

Automatic Removal of “Garbage Strings” in OCR Text: An Implementation

Kazem TAGHVA, Tom NARTKER, Allen CONDIT, and Julie BORSACK
Information Science Research Institute, University of Nevada, Las Vegas
Las Vegas, NV 89154-4021, USA

ABSTRACT

In many of our OCR/IR experiments, we found that “garbage strings” generated by OCR devices caused complications for information retrieval systems. In particular, index sizes more than doubled, formulas used to determine statistical evidence were distorted, and document ranking was affected.

To help correct these problems, we implemented a set of rules to automatically “clean up” the OCR text prior to indexing. We have applied this system, called `rmgarbage`, in several of our experiments and it is currently being applied by the Department of Energy (DOE) to their large scale Licensing Support Network (LSN) which consists primarily of OCR text.

Keywords: OCR, errors, information retrieval, garbage strings, graphic text

1. INTRODUCTION

The ability to provide easy access to electronic information has spawned a countless number of government initiatives, corporate projects, and research undertakings to make collections of documentation available, such as *The Making of America* at the University of Michigan. More often than not, populating these data sets will require the use of commercial OCR systems.

OCR, which typically encompass both automatic page segmentation and character recognition, is an accepted technology that eliminates the need for manual zoning and rekeying of perhaps millions of pages of text. We found though through our research that generalized automatic zoning is not flawless. If there is any indication that text exists in a zone, the OCR device may tag it for subsequent character recognition and try to translate zone graphics into “words.” This process results in what we call “graphic text” or “garbage strings.”

Depending on the eventual use of the OCR output, several complications become apparent. In this paper, we report on some of the effects of garbage strings, a software implementation designed at Information Science Research Insti-

tute (ISRI) that automatically removes graphic text, and then we discuss some settings where we show the usefulness of our system, `rmgarbage`.

2. COMPLICATIONS OF GARBAGE STRINGS

Most of our research at ISRI has been focused on the implications of using OCR generated text with various technologies that previously had assumed clean ASCII as input [6, 4, 5, 8]. Our most notable discovery was the fact that average precision and recall is not affected when OCR text is loaded directly into an IR system. This was a consequential result, in particular for large paper collections, because re-keying millions of pages is a tedious and expensive task. However, as our studies continued, we found that some characteristics of OCR text cause side effects that may not be readily apparent.

First, OCR devices typically do not generate formatted text. In fact, unless the intention for the text pages is solely for viewing, producing a replica of an input page would be inadequate. If you consider the complexity of formatted pages with multicolumn text, captions, inline tables, graphics, formulas and the like, you can see how the continuity of the text body is lost.

The purpose then of an OCR device is to produce usable electronic text, which in most cases, is not acceptable for display. This is particularly true when graphic text is generated. Figure 1 shows an image page with graphics; the corresponding OCR text is shown in Figure 2.

Second, if the purpose of the OCR text will be for searching, then it will be loaded into a retrieval system. Typically, for clean electronic documents, the overhead associated with a loaded collection ranges from 20% to as much as 100% of the original collection size. For our collection this percentage ballooned to nearly five fold when the OCR text was used that contained garbage strings. For large collections, this inflated size can affect a system’s reliability, efficiency, and response times and nearly all of this overhead will have no retrieval value. Table 1 gives some statistics for a probabilistic system that we used in several of our experiments [3].

Third, any system that relies on statistical data from a

T.B.M. Rock Quality Estimate

Start Station: 4585 End Station: 4590 Engineer: Ken Donnelson
 Thermo-Mechanical Unit: Tsw2 Stratigraphic Unit: Tptpmn
 Ground Support: Class 4, Ground Support
 Important Geology: Left side blocky.
 Comment: Near ss 908-912/913, Right side lagged.

Date: 03/22/96

Q Factors

RQD 90.60
 Joint Set # 6.00
 Joint Roughness # 2.10
 Joint Alteration # 2.00
 Water Reduction Factor 1.00
 Stress Reduction Factor 2.50

