Hedge Trimmer: A Parse-and-Trim Approach to Headline Generation

Anonymous

Anonymous

Abstract

This paper presents Hedge Trimmer, a HEadline GENeration system that creates a headline for a newspaper story using linguistically-motivated heuristics to guide the choice of a potential headline. We present feasibility tests used to establish the validity of an approach that focuses on the earlier part of the story for producing a headline. In addition, we describe experimental results that demonstrate the effectiveness of our linguistically-motivated approach over a HMM-based model, using both human evaluation and automatic metrics for comparing the two approaches.

1 Introduction

In this paper we present Hedge Trimmer, a HEaDline GEneration system that creates a headline for a newspaper story by removing constituents from a parse tree until a length threshold has been reached. Linguistically-motivated heuristics guide the choice of which constituents of a story should be preserved, and which ones should be deleted. Our focus is on headline generation for English newspaper texts, with an eye toward the production of document surrogates—for cross-language information retrieval—and the generation of readable headlines from speech broadcasts.

In contrast to original newspaper headlines, which are often intended only to catch the eye, our approach produces informative abstracts describing the main theme or event of the newspaper article. We claim that the construction of informative abstracts requires access to deeper linguistic knowledge, in order to make substantial improvements over purely statistical approaches.

The next section presents previous work in the area of automatic generation of abstracts. Following this, we present feasibility tests used to establish the validity of an approach that focuses on the earlier part of the story for producing a headline.

Next, we describe our algorithm. Finally, we describe our automatic and human evaluations, comparing Hedge Trimmer to a probabilistic model for automatic headline generation (Zajic et al, 2002). We demonstrate the effectiveness of our linguistically-motivated approach over the probabilistic model, using both human evaluation and automatic metrics.

2 Previous Work

Other researchers have investigated the topic of automatic generation of abstracts, but the focus has been different, e.g., sentence extraction (Edmundson, 1969; Johnson et al, 1993; Kupiec et al., 1995; Mann et al., 1992; Teufel and Moens, 1997; Zechner, 1995), processing of structured templates (Paice and Jones, 1993), sentence compression (Knight and Marcu, 2001; Luhn, 1958), and generation of abstracts from multiple sources (Radev and McKeown, 1998). We focus instead on the construction of headline-style abstracts from a single story.

The approach we use in Hedge is most similar to that of (Knight and Marcu, 2001), where a single sentence is shortened using statistical compression. As in this work, we select headline words from story words in the order that they appear in the story—in particular, the first sentence of the story. However, we use linguistically motivated heuristics for shortening the sentence; there is no statistical model, which means we do not require any prior training on a large corpus of story/headline pairs.

Linguistically motivated heuristics have been used by (McKeown et al, 2002) to distinguish constituents of parse trees which can be removed without affecting grammaticality or correctness. GLEANS (Daumé et al, 2002) uses parsing and named entity tagging to fill values in headline templates.

Consider the following excerpt from a news story:
(1) **Story Words:** According to a now-finalized blueprint described by U.S. officials and other sources, the Bush administration plans to take complete, unilateral control of a post-Saddam Hussein Iraq, with an interim administration headed by a yet-to-be named American civilian who would direct the reconstruction of the country and the creation of a "representative" Iraqi government.

Generated Headline: Bush administration plans to take complete, unilateral control of post-Saddam Hussein Iraq.

In this case, the words in bold form a fluent and accurate headline for the story. Italicized words are deleted based on information provided in a parse-tree representation of the sentence.

In this paper, we present our technique for producing headlines using a parse-and-trim approach based on the BBN Parser. As described in Miller et al. (1998), the BBN parser builds augmented parse trees according to a process similar to that described in Collins (1997). The BBN parser has been used successfully for the task of information extraction in the SIFT system (Miller et al., 2000).

We first discuss the results of our feasibility testing—illustrating that our approach is a promising path to follow. Next, we describe the application of the parse-and-trim approach to the problem of headline generation. After this, we discuss the linguistically-motivated heuristics we use to produce results that are more headline-like. Finally, we discuss two evaluations—one by human and one by machine—for assessing the coverage and general utility of our approach to automatic generation of headlines.

### 3 Feasibility Testing

Our approach is based on the selection of words from the original story, in the order that they appear in the story, and allowing for morphological variation. To determine the feasibility of our headline-generation approach, we first attempted to apply our “select-words-in-order” technique by hand. We examined 73 stories from the TIPSTER corpus and found that it was possible to produce a fluent and accurate informative headline for all of the stories. Two researchers each constructed a headline for each of the 73 stories, using words or morphological variants of words from the stories in order.

