# Headline extraction based on a combination of uni- and multidocument summarization techniques

Wessel Kraaij, Martijn Spitters, Anette Hulth

TNO-TPD
P.O. Box 155, 2600 AD Delft
The Netherlands

**Abstract**

The TNO system for multi-document summarisation is based on an extraction approach. For headline generation, we chose to extend our system to extract the most informative topical noun phrase. The cluster topic is defined as the most frequent term occurring in the most salient document sentences. The core of our system is a probabilistic model, which estimates the log-odds of salience based on a number of features including sentence position, sentence length, cue phrases and a language model based content score. The parameters of the model were estimated on annotated training data.

## 1 Introduction

This paper describes the design and development of a system for multi-document summarisation based on probabilistic methods. Document summarisation is a rather new research area at TNO TPD. For DUC 2001 we built a system from scratch based on statistical language models that we had successfully applied in different IR tasks [Kraaij et al., 2000, Hiemstra et al., 2001, Spitters and Kraaij, 2001]. Our aim for DUC 2002 was to extend the system with abstraction and/or compaction functionality. However we had to refrain from this ambition given the short preparation time. Instead we focused on an assessment of the relative contribution of the different features of last year's system. Secondly, since a considerable amount of headlines are noun phrases, we designed a method to extract the most topical informative noun phrase from a cluster.

## 2 A revised Design of a probabilistic sentence extraction system

Previous research on extraction based approaches has shown the effectiveness of several non-content or *surface* features for the determination of a sentence's salience: position in text, the usage of cue phrases, sentence length etc. [Edmundson, 1969]. Starting point for DUC2002 was our DUC2001 sentence extraction algorithm, which is a hybrid system based on unigram language models for scoring sentences in relation to the cluster context an a naive Bayes classifier based on non-content features like sentence length[Kraaij et al., 2001].

The combined model determines a salience value for each extracted sentence. This ranked list of sentences forms the input for the summary generation module. This module tries to generate a summary which consists of the most salient sentences, with minimal redundancy and maximal coherence/readability. Since we do not have a deeper meaning representation of the extracted sentences, we can only use very shallow techniques to meet the latter constraints. The maximum marginal relevance (MMR) criterion [Carbonell and Goldstein, 1998] was adapted for our system in order to minimise redundancy of the produced summaries.

## 2.1 Unigram language model for content based salience

Since ordering sentences with respect to a cluster model is conceptually not far from ordering stories with respect to a certain topic model (in a TDT context), we apply similar techniques. The only difference is that we use a mixture model, consisting of the cluster model and a model estimated on the document. The mixture is intended to model the intuition that candidate summary sentences should either be highly relevant for the document or for the cluster. For single document summaries, the content model just consists of the document model.

The salience of a sentence $S = T_1, T_2, ..., T_n$ w.r.t. a document $D_k$ (the within document salience) can be modeled as the probability that the sentence is generated by a unigram model corresponding to that document, assuming independence between the individual terms.

$$P(T_1, T_2, ..., T_n|D_k) = \prod_{i=1}^{n} P(T_i|D_k) \tag{1}$$

The salience of a sentence with respect to a cluster of documents (the within cluster salience) can be modeled along the same lines (i.e. replace document $D_k$ by cluster $C_j$ in formula (1)).

Our hypothesis about multi-document summarisation is that a "good" sentence is both salient for a document and for the corresponding cluster. $P(S|D_k)$ and $P(S|C_j)$ are combined in a straightforward way: by linear interpolation. This results in the following mixture model:

$$P(S|D_k, C_j) = \prod_{i=1}^{n} (\lambda P(T_i|D_k) + (1 - \lambda)P(T_i|C_j) \tag{2}$$

As a final step we applied two normalisation steps in order to be able to use the probability as a metric across sentences and applied a logarithm in order to convert the product into a summation.

$$NLLR_{\text{content}}(S) = NLLR(S|D_k, C_j) = 1/n \sum_{i=1}^{n} \log(\frac{\lambda P(T_i|D_k) + \mu P(T_i|C_j) + (1 - \mu - \lambda)P(T_i)}{P(T_i)}) \tag{3}$$

Formula (3) shows the final model which can be paraphrased as the (geometric) mean of the log likelihood ratio of a sentence given the mixture model and given the background model.

The conditional probabilities were estimated using maximum likelihood procedures. The documents, clusters and corpus of general English were stemmed (Porter) and stopped in order to reduce morphological variation and eliminate non-content words.

## 2.2 Assessment of the topic model score distribution

For training purposes, we had manually annotated the sentences of five clusters for salience w.r.t. the cluster (we will refer to this set as DUC2001-TNO). We plotted score distributions for within cluster salience and within document salience for both salient and non salient sentences and discovered that indeed there is a monotonically increasing relationship between the score and the probability of salience of a sentence. But this relationship is restricted to the low to medium score ranges. After an optimum, the probability of salience decreases again, indicating that sentences with very high average LR score per term are not so good summary candidates. A closer inspection revealed that often marginally relevant proper names "inflated" the scores. But still the score distribution in the salient and non-salient sentences is different, so the score can be used to infer salience.

We decided to quantize the topic model scores as well and treat them like features in a Naive Bayes framework.

