of TREC texts and their summaries created by various systems. After classification, the agreement between the classifications of texts and their corresponding summaries is measured; the greater the agreement, the better the summary captures that which causes the full text to be classified as it is. In the Ad Hoc task, testers classified query-based summaries as Relevant or Not Relevant to the query. The agreement of texts and summaries classified in each category reflects the quality of the summary. Space constraints prohibit full discussion of the results; some interesting findings are that, for newspaper texts, all extraction systems performed equally well (and no better than the lead method) for generic summarization, and that IR methods produced the best query-based extracts. Still, despite the fact that all the systems performed extracts only, thereby simplifying much of the scoring process to IR-like recall and precision measures against human extracts, the wealth of material and the variations of analysis contained in Firmin and Chrzanowski (1999) underscore how little is still understood about summarization evaluation. This conclusion is strengthened in a fine paper by Donaway, Drummey, and Mather (2000) who show how summaries receive different scores with different measures, or when compared to different (but presumably equivalent) ideal summaries created by humans.

Recognizing these problems, Jing et al. (1998) compare several evaluation methods, intrinsic and extrinsic, on the same extracts. With regard to inter-human agreement, they find fairly high consistency in the news genre, as long as the summary (extract) length is fixed as relatively short (there is some evidence that other genres will deliver less consistency (Marcu 1997)). With regard to summary length, they find great variation. Comparing three systems, and comparing five humans, they show that the humans' ratings of systems, and the perceived ideal summary length, fluctuate as summaries become longer.

32.5.2 Two basic measures

Much of the complexity of summarization evaluation arises from the fact that it is difficult to specify what one really needs to measure, and why, without a clear formulation of what precisely the summary is trying to capture. We outline some general considerations here.

In general, to be a summary, the summary must obey two[1] requirements:

- it must be shorter than the original input text;
- it must contain the important information of the original (where importance is defined by the user), and not other, totally new, information.

One can define two measures to capture the extent to which a summary *S* conforms to these requirements with regard to a text *T*:

(32.4) **Compression Ratio**: *CR* = (*length S*) / (*length T*)
(32.5) **Retention Ratio**: *RR* = (*info in S*) / (*info in T*)

However we choose to measure the length and the information content, we can say that a good summary is one in which *CR* is small (tending to zero) while *RR* is large (tending to unity). We can characterize summarization systems by plotting the ratios of the summaries produced under varying conditions. For example, Fig. 32.2(*a*) shows a fairly normal growth curve: as the summary gets longer (grows along the *x* axis), it includes more information (grows also along the *y* axis), until it equals the original. Fig. 32.2(*b*) shows a more desirable situation: at some special point, the addition of just a little more text to the summary adds a disproportionately large amount of information. Fig. 32.2(*c*) shows another: quite early, most of the important material is included in the summary; as the length grows, the added material is less interesting. In both the latter cases, summarization is useful.

Measuring length. Measuring length is straightforward; one can count the number of words, letters, sentences, etc. For a given genre and register, there is a fairly good correlation among these metrics, in general.

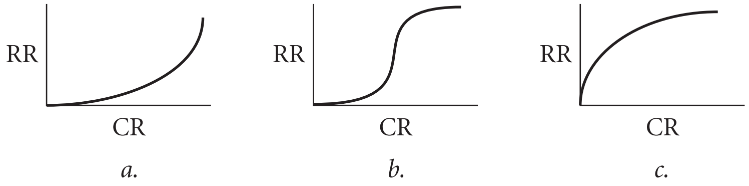

Fig. 32.2 Compression Ratio (*CR*) vs. Retention Ratio (*RR*)

[1] Ideally, it should also be a coherent, readable text, though a list of keywords or text fragments can constitute a degenerate summary. Readability is measured in several standard ways, for purposes of language learning, machine translation, and other NLP applications.

Measuring information content. Ideally, one wants to measure not information content, but *interesting* information content only. Although it is very difficult to define what constitutes *interestingness*, one can approximate measures of information content in several ways. We describe four here.

The *Expert Game*. Ask experts to underline and extract the most interesting or informative fragments of the text. Measure recall and precision of the system's summary against the human's extract, as outlined in section 32.3.1.

The *Classification Game*. Two variants of this extrinsic measure were implemented in the TIPSTER-SUMMAC evaluation (Firmin Hand and Sundheim 1998; Firmin and Chrzanowski 1999); see section 32.5.1.