Of the 146 headlines, 2 did not meet the “select-words-in-order” criteria because of word reordering. We found that at least one fluent and accurate headline meeting the criteria was created for each of the stories. Further, we discovered that, with no instructions about sentence boundaries, the researchers constructed headlines entirely of words from the first sentence 80.1% of the time, and at a finer grain, 86.7% of the headline words were chosen in order from the first sentence. We conclude that our approach demonstrates promise for stories that are written as paragraphs of prose.

As part of this initial feasibility evaluation, we observed that only 8.9% of our 146 human-generated headlines used words beyond the first sentence, and none of the 144 valid headlines used words from beyond the fourth sentence. Figures 1 and 2 show the percentages of headlines and headline words selected from the first through fifth sentences.

The average length of the headlines was 10.76 words. Stories whose headlines required the later sentences tended to be human-interest stories with attention-grabbing introductions or they appeared to be excerpts from the middle of larger stories. Thus, in our current model, we adopt the additional constraint that headline words must be chosen from the first sentence of the story, using a threshold of 10 headline words.

---

**Figure 1**

![Percentage of Human-Generated Headlines with Words Drawn n (in order) Entirely from First N Sentences](image-url)
4 Approach

The input to Hedge is a story, whose first sentence is immediately passed through the BBN parser. The parse-tree result is passed to a linguistically-motivated module that selects story words to form headlines by means of three simple steps:

1. Choose lowest leftmost S with NP,VP
2. Remove low content units
   - some determiners
   - time expressions
3. Iterative shortening:
   - XP Reduction
   - Remove preposed adjuncts
   - Remove trailing PPs
   - Remove trailing SBARs

We discuss each of these three steps in turn.

4.1 Choose the Correct S Node

The first step relies on what is referred to as the Projection Principle in linguistic theory (Chomsky, 1981): Predicates project a subject (both dominated by S) in the surface structure. Our human-generated headlines generally conformed to this rule; thus, we adopted it as a constraint in our algorithm.

An example of the application of step 1 above is the following, where boldfaced material from the parse tree representation is retained and italicized material is eliminated:

(2) Input: Rebels agree to talks with government officials said Tuesday.

Parse: /S [S [NP Rebels] [VP agree to talks with government]] officials said Tuesday.

Output of step 1: Rebels agree to talks with government.

When the parser produces a correct tree, this step provides a grammatical headline. However, the parser often produces an incorrect output. Human inspection of our 624-sentence DUC-2003 evaluation set revealed that there were two such scenarios, illustrated by the following cases:

(3) [S [SBAR What started as a local controversy] [VP has evolved into an international scandal.]]

(4) [NP [NP Bangladesh] [CC and] [NP [NP India] [VP signed a water sharing accord.]]]

In the first case, an S exists, but it does not conform to the requirements of step 1. This occurred in 2.6% of the sentences in the DUC-2003 evaluation data. We resolve this by selecting the lowest leftmost S, i.e., the entire string “What started as a local controversy has evolved into an international scandal” in the example above.

In the second case, there is no S available. This occurred in 3.4% of the sentences in the evaluation data. We resolve this by selecting the root of the parse tree; this would be the entire string “Bangladesh and India signed a water sharing accord” above. No other parser errors were encountered in the DUC-2003 evaluation data.

4.2 Removal of Low Content Nodes

Step 2 of our algorithm eliminates low-content units. We start with the simplest low-content units: the determiners a and the. Beyond these, we found that the human-generated headlines did not include time expressions which, although certainly not content-free, do not contribute toward conveying the overall “who/what content” of the story. Since our goal is to provide an informative headline (i.e., the action and its participants), the identi-
fication and elimination of time expressions provided a significant boost in the performance of our automatic headline generator.

We identified time expressions in the stories using BBN’s Identifinder (Bikel et al, 1999). We implemented the elimination of time expressions as a two-step process:

- Use Identifinder to mark time expressions
- Remove [PP … [NP [X] …] …] and [NP [X]] where X is tagged as part of a time expression

The following examples illustrate the application of this step:

(5) **Input:** The State Department on Friday lifted the ban it had imposed on foreign fliers.

**Parse:** [Det The] State Department [PP [IN on] [NP [NNP Friday]]] lifted [Det the] ban it had imposed on foreign fliers.

**Output of step 2:** State Department lifted ban it has imposed on foreign fliers.

(6) **Input:** An international relief agency announced Wednesday that it is withdrawing from North Korea.

