n-grams of Seeds: A Hybrid System for Corpus-Based Text Summarization

René Schneider

DaimlerChrysler AG
Research and Technology
Dialogue Systems (RIC/AD)
rene.schneider@daimlerchrysler.com

Abstract

This paper presents a hybrid system for automatic text summarization which combines statistical and knowledge-based methods. In particular, it demonstrates how two corpus-based learning and indexing algorithms, namely an n-gram and a seed-oriented approach, may be combined to bring out the best of both approaches. This system selects sentences from an input text to constract a highly compressed, generic, and informative summary. The hybrid algorithm described here was developed and tested with a corpus of movie reviews collected from several on-line data bases.

1. Introduction

In recent years, text summarization has become a field of growing interest within the area of language engineering with a large variety of applications. For many systems it is no longer a “nice to have” but rather an indispensable “must.” Besides, it is one of the fields in natural language processing where many methodologies come together and statistical, rule-based, and symbolic strategies claim their rights. In this paper we will show how these different strategies may be combined into a hybrid summarization engine.

1.1. Scenarios

In the near future every surfer in the world wide web will expect a search engine not only to present the results in an appropriate ranking but also to offer the option of at least basic summaries.

This requirement has to be fulfilled in most of the information systems, especially multi-modal information systems, where the text or text summaries that are displayed on a screen force the user to read aloud longer text passages. This read-off talk produces new input for the speech recognizer or barge-in for the information system. To prevent this, it is better to output small text passages or summaries via the synthesis module.

This mode of transaction will also play a more and more dominant role in the mobile environment, i.e. in cars, where every interaction between the driver and the system is done via a dialogue system and a text-to-speech system. Here, language technology has to deliver solutions to the driver distraction dilemma, i.e. to limit interaction and superfluous information by keeping texts short and concise. For text summarization this means that the process of summarizing is characterized by a very high compression rate which in several cases may reduce the summary to only one or two sentences.

1.2. Definitions

Following the definitions given in several standard books (e.g. Mani, 2001), the actual system described in this paper produces extracts (as opposed to abstracts) from *sentences* in German movie reviews. The sentence fragments with the highest significance values are extracted to form a summary with a high compression rate, for the reasons given at the end of Section 2.1. Since there are no criteria for user adaptation so far, extracts are generically oriented (as opposed to being focussed) with each summary being informative (as opposed to being indicative or evaluative), which tries to reflect the essence of the original text as objectively (as opposed to critically) as possible.

1.3. The Corpus

The actual work was not started until after a corpus of plot descriptions[1] from several movie-review data bases online-available was built. Considering *Netiquette* (e.g. web-robot identification and polling rhythm), raw text corpora of representative size for scientific use may nowadays be generated in about one or two days. In our case, 4,792 movie reviews were downloaded and stored from several www servers. For each type of HTML-document, a filter was implemented to strip away any non-relevant and superfluous tags and signs. Using these raw texts, two learning and weighting methods were applied to construct a ranked list of sentences.

[1] The example extracts in this paper were generated from the following original movie review: “Der elfjährige Billy Elliot (Jamie Bell) lebt mit seinem Vater (Gary Lewis), seinem älteren Bruder (Jamie Draven) und der Großmutter (Jean Haywood) in einem kleinen Ort in Nordengland zur Zeit des großen Streikes der 80er Jahre. Nachmittags muss sich die Boxklasse die Turnhalle mit der Ballettklasse teilen. Dabei wird Billy von den weichen Bewegungen der Tänzerinnen in den Bann gezogen. Heimlich tauscht er seine Boxhandschuhe gegen Ballettschläppchen ein. Er wird von der energischen Tanzlehrerin Mrs. Wilkinson (Julie Walters) auch in die Gruppe aufgenommen, obwohl ihm das Geld für den Unterricht fehlt. Von Billys Talent überzeugt, will sie ihn für ein Vortanzen an der Akademie in London vorbereiten. Doch sein Vater ist – als er von Elliots Passion erfährt - gar nicht begeistert. Viele Tanzfilme verherrlichen die darstellende Kunst und übertreiben gerne mit groß angelegten Choreographien. Stephen Daldry erzählt die Geschichte eines Jungen, der seiner Leidenschaft, dem Tanzen, trotz enormer Vorurteile und Widerstände, nachgehen will. In Jamie Bell hat er eine ideale Besetzung dafür gefunden, denn der Junge besitzt die Fähigkeit, trotz seiner klassischen Ausbildung, wie ein ganz normaler Junge von der Straße zu tanzen – eben nur besser. "Billy Elliot" ist weder kitschig, noch unrealistisch geraten und ist deshalb ein sehr gelungener Film.”