## 2.3 Naive Bayes classifier for surface based salience

We also used the DUC2001-TNO training set to reassess the effectiveness of the surface features that we had used for the DUC 2001 system. Surface features encode information which is not related to the

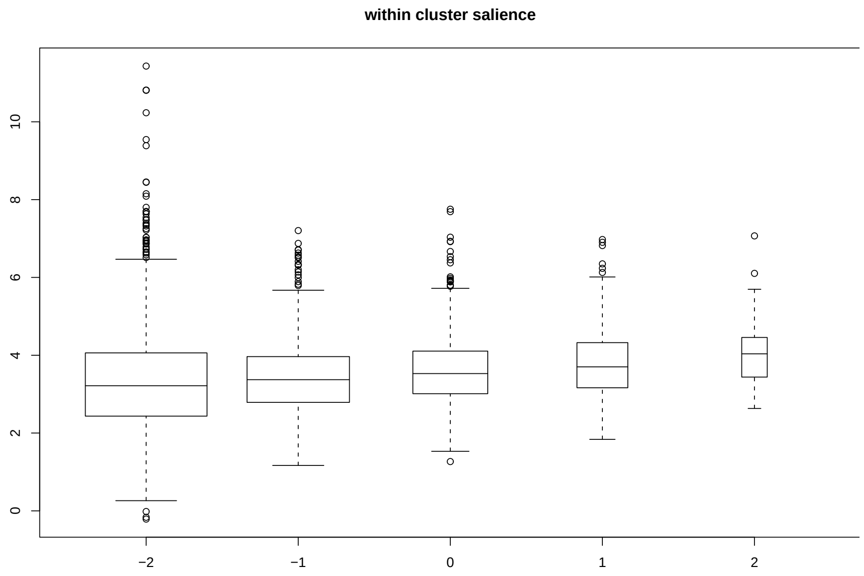

Figure 1: Boxplot of the LR scores for different salience levels

content of a particular document cluster, but which is still useful to predict whether a sentence would be a good summary sentence. A good example is sentence position. Often, the first sentence of a document is a good summary candidate.

Like last year, we decided to work with a Naive Bayes classifier. NB classifiers are extremely easy to implement, but of course care has to be taken with the definition of classes. Enough training data has to be available for robust MLE estimates, so sometimes it is better to work with a smaller amount of classes for a certain parameter (introducing bias) to achieve a higher robustness of the system. Also, Naive Bayes assumes conditional independence between the features, so we have to avoid heavily dependent features like sentence number and reverse sentence number. As an example, we encoded both the sentence number (distance to start of the text) and reverse sentence number (distance to end of the text) in one single feature with values: “first sentence”, “sentence 2-4”, “middle sentence” , “last sentence”, “3 – 1 but last sentence” . This approach effectively avoids dependency and captures the increased probability of salience at the start and end of a document. We also used sentence length (4 classes) and cue phrases (both negative and positive) as features. We trained the NB classifier on the DUC2001-TNO training collection.

The features were combined in the following way to compute the odds of salience (or being a summary candidate sentence).

$$O(s \in S|\bar{\mathbf{x}}) = \frac{P(s \in S|\bar{\mathbf{x}})}{P(s \notin S|\bar{\mathbf{x}})} = \frac{P(\bar{\mathbf{x}}|s \in S)P(s \in S)}{P(\bar{\mathbf{x}}|s \notin S)P(s \notin S)} \tag{4}$$

In formula (4), $\bar{\mathbf{x}}$ is a shorthand for $P(\bar{\mathbf{X}} = \bar{\mathbf{x}})$ i.e. $\bar{\mathbf{X}}$ is a random variable with as value a vector of features $\bar{\mathbf{x}} = (x_1, x_2, ..., X_k)$. Note that $s \in S$ refers here to the probability that a sentence $s$ is part of a summary $S$. The next step is to assume that the features are conditionally independent ($P(\bar{\mathbf{x}}|s \in S) = \prod_i P(x_i|s \in S)$), this is the so-called Naive Bayes assumption. This assumption makes it possible to approximate the probability $P(\bar{\mathbf{x}}|s \in S)$ using maximum likelihood procedures on only a small data set. This leads us to formula 5:

$$O(s \in S|\bar{\mathbf{x}}) = \prod_{j=1}^{k} \frac{P(x_j|s \in S)}{P(x_j|s \notin S)} \frac{P(s \in S)}{P(s \notin S} \tag{5}$$

The prior odds term $\frac{P(s \in S)}{P(s \notin S}$ can be ignored for ranking purposes since it is a constant.

"Training" the naive Bayes classifier consisted of estimating the ratios $P(x_j|s \in S)/P(x_j|s \in S)$ on the 4096 annotated sentences. For our DUC2002 submission, we interpreted only the sentences marked as highly salient as positive training examples, in order to bias the system for short summaries. Also the sentence length feature was further biased towards shorter summaries, by interpolating with the odds of relevance given a certain length class computed on the 200 word summaries of the DUC2001 multi-document model abstracts. We hoped to include some information about the multi-document training data using this method.

| $x_j$ | $P(x_j\|s \in S)/P(x_j\|s \notin S)$ |
|---|---|
| pcp=1 | 2.97 |
| pcp=0 | 0.88 |
| ncp=1 | 0.10 |
| ncp=0 | 1.14 |
| sp=first | 12.53 |
| sp=start2-4 | 4.31 |
| sp=middle | 0.68 |
| sp=last2-4 | 1.38 |
| sp=last | 0.55 |
| len<5 | 0.58 |
| len<10 | 1.30 |
| len<15 | 1.08 |
| len>15 | 0.82 |
| lr<4 | 0.67 |
| lr<5 | 1.43 |
| lr<6 | 1.71 |
| lr>6 | 0.71 |

Table 1: Predictive power of the features. pcp: positive cue phrase, ncp: negative cue phrase, sp: sentence position, len: length, lr: content score

Table 1 shows the predictive power of the different feature values. The feature "first sentence" has a very strong predictive power. Cue phrases exhibit a smaller but significant correlation with salient and non salient sentences. The sentence length feature is especially effective to predict non-salience. This is probably due to the relative high frequency of one, two word "sentences" in the annotated corpus. Sometimes these are real sentences, but often these short sentences are due to a sentence segmentation problem, e.g. a list of senate members is formatted with semicolons. The large variety of interpunction in the training collection (e.g. broadcast transcripts) made it impossible to reach a perfect accuracy using generic splitting rules.