**Parse:** [Det An] international relief agency announced [NP [NNP Wednesday]] that it is withdrawing from North Korea.

**Output of step 2:** International relief agency announced that it is withdrawing from North Korea.

We found that 53.2% of the stories we examined contained at least one time expression which could be deleted. A human inspection of 50 deleted time expressions showed that 38 were desirable deletions, 10 were locally undesirable because they introduced an ungrammatical fragment,1 and 2 were undesirable because they removed a potentially relevant constituent. However, even an undesirable deletion often pans out for two reasons: (1) the ungrammatical fragment is frequently deleted later by some other rule; and (2) every time a constituent is removed it makes room under the threshold for some other, possibly more relevant constituent. Consider the following examples.

(7) At least two people were killed Sunday.

(8) At least two people were killed when single-engine airplane crashed.

Example (7) was produced by a system which did not remove time expressions. Example (8) shows that if the time expression Sunday were removed, it would make room below the 10-word threshold for another important piece of information.

### 4.3 Iterative Shortening

The final step, iterative shortening, removes linguistically peripheral material—through successive deletions—until the sentence is shorter than a given threshold. We took the threshold to be 10 for the DUC task, but it is a configurable parameter. Also, given that the human-generated headlines tended to retain earlier material more often than later material, much of our iterative shortening is focused on deleting the rightmost phrasal categories until the length is below threshold.

There are four types of iterative shortening rules. The first type is a rule we call “XP-over-XP,” which is implemented as follows:

In constructions of the form [XP [XP …] …] remove the other children of the higher XP, where XP is NP, VP or S.

The motivation for this rule is that the human-produced headlines primarily include head words with short lexical modifiers, not phrasal-level modifiers. The rule is applied iteratively, from the deepest rightmost applicable node backwards, until the length threshold is reached.

The impact of XP-over-XP can be seen in these examples of NP-over-NP, VP-over-VP, and S-over-S, respectively:

(9) **Input:** A fire killed a firefighter who was fatally injured as he searched the house.

---

1 Two examples of genuinely undesirable time expression deletion are:

- The attack came on the heels of [New Year’s Day].
- [New Year’s Day] brought a foot of snow to the region.
The impact of this type of shortening can be seen in the following example:

(10) Input: Illegal fireworks injured hundreds of people and started six fires.

Parse: [S [NP [NP [VP in-

jured hundreds of people] [CC and] [VP started six fires]]]]

Output of VP-over-VP: Illegal fireworks injured hundreds of people

(11) Input: A company offering blood cholesterol tests in grocery stores says medical technology has outpaced state laws, but the state says the company doesn’t have the proper licenses.

Parse: [S [Det A] company offering blood cholesterol tests in grocery stores says [S [S medical technology has outpaced state laws], [CC but] [S [Det the] state says [Det the] company doesn’t have [Det the] proper licenses.]]]

Output of S-over-S: Company offering blood cholesterol tests in grocery store says medical technology has outpaced state laws

The third and fourth types of iterative shortening are the removal of preposed PPs and SBARs, respectively:

- Remove PPs from deepest rightmost node backward until length is below threshold.
- Remove SBARs from deepest rightmost node backward until length is below threshold.

These rules are applied with a backoff option to avoid over-trimming the parse tree. First the PP shortening rule is applied. If the threshold has been reached, no more shortening is done. However, if the threshold has not been reached, the system reverts to the parse tree as it was before any PPs were removed, and applies the SBAR shortening rule. If the threshold still has not been reached, the PP rule is applied to the result of the SBAR rule.

Other sequences of shortening rules are possible. The one above was observed to produce the best results on a 73-sentence development set of stories from the TIPSTER corpus. The intuition is that, when removing constituents from a parse tree, it’s best to remove smaller portions from each iteration, to avoid producing trees with undesirably few words. PPs tend to represent small parts of the tree while SBARs represent large parts of the tree. Thus we try to reach the threshold by removing small constituents, but if we can’t reach the threshold that way, we restore the small constituents,
remove a large constituent and resume the deletion of small constituents.

The impact of these two types of shortening can be seen in the following examples:

(13) **Input:** More oil-covered sea birds were found over the weekend.

**Parse:** [S More oil-covered sea birds were found [PP over the weekend]]

**Output of PP Removal:** More oil-covered sea birds were found over the weekend.

(14) **Input:** Visiting China Interpol chief expressed confidence in Hong Kong’s smooth transition while assuring closer cooperation after Hong Kong returns.

**Parse:** [S Visiting China Interpol chief expressed confidence in Hong Kong’s smooth transition [SBAR while assuring closer cooperation after Hong Kong returns]]

**Output of SBAR Removal:** Visiting China Interpol chief expressed confidence in Hong Kong’s smooth transition

5 Evaluation

We conducted two evaluations. One was an informal human assessment and one was formal automatic evaluation.

