Multi-Document Summarization by Capturing the Information Users are Interested in

Elena Lloret
University of Alicante
Apdo. de Correos 99
E-03080, Alicante, Spain
elloret@dlsi.ua.es

Laura Plaza
Universidad Complutense de Madrid
C/Prof. José García Santesmases, s/n
28040 Madrid, Spain
lplazam@fdi.ucm.es

Ahmet Aker
University of Sheffield
211 Portobello
Sheffield, S1 4DP, UK
a.aker@dcs.shef.ac.uk

Abstract

This paper proposes a method for automatically generating summaries taking into account the information in which users may be interested. Our approach relies on existing model summaries from tourist sites and captures from them the type of information humans use to describe places around the world. Relational patterns are first extracted and categorized by the type of information they encode. Then, we apply them to the collection of input documents to automatically extract the most relevant sentences and build the summaries. In order to evaluate the performance of our approach, we conduct two types of evaluation. On the one hand, we use ROUGE to assess the information contained in our summaries against existing human written summaries, whereas on the other hand, we carry out a human readability evaluation. Our results indicate that our approach achieves high performance both in ROUGE and manual evaluation.

1 Introduction

The amount of information currently available is growing at an exponential rate. Information presented in different formats (text, images, audio, video) needs to be carefully processed in order to allow users to manage it efficiently and effectively. Text summarization (TS) can provide many advantages to users, since TS systems are able to generate a brief summary of one or several documents by selection and/or generalization of what is important in the source (Spärck Jones, 2007).

However, TS is an especially challenging Natural Language Processing (NLP) task, since the generation of summaries depends on a wide range of issues, such as the summarization input, output or purpose. In particular, the type of text or domain we deal with is of great importance in TS, since each domain has its particular features, and they need to be treated accordingly. For instance, when summarizing newswire text, the reader is mainly concerned about the *who*, *what*, *when*, *where* and *why* of the fact reported in the news item; when summarizing a research paper, the reader is mostly interested in the *problem* being faced, the *method* proposed to solve it and the *results* achieved. Therefore, being capable of knowing what a user would like to read in a summary will allow the summaries to be biased towards such information. The order in which this information is shown in the source documents is also important (Barzilay et al., 2002), and thus this same order should be kept in the summary. Continuing with the newswire example, the information in news articles may be presented in chronological order, in a cause-effect manner, etc., so that this logical order ensures the coherence of the text.

In this paper, we suggest an approach to automatically generate extractive summaries from a set of documents. Our approach exploits the information in existing model summaries to capture what is salient regarding a certain document type or domain (in particular, documents describing tourist places such as a church, bridge, tower or a mountain). Then, this information is used to extract the most important sentences from the input documents. Moreover, our approach also takes into consideration the order in which the information is usually presented in the model summaries and reuse this information to order sentences in the automatic summary.

2 Related Work

A great number of techniques have been proven to be effective for generating summaries automatically. Such approaches include template creation (Oakes and Paice, 1999), statistical techniques (Teng et al., 2008; Lloret and Palomar, 2009), discourse analysis (Marcu, 1999; Teufel

Proceedings of Recent Advances in Natural Language Processing, pages 77–83,
Hissar, Bulgaria, 12-14 September 2011.

and Moens, 2002), graph-based methods (Mihalcea, 2004; Plaza et al., 2008), and machine learning algorithms (Fattah and Ren, 2008; Schilder and Kondadadi, 2008).

Moreover, new scenarios, such as the generation of summaries that can be used as image captions (Aker and Gaizauskas, 2009; Plaza et al., 2010; Aker and Gaizauskas, 2010a), have recently drawn special attention in recent years. In particular, this image caption generation task has been automatically approached by analyzing image-related text from the immediate context of the image, for instance, the surrounding text in HTML documents (Mori et al., 2000; Deschacht and Moens, 2007). In these approaches, named entities and other noun phrases in the image-related text are identified and assigned to the image as captions.

Similar to these approaches, our aim is to produce summaries capable of providing a brief description for an image of an object related to the tourist domain, for instance the *Eiffel Tower*. Instead of analyzing the text surrounding the image (which may be not available), we use documents obtained from the web using the place name as query. In order to achieve this goal, we rely on the corresponding human written descriptions or summaries to capture which information a user would be interested in when describing an object of the type shown in the image. This information is extracted in the form of dependency patterns, and next used for selecting from the web-documents the most suitable sentences to appear in the summary. To our knowledge, capturing the types of information people include in human summaries via dependency patterns, and applying them on the input documents to generate automated summaries has not been previously investigated.

