Machine Learning Approach To Augmenting News Headline Generation

Ruichao Wang
Dept. of Computer Science
University College Dublin
Ireland
rachel@ucd.ie

John Dunnion
Dept. of Computer Science
University College Dublin
Ireland
John.Dunnion@ucd.ie

Joe Carthy
Dept. of Computer Science
University College Dublin
Ireland
Joe.Carthy@ucd.ie

Abstract

In this paper, we present the HybridTrim system which uses a machine learning technique to combine linguistic, statistical and positional information to identify topic labels for headlines in a text. We compare our system with the Topiary system which, in contrast, uses a statistical learning approach to finding topic descriptors for headlines. The Topiary system, developed at the University of Maryland with BBN, was the top performing headline generation system at DUC 2004. Topiary-style headlines consist of a number of general topic labels followed by a compressed version of the lead sentence of a news story. The Topiary system uses a statistical learning approach to finding topic labels. The performance of these systems is evaluated using the ROUGE evaluation suite on the DUC 2004 news stories collection.

1 Introduction

In this paper we present an approach to headline generation for a single document. This headline generation task was added to the annual summ5arisation evaluation in the Document Understanding Conference (DUC) 2003. It was also included in the DUC 2004 evaluation plan where summary quality was automatically judged using a set of n-gram word overlap metrics called ROUGE (Lin and Hovy, 2003).

Eighteen research groups participated in the headline generation task at DUC 2004, i.e. Task 1: very short summary generation. The Topiary system was the top performing headline system at DUC 2004. It generated headlines by combining a set of topic descriptors with a compressed version of the lead sentence, e.g. ***KURDISH TURKISH SYRIA:*** *Turkey sent 10,000 troops to southeastern border*. These topic descriptors were automatically identified using a statistical approach called Unsupervised Topic Discovery (UTD) (Zajic et al., 2004). The disadvantage of this technique is that meaningful topic descriptors will only be identified if this technique is trained on the corpus containing the news stories that are to be summarised. In addition, the corpus must contain clusters of related news stories to ensure that reliable cooccurrence statistics are generated.

In this paper we compare the UTD method with an alternative topic label identifier that can be trained on an auxiliary news corpus, and observe the effect of these labels on summary quality when combined with compressed lead sentences. Our topic labeling technique works by combining linguistic and statistical information about terms using the C5.0 (Quinlan, 1998) machine learning algorithm, to predict which words in the source text should be included in the resultant gist with the compressed lead sentence. In this paper, we compare the performance of this system, HybridTrim, with the Topiary system and a number of other baseline gisting systems on a collection of news documents from the DUC 2004 corpus (DUC, 2003).

2 Topiary System

In this section, we describe the Topiary system developed at the University of Maryland with BBN Technologies. As already stated, this system was the top performing headline generation system at DUC 2004. A Topiary-style headline consists of a set of topic labels followed by a compressed version of the lead sentence. Hence, the Topiary system views headline generation as a two-step process: first, create a compressed version of the lead sentence of the source text, and second, find a set of topic descriptors that adequately describe the general topic of the news story. We will now look at each of these steps in more detail.

Dorr et al. (2003) stated that when human subjects were asked to write titles by selecting words in order of occurrence in the source text, 86.8% of these headline words occurred in the first sentence of the news story. Based on this result Dorr, Zajic and Schwartz, concluded that compressing the lead sentence was sufficient when generating titles for news stories. Consequently, their DUC 2003 system HedgeTrimmer used linguistically-motivated heuristics to remove constituents that could be eliminated from a parse tree representation of the lead sentence without affecting the factual correctness or grammaticality of the sentence. These linguistically-motivated trimming rules (Dorr et al., 2003; Zajic et al., 2004) iteratively remove constituents until a desired sentence compression rate is reached.

The compression algorithm begins by removing determiners, time expressions and other low content words. More drastic compression rules are then applied to remove larger constituents of the parse tree until the required headline length is achieved. For the DUC 2004 headline generation task systems were required to produce headlines no longer than 75 bytes, i.e. about 10 words. The following worked example helps to illustrate the sentence compression process.[1]

Lead Sentence: The U.S. space shuttle Discovery returned home this morning after astronauts successfully ended their 10-day Hubble Space telescope service mission.

Parse: (S (NP (NP The U.S. space shuttle) Discovery) (VP returned (NP home) (NP this morning)) (SBAR after (S (NP astronauts) (VP (ADVP successfully) ended (NP their 10-day Hubble Space telescope service mission)))))

1. Choose leftmost S of parse tree and remove all determiners, time expressions and low content units such as quantifiers (e.g. each, many, some), possessive pronouns (e.g. their, ours, hers) and deictics (e.g. this, tese, those):

Before: (S (NP (NP *The* U.S. space shuttle) Discovery) (VP returned (NP home) (NP *this morning*)) (SBAR after (S (NP astronauts) (VP (ADVP successfully) ended (NP their 10-day Hubble Space telescope service mission)))))

After: (S (NP (NP U.S. space shuttle) Discovery) (VP returned (NP home)) (SBAR after (S (NP astronauts) (VP (ADVP successfully) ended (NP 10-day Hubble Space telescope service mission)))))

2. The next step iteratively removes constituents until the desired length is reached. In this instance the algorithm will remove the trailing SBAR.