## 3 Extraction of noun phrases for headline summaries

For the task of assigning a headline to each cluster, we had basically two options: we could either create a new system tailored to the task; or we could utilize the existing summarization system to extract salient sentences. We decided to go for the latter, and more specifically we focused our experiments on assigning an extracted maximal noun phrase to the cluster, with the help of our summarization system and a noun phrase extractor developed in the TwentyOne project [ter Stal et al., 1998].

The first step in our method is to locate the NP:s describing the topic of the cluster. To do this, we choose one word, which we call the "trigger word", that must be contained in a potential phrase. To decide on the trigger word, we automatically summarize each single document in a cluster, and then take the highest ranked sentence for every document. These sentences constitute our "trigger word pool". To this pool, we also add the titles of the documents. To find the trigger word, we calculate the word frequency in the pool, using the same stoplist (which is rather extensive) as we use in the actual summarization. The highest ranked word then becomes the trigger word for the cluster at hand.

The second and last step is to select which of all noun phrases containing the trigger word is the most appropriate headline. This time we use the multi-document summarizer and summarize the whole cluster. This gives us all cluster sentences ranked in order of salience. Thereafter, the NP including the trigger word, in a sentence as highly ranked as possible is chosen. As to not get too short cluster headlines, a phrase must contain more than one word, not counting determiners. We also noticed that too long NP:s are not very appropriate, and the task constraint of ten words suits our method fine. If there are two noun phrases containing the trigger word in one sentence, we select the longest one. The same goes for a draw between the trigger words: In case several terms have the same frequency, the longest NP is selected. (Often when the frequency is identical, the two words belong to the same NP.)

When selecting the trigger word, all documents contribute equally to the pool. A document that is peripheral to the theme is trivially handled by the frequency count. We have not noticed any problems in finding an appropriate trigger word. When running our system on the thirty test clusters of last year, the highest ranked term is always central to the subject of the clusters, and they are all part of a noun phrase (in fact, almost all are (proper) nouns). We experimented with letting the trigger word pool consist of the full texts instead of the selected unisummary sentence, but in that case the output was not always desirable, with a non-topical word at the top. Removing the titles from the pool, on the other hand, does not alter the outcome, but including them creates a larger gap between the top ranked word and the others, thus making the selection more reliable. We also experimented with stemming. This only affected the selection of NP:s, and we preferred the headlines selected without stemmed trigger words.

It is, nevertheless, a bigger problem to select the most appropriate of the potential NP:s, with the most common being that the headline is to specific. This is even more problematic when the theme of a cluster is more general than the content of the individual documents. This difficulty holds also for longer extracted summaries, but the problem may be easier disguised when sentences from different documents are presented. At least it should be obvious to a user that the cluster is about, for example, different military planes crashing and not only about an F-14 jet fighter. That this is indeed a problem, can be seen when looking at how our headlines perform in relation to the four cluster categories (cf. Section 4.2).

# 4 Results of Multi-document abstract task

TNO participated with the same system in both the multi-document extract and abstract task. In this section we will present the results in the abstract task.

## 4.1 Official results

Tables 2- 8 give a condensed overview of the results of our system in the abstract task. Each row of the table lists the results for a particular subtask (headline, 50, 100 and 200 word summary respectively). The first column lists the average result of the manual abstracts produced by two NIST assessors with respect to the model summary. The second and third column list the result for the two automatic baseline systems, `lead` and `coverage`, the next column gives the average result for all submitted systems and the last column represents the TNO system. All data points are complemented with the absolute difference with the average manual performance.

Table 2 shows the average number of quality questions that were answered with a non zero answer. The quality questions concern aspects like grammaticality and coherence. A zero answer means that

| | avman | b2(lead) | b3(cov) | avsys | tno |
|---|---|---|---|---|---|
| M-050 | 0.62 | 0.81 (+0.19) | 0.73 (+0.11) | 1.76 (+1.14) | 1.27 (+0.65) |
| M-100 | 0.67 | 0.93 (+0.26) | 1.54 (+0.87) | 2.24 (+1.57) | 2.05 (+1.38) |
| M-200 | 0.83 | 1.19 (+0.36) | 2.20 (+1.37) | 2.83 (+2.00) | 2.83 (+2.00) |

Table 2: count quality non-0

the abstract is good. Automatic systems perform worse than the baseline systems and our system is no exception. Baseline do quite OK on the shorter abstracts (headlines were not evaluated on this aspect)

| | avman | b2(lead) | b3(cov) | avsys | tno |
|---|---|---|---|---|---|
| M-050 | 0.40 | 0.42 (+0.02) | 0.59 (+0.19) | 0.78 (+0.38) | 0.78 (+0.38) |
| M-100 | 0.39 | 0.56 (+0.17) | 0.89 (+0.50) | 0.96 (+0.57) | 1.00 (+0.61) |
| M-200 | 0.58 | 0.60 (+0.02) | 0.99 (+0.41) | 1.04 (+0.46) | 1.03 (+0.45) |

Table 3: mean score quality non-0

Table 3 summarizes the same data, but lists the mean scores for those questions that were answered with a non-zero value. We can see here that the lead baseline is better than the coverage baseline, since a complete portion from a document is of course more coherent than a concatenation of fragments from different documents. Automatic systems do not differ dramatically from the coverage baseline system.