3 Dependency Pattern Models

Knowing the types of information humans use to describe a specific topic can help automatic procedures to produce high quality summaries about that topic. Our topics are place or object names around the world, for instance Edinburgh Zoo (see Section 3.1). We use dependency relational patterns for capturing the types of information humans include when describing them. In Section 3.2 we describe the acquisition of these relational patterns and in Section 3.3 we highlight the strategy we followed to categorize those patterns by the type of information they encode.

3.1 Data

As corpus, we use the document's collection described in Aker and Gaizauskas (2010b). This collection contains 310 images with manually assigned place names. Each image has up to 4 model summaries (932 in total) which were created manually from the information in an online social site, *VirtualTourist.com*. The summaries contain a minimum of 190 and a maximum of 210 words and are expected to contain the type of information a user wants to know about an object.

Each image in the collection was associated to the top 30 web-documents that were gathered using the Yahoo! search engine[1] and the place names as queries. We use these web-documents to generate the automated image summaries/descriptions (see Section 4).

3.2 Dependency Patterns

The model summaries were used to learn models for capturing the types of information users include in descriptions of images. To construct them we adopted the dependency relational patterns extraction described by Aker and Gaizauskas (2010a). As a result, we build what we call a *Dependency Pattern Model (DpM)*. Our patterns are derived from dependency trees. The dependency trees are obtained using the Stanford parser.[2]

First, we pre-process each model summary by applying sentence splitting, named entity tagging[3] and replacing any occurrence of a string denoting the object type (e.g. *church, bridge*) by the term "OBJECTTYPE".[4] Next, we apply the Stanford parser to parse the sentences and extract patterns where each pattern is composed of a verb and two other words being in direct or indirect relation with the verb.

For illustration consider the sentence shown in Table 1. The first two rows of the table show the original sentence and its form after named entity tagging and replacing the string denoting the object type (*bridge*) with "OBJECTTYPE". The final two rows of the table show the output of the Stanford dependency parser and the relational patterns identified for this example. For each verb identified, we extracted two further words being

[1] http://search.yahoo.com/

[2] http://nlp.stanford.edu/software/lex-parser.shtml

[3] For performing shallow text analysis the OpenNLP tools (http://opennlp.sourceforge.net/) were used.

[4] There are in total 107 object types. This list is used as a lookup when processing the sentences.

Original sentence: The bridge was built in 1876 by W. W.
Input to the parser: The OBJECTTYPE was built in DATE by W. W.
Output of the parser: *det(OBJECTTYPE-2, The-1), nsubjpass(built-4, OBJECTTYPE-2), auxpass(built-4, was-3), prep-in(built-4, DATE-6), nn(W-10, W-8), agent(built-4, W-10)*
Patterns: The OBJECTTYPE built, OBJECTTYPE was built, OBJECTTYPE built DATE, OBJECTTYPE built W, was built DATE, was built W

Table 1: Example sentence for dependency pattern.

in direct or indirect relation to the current verb. Two words are directly related if they occur in the same relational term. The verb *built-4*, for instance, is directly related to *DATE-6* because they both are in the same relational term *prep-in (built-4, DATE-6)*. Two words are indirectly related if they occur in two different terms but are linked by a word that occurs in those two terms. The verb *was-3* is, for instance, indirectly related to *OBJECTTYPE-2* because they are both in different terms but linked with *built-4* that occurs in both terms. For instance, for the term *nsubjpass (built-4, OBJECTTYPE-2)* we use the verb *built* and extract patterns based on this. *OBJECTTYPE* is in direct relation to *built* and *The* is in indirect relation to *built* through *OBJECTTYPE*. So a pattern from these relations is *The OBJECTTYPE built*. The next pattern extracted from this term is *OBJECTTYPE was built*. This pattern is based on direct relations. The verb *built* is in direct relation to *OBJECTTYPE* and also to *was*. We continue this process until we cover all direct relations with *built* resulting in two more patterns (*OBJECTTYPE built DATE* and *OBJECTTYPE built W*).