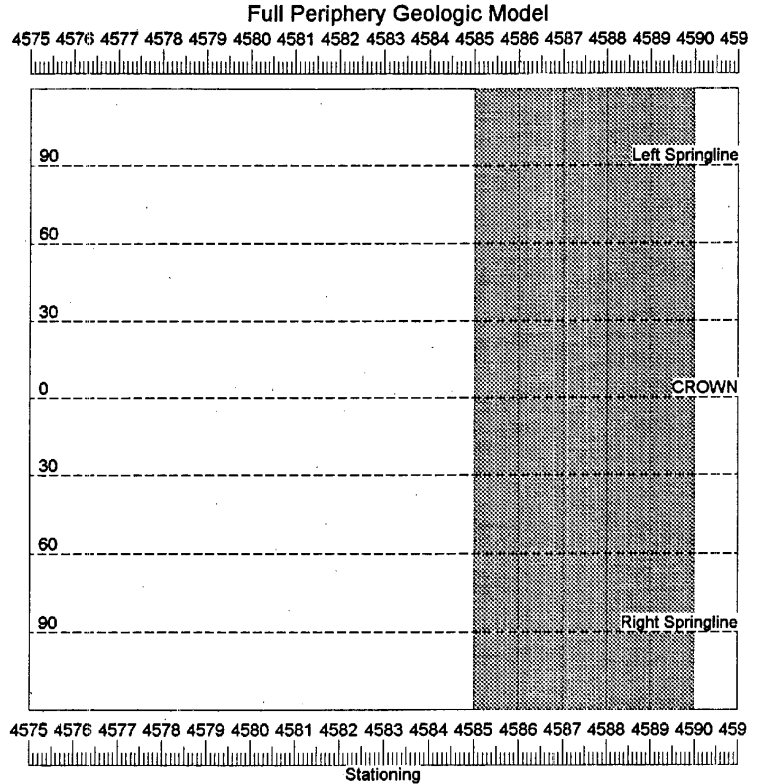
Q 6.342
 Rock Quality Fair

RMR Factors

RQD (average) 20.0
 Intact Rock Strength 12.0
 Joint Spacing 10.0
 Joint Condition 14.2
 Ground Water 15.0
 Joint Orientation Adjust. -2.0

RMR Index 69.2
 Rock Quality Good

RQD/Jn 15.1
 Jr/Ja 1.1

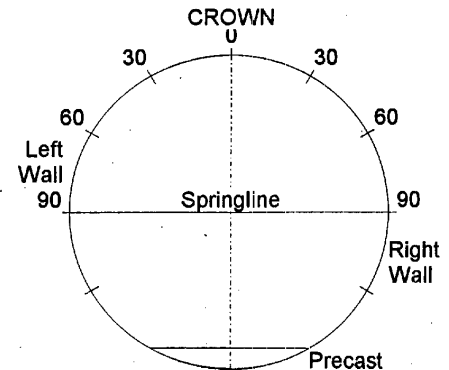


ESF Ground Support Recommendation
 Class 2 ground support (refer to drawing 40154)
 Rockbolts nominal 1m X 1m spacing w/ WWF
 (Spot bolt as necessary) Pins and channel as needed.
 Or: W6X20 steel sets spaced 1220mm w/WWF

NGI Ground Support Recommendation*

Cat	Q	RQD/Jn	Jr/Ja	Support type	Note
17	10-4	>30	-	sb (utg)	
		>=10, <=30	-	B (utg) 1-5 m	
		>10	-	B (utg) 1-5 m +S 2-3 cm	

* Barton et. al. (1974)



File: C:\TBM\MD6
 Code: TBM.EXE Ver: 4.1. Agapito Associates, Inc.

Figure 1: Page image with text and graphics

Statistic	Corrected Collection	OCR Collection
database size (bytes)	15,686,772	40,918,148
average number of terms per document	6,114	8,583
number of unique index terms	78,494	387,276

Table 1: Statistics for Corrected and OCR Collections

collection for retrieval, categorization, ranking, or other similar document manipulation task can be affected by an increase in garbage strings[3]. Again, in studies performed with a probabilistic retrieval system, we found that skewed term occurrence counts affected formulas used in determining document relevance. For example, document length is used to normalize term frequencies. As shown in Table 1, in our experiments, the OCR collection’s average document lengths increased by almost 34% when compared to the manually typed collection. Other term distribution values were also affected. Maximum document term frequency (*maxtf*), which is used in this probabilistic system to assign concepts to documents and rank a document’s relevance to queries, caused erratic results.

Many “garbage strings” are short—one or two characters in length. If a graphic border gets interpreted as a particular character (e.g., I), it may very well have the highest occurrence count in a document. The *maxtf* value impacts every document-to-term assignment and hence a document’s relevance to incoming queries. We found this same kind of variability in a vector-space system used in our experimentation.

Information retrieval systems[1, 2] are not the only systems that rely on term statistics to function properly. More recently, we have looked at the effect of OCR generated text on document categorization[8]. Like IR, we found that in general, average precision and recall are not affected in the categorization process. But we also found that garbage strings can affect a categorization system’s ability to perform its function correctly. In particular, after automatically removing almost 20,000 garbage strings with *rmgarbage* from a single category, documents were correctly categorized. For large scale categorization, locating good quality training documents can improve performance dramatically.

3. RMGARBAGE IMPLEMENTATION

For the reasons enumerated in Section 2, we believed that removing garbage strings would be an important preprocessing step. Since the objective of OCR is to *eliminate* manual intervention, it seemed futile to have to visually inspect the original image and manually cleanup garbage strings after OCR. Our solution was the development of a few generalized rules that could identify garbage strings

and remove them from the document text.

In this section, we describe the rules in our system, *rmgarbage*, and show examples of the types of strings that are removed. This system has already shown its usefulness in our experimentation and is currently being applied by the Department of Energy (DOE) to their large scale Licensing Support Network (LSN).

For the purpose of demonstration, we ran 462 pages through the most recent version (v10) of *Scansoft DevKit 2000*. We employed both automatic zoning and the available *POWR+* module which performs a voting algorithm using the Calera and Recognita-derived engines. Pages were selected because they contained both text and graphic material or tables. In this way, we give some idea of the state-of-the-art of OCR technology.