The *Shannon Game*. In information theory (Shannon 1951), the amount of information contained in a message is measured by $-p \log p$, where p is, roughly speaking, the probability of the reader guessing the message (or each piece thereof, individually). To measure the information content of a summary S relative to that of its corresponding text T, assemble three sets of testers. Each tester must create T, guessing letter by letter. The first set reads T before starting, the second set reads S before starting, and the third set reads nothing, For each set, record the number of wrong guesses g_{wrong} and total guesses g_{total}, and compute the ratio $R = g_{wrong}/g_{total}$. The quality of S can be computed by comparing the three ratios. R_{none} quantifies how much a tester could guess from world knowledge (and should hence not be attributed to the summary), while R_T quantifies how much a tester still has to guess, even with 'perfect' prior knowledge. The closer R_S is to R_T, the better the summary.[2]

The *Question Game*. This measure approximates the information content of S by determining how well it allows readers to answer questions drawn up about T. Before starting, one or more people create a set of questions based on what they consider the principal content (author's view or query-based) of T. Then the testers answer these questions three times in succession: first without having read either S or T, second after having read S, and third after having read T. After each round, the number of questions answered correctly is tallied. The quality of S can be computed by comparing the three tallies, as above. The closer the testers' score for S is to their score for T, the better the summary. The TIPSTER-SUMMAC summarization evaluation (Firmin Hand and Sundheim 1998) contained a tryout of the Question Game.

Further Reading and Relevant Resources

Mani (2001) provides a thorough overview of the field, and Mani and Maybury (1999) provide a most useful collection of twenty-six papers about summarization, includ-

[2] In 1997, the author performed a small experiment using the Shannon Game, finding an order of magnitude difference between the three contrast sets.

ing many of the most influential. Recent workshop proceedings are Hovy and Radev (1998); Hahn et al. (2000); Goldstein and Lin (2001); DUC (2001) and DUC (2002). Useful URLs are at http://www.cs.columbia.edu/~radev/summarization/.

Acknowledgements

Thanks to Chin-Yew Lin, Daniel Marcu, Hao Liu, Mike Junk, Louke van Wensveen, Thérèse Firmin Hand, Sara Shelton, and Beth Sundheim.

References

Aone, C., M. E. Okurowski, J. Gorlinsky, and B. Larsen. 1999. 'A scalable summarization system using robust NLP'. In Mani and Maybury (1999), 71–80.

Banko, M., V. O. Mittal, and M. J. Witbrock. 2000. 'Headline generation based on statistical translation.' *Proceedings of the 38th Annual Conference of the Association for Computational Linguistics* (*ACL 2000*) (Hong Kong), 318–25.

Barzilay, R. and M. Elhadad. 1999. 'Using lexical chains for text summarization'. In Mani and Maybury (1999), 111–21.

——K. R. McKeown, and M. Elhadad 1999. 'Information fusion in the context of multi-document summarization'. *Proceedings of the 37th Conference of the Association of Computational Linguistics* (*ACL '99*) (College Park, Md.), 550–7.

Baxendale, P. B. 1958. 'Machine-made index for technical literature: an experiment'. *IBM Journal*, 3, 54–361.

Brandow, R., K. Mitze, and L. Rau. 1995. 'Automatic condensation of electronic publishing publications by sentence selection'. *Information Processing and Management* 31(5), 675–85. Also in Mani and Maybury (1999), 293–304.

Buckley, C. and C. Cardie. 1997. 'Using EMPIRE and SMART for high-precision IR and summarization'. *Proceedings of the TIPSTER Text Phase III 12-Math Workshop*. San Diego, USA.

Carbonell, J., Y. Geng, and J. Goldstein. 1997. 'Automated query-relevant summarization and diversity-based reranking'. *Proceedings of the IJCAI-97 Workshop on AI in Digital Libraries*. San Mateo, Calif.: Morgan Kaufmann, 12–19.

DeJong, G. J. 1978. *Fast skimming of news stories: the FRUMP system*'. Ph.D. thesis, Yale University.

Donaway, R. L., K. W. Drummey, and L. A. Mather. 2000. 'A comparison of rankings produced by summarization evaluation measures.' *Proceedings of the NAACL Workshop on Text Summarization* (Seattle), 69–78.

Donlan, D. 1980. 'Locating main ideas in history textbooks'. *Journal of Reading*, 24, 135–40.

DUC. 2001. *Proceedings of the Document Understanding Conference (DUC) Workshop on Multi-Document Summarization Evaluation*, at the SIGIR-01 Conference. New Orleans, USA. http://www.itl.nist.gov/iad/894.02/projects/duc/index.html.