Before: (S (NP (NP U.S. space shuttle) Discovery) (VP returned (NP home)) *(SBAR after (S (NP astronauts) (VP (ADVP successfully) ended (NP 10-day Hubble Space telescope service mission)))))*)

After: U.S. space shuttle Discovery returned home.

Like the 'trailing SBAR' rule, the other iterative rules identify and remove non-essential relative clauses and subordinate clauses from the lead sentence. A more detailed description of these rules can be found in Dorr et al. (2003) and Zajic et al. (2004) In this example, we can see that after compression the lead sentence reads

[1] The part of speech tags in the example are explained as follows: **S** represents a simple declarative clause; **SBAR** represents a clause introduced by a (possibly empty) subordinating conjunction; **NP** is a noun phrase; **VP** is a verb phrase; **ADVP** is an adverbial phrase.

more like a headline. The readability of the sentence in this case could be further improved by replacing the past tense verb 'returned' with its present tense form; however, this refinement is not currently implemented by the Topiary system or by our implementation of this compression algorithm.

As stated earlier, a list of relevant topic words is also concatenated with this compressed sentence resulting in the final headline. The topic labels are generated by the UTD (Unsupervised Topic Discovery) algorithm (Zajic et al., 2004). This unsupervised information extraction algorithm creates a short list of useful topic labels by identifying commonly occurring words and phrases in the DUC corpus. So for each document in the corpus it identifies an initial set of important topic names for the document using a modified version of the *tf.idf* metric. Topic models are then created from these topic names using the OnTopic™ software package. The list of topic labels associated with the topic models closest in content to the source document are then added to the beginning of the compressed lead sentence produced in the previous step, resulting in a Topiary-style summary.

One of the problems with this approach is that it will only produce meaningful topic models and labels if they are generated from a corpus containing additional on-topic documents on the news story being summarised. In the next section, we explore two alternative techniques for identifying topic labels, where useful summary words are identified 'locally' by analysing the source document rather than 'globally' using the entire DUC corpus, i.e. the UTD method.

3 C5.0

C5.0 (Quinlan, 1998) is a commercial machine learning program developed by RuleQuest Research and is the successor of the widely used ID3 (Quinlan, 1983) and C4.5 (Quinlan, 1993) algorithms developed by Ross Quinlan. C5.0 is a tool for detecting patterns that delineate categories. It subsequently generates decision trees based on these patterns. A decision tree is a classifier represented as a tree structure, where each node is either a leaf node, a classification that applies to all instances that reach the leaf (Witten, 2000), or a non-leaf node, some test is carried out on a single attribute-value, with one branch and sub-tree for each possible outcome of the test. A decision tree is a powerful and popular tool for classification and prediction and can be used to classify an instance by starting at the root of the tree and moving down the tree branch until reaching a leaf node. However, a decision tree may not be very easy to understand. An important feature of C5.0 is that it can convert trees into collections of rules called rulesets. C5.0 rulesets consist of unordered collections of simple if-then rules. It is easy to read a set of rules directly from a decision tree. One rule is generated for each leaf. The antecedent of the rule includes a condition for every node on the path from the root to that leaf, and the consequent of the rule is the class assigned by the leaf. This process produces rules that are unambiguous in that the order in which they are executed is irrelevant (Witten, 2000).

C5.0 has been used for text classification in a number of research projects. For example, Akhtar et al. (2001) used C5.0 for automatically marking up XML documents, Newman et al. (2005) used it for generating multi-document summary, while Zhang et al. (2004) applied this approach to World Wide Web site summarisation.

4 HybridTrim System

The HybridTrim system uses our implementation of the Hedge Trimmer algorithm and the C5.0 (Quinlan, 1998) machine learning algorithm to create a decision tree capable of predicting which words in the source text should be included in the resultant gist.