3.3 Pattern Categorization

We next categorized the relational patterns by the type of information they encode. For doing this we first performed an analysis of the human written model summaries and recorded for each sentence the kind of information it contains about the object. Then, we manually categorized this information into the following categories:

- **type**: sentences containing the "type" information of the object such as *XXX* is a *bridge*.
- **year**: sentences containing information about, for instance, when the object was built, in case of mountains, for instance, when it was first climbed.
- **location**: sentences containing information about where the object is located.
- **background**: sentences containing some general information about the object (e.g., its history).
- **surrounding**: sentences containing information about what other objects are close to the main object.
- **visiting**: sentences containing information about, e.g., visiting times, prices, etc.

We then assigned each relational pattern to one of the above categories, provided the pattern occurred five or more times in the object type corpora. In total there were 800 relational patterns that satisfied this restriction. We used three people to assign these patterns to one of the categories described above. Finally, we selected those patterns in which the three humans agreed on the same category they should belong to (400 patterns in total).

4 Generating Summaries

The proposed approach for generating summaries takes as input the set of documents describing an image's location to be summarized and the query used to retrieve them. The summaries are created in a two step process: first, several features from the document sentences are extracted, and they are used to compute different scores for each sentence (Section 4.1). Second, the sentences are assigned to the categories their patterns are associated with and ranked according to their scores. This ranking is used to analyzed different strategies for building summaries, focusing on the type of information users may be more interested in (Section 4.2).

4.1 Feature Extraction and Sentence Scoring

In the first step of our summarization approach, we propose several features and functions for scoring sentences. Given the set of documents to summarize, we first obtain the dependency patterns for each sentence along with the frequency of these patterns in the model summaries (the so called *DpM*). This information is then used to build the two following vector representations for each sentence:

- ***Binary vector (BinVec):*** A vector of six positions, each position representing one of the pattern categories described in Section 3.3. Each position gets a binary score depending on whether or not a pattern from that category is found in the sentence.
- ***Frequency vector (FreqVec):*** Each category position is set to the number of pattern occurrences in the sentence belonging to that category.

For example, the sentence *"Karnak temple is the biggest temple in Egypt owing its monumental size to 1300 years of construction"* contains the patterns *[is the OBJECTTYPE, is biggest OBJECTTYPE, is OBJECTTYPE location]* as defined in the *DpM*. The two first patterns belong to the category "type", while the third one belongs to the "location" category. Thus, this sentence is represented by the binary vector *[1 0 1 0 0 0]* and the frequency vector *[2 0 1 0 0 0]*. We next extract the following features for scoring sentences:

- ***Pattern Frequency (PattFreq):*** is the sum of occurrence frequencies of dependency patterns in *DpM* detected also in the sentence *S*, as shown in Equation 1.

$$PattFreq(S) = \sum_{p \epsilon S} FreqDpM(p) \quad (1)$$

- ***Category Frequency (CatFreq):*** is computed by multiplying each category position in the frequency vector by the number of dependency patterns in the *DpM* belonging to that category and adding these partial results, as shown in Equation 2.

$$CatFreq(S) = \sum_{i=1}^{6} FreqVec(S,i) \times FreqDpM(Cat_i) \quad (2)$$

- ***Category Occurrence (CatOcc):*** is computed in a similar fashion to *CatFreq* but using the binary vector instead of the frequency vector, as shown in Equation 3.

$$CatFreq(S) = \sum_{i=1}^{6} BinVec(S,i) \times FreqDpM(Cat_i) \quad (3)$$

- ***Object Similarity (ObjSim):*** Sentence similarity to the object being described is derived from two further similarities: ***Query Similarity (QuerySim)*** and ***Object Type Similarity (ObjTypeSim)***. *QuerySim* is calculated as the normalized cosine similarity over the vector representation of the sentence and the query. *ObjTypeSim* is a binary value indicating the presence of the object type name (e.g., "temple", "church") in the sentence. We combine these two similarities so that if both are equal to '0', then *ObjSim* is set to '0'; if only one of these similarities is higher than '0', then *ObjSim* is set to the non-zero similarity value; otherwise, if both similarities are higher than '0', *ObjSim* is set to $QuerySim \times ObjTypeSim$.

Using the previous features, we compute three different scores for each sentence. We refer to these scores as ***Pattern Frequency Score (PattFreqScore)***, ***Category Frequency Score (CatFreqScore)*** and ***Category Occurrence Score (CatOccScore)***. To obtain these scores, we multiply, respectively, the sentence values for the *PattFreq*, *CatFreq* and *CatOcc* features by the *ObjSim* feature value.