2. Two Learning and Weighting Methods

For the system presented here, we developed two different corpus-based learning algorithms for generating text specific features based on a representative training corpus, as described in Figure 2.1:

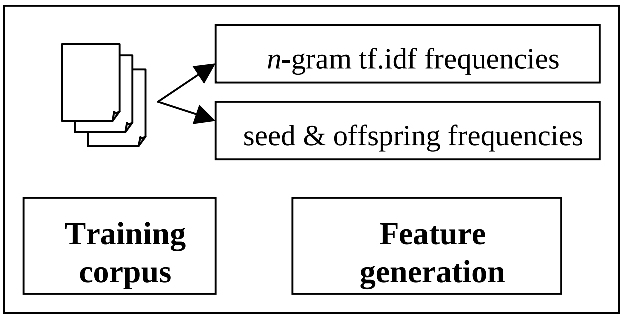

Figure 2.1: Learning from corpora

- The first algorithm is based on an *n*-gram approach that calculates for every 4-gram a specific value based on its tf.idf (text frequency divided by incremented document frequency) in the training corpus,
- The second algorithm extracts concordances which match a very small number of strings that were determined to be significant members of domain-specific sentences in the corpus. These strings (approximately three dozen) represent seed words. The words in this seed list are matched with the whole training corpus. When a match is made between a seed word and a word in the corpus, the four preceeding and the four succeeding words are also extracted for further exploitation.

As a function of the *n*-gram and seed based frequencies, a statistical value is assigned to each sentence of the text in order to enable a limited number of sentence candidates to be selected for the summarization engine.

2.1. The *n*-gram Based Approach

For every text, all word forms of the training texts are transformed into topic specific lists of 4-grams together with their frequencies. An *n*-gram is a sequence of 4 contiguous characters including blanks but excluding punctuation marks, which have already been stripped. Previous works (Bayer et al., 1997) have shown that the 4-gram approach produces better results than 3-grams, where fewer features are generated. On the other hand, the memory requirements and complexity of 5-grams are generally unacceptable.

Since the summarization engine works with sentences, we have to assign a value to each sentence to estimate its significance within a given text. For the *n*-gram approach, we compute the arithmetic mean from the tf.idf (text frequency / inverse document frequency) of all 4-grams of a sentence. As stated in (Manning, Schütze, 1999), tf.idf has shown in many cases to be a tried and tested heuristic for characterizing a string *i* (in this case a 4-gram) in a document *j* by its term occurrence weighting tf_{ij}, its document frequency weighting df_i and (if desired) its normalization. For our investigation we tested several normalization procedures and finally decided to use the logarithmic occurrence count weighting, since it produced the best results. The weight is calculated as:

$$\mathrm{weight}(i,j) = (1 + \log(\mathrm{tf}_{i,j}))\ \log N/\mathrm{df}_i$$

where N is the total number of documents in the corpus.

Generally speaking, this method assigns high values (indicating a high degree-of-interest) to sentences that contain *n*-grams with a low corpus frequency. Table 2.1 shows a ranked list of the three best weighted sentences from our example movie review.

Average 4-gram weight	*Sentence*
3..67	Der elfjährige Billy Elliot (Jamie Bell) lebt mit seinem Vater (Gary Lewis), seinem älteren Bruder (Jamie Draven) und der Großmutter (Jean Haywood) in einem kleinen Ort in Nordengland zur Zeit des großen Streikes der 80er Jahre.
3.63	Nachmittags muss sich die Boxklasse die Turnhalle mit der Ballettklasse teilen.
3.62	"Billy Elliot" ist weder kitschig, noch unrealistisch geraten und ist deshalb ein sehr gelungener Film.