| | avman | b2(lead) | b3(cov) | avsys | tno |
|---|---|---|---|---|---|
| M-010 | 0.49 | 0.00 (-0.49) | 0.00 (-0.49) | 0.19 (-0.30) | 0.13 (-0.36) |
| M-050 | 0.36 | 0.14 (-0.22) | 0.14 (-0.22) | 0.16 (-0.20) | 0.12 (-0.24) |
| M-100 | 0.34 | 0.13 (-0.21) | 0.20 (-0.14) | 0.17 (-0.17) | 0.16 (-0.18) |
| M-200 | 0.36 | 0.13 (-0.23) | 0.22 (-0.14) | 0.20 (-0.16) | 0.21 (-0.15) |

Table 4: mean coverage

Table 6 gives insight in the quality of the peer units that were left unmarked. Of course, a single perfect model abstract does not exist, making it very well possible that automatic abstracts contain relevant material which was not chosen by the assessor who produced the model summary. Fortunately, most of the unmarked material in the 50,100 and 200 manual peer summaries was judged relevant. We can conclude that there is a quite high level of inter assessor agreement on relevance. However, it is striking that also most of the unmarked lead baseline material is relevant.

The unit ratio gives some insight in the differences between the number of sentences produced by the reference manual abstract and the automatic abstracts. Most automatic abstracts contain fewer units, which is probably due to the fact that they do abstraction and have no functionality to trim unnecessary phrase elements.

Table 8 lists the fraction of unrelated unmarked peer units, which can be thought of the fraction of noise in the total abstract. The TNO system is performing disappointingly here, since it has more noise than the average automatic systems.

## 4.2 Cluster Categories

We have studied how our system performed in relation to the four category types, to which each cluster belongs. For the ten word summary the most difficult type — as judge by mean coverage — is the different events of the same type (category 3), while the easiest one is documents about a single event, not being a natural disaster (category 2). To assign an appropriate headline for a set of document each

| | avman | b2(lead) | b3(cov) | avsys | tno |
|---|---|---|---|---|---|
| M-010 | 0.33 | 0.00 (-0.33) | 0.00 (-0.33) | 0.12 (-0.21) | 0.08 (-0.25) |
| M-050 | 0.24 | 0.10 (-0.14) | 0.10 (-0.14) | 0.11 (-0.13) | 0.10 (-0.14) |
| M-100 | 0.23 | 0.08 (-0.15) | 0.14 (-0.09) | 0.12 (-0.11) | 0.11 (-0.12) |
| M-200 | 0.24 | 0.09 (-0.15) | 0.15 (-0.09) | 0.14 (-0.10) | 0.14 (-0.10) |

Table 5: mean length adjusted coverage

| | avman | b2(lead) | b3(cov) | avsys | tno |
|---|---|---|---|---|---|
| M-010 | 0.30 | 0.00 (-0.30) | 0.00 (-0.30) | 0.25 (-0.05) | 0.26 (-0.04) |
| M-050 | 0.89 | 0.64 (-0.25) | 0.57 (-0.32) | 0.60 (-0.29) | 0.46 (-0.43) |
| M-100 | 0.86 | 0.68 (-0.18) | 0.75 (-0.11) | 0.69 (-0.17) | 0.60 (-0.26) |
| M-200 | 0.86 | 0.80 (-0.06) | 0.73 (-0.13) | 0.75 (-0.11) | 0.73 (-0.13) |

Table 6: fraction of unmarked peers that are relevant

| | avman | b2(lead) | b3(cov) | avsys | tno |
|---|---|---|---|---|---|
| M-010 | 1.02 | 0.00 (-1.02) | 0.00 (-1.02) | 0.89 (-0.13) | 0.87 (-0.15) |
| M-050 | 0.76 | 0.56 (-0.20) | 0.44 (-0.32) | 0.52 (-0.24) | 0.44 (-0.32) |
| M-100 | 0.69 | 0.52 (-0.17) | 0.43 (-0.26) | 0.44 (-0.25) | 0.48 (-0.21) |
| M-200 | 0.64 | 0.49 (-0.15) | 0.39 (-0.25) | 0.40 (-0.24) | 0.51 (-0.13) |

Table 7: unit ratio

| | avman | b2(lead) | b3(cov) | avsys | tno |
|---|---|---|---|---|---|
| M-010 | 0.11 | 0.00 (-0.11) | 0.00 (-0.11) | 0.31 (+0.20) | 0.34 (+0.23) |
| M-050 | 0.03 | 0.19 (+0.16) | 0.13 (+0.10) | 0.17 (+0.14) | 0.24 (+0.21) |
| M-100 | 0.06 | 0.20 (+0.14) | 0.08 (+0.02) | 0.13 (+0.07) | 0.18 (+0.12) |
| M-200 | 0.05 | 0.14 (+0.09) | 0.07 (+0.02) | 0.10 (+0.05) | 0.13 (+0.08) |

Table 8: fraction of unrelated unmarked peer units

discussing a separate activity, although conceptually related — as is the case for category 3 — is naturally rather difficult, and would most likely require some form of semantic generalization.

As for the longer abstracts, the biography type (category 4) is the one obtaining the most low scores. One explanation could be that such a cluster may contain a lot of various information about one person, and it is therefore less likely that the system selects the same information as the person constructing the model summary does. As for the headline assignment, getting the name of the person in question correct is usually enough to get at least a low mean coverage score. It is also the case that the performance of the summarizer increases for category 4 as the summary length increases (in contrast to category 2 or 3, where it decreases).