4.2 Sentence Selection

The goal of this step is to select the most relevant sentences according to what users are interested in and ordering them to build the final summary. Since the dependency patterns are grouped into six different categories of information, we can select the sentences for the summary from these categories so that we ensure that the summary covers most relevant information while reducing redundancy. We first assign each sentence to the category its patterns are associated with. Since a sentence may contain patterns from more than one category, we test two strategies for assigning sentences to categories:

- The sentence is assigned to its most frequent category (as represented in its frequency vector). If several categories present the same frequency, then the sentence is assigned to all of them. We name this strategy the ***Most Frequent Category (MostFreqCat)***.
- The sentence is assigned to all categories for which a pattern has been found in it. We refer to this strategy as ***All Categories (AllCat)***.

Using these two strategies, we generate summaries by including the best scored sentence from the category "type", then "year", then "location", then "background", then "surrounding" and then "visiting". For the categories "background" and "visiting", respectively, the top three and two sentences are included. If the summary does not reach the desired summary length, we fill the summary with additional sentences from the "background" category. The reason why we fill in the summary with "background" sentences is that they provide general information about the topic, being useful when user are interesting in additional facts about the object to be summarized. Moreover, it is worth noting that we make sure not to add to the summary any sentence that is already part of it.

5 Evaluation

According to the two sentence selection strategies and the three scores computed for each sentence (Section 4), we generated 6 different types of 200-word summaries from the documents describing each image in the corpus. Table 2 shows two examples of summaries about the Vatican Museums. The one at the top is generated following the ***All Categories*** strategy for selecting sentences after computing the ***Category Frequency Score*** for each one, whereas the second one is an example of human made summary for the same object.

We next evaluated the automatic summaries both quantitatively and qualitatively.

5.1 Quantitative Evaluation

We use ROUGE (Lin, 2004) to assess the automatic summaries in comparison to the human written ones available in the image captioning corpus. ROUGE is a well-known evaluation method for summarization which is based on the common number of n-grams between a peer and one or several model summaries. The metrics taken into consideration for this evaluation are ROUGE-1 (R-1), ROUGE-2 (R-2) and ROUGE-SU4 (R-SU4). R-1 and R-2 compute the number of unigrams and bigrams, respectively, that coincide in the automatic and model summaries. R-SU4 measures the overlap of skip-bigrams between them allowing a skip distance of 4 words.

We first evaluate the automatic summaries in order to analyze which strategy and feature is capable of obtaining the best results. These results can be seen in Table 3. A paired t-test is used to account for the statistical significance of the results with a 95% confidence interval. Then, we select the best performing approach (***AllCat-CatFreqScore***) and we set up a comparative framework with current summarization approaches that have been tested on the same data. These results are shown in Table 4. In this framework, we establish an upper bound consisting of evaluating one human written summary against the remaining human written ones for the same place name. In addition, a semantic-graph based summarizer and a statistical-based one are also used for comparison because they have been successfully tested within the image captioning domain in previous research (Plaza et al., 2010).

Summarization Approach	R-1	R-2	R-SU4
AllCat-PattFreqScore	0.39960	0.09961	0.15463
AllCat-CatFreqScore	**0.40239**	**0.10045**	**0.15600**
AllCat-CatOccScore	0.40141	0.10041	0.15555
MostFreq-PattFreqScore	0.39982	0.09897	0.15371
MostFreq-CatFreqScore	0.40103	0.09976	0.15441
MostFreq-CatOccScore	0.39869	0.09742	0.15289

Table 3: ROUGE recall results for the summaries.

Summarization Approach	R-1	R-2	R-SU4
Human	0.42083	0.11191	0.16655
AllCat-CatFreqScore	0.40239	0.10045	0.15600
Semantic-graphs	0.37971	0.08950	0.14290
Statistical summarizer	0.35875	0.08551	0.13371

Table 4: Comparison of summarization approaches (automatic vs. human summaries).

5.2 Qualitative Evaluation

We also performed a manual readability assessment of a set of 50 randomly-selected summaries from our best approach (***AllCat-CatFreqScore***). We asked three people to evaluate the summaries according to the following criteria: grammaticality, redundancy, clarity, focus and coherence, following the evaluation guidelines in DUC conferences (Dang (2006)). Then, these values were mapped into a quantitative scale where the maximum value is 5 and the lowest is 1. The average scores for each criterion are shown in Table 5. For comparison we also show the readability scores for the human written summaries of the image descriptions reported in Aker and Gaizauskas (2010b).