We define a *string* as a stream of ASCII characters separated by whitespace. The following rules define “garbage strings” which are automatically removed from the input file. A few examples follow each rule.

- (L) If a *string* is longer than 40 characters, it is garbage:

```
L PIIIn-9Ir11Cftmf11F1111M11W11h1ng...
L I1111111111I11111111111111111111...
L u11Lu1111~1Iu111~11~1~11nnlu111...
```

- (A) If a string’s ratio of alphanumeric characters to total characters. is less than 50%, the string is garbage:

```
A .M~y~l~ic~.I~
A _____J.~:ys~, .<F9>j}ss.
A 14.9tv="~;<F9>ia.~:..
```

- (R) If a string has 4 identical characters in a row, it is garbage:

```
R 11111111111111111111111111111111
R Pnlhrrrr
R 11111k1U1M.il.uu4ailuidtji
```

- (V) If a string has nothing but alphabetic characters, look at the number of consonants and vowels. If the number of one is less than 10% of the number of the

4. CONCLUSION

Table 2: Strings that would ordinarily be removed

Page count	462
Total Input strings	102,102
Strings removed	3,661
Erroneously removed strings	165
rmgarbage accuracy	96%

Table 3: Page Statistics

other, then the string is garbage. For example, 10 consonants and 1 vowel passes, but 11 consonants and 1 vowel would be removed as garbage:

```
V CslwWkrm  
V Tptpmn  
V Thlrld
```

- (P) Strip off the first and last characters of a string. If there are two distinct punctuation characters in the result, then the string is garbage, so a ,bc/defg is garbage, ab ,cde ,fg is not.

```
P btkvdy@us1s<F9>8  
P w.a.e~tctet~oe~  
P <F9><F9>iA,111f11w1~f111~N
```

- (C) If a string begins and ends with a lowercase letter, then if the string contains an uppercase letter anywhere in between, then it is removed as garbage:

```
C bAa  
C aepauWetelectronic  
C sUatigraphic
```

Because domain specific collections manifest certain characteristics, we have the ability to add exceptions to these rules in the form of regular expressions. So even though the strings in Table 2 would ordinarily be removed, with properly defined regular expressions for the LSN collection, they will not be removed. Further, specific expressions can also be tagged for certain removal. Table 3 gives statistics on how well rmgarbage removes garbage strings from OCR text.

Rmgarbage is actually just one module of a much more complex automatic OCR correction system we call MANICURE (Markup ANd Image-based Correction Using Rapid Editing)[7]. Note the string “sUatigraphic” that was removed because of the last rule (C). This term would likely have been first corrected to “stratigraphic” and then not removed by rmgarbage if the complete system had been run on this page.

OCR technology has improved immensely since ISRI first began its studies in the early 1990’s. The vast majority of documents can be automatically zoned and recognized and will produce usable electronic text. But as we have pointed out, there are still some concerns one should have with regard to using OCR directly.

The eventual use of the OCR output can help dictate the kind of post OCR processing that may be required. Our goals were specific to the needs of the LSN. Their requirements were to ensure that superfluous text (garbage strings) was kept in check and that all relevant documents were retrievable. Since images were being used for display, clean formatted text was not essential. Rmgarbage and the other modules of the MANICURE system improve the OCR text to meet these requirements.

5. REFERENCES

- [1] J. P. Callan, W. B. Croft, and S. M. Harding. The IN-QUERY retrieval system. In *Proc. 3rd Intl. Conf. on Database and Expert Systems Applications*, pages 78–83, 1992.
- [2] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Inf. Proc. and Management*, 24(5):513–523, 1988.
- [3] Kazem Taghva, Julie Borsack, and Allen Condit. Results of applying probabilistic IR to OCR text. In *Proc. 17th Intl. ACM/SIGIR Conf. on Research and Development in Information Retrieval*, pages 202–211, Dublin, Ireland, July 1994.
- [4] Kazem Taghva, Julie Borsack, and Allen Condit. Effects of OCR errors on ranking and feedback using the vector space model. *Inf. Proc. and Management*, 32(3):317–327, 1996.
- [5] Kazem Taghva, Julie Borsack, and Allen Condit. Evaluation of model-based retrieval effectiveness with OCR text. *ACM Transactions on Information Systems*, 14(1):64–93, January 1996.
- [6] Kazem Taghva, Julie Borsack, Allen Condit, and Srinivas Erva. The effects of noisy data on text retrieval. *J. American Soc. for Inf. Sci.*, 45(1):50–58, January 1994.
- [7] Kazem Taghva, Allen Condit, Julie Borsack, John Kilburg, Changshi Wu, and Jeff Gilbreth. The MANICURE document processing system. In *Proc. IS&T/SPIE 1998 Intl. Symp. on Electronic Imaging Science and Technology*, San Jose, CA, January 1998.
- [8] Kazem Taghva, Thomas A. Nartker, and Julie Borsack. Recognize, categorize, and retrieve. In *Proc. of the*

Symposium on Document Image Understanding Technology, pages 227–232, Columbia, MD, April 2001.
Laboratory for Language and Media Processing, University of Maryland.