Table 2.1: Top three sentences (with scores) according to *n*-gram approach

2.2. The Seed Based Approach

In information extraction (Riloff, Jones, 1999) seed words, i.e. a number of carefully preselected words, are used to learn extraction patterns from raw training corpora. Text summarization (and especially extract generation) can be seen as a special case of information extraction. Similar to the work of Riloff and Jones, we exploit the extraction patterns to find more words of interest and collect their frequencies in corresponding lists[2].

In our investigation the seed words for the movie domain consist of the approximately three dozen substrings shown in Figure 2.2. As can be easily seen, the majority belongs to words describing the movie genre:

werk, komoedi, film, geschicht, litera, drama, klassi, movie, epos, geschicht, maerch, debut, thriller, psycho, roman, satir, dokumenta, action, zeichentrick, trick, anima, histori, krimi, tragik, science, horror, fantas, abenteuer, musical, tanz

Figure 2.2: seeds

In the first processing step, whenever one of these strings (see Table 2.2) appears, we cut out a text window or extraction pattern, with the four preceeding and four succeeding words, regardless of any punctuation. If identity of a seed in a word appears (that we named *extended seeds*) the frequency value of this word is incremented in the corresponding list.

[2] Since text summarization often deals with the preferences of a user, it should be stressed that seeds indicating the users interests may be a good starting point for user-focussed learning procedures.

Content words or autosemantica are determined with a shallow suffix analysis based on a small suffix lexicon. All function words are excluded from further consideration. Any remaining *words-of-interest* are determined as a function of their distance to the initial seed. The frequencies of all these words are incremented and stored in eight additional frequency lists corresponding to their location in the concordance to the left (L4-L1) or right (R1-R4) of the extended seed.

predecessor *L4-L1*	*extended seed*	*successors* *R1-R4*
vincenzo natalie ein fulminantes	erstlings**werk**	sein intelligenter genrefilm zwischen
einer der innovativsten	**zeichentrick**filme	die je realisiert wurden
im zeitalter des internets	erst**klassi**g	besetzt mit tom hanks
charles aznavour zu einem	**klassi**ker	*End_of_text*

Table 2.2: Pattern exploitation

The second processing step examines the L1 predecessor of each extended seed and then collects those word pairs or collocations whose first elements are these L1 words. The second elements of these pairs are called *offsprings*. Since it has been shown that adjective/noun collocations can greatly benefit content extraction, we look for such pairs among the set of L1/offspring collocations.

Table 2.3 shows some successors or offsprings for the seed preceeding word from the first example in Table 2.2. Once again, string matching is based on stems and not on full words.

L1	*offspring*
fulminante	wirkung
fulminanter	sieg
fulminantes	regiedebüt

Table 2.3: "Planting offsprings"

These two steps just described produce ten different frequency lists on the "seed" side of our feature extraction: one with the incremented frequency of the extended seeds, one for the offsprings, and one a piece for each of the frequencies of the four predecessors, L4-L1, and for each of the four successors, R1-R4.

The "weight" of each word in a given sentence is computed by adding up its frequencies in each of the ten lists where it occurs. These word weights are then summed over all the words in the sentence and then divided by the total number of occurrences in all ten tables. This final value is the "seed weight" of the sentence. Table 2.4 shows the calculation of this sentence weight for a typical sentence.