——2002. *Proceedings of the Document Understanding Conference (DUC) Workshop on Multi-Document Summarization Evaluation*, at the ACL-02 Conference. Philadelphia, USA (forthcoming).

Edmundson, H. P. 1969. 'New methods in automatic extraction'. *Journal of the ACM*, 16(2), 264–85. Also in Mani and Maybury (1999), 23–42.

Firmin, T. and M. J. Chrzanowski. 1999. 'An evaluation of text summarization systems.' In Mani and Maybury (1999), 325–35.

Firmin Hand, T., and B. Sundheim. 1998. 'TIPSTER-SUMMAC summarization evaluation'. *Proceedings of the TIPSTER Text Phase III Workshop*, Washington, DC.

Fukumoto, F. and Y. Suzuki. 2000. 'Extracting key paragraph based on topic and event detection: towards multi-document summarization.' *Proceedings of the NAACL Workshop on Text Summarization* (Seattle), 31–9.

Fukushima, T., T. Ehara, and K. Shirai. 1999. 'Partitioning long sentences for text summarization'. *Journal of the Society of Natural Language Processing of Japan*, 6(6), 131–47 (in Japanese).

Goldstein, J. and C.-Y. Lin (eds.). 2001. *Proceedings of the NAACL Workshop on Text Summarization*. Pittsburgh, USA.

——M. Kantrowitz, V. Mittal, and J. Carbonell. 1999. 'Summarizing text documents: sentence selection and evaluation metrics'. *Proceedings of the 22nd International ACM Conference on Research and Development in Information Retrieval (SIGIR '99)* (Berkeley), 121–8.

——V. Mittal, J. Carbonell, and M. Kantrowitz. 2000. 'Multi-document summarization by sentence extraction'. *Proceedings of the NAACL Workshop on Text Summarization* (Seattle), 40–8.

Hahn, U. and U. Reimer. 1999. 'Knowledge-based text summarization: salience and generalization operators for knowledge base abstraction'. In Mani and Maybury (1999), 215–32.

——C.-Y. Lin, I. Mani, and D. Radev (eds). 2000. *Proceedings of the NAACL Workshop on Text Summarization* (Seattle).

Hirst, G., C. DiMarco, E. H. Hovy, and K. Parsons. 1997. 'Authoring and generating health-education documents that are tailored to the needs of the individual patient'. *Proceedings of the 6th International Conference on User Modelling (UM '97)* (Sardinia). http://um.org.

Hovy, E. H. and C.-Y. Lin. 1999. 'Automating text summarization in SUMMARIST'. In Mani and Maybury (1999), 81–97.

——and D. Radev (eds.), 1998. *Proceedings of the AAAI Spring Symposium on Intelligent Text Summarization*. Stanford, Calif.: AAAI Press.

Jing, H. and K. R. McKeown. 1999. 'The decomposition of human-written summary sentences'. *Proceedings of the 22nd International ACM Conference on Research and Development in Information Retrieval (SIGIR-99)* (Berkeley), 129–36.

——R. Barzilay, K. McKeown, and M. Elhadad. 1998. 'Summarization evaluation methods: experiments and results'. In Hovy and Radev (1998), 60–8.

Knight, K. and D. Marcu. 2000. 'Statistics-based summarization—step one: sentence compression'. *Proceedings of the Conference of the American Association for Artificial Intelligence (AAAI)* (Austin, Tex.), 703–10.

Kubota Ando, R., B. K. Boguraev, R. J. Byrd, and M. S. Neff. 2000. 'Multi-document summarization by visualizing topical content'. *Proceedings of the NAACL Workshop on Text Summarization* (Seattle), 79–88.

Kupiec, J., J. Pedersen, and F. Chen. 1995. 'A trainable document summarizer'. *Proceedings of the 18th Annual International ACM Conference on Research and Development in Information Retrieval (SIGIR)* (Seattle), 68–73. Also in Mani and Maybury (1999), 55–60.

Lehnert, W. G. 1981. 'Plot units and narrative summarization'. *Cognitive Science*, 5(4). See also 'Plot units: a narrative summarization strategy', in Mani and Maybury (1999), 177–214.

Lin, C.-Y. 1999. 'Training a selection function for extraction'. *Proceedings of the 8th International Conference on Information and Knowledge Management* (*CIKM*) (Kansas City), 1–8.

——and E. H. Hovy. 1997. 'Identifying topics by position'. *Proceedings of the Applied Natural Language Processing Conference* (*ANLP '97*) (Washington), 283–90.