Criterion	AllCat-CatFreqScore	Image Descriptions
Grammaticality	4.19	4.72
Redundancy	3.74	4.92
Clarity	4.41	4.90
Focus	3.81	4.88
Coherence	3.21	4.86

Table 5: Results for the readability evaluation.

AllCat-CatFreqScore summary: The Vatican Museums (Italian: Musei Vaticani), in Viale Vaticano in Rome, inside the Vatican City, are among the greatest museums in the world, since they display works from the immense collection built up by the Roman Catholic Church throughout the centuries. The building was used as a prison until 1870, but now houses a museum. It is easy to find located across the street from the entrance to the Vatican Museum and a short walk from St Peter&'s Basilica. The closest Metro stop to the museum entrance is Cipro-Musei Vaticani near Piazza Santa Maria delle Grazie, where there is also a parking garage. The most popular areas open to tourists are the Basilica of St. Peter and the Vatican Museums. This museum is named after Pope Pius VII (whose last name was Chiaramonti before his election as pope), who founded it in the early 1800s. [. . .]

Human written summary: Not everyone who visits the Vatican is aware that it is a sovereign state and has been since 1929. The Pope rules it as Europe's only absolute monarch! It includes St. Peter's Cathedral, The Vatican Gardens, The Vatican Museums, and the famed Sistine Chapel. All of these should be on your agenda for a visit, especially the Sistine Chapel. Go early because you will, no doubt, have to stand in line. The last person to enter is at 1:00 PM. So, it's better to see it first and then see the Cathedral. Michelangelo did the ceiling for Pope Julius II, and it shows the Creation of the World and The Fall of Man. It was restored in the 1980s. [. . .]

Table 2: Examples of an automatic and a model summary fragments.

5.3 Discussion

It can be seen from Table 3 that the best approach for automatically generating summaries is the one in which the score of a sentence is computed using the category frequency, and sentence selection involves considering all categories of information that the sentence includes (*AllCat-CatFreqScore*). This strategy obtains a recall value for R-1 of 0.40239. Moreover, this value is statistically significant with respect to the other approaches except for the *AllCat-CatOccScore*. Regarding R-2 and R-SU4, this approach also achieves the best results compared to the others but the results in these cases are not statistically significant, except for *MostFreqc-CatOccScore* for R-SU4.

Concerning the comparison with other systems, our approach significantly improves the results obtained by the semantic-graphs and statistical based summarizers for all ROUGE metrics.

On the other hand, it is important to stress upon the fact that the human written summaries were generated from external sources and written following an abstractive paradigm (i.e., they include material that is not explicitly present in the source documents), whereas our proposed method is an extractive one (i.e., it selects sentences from the source documents). As a consequence, the chances to have common sentences between our summaries and the human-made ones decrease, as well as the corresponding ROUGE scores.

Regarding the readability assessment, Table 5 showed that our approach obtains close results to the human performance in Aker and Gaizauskas (2010b). However, the coherence criteria is the poorest in performance and should be improved. We plan to face this problem in the future.

6 Conclusions and Future Work

This paper presented the analysis of several approaches to automatically generate summaries from a set of documents related to tourist sites. For generating such summaries, we took into account the type of information users reflect when writing summaries of this particular domain. Therefore, we analyzed a collection of model summaries in order to determine which information would be relevant to extract from the source documents. In this manner, we performed dependency pattern identification and categorization and then used this information to suggest three score schemes to represent the sentences in the source documents, as well as two strategies for automatically assigning each sentence to a category. In order to build the final summary, sentences pertaining to each of the categories were selected in turn, taking also into account the order in which such sentences are placed in the summary. We used ROUGE for evaluating all the proposed approaches, and we also compared the performance of our summaries with the human written ones. The results obtained are very encouraging, our summaries being comparable to the human written ones. We believe that the differences of the results between our summaries and the human written ones are partly due to the manner of generating summaries. While ours were produced following an extractive paradigm which selects sentences from documents, the human written models are in fact abstracts, and this means that some of the vocabulary in them may not appear in the source documents or has been paraphrased. Furthermore, the readability evaluation also shows that our approach performs well with respect to some criteria, such as grammaticality, clarity and focus, but we have to pay special attention to the coherence of the summaries.