For example the ninth word in the sentence, "Höhen," occured with a count of 1 as offspring, 2 in the R2 position, 3 in the R3 position and 4 in the R4 position. The sum of the word weights is 922, the total number of occurrences all words in all ten tables is 21. Note that this last number is *not* the number of words in a sentence, which is 13.

sentence	*word weight*	*list count*
Das	0	0
eingespielte	0	0
Darsteller-	$41_{off}+49_{L4}+46_{L3}+44_{L2}+51_{R1}+52_{R2}+55_{R3}+57_{R4}$	8
Ensemble	2_{off}	1
durchleidet	2_{off}	1
im	0	0
Stakkato	0	0
die	0	0
Höhen	$1_{off}+2_{R2}+3_{R3}+4_{R4}$	4
und	0	0
Tiefen	$3_{off}+4_{R2}+5_{R3}+6_{R4}$	4
des	0	0
Lebens.	$159_{off}+165_{L1}+171_{Ll2}$	3
sum	922	21
seed weight:	922/21 = 43.9	

Table 2.4: Seed-based weight calculation

Table 2.5 shows the top three sentences and their seed weights for our example text:

sentence seed weight	*Sentence*
57.26	"Billy Elliot" ist weder kitschig, noch unrealistisch geraten und ist deshalb ein sehr gelungener Film.
34.15	Stephen Daldry erzählt die Geschichte eines Jungen, der seiner Leidenschaft, dem Tanzen, trotz enormer Vorurteile und Widerstände, nachgehen will.
8.97	In Jamie Bell hat er eine ideale Besetzung dafür gefunden, denn der Junge besitzt die Fähigkeit, trotz seiner klassischen Ausbildung, wie ein ganz normaler Junge von der Straße zu tanzen – eben nur besser.

Table 2.5: Top three sentences (with scores) according to seed approach

2.3. Comparison of Both Approaches

This section compares these two methods, points out their relative advantages and disadvantages and shows how they can enhance each other: The *n*-gram approach is totally data-driven and both domain and language independent. It has proved in the past to apply to any alphabetically written languages. With these *n*-gram weights the summarization engine can determine which sentences are specific and distinctive to the input text.

The seed based approach is expectancy-driven. Just as the summarization results for the *n*-gram approach depend on the corpora used in learning, so the results of the seed based approach depend on what seeds are preselected. However, unlike the *n*-gram approach which is fully automatic once the corpora have been selected, in the seed based approach a manual selection of seeds for the domain and language of the corpora must first be made.

As opposed to *n*-grams, seed-weighted sentences characterize a text in relation to other texts within a given domain or genre and emphasize text similarities rather than differences.

In other words, *n*-grams tell us something about the uniqueness of a text, whereas seeds give hints about what a text has in common with other texts of the same domain. Effectively, *n*-grams and seeds represent two sides of the same coin, since the interest in generic text summarization generally lies in knowing something about the differences and similarities among related documents. This is especially true for movie reviews since they try to work out the characteristics of the movie itself and set it into relation to previous movies of the same director, actors and so forth.

The only remaining question is how to merge these two strategies. In other words, how can we choose the best sentences from both methods? The following section will show how these different approaches may be combined into a unified hybrid algorithm.

3. A Hybrid Summarizer

3.1. Overview of the System

This section presents the overall architecture of the system. The major steps are shown in Figure 3.1. First the text is segmented into individual sentences and these are then normalized. Next each sentence is evaluated with each of the two methods described above and given a relative-importance index. The next step is the heart of the hybridization method: From the last step we have two ranked lists of the sentences of the input text: one based on the seed method and the other based on the *n*-gram method. In this step the two lists are merged into a single ranked list based on a hybrid criterion as described in Section 3.3.

Afterwards the appropriate number of sentences for the summary are selected and reordered. Finally smoothing techniques, such as anaphora resolution, are applied.