We also studied how the other five systems that submitted ten word abstracts performed in relation to the category types, as well as on the 100 hundred word summaries. As for the headlines, the other systems' best and worst categories are in most cases different from ours.

Category 2 is basically a subtype to category 1 (a single event of a natural disaster), and these two types are those for which our system has its best performance for the 50, 100 and 200 word summaries. These are actually the types for which all systems have their best performance on the one hundred word summary (as mentioned previously, this was the length, apart form the headlines, that we checked). Assumingly, many sentences describing more or less the same thing is favorable for the way sentences are selected.

## 4.3 Judgement Quality

When inspecting the judgments for the headline assignment, i.e., the ten word summaries, we discovered many inconsistencies. We will here give examples of some of those.

The model summary for cluster 65 is “Bush, friends support Quayle despite errors and political inexperience”. “The story of Dan Quayle” is not marked and not related, while “Danny Quayle from Polk Street” has mean coverage 0.2. For cluster 102, the headline “[T]he death of Lucille Ball” is related but does not share any content with the model “Lucille Ball, TV's comic genius, had major impact on the industry”. For another cluster, one headline has mean coverage 0.6, while another headline, which is identical apart from containing even more information has 0.2 (cluster 61).

Also, the notion of relatedness seem to vary among the assessors. For example, the proposed headline (that does not share any content with the model) “African National Congress leader Nelson Mandela” is not related to the model headline “Mandela released from prison; end of apartheid in South Africa near”, while “people in cars and homes” is related to “Powerful storm causes death and destruction across the United States”.

Another remark concerns the number of model units. Does the headline “Ferry capsizings from overloading, hitting reefs, high seas, etc., kill hundreds” really consist of five model units? This is not unimportant, as the mean coverage depends on the number of model units.

We are well aware of the fact that judging is difficult, and the question is rather how much inconsistency that may be tolerated before the results become flaw. Since there are 59 clusters, we can assume that inconsistencies are averaged out, but the results are not very reliable when looking at how a system performs for single clusters. This type of evaluation is therefore not suitable for in-depth analyses of how a system's performance may be improved.

## 4.4 Blind Tests

To see whether the performance according to the evaluation results for the different systems correspond to the intuition of what constitutes a “good” summary, we conducted two blind tests. For three randomly selected clusters, we collected the six headlines and the eight 100 word abstracts submitted by the participating teams, as well as the manually constructed ones (two for each summary type and cluster). We let three persons individually rank the headlines and the abstracts respectively in order of how “good” they were. The persons did not have access to the actual documents in the clusters, but could base their opinion only on the material at hand. It should be noted that this ranking task differs from the DUC evaluation in two respects. Firstly, no summary could be just as good (or bad) as another. Secondly, no manual summary was considered being the model.

As the data is too sparse, we have not checked for the statistical significance of the ranking results. However, the inter-judge agreement seem to be strikingly high. For the headlines the same systems are found at the top positions and the bottom positions respectively for each cluster. Remarkably, the ranking corresponds closer to the overall performance of the automatic systems when averaged over all 59 clusters, than to the judgments made by the assessors for these three clusters. This indicates that even if a system gets a bad score for a cluster, this may be due to the fact that the model headline and the system's headline do not coincide, not because the system's headline is bad. And in the end a system capable of assigning "good" headlines will get a better total result. (Here we can see a parallel to the inconsistencies discussed above, where single "misjudgments" are evened out over a large set of clusters.) An interesting point is that more often was an automatically constructed headline ranked the highest, and not a manual one. (This does actually not correspond to the overall performance, where the manual headlines have the highest mean coverage.)

As for the ranking of the 100 word summaries, the picture is not as clear, although the tendency seem to be the same. As the summaries are longer, the differences are probably evened out already for the individual summary. This is confirmed by studying a histogram over the mean coverage scores for all clusters and for all systems for the two abstract sizes. There we can see that the score varies more for the headlines than for the longer summaries.

# 5 Results of the Multi-document extract task

This section will present the results of the extract task, which is a new task. The aim of the task is to study the performance of sentence extraction systems in a systematic way, by working with pre-segmented documents. The advantage of this set-up is that the judgements of the task can potentially be reused to train extraction systems.

## 5.1 Results of the submitted system

The results for the extract task were disappointing. We decided to run some post-hoc experiments to check the performance of our DUC 2002 system with the DUC 2001 system. The results of these experiments will be discussed in Section 5.4. Since the level of agreement between the two manual extracts was so low, we decided to do a small experiment in order to get an idea for the difficulty of the task. This experiment will be described in the following section.

## 5.2 Evaluation Measures

Seven teams did both the extract and the abstract task. For the 200 word summaries, the correlation coefficient between the average precision for all 59 clusters (calculated over the 116 model extracts) and the mean coverage for the 59 model abstracts is 0.91. If looking at how the seven systems are ranked by the two measures, the top three systems are the same, as are the three bottom ones. The seventh system stays in the middle (ours…). As for recall, the correlation with mean coverage is 0.96, and the rankings are identical. In other words, if a system performs well in the extract task, it is also very likely to perform well in the other task. Taking this a step further, this means that the simpler precision and recall calculations can be enough to decide how a system would score if assessed by a human comparing it to a man-made abstract. And as most systems still do extracting, re-usable man-made extracts are indeed valuable for system development. However, this does not hold for single clusters: the correlation between mean coverage and precision when calculated for each cluster for the seven systems, and then averaged over all cluster is but 0.31. The corresponding figure for recall is 0.27.

As an alternative to the measures used in the DUC evaluation, we decided to experiment with the *relative utility* measure, proposed by [Radev et al., 2000].