To identify pertinent topic labels the algorithm follows a two-step process: the first step involves creating an intermediate representation of a source text, and the second involves transforming this representation into a summary text. The intermediate representation we have chosen is a set of features, that we feel are good indicators of possible 'summary words'. We focus our efforts on the content words of a document, i.e. the nouns, verbs and adjectives that occur within the document. For each occurrence of a term in a document, we calculate several features: the *tf*, or term

frequency of the word in the document; the *idf*, or inverse document frequency of the term taken from an auxiliary corpus (TDT, 2004); and the relative position of a word with respect to the start of the document in terms of word distance. We also include binary features indicating whether a word is a noun, verb or adjective and whether it occurs in a noun or proper noun phrase. The final feature is a lexical cohesion score calculated with the aid of a linguistic technique called lexical chaining. Lexical chaining is a method of clustering words in a document that are semantically similar with the aid of a thesaurus, in our case WordNet. Our chaining method identifies the following word relationship (in order of strength): repetition, synonymy, specialisation and generalisation, and part/whole relationships. Once all lexical chains have been created for a text then a score is assigned to each chained word based on the strength of the chain in which it occurs. More specifically, as shown in Equation (1), the chain strength score is the sum of each strength score assigned to each word pair in the chain.

$$Score(chain) = \sum((reps_i + reps_j) * rel(i, j)) \quad (1)$$

where $reps_i$ is the frequency of word *i* in the text, and *rel(i,j)* is a score assigned based on the strength of the relationship between word *i* and *j*. More information on the chaining process and cohesion score can be found in Doran et al. (2004a) and Stokes (2004).

Using the DUC 2003 corpus as the training data for our classifier, we then assigned each word a set of values for each of these features, which are then used with a set of gold standard human-generated summaries to train a decision tree summarisation model using the C5.0 machine learning algorithm. The DUC 2003 evaluation provides four human summaries for each document, where words in the source text that occur in these model summaries are considered to be positive training examples, while document words that do not occur in these summaries are considered to be negative examples. Further use is made of these four summaries, where the model is trained to classify a word based on its summarisation potential. More specifically, the appropriateness of a word as a summary term is determined based on the class assigned to it by the decision tree. These classes are ordered from strongest to weakest as follows: 'occurs in 4 summaries', 'occurs in 3 summaries', 'occurs in 2 summaries', 'occurs in 1 summary', 'occurs in none of the summaries'. If the classifier predicts that a word will occur in all four of the human generated summaries, then it is considered to be a more appropriate summary word than a word predicted to occur in only three of the model summaries. This resulted in a total of 103267 training cases, where 5762 instances occurred in one summary, 1791 in two, 1111 in three, 726 in four, and finally 93877 instances were negative. A decision tree classifier was then produced by the C5.0 algorithm based on this training data.

To gauge the accuracy of our decision tree topic label classifier, we used a training/test data split of 90%/10%, and found that on this test set the classifier had a precision (true positives divided by true positives and false positives) of 63% and recall (true positives divided by true positives and false negatives) of 20%.

5 Evaluation and Results

In this section we present the results of our headline generation experiments on the DUC 2004 corpus.[2] We use the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) metrics to evaluate the quality of our automatically generated headlines. In DUC 2004 task 1, participants were asked to generate very short (<=75 bytes) single-document summaries for documents on TDT-defined events.

The DUC 2004 corpus consists of 500 Associated Press and New York Times newswire documents. The headline-style summaries created by each system were evaluated against a set of human generated (or model) summaries using the ROUGE metrics. The format of the evaluation was based on six scoring metrics: ROUGE-1, ROUGE-2, ROUGE-3, ROUGE-4, ROUGE-LCS and ROUGE-W. The first four metrics are based on the average n-gram match between a set of model summaries and the system-generated summary for each document in the corpus. ROUGE-LCS calculated the longest common

[2] Details of our official DUC 2004 headline generation system can be found in Doran et al. (2004b). This system returned a list of keywords rather than 'a sentence + keywords' as a headline. It used a decision tree classifier to identify appropriate summary terms in the news story based on a number of linguistic and statistical word features.

sub-string between the system summaries and the models, and ROUGE-W is a weighted version of the LCS measure. So for all ROUGE metrics, the higher the ROUGE value the better the performance of the summarisation system, since high ROUGE scores indicate greater overlap between the system summaries and their respective models. Lin and Hovy (2003) have shown that these metrics correlated well with human judgments of summary quality, and the summarisation community is now accepting these metrics as a credible and less time-consuming alternative to manual summary evaluation. In the official DUC 2004 evaluation all summary words were stemmed before the ROUGE metrics were calculated; however, stopwords were not removed. No manual evaluation of headlines was performed.

5.1 ROUGE Evaluation Results

Table 1 shows the results of our headline generation experiments on the DUC 2004 collection. Seven systems in total took part in this evaluation, three Topiary-style headline generation systems and four baselines: the goal of our experiments was to evaluate linguistically-motivated heuristic approaches to title generation, and establish which of our alternative techniques for padding Topiary-style headlines with topic labels works best.

Since the DUC 2004 evaluation, Lin (2004) has concluded that certain ROUGE metrics correlate better with human judgments than others, depending on the summarisation task being evaluated, i.e. single document, headline, or multi-document summarisation. In the case of headline generation, Lin found that ROUGE-1, ROUGE-L and ROUGE-W scores worked best and so only these scores are included in Table 1.