Luhn, H. P. 1959. 'The automatic creation of literature abstracts'. *IBM Journal of Research and Development*, 159–65. Also in Mani and Maybury (1999), 15–22.

Mani, I. 2001. *Automatic Summarization*. Amsterdam: John Benjamins.

——and E. Bloedorn. 1997. 'Multi-document summarization by graph search and matching'. *Proceedings of AAAI-97* (Providence), 622–8.

——B. Gates, and E. Bloedorn. 1999. 'Improving summaries by revising them'. *Proceedings of the 37th Conference of the Association of Computational Linguistics* (*ACL '99*) (College Park, Md.), 558–65.

——and M. Maybury (eds.). 1999. *Advances in Automatic Text Summarization*. Cambridge, Mass.: MIT Press.

Marcu, D. 1997. *The rhetorical parsing, summarization, and generation of natural language texts*. Ph.D. thesis, University of Toronto.

——1998. 'Improving summarization through rhetorical parsing tuning'. *Proceedings of the COLING-ACL Workshop on Very Large Corpora* (Montreal), 10–16.

——1999. 'The automatic construction of large-scale corpora for summarization research'. *Proceedings of the 22nd International ACM Conference on Research and Development in Information Retrieval* (*SIGIR '99*) (Berkeley), 137–44.

——and L. Gerber. 2001. 'An inquiry into the nature of multidocument abstracts, extracts, and their evaluation'. *Proceedings of the Workshop on Text Summarization at the 2nd Conference of the North American Association of Computational Linguistics* (Pittsburgh), 1–8.

Miike, S., E. Itoh, K. Ono, and K. Sumita. 1994. 'A full-text retrieval system with dynamic abstract generation function'. *Proceedings of the 17th Annual International ACM Conference on Research and Development in Information Retrieval* (*SIGIR*), 152–61.

Mitra, M., A. Singhal, and C. Buckley. 1997. 'Automatic text summarization by paragraph extraction'. *Proceedings of the Workshop on Intelligent Scalable Summarization at the ACL/EACL Conference* (Madrid), 39–46.

Morris, A. G. Kasper, and D. Adams. 1992. 'The effects and limitations of automatic text condensing on reading comprehension performance'. *Information Systems Research*, 3(1), 17–35.

Ono, K., K. Sumita, and S. Miike. 1994. 'Abstract generation based on rhetorical structure extraction'. *Proceedings of the 15th International Conference on Computational Linguistics* (*COLING '94*) (Kyoto), i, 344–8.

Paice, C. D. 1990. 'Constructing literature abstracts by computer: techniques and prospects'. *Information Processing and Management*, 26(1), 171–86.

Quinlan, J. R. 1986. 'Induction of decision trees'. *Machine Learning*, 81–106.

Radev, D. R. 1999. *Generating natural language summaries from multiple on-line source: language reuse and regeneration*. Ph.D. thesis, Columbia University.

Rau, L. S. and P. S. Jacobs. 1991. 'Creating segmented databases from free text for text retrieval'. *Proceedings of the 14th Annual ACM Conference on Research and Development in Information Retrieval* (*SIGIR*) (New York), 337–46.

Salton, G., A. Singhal, M. Mitra, and C. Buckley. 1997. 'Automatic text structuring and summarization'. *Information Processing and Management*, 33(2), 193–208. Also in Mani and Maybury (1999), 341–56.

Shannon, C. 1951. 'Prediction and entropy of printed English'. *Bell System Technical Journal*, Jan., 50–64.

Spärck Jones, K. 1999. 'Automatic summarizing: factors and directions'. In Mani and Maybury (1999), 1–13.

——and J. R. Galliers. 1996. *Evaluating Natural Language Processing Systems: An Analysis and Review*. New York: Springer.

Strzalkowski, T., G. Stein, J. Wang, and B. Wise. 1999. 'A robust practical text summarizer'. In Mani and Maybury (1999), 137–54.

Teufel, S. and M. Moens. 1997. 'Sentence extraction as a classification task'. *Proceedings of the ACL Workshop on Intelligent Text Summarization* (Madrid), 58–65.

————1999. 'Argumentative classification of extracted sentences as a first step toward flexible abstracting'. In Mani and Maybury (1999), 155–75.

Witbrock, M., and V. Mittal. 1999. 'Ultra-summarization: a statistical approach to generating highly condensed non-extractive summaries'. *Proceedings of the 22nd ACM Conference on Research and Development in Information Retrieval* (*SIGIR*) (Berkeley), 315–16