A measure like precision and recall only rewards sentences identical to those selected by the human assessor. In contrast, relative utility is a way to give scores to all sentences in a summary, depending on how “summary worthy” each one is.

We selected one cluster (d062, as it does not contain that many sentences), and let three persons assign a score from one to ten to each sentence of how suitable it would be to include in a summary. The idea of [] is that this scoring is used to automatically construct, for each judge, a best extract of desired length by taking the top *n* per cent of the sentences. Thereafter, a system’s (or several systems’) performance may be computed by comparing the system’s extract to the judges’ extract. This is done by calculating the relative utility, by giving each sentence in the system’s extract the sentence score of each judge. The total score per judge for the whole extract is thereafter normalized with the score of the judge’s own extract. The final step is to average the scores over all judges.

The relative utility assumes that all extracts have the same number of sentences. A user will supposedly state the compression rate in per cent of the whole document, and relative utility transforms this to the percentage of sentences. While this may still be feasible for a single document — where one can imagine that the user knows how long a document is — it is less practical for a cluster of documents. When constructing summaries consisting of a certain number of words, the number of sentences will be different for different extracts (both for the judges’ and the systems’). We calculated the relative utility for the ten submitted 200 word extracts as well as for the two manual extracts for cluster 62 in two different ways: we added all scores for an extract, not considering the number of sentences; and we added the scores and then normalized with the number of sentences in the extract at hand (in this case, the judges’ extracts are also normalized). The correlation between these two results for the ten submitted extracts as well as the two model extracts is 0.55. The two different approaches thus give a different outcome of the performance of the systems. A problem when not normalizing is that the value does not necessarily fall between zero and one, as an extract can get a higher total score than the judge’s extract. It also means that two short sentences with lower scores can be just as good as a longer sentence with a higher score. On the other hand, when the total value is normalized, the number of sentences are given to high importance, and a system that happens to include a low score sentence will be severely punished. To illustrate: If one system picks two sentences with the scores 10 and 9, the average per sentence is 9.5, while if another system selects three sentences with the scores 10, 9, and 1, the average will only be 6.7, although it may even contain the same two informative sentences. In other words, the normalized version is a kind of precision, while the unnormalized version is a kind of recall. So maybe both measures are interesting (like precision and recall are each others complement).

As several sentences may contain the same information and get the same scores, this can lead to redundancy in an automatically constructed judge extract. To explore the influence of this on the relative utility calculations, we let the three persons manually construct two extracts each, one of 200 words and one of 400 words. (Another motivation was that we wanted to check consistency between our extracts and the manual extracts, to confirm our hypothesis that many good extracts are possible. This was indeed the case.) Of the six manually constructed extracts, five had a higher total sentence score than the corresponding automatically constructed judge extracts. This was due to the fact that more sentences were included (within the word limit), as lower scored sentences were sometimes preferred over higher scored ones. However, when calculating the relative utility using these manually constructed extracts instead of the automatically constructed ones, the difference in scores were averaged out. It is thus not a problem if the judges’ extracts contain redundancy (whether normalized with the sentence length or not).

## 5.3 Relative Utility vs. Precision, Recall and Mean Coverage

As previously mentioned, we calculated the correlation between the mean coverage and the precision and recall for the teams participating in both the extraction task and the abstraction task. This means that we compared different texts, as the teams have submitted different material to the two tasks.

To see how the evaluation measures correlate, we constructed an extract from the 200 word model summary of cluster d062. (This extract became closer to 300 words, though.) We also identified the original sentences for the eight abstracts from the participating systems in this task.

To briefly summarize what we found for the eight abstracts turned into extracts for this cluster, as judged by the model summary turned into a sentence extract: The non-normalized relative utility has a rather high correlation with the precision and recall for these constructed extracts. The sentence normalized relative utility has a very low correlation with both precision and recall. The precision and recall are in turn highly correlated to the average precision and average recall respectively when using the two original model extracts as keys instead of the one constructed from the model abstract.

The correlation coefficient between the mean coverage and the precision and recall for the constructed extracts is as low as 0.33 and 0.26 — thus showing that for at least this cluster the correlation between the mean coverage and the other two measures is rather low, when judging the same material just using different measures. Mean coverage has in addition a low correlation with non-normalised relative utility, while it is even lower with normalised relative utility. As stated earlier, the non-normalised version can be seen as a kind of recall, and mean coverage is also favoring recall.

## 5.4 Post Hoc extraction experiments

After the official DUC evaluation we performed several extraction experiments to gain more insight into a number of important issues. First of all we wanted to discover the relative contribution to performance of each individual feature and to see the effect of different feature value classes. Second, we wanted to see the influence of estimating the parameters of our probabilistic salience model on a larger training set than the relatively small one we had been using. This experiment was possible thanks to the release of a set of 146 manually created extracts for the DUC01 training data by John M. Conroy. Third, obviously, we wanted to compare our new system to our 2001 approach.

### 5.4.1 Redundancy reduction

By coincidence we found that our redundancy reduction method, based on the MMR algorithm by Carbonell and Goldstein [Carbonell and Goldstein, 1998] did not have any added value. After we found and fixed a bug in the code for this algorithm, the performance of the system did not improve. Varying the parameter which determines to what extent the distance to a summary sentence contributes to the final sentence score did not help. Therefore, we decided not to apply sentence reranking in our post hoc runs. We need to further investigate this issue in our future work.

### 5.4.2 Relative contribution of the features

To discover the relative contribution of each individual feature to the performance of the extraction system, we performed some runs where we left out certain features. The results of these runs are presented in Table 9. Cpn is the negative cue phrase feature, cpp the positive cue phrase feature, sp means sentence position, sl is the sentence length feature, and lrc is the combined likelihood ratio (a combination of the within-document and within-cluster salience of a certain sentence).