	Systems	**R-1**	**R-L**	**R-W**
Combination Systems	TFTrim	0.279	0.213	0.126
	HybridTrim	0.274	0.214	0.127
	Topiary	0.249	0.20	0.119
Baseline Systems	TF	0.244	0.171	0.098
	Hybrid	0.219	0.176	0.102
	Trim	0.201	0.183	0.101
	UTD	0.159	0.130	0.078

Table 1. ROUGE scores for headline generation systems

As the results show the best performing topic labeling techniques are the TF and Hybrid systems. TF system is a baseline system that chooses high frequency content words as topic descriptors. Hybrid system is our decision tree classifier described in the previous section.

Both of these systems outperform the Topiary's UTD method. The top three performing systems in this table combine topic labels with a compressed version of the lead sentence. Comparing these results to the Trim system (that returns the reduced lead sentence only), it is clear that the addition of topic descriptors greatly improves summary quality. The performance of the baseline TFTrim system and the HybridTrim system are very similar for all Rouge metrics; however, both systems outperform the Topiary headline generator.

6 Conclusions and Future work

The results of our experiment have shown the TFTrim system (the simplest of the three Topiary-style headline generators examined in this paper) is the most appropriate headline approach because it yields high quality short summaries and, unlike the Topiary and HybridTrim systems, it requires no prior training. This is an interesting result as it shows that a simple tf weighting scheme can produce as good, if not better, topic descriptors than the statistical UTD method employed by the University of Maryland and our own statistical/linguistic approach to topic label identification.

In future work, we intend to proceed by improving the sentence compression procedure described in this paper. We are currently working on the use of term frequency information as a means of improving the performance of the Hedge Trimmer algorithm by limiting the elimination of important parse tree components during sentence compression.

References

B. Dorr, D. Zajic, and R. Schwartz. 2003. *Hedge Trimmer: A Parse-and-Trim Approach to Headline Generation*. In the Proceedings of the Document Understanding Conference (DUC).

C-Y Lin and E. Hovy. 2003. *Automatic Evaluation of Summaries using n-gram Co-occurrence Statistics*, Proceedings of HLT/NACCL.

C-Y Lin. 2004. *ROUGE: A Package for Automatic Evaluation of Summaries*, In the Proceedings of

the ACL workshop, Text Summarization Branches Out, Barcelona, Spain, pp. 56-60.

DUC. 2003. *http://www-nlpir.nist.gov/projects/duc/*, Accessed March 2005.

D. Zajic and B. Dorr and R. Schwartz. 2004. *BBN/UMD at DUC-2004: Topiary*, Proceedings of the Document Understanding Conference (DUC).

I. H. Witten and E. Frank. 2000. *Data Mining, Practical Machine Learning Tools and Techniques with Java Implementations.* Morgan Kaufann Publishers. ISBN 1-55860-552-5.

N. Stokes. 2004. *Applications of Lexical Cohesion Analysis in the Topic Detection and Tracking domain.* Ph.D. Thesis, Dept. of Computer Science, University College Dublin.

R. Quinlan. 1983. *Learning efficient classification procedures and their application to chess end games, in Machine Learning*. An Artificial Intelligence Approach edited by R.S. Michalski, J.G. Carbonell and T.M. Mitchell, Tioga, Palo Alto, CA, 1983, pp.463-482.

R. Quinlan. 1993. *C4.5: Programs for Machine Learning.* Morgan Kaufmann Publishers, San Mateo, California ISBN 1-55860-238-0.

R. Quinlan. 1998. *C5.0: An Informal Tutorial*, *http://www.rulequest.com/see5-unix.html*, Accessed March 2005.

S. Akhtar, J. Dunnion and R. Reilly. 2001. *Automating XML mark-up*, Joint International Conference of the Association for Computers and the Humanities (ACH) and the Association for Literary and Linguistic Computing (ALLC), New York.

TDT Pilot Study Corpus. 2004. *http://www.nist.gov/speech/tests/tdt.*

W. Doran, N. Stokes, J. Dunnion, and J. Carthy. 2004a. *Assessing the Impact of Lexical Chain Scoring Methods and Sentence Extraction Schemes on Summarization*. In the Proceedings of CICLing, Seoul.

W. Doran, N. Stokes, E. Newman, J. Dunnion, J. Carthy, and F. Toolan. 2004b. *News Story Gisting at University College Dublin.* In the Proceedings of the Document Understanding Conference (DUC).

Y. Zhang, N. Zincir-Heywood, E. Milios. 2004. *World Wide Web Site Summaisation.* To Appear in Web Intelligence and Agent Systems: An International Journal (The Web intelligence Consortium), 2(1), pages 39-53.