5.1 Bleu: Automatic Evaluation

We undertook a Bleu evaluation (Papineni et al., 2002) on the 73 articles from the TIPSTER corpus and 624 articles from the DUC2003 evaluation—comparing Hedge Trimmer to a probabilistic/HMM automatic headline generator (Zajic et al., 2002). The latter system, which was based on the concept of finding the headline most likely to have generated a given story, produced headlines consisting of words from the story, or morphological variants thereof. As reference sets we used the 146 human-generated headlines described in Section 3, and the 2496 manual abstracts prepared for the DUC2003 evaluation.

<table>
<thead>
<tr>
<th></th>
<th>TIPSTER</th>
<th>DUC2003</th>
</tr>
</thead>
<tbody>
<tr>
<td>HMM60</td>
<td>0.2612</td>
<td>0.0710</td>
</tr>
<tr>
<td>HMM28/30</td>
<td>0.2625</td>
<td>0.0729</td>
</tr>
<tr>
<td>HedgeTr</td>
<td>0.2896</td>
<td>0.0876</td>
</tr>
</tbody>
</table>

Table 1

The probabilistic system can be configured to select headline words from the first N words of the story, with the default value of N as 60. In order to make the probabilistic system more comparable to Hedge Trimmer, we ran it both with the default setting of N, and with N set to the average length of the first sentences of the test data, rounded up to the nearest integer. For TIPSTER the average first sentence length was 27.6 words, and for DUC2003 it was 29.6 words. The results are shown in Table 1.

5.2 Human Evaluation

Human evaluation indicates significantly higher scores than might be guessed from the automatic evaluation. For the 73 stories from the TIPSTER corpus, the output of Hedge Trimmer and the probabilistic system was evaluated by one human. Each headline was evaluated as GOOD, OK or BAD. Headlines were evaluated as GOOD if they were fluent, informative and correct with respect to the article. Headlines were evaluated as OK if they contained content words that correctly indicated the topic of the article. Otherwise the headline was evaluated as BAD. Table 2 shows the results of this informal evaluation.

<table>
<thead>
<tr>
<th></th>
<th>GOOD</th>
<th>OK</th>
<th>BAD</th>
</tr>
</thead>
<tbody>
<tr>
<td>HMM60</td>
<td>20</td>
<td>18</td>
<td>35</td>
</tr>
<tr>
<td>HedgeTr</td>
<td>33</td>
<td>25</td>
<td>15</td>
</tr>
</tbody>
</table>

Table 2

The types of problems exhibited by the two systems are qualitatively different. The probabilistic system is more likely to produce an ungrammatical result or omit a necessary argument, as in the examples below.

(15) **HMM60:** Nearly drowns in satisfactory condition satisfactory condition.
(16) **HMM60:** A county jail inmate who noticed.

In contrast, the parser-based system is more likely to fail by producing a grammatical but semantically useless headline.

(17) **HedgeTr:** It may not be everyone’s idea especially coming on heels.

Finally, even when both systems produce acceptable output, Hedge Trimmer usually produces headlines which are more fluent or include more useful information.

(18) **HMM60:** New Year’s eve capsizing

(19) **HedgeTr:** Sightseeing cruise boat capsized and sank.

(20) **HMM60:** hundreds of Tibetan students demonstrate in Lhasa.

(21) **HedgeTr:** Hundreds demonstrated in Lhasa demanding that Chinese authorities respect culture.

### 6 Conclusions and Future Work

We have shown the effectiveness of constructing headlines by selecting words in order from a newspaper story. The practice of selecting words from the early part of the document has been justified by analyzing the behavior of humans doing the task, and by automatic evaluation of a system operating on a similar principle.

We have compared two systems that use this basic technique, one taking a statistical approach and the other a linguistic approach. The results of the linguistically motivated approach show that we can build a working system with minimal linguistic knowledge and circumvent the need for large amounts of training data. We should be able to quickly produce a comparable system for other languages, especially in light of current multilingual initiatives that include automatic parser induction for new languages, e.g. the TIDES initiative.

We plan to enhance Hedge Trimmer by using a language model of Headlinese, the language of newspaper headlines (Märth 1980) to guide the system in which constituents to remove. Also we plan to allow for morphological variation in verbs to produce the present tense headlines typical of Headlinese.

Hedge Trimmer will be installed in a translational detection system for enhanced display of document surrogates for cross-language question answering. This system will be evaluated in up-coming iCLEF conferences.

### 7 Acknowledgements

Acknowledgements withheld to preserve anonymity.

### 8 References


---

**Note:** The text appears to be incomplete and contains a mix of numbered references and text fragments. The full context or the full text is not visible in the provided image.