In the short term, it would be interesting to use the same strategy to generate summaries in other domains and analyze whether it is feasible and appropriate. Furthermore, in the long term we plan to improve our best approach by automating the pattern categorization stage. Moreover, in order to overcome the lack of coherence of the generated summaries, the benefits of anaphora resolution over the documents, as well as sentence fusion or simplification should be analyzed in the future.

Acknowledgments

This research is funded by Generalitat Valenciana (projects PROMETEO/2009/119 and ACOMP/2011/001); the Spanish Government through the FPI and FPU programs and projects TEXT-MESS (TIN2006-15265-C06-01) and TIN2009-14659-C03-01; and by the TRIPOD project supported by the European Commission (Contract No. 045335).

References

Ahmet Aker and Robert Gaizauskas. 2009. Summary generation for toponym-referenced images using object type language models. In *Proc. of the International Conference RANLP-2009*, pages 6–11.

Ahmet. Aker and Robert Gaizauskas. 2010a. Generating image descriptions using dependency relational patterns. In *Proc. of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1250–1258.

Ahmet Aker and Robert Gaizauskas. 2010b. Model summaries for location-related images. In *Proceedings of LREC 2010.*

Regina Barzilay, Noemie Elhadad, and Kathleen R. McKeown. 2002. Inferring strategies for sentence ordering in multidocument news summarization. *Journal of Artificial Intelligent Research*, 17:35–55.

H.T. Dang. 2006. Overview of DUC 2006. *National Institute of Standards and Technology.*

K. Deschacht and M.F. Moens. 2007. Text Analysis for Automatic Image Annotation. *Proc. of the 45 th ACL.*

Mohamed Abdel Fattah and Fuji Ren. 2008. Probabilistic neural network based text summarization. In *Proc. of the International Conference on Natural Language Processing and Software Engineering*, pages 43–48.

Chin-Yew Lin. 2004. ROUGE: a package for automatic evaluation of summaries. In *Proc. of ACL Text Summarization Workshop*, pages 74–81.

Elena Lloret and Manuel Palomar. 2009. A Gradual Combination of Features for Building Automatic Summarisation Systems. In *Proc. of the 12th International Conference on Text, Speech and Dialogue (TSD)*, pages 16–23.

Daniel Marcu. 1999. Discourse trees are good indicators of importance in text. In *Inderjeet Mani and Mark Maybury, editors,* Advances in Automatic Text Summarization, pages 123–136. MIT Press.

Rada Mihalcea. 2004. Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *Proc. of the ACL 2004 on Interactive poster and demonstration sessions*, page 20.

Y. Mori, H. Takahashi, and R. Oka. 2000. Automatic word assignment to images based on image division and vector quantization. In *Proc. of RIAO 2000: Content-Based Multimedia Information Access.*

Michael P. Oakes and Chris D. Paice. 1999. The automatic generation of templates for automatic abstracting. In *BCS-IRSG Annual Colloquium on IR Research.*

Laura Plaza, Alberto Díaz, and Pablo Gervás. 2008. Concept-graph based biomedical automatic summarization using ontologies. In *Proc. of the 3rd Textgraphs workshop on Graph-based Algorithms for NLP*, pages 53–56.

Laura Plaza, Elena Lloret, and Ahmet Aker. 2010. Improving automatic image captioning using text summarization techniques. In *Proc. of the 13th International Conference on Text, Speech and Dialogue*, pages 165–172.

Frank Schilder and Ravikumar Kondadadi. 2008. FastSum: Fast and accurate query-based multi-document summarization. In *Proceedings of ACL-08: HLT, Short Papers*, pages 205–208.

Karen Spärck Jones. 2007. Automatic summarising: The state of the art. *Information Processing & Management*, 43(6):1449–1481.

Zhi Teng, Ye Liu, Fuji Ren, Seiji Tsuchiya, and Fuji Ren. 2008. Single document summarization based on local topic identification and word frequency. In *Proc. of the 2008 Seventh Mexican International Conference on Artificial Intelligence*, pages 37–41.

Simone Teufel and Marc Moens. 2002. Summarizing scientific articles: experiments with relevance and rhetorical status. *Computational Linguistis*, 28(4):409–445.