As the figures in Table 9 show, the sentence position feature is the tip for the top. Using just this feature produces a precision which is almost as high as for the run in which all features are used, and a recall which is even better than for the run with all features. Section (see 2.3) describes the five feature value classes we defined for this feature. Table 1 shows that the ratio $P(x_j|s \in S)/P(x_j|s \notin S)$ is extremely high for the 'sp=first' (sentence S is the first sentence of a document) value of the sentence position feature. Thus, whether or not a certain sentence is the first sentence of a document largely determines the final score for that sentence. Often, the first sentence of a document summarizes the event or subject described in that document and is therefore a good (in most cases the best) sentence for an extract.

Another noticeable result is the considerable negative influence of the negative cue phrase feature. We found that the value class division we defined for this feature ('one or more negative cue phrases' or 'no negative cue phrases') was not specific enough. We defined new classes for this feature ('no negative cue phrase', 'one negative cue phrase', and 'more than one negative cue phrase') which yielded better results, as shown in Table 9 (cn3,cp,sl,sp,lrc).

| *n*-word summary | features | Precision | Recall |
|---|---|---|---|
| 200 | cp,cn,sl,sp,lrc | 0.220 | 0.166 |
|  | cp,sl,sp,lrc | 0.224 | 0.174 |
|  | cn,sl,sp,lrc | 0.218 | 0.166 |
|  | sl,sp,lrc | 0.224 | 0.168 |
|  | sl,lrc | 0.074 | 0.052 |
|  | lrc | 0.052 | 0.052 |
|  | cp,cn,sl,sp | 0.218 | 0.166 |
|  | cp,cn,sp,lrc | 0.206 | 0.166 |
|  | cp,cn,sl,lrc | 0.054 | 0.038 |
|  | sp | 0.204 | 0.169 |
|  | cn3,cp,sl,sp,lrc | 0.216 | 0.169 |
| 400 | cp,cn,sl,sp,lrc | 0.277 | 0.229 |
|  | cp,sl,sp,lrc | 0.285 | 0.240 |
|  | cn,sl,sp,lrc | 0.278 | 0.228 |
|  | sl,sp,lrc | 0.282 | 0.238 |
|  | sl,lrc | 0.146 | 0.112 |
|  | lrc | 0.084 | 0.091 |
|  | cp,cn,sl,sp | 0.282 | 0.230 |
|  | cp,cn,sp,lrc | 0.256 | 0.233 |
|  | cp,cn,sl,lrc | 0.126 | 0.096 |
|  | sp | 0.262 | 0.248 |
|  | cn3,cp,sl,sp,lrc | 0.284 | 0.242 |

Table 9: the contribution of the different features to precision and recall for the extraction task

| *n*-word summary | Pos. training set | *N* sentences | Precision | Recall |
|---|---|---|---|---|
| 200 | annotClass=2 | 74 | 0.144 | 0.136 |
| | annotClass=1+2 | 368 | 0.220 | 0.166 |
| | annotClass=0+1+2 | 1005 | 0.206 | 0.155 |
| | annotClass=-1+0+1+2 | 2197 | 0.146 | 0.111 |
| 400 | annotClass=2 | 74 | 0.184 | 0.195 |
| | annotClass=1+2 | 368 | 0.277 | 0.229 |
| | annotClass=0+1+2 | 1005 | 0.287 | 0.236 |
| | annotClass=-1+0+1+2 | 2197 | 0.221 | 0.180 |

Table 10: the influence of extending the positive training set with sentences from different annotation classes

| *n*-word summary | Training set | Precision | Recall | F-Measure |
|---|---|---|---|---|
| 200 | TNOannotClass=1+2 | 0.220 | 0.166 | 0.189 |
| | Conroy | 0.232 | 0.181 | 0.203 |
| | Merge | 0.232 | 0.182 | 0.204 |
| 400 | TNOannotClass=1+2 | 0.277 | 0.229 | 0.251 |
| | Conroy | 0.280 | 0.242 | 0.260 |
| | Merge | 0.286 | 0.246 | 0.264 |

Table 11: the effect of using different training material

### 5.4.3 Training data

We estimated the parameters of our probabilistic sentence salience model on a manually annotated set of five clusters from the DUC01 training data. About 3900 sentences from these clusters were scored on salience using a scale of five values, from ‘completely irrelevant (-2)’ to ‘highly relevant (2)’. For the parameter estimations for our DUC01 system, we regarded the sentences scored with values 1 and 2 as positive examples. Our DUC01 system showed a bias towards longer summaries. Because we wanted to focus on extracting shorter summaries this year, we restricted our positive set to the highly relevant sentences. However, because the portion of sentences scored as ‘highly relevant’ was very small, the parameter estimations based on these data were not reliable. We performed a post evaluation run which shows that our decision of restricting our positive training set was injudicious. Table 10 shows the effect of extending the positive training set with sentences from the different annotation classes. The number of positive training examples seems to be very important.

John M. Conroy provided 146 manually created extracts covering approximately half of the DUC01 training documents. These extracts can very well be used as training data for an extraction-based summarization system by regarding all sentences in the manually created extracts as positive training examples. Because we used a different sentence splitter than Conroy, a limited number of sentences could not be mapped to the Conroy annotations. All in all we could use 6835 sentences for training, of which 967 were part of one of the manually generated extracts. Table 11 shows the effect of training on the Conroy data (and of merging the Conroy annotations with our own). Again, we can see that a larger training set leads to better parameter estimations and consequently to better extraction results.