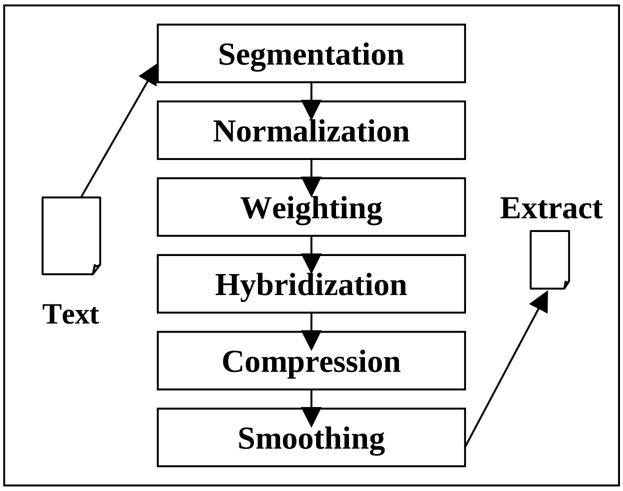

Figure 3.1: Overview of the system

3.2. Initial Steps

Before any processing is begun, the number of sentences *m* considered to be appropriate for the extract is computed as follows: This number is 20% of the total number of sentences, but not less than 2 nor more than 6 sentences. This high compression rate is suitable for all transmissions in a mobile and possibly distracting and noisy environment.

Initially the input text is segmented, normalized and indexed as described above. The normalization ensures identical feature extraction to that obtained during learning. As indicated above, the n-gram ranking is derived from the mean tf.idf weights and the seed ranking is based on the mean frequency of word occurrences.

3.3. Hybridization

The next and decisive step consists of choosing those sentences which will be part of the extract. We exclude certain sentences based on length and well-formedness. For the sake of illustration consider the set of all sentences in the input text to be *T* and the set of those sentences selected for the extract to be *E* (see Figure 3.2).

We now select the *m* highest ranked sentences from the seed approach and call this set *S*, and the *m* highest ranked sentences from the *n*-gram approach which we call *N*. The first sentences to be put into set *E* are the intersection of *N* and *S*. Then we fill in the remaining sentences in *E* by alternately selecting the highest ranked sentence remaining in *S* and then in *N*.

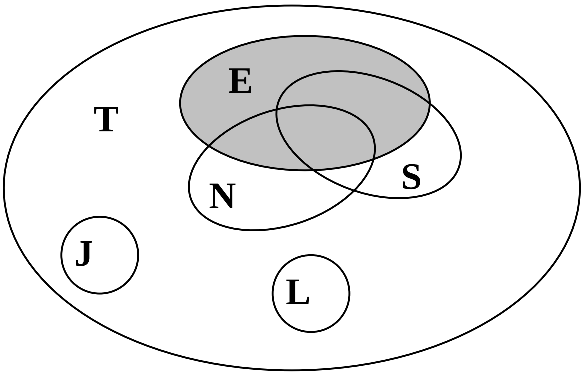

Figure 3.2: Set-theoretical view of hybridization

The motivation for the pre-exclusion of certain sentences mentioned above are as follows:

1. We designate ill-formed sentences as “junk” and the set as *J*. For the time being those sentences which contain no function words are junk. This simple routine seems to be sufficient for our purposes since the main goal is to exclude ill-formed sentences from becoming candidates for extraction. A frequent example of this is a badly tagged and therefore unstripped HTML tag. Such sentences typically have a very high *n*-gram score, which is why they must be excluded.
2. We also exclude very long and very short sentences. We designate this set as *L*. Here a long sentence is more than 40 words. Such sentences are not helpful or needed in highly compressed and orally transmitted extracts. Also, a very long length sometimes means that a sentence divider is missing. Short sentences are defined 3 or less words. They often contain anaphora and thus have no meaning without reference to prior sentences.

Also, such short sentences normally do not contain any useful information for an extract.

The two methods taken by themselves, seeds and *n*-grams, produce scores which cannot be related to each other. It therefore seems reasonable to first choose those sentences for the extract which scored high with both methods. This is the motivation for the intersection of *N* and *S* as described above.

Of the two methods the seed approach seems to always yield slightly better results than the *n*-gram method. On the other hand, the extract should not exclude good *n*-gram sentences out of principle. For this reason the remaining sentences which do not belong to both *S* and *N* are chosen alternately from *S* and then from *N*, but starting with *S*. Since extracts contain at most six sentences and typically the first or first two sentences belong to the intersection, we have four to five sentences to fill in. Figure 3.3 shows the result of the hybridisation step for our example text[3]:

> Der elfjährige Billy Elliot (Jamie Bell) lebt mit seinem Vater (Gary Lewis), seinem älteren Bruder (Jamie Draven) und der Großmutter (Jean Haywood) in einem kleinen Ort in Nordengland zur Zeit des großen Streikes der 80er Jahre.
> Stephen Daldry erzählt die Geschichte eines Jungen, der seiner Leidenschaft, dem Tanzen, trotz enormer Vorurteile und Widerstände, nachgehen will.
> "Billy Elliot" ist weder kitschig, noch unrealistisch geraten und ist deshalb ein sehr gelungener Film.

Figure 3.3: Resulting extract using hybrid algorithm

3.4. Smoothing

After the sentences for the extract have been chosen, they are output in their order in the original input text. The final step before completing the extract is anaphora resolution, which is generally indispensable for text summarization. Currently anaphora resolution is limited to the first sentence of the extract. This resolution consists of inserting an additional sentence in front of this first sentence. This problem will be further investigated later.

4. Future Work

The work on the system is still ongoing and thus many improvements and tests must be made before the final prototype is finished. As mentioned above, anaphora resolution is a major problem. Another field of work is to establish better criteria for identifying junk sentences.

In the n-gram approach the normalization of the tf.idf weighting needs to be improved. The word weight in the seed approach (see Table 2.4 above) can be improved by weighting each term in the sum according to its distance from the seed.

Another interesting question is the automatic derivation of the seeds from training corpora. We have observed that the “corpus distribution”, i.e. the document *df* divided by the corpus frequency *cf*, of the vast majority of seeds is 1 or slightly less. This means they usually appear only once or twice in a document. Unfortunately this is also true for many other words, so this is only one criterion. Other criteria for seed detection have to be found. Nevertheless this corpus distribution can be used as an additional criterion for the quality of the manually selected seeds.

Finally we want to implement an evaluation routine. Nevertheless, evaluation in text summarization is a difficult matter, since different people have different opinions as to which sentences in a text are the most important. Informal tests within the department have confirmed this fact. To evaluate the system presented, we have started to implement a test routine: The system is trained on a large news corpus, along with abstracts written by the author of the text. These abstracts and the automatically derived extracts will be compared by human evaluation and also with a statistical method which will evaluate the similarity of the author generated abstract and the machine generated extract.

5. Conclusions

The work described in this paper is based on two corpus-based learning methods, n-gram and seed based, and two sentence-based weighting methods, namely the tf.idf and word-of-interest frequencies. The system is enhanced with several rule-based components to improve the sentence merger of the results from the two weighting approaches. The whole system requires a minimal amount of a priori linguistic knowledge: a carefully selected list of seeds, a list of function words as well as anapher, abbreviation, and suffix inventories for the language we are working with.

The work done so far has been focussed on how to construct a hybrid system from diverse methods to construct highly compressed summaries, which are required in multi-modal and distracting mobile environments. The results achieved through the combination of the two techniques are promising and will be evaluated and further refined.

6. References

Bayer, Th., H. Mogg-Schneider, I. Renz, H. Schäfer, 1997. Daimler Benz Research: System and Experiments Routing and Filtering. In *Proceedings of the 6th Text REtrieval Conference (TREC-97).*

Mani, I., M. Maybury, 1999. *Advances in Text Summarization*. MIT Press.

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

Manning, C., H. Schütze, 1999. *Foundations of Statistical Natural Language Processing*. MIT Press.

Riloff, E., R. Jones, 1999. Learning Dictionaries for Information Extraction by Multi-Level Bootstrapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99).*

[3] Resulting abstract using hybrid algorithm (translated): The eleven year old Billy Elliot (Jamie Bell) lives with his father (Gary Lewis), his older brother (Jamie Draven) and his grandmother (Jean Haywood) in a little town in North England during the big strikes in the 80’s. Stephen Daldry tells the story of a boy who wants to persue his passion for dancing in spite of enormous prejudices and resistance. "Billy Elliot" is neither corny nor unrealistic and for this reason a very successful film.