### 5.4.4 DUC01 versus DUC02

When we compare the results of our DUC01 system with the results of this year’s system (trained on the same data, redundancy reduction switched off), we can see a substantial improvement of both precision and recall for different summary lengths. Table 12 shows the results.

| *n*-word summary | System | Precision | Recall | F-Measure |
|---|---|---|---|---|
| 200 | TNO DUC01 | 0.206 | 0.132 | 0.161 |
| | TNO DUC02 | 0.220 | 0.166 | 0.189 |
| 400 | TNO DUC01 | 0.260 | 0.200 | 0.226 |
| | TNO DUC02 | 0.277 | 0.229 | 0.251 |

Table 12: the TNO ’01 approach versus this year’s system

# 6 Conclusions

For DUC2002 we redesigned our system. For DUC2001, language models were a central feature. However, we found that language models favour sentences with rare terms, which is undesirable for a summarizer. However, the content of a sentence can help to predict whether a sentence should be extracted. The central architecture of the new system is a Naive Bayes classifier, based on a number of features: sentence position, positive and negative cue phrases, sentence length and language model score. We compute the log-odds of salience in order to avoid the inconsistent statistical independence assumption of [Kupiec et al., 1995]. This choice was justified by comparing the results of the old and new architecture on the extraction task.

A second change with regards to DUC2001 was the choice to train our summarizer only on highly salient sentences. This was motivated by the fact that our DUC2001 system was biased for longer summaries. This choice turned out to deteriorate results in a significant way. Retraining the summarizer (for the extract task) on both salient and highly salient sentences, brought performance on par with the other best performing automatic systems. This effect can be explained by the fact that the amount of highly salient sentences is quite small, so small that the model parameters could not be estimated in a reliable way. We also retrained on a somewhat larger dataset provided by John Conroy and withe the merged training sets. Results improved a little bit with eaxh extension of the training set. As any ML approach, performance is dependent on the size of the training set. Our extraction system yields in fact better results than the first sentence (coverage) baseline system. However, also for our system, the sentence position is by far the strongest feature.

Unfortunately, the retrained summarizer cannot be reevaluated for the abstract data, since this is a manual process. This brings us to an important point: although the extract evaluation set of DUC2002 can be critisized because the inter assessor agreement is quite low (many different but sufficiently good extracts are possible), this evaluation set has already enabled several groups to do very interesting post-hoc experiments. We would recommend however, to do graded assessments of at least the submitted and manual extracts, since this would produce a pool of assessments enabling the calculation of relative utility. We think that relative utility is a more robust evaluation metric, which is less sensitive to a specific golden standard. Relative utility has some disadvantages though, it should be extended or complemented with a metric for redundancy. Also, relative utility is not designed to compare summaries that were produced to meet a fixed word length criterion.

The TNO system has also been extended to produce headlines in the form of extracted noun phrases. Manual assessment of the produced headlines for the training data gave quite satisfactory results for more than 60% of the clusters. The official evaluation on the DUC2002 testset yielded acceptable results. Manual assessment according to the same standards as used for the assessment of the headlines for the training data yielded the following results: good: 34%, acceptabel: 29%, bad: 37 %. It is unclear though, whether the assessment of the headlines conforms to our intuition. Some noun-phrase headlines, which were clearly to the point, received a zero credit.

Although abstracting should be the task to aim for, supporting extraction evaluations is valuable in the meantime as it easier can be re-used and we propose that if someone is constructing manual extracts, it might be worth-while to score each sentence for summary worthiness as well.

# References

[Carbonell and Goldstein, 1998] Carbonell, J. G. and Goldstein, J. (1998). The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *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.

[Edmundson, 1969] Edmundson, H. (1969). New methods in automatic abstracting. *Journal of the Association for Computing Machinery*, 16(2).

[Hiemstra et al., 2001] Hiemstra, D., Kraaij, W., Pohlmann, R., and Westerveld, T. (2001). Twenty-one at clef-2000: Translation resources, merging strategies and relevance feedback. In Peters, C., editor, *Proceedings of the CLEF 2000 Cross-Language Text Retrieval System Evaluation Campaign*.

[Kraaij et al., 2000] Kraaij, W., Pohlmann, R., and Hiemstra, D. (2000). Twenty-one at TREC-8: using language technology for information retrieval. In *The Eighth Text Retrieval Conference (TREC-8)*. National Institute for Standards and Technology.

[Kraaij et al., 2001] Kraaij, W., Spitters, M., and van der Heijden, M. (2001). Combining a mixture language model and naive bayes for multi-document summarisation. In *Notebook papers of the SIGIR 2001/ DUC 2001 workshop*.

[Kupiec et al., 1995] Kupiec, J., Pedersen, J. O., and Chen, F. (1995). A trainable document summarizer. In *SIGIR'95, Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Seattle, Washington, USA.*, pages 68–73. ACM Press.

[Radev et al., 2000] Radev, D. R., Jing, H., and Budzikowska, M. (2000). Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation and user studies. In *ANLP/NAACL 2000 Workshop*, pages 21–29.

[Spitters and Kraaij, 2001] Spitters, M. and Kraaij, W. (2001). Using language models for tracking events of interest over time. In *Proceedings of LMIR 2001*, pages 60–65, Pittsburgh, USA.

[ter Stal et al., 1998] ter Stal, W., Beijert, J.-H., de Bruin, G., van Gent, J., de Jong, F., Kraaij, W., Netter, K., and Smart, G. (1998). Twenty-one: cross-language disclosure and retrieval of multimedia documents on sustainable development. *Twenty-One: cross-language disclosure and retrieval of multimedia documents on sustainable development*, 30(13).