

# Automatic Reformatting of OCR Text from Biomedical Journal Articles

Glenn M. Ford, Susan E. Hauser, George R. Thoma

National Library of Medicine

Bethesda, Maryland 20894

## Abstract

*The goal of the Medical Article Record System (MARS), being developed by the National Library of Medicine, is to reduce the manual keyboard entry of bibliographic citation fields for the MEDLINE database by automatically identifying and converting information from bit-mapped images of biomedical journal article pages to ASCII data. An important element of this automatic conversion requires reformatting the title, author and affiliation fields from the output of the Optical Character Recognition (OCR) process in MARS to the formats specified by MEDLINE conventions. This paper outlines the methods developed to implement the reformatting process.*

## 1 Introduction

The MARS system in its first version now operating at the National Library of Medicine automatically extracts article abstracts from the bitmapped images of journal articles, but relies on the manual keyboard entry of all the other fields required in the MEDLINE database. A second generation MARS system is being designed to automate the entry of other fields, focussing primarily at present on the article title, author names and their institutional or organizational affiliation. Following the scanning stage, the OCR system converts the image contents to text, and algorithms segment the page image (autozoning), and automatically label the zones. Reformatting follows the zone labeling stage so that the zone contents adhere to MEDLINE's syntactic conventions.

### 1.1 Institutional Affiliations

Institutional affiliations of the authors are reformatted by finding the best match between the OCR text and a list of about 130,000 correctly formatted affiliations obtained from the current production version of MARS. Simple string matching is not promising because of the myriad arrangements in which affiliations can be

expressed. Most journals show the affiliations of all authors, but by convention only the affiliation of the first author is entered into MEDLINE. However, the text string corresponding to the first affiliation may be scattered throughout the OCR text for the affiliation field. As an example, when multiple authors are affiliated with different departments within the same institution, the printed affiliation may be "Department A, Department B, Department C, Institution XYZ," while the correct MEDLINE entry is "Department A, Institution XYZ." The problem is further confounded by OCR errors, especially errors in detecting superscripts and subscripts. To find a match, the entire OCR text of the affiliation field is compared with every entry in the list of existing affiliations. A matching score for each of the existing affiliations is calculated on the basis of partial token matches, distance between token matches and customized soundex matching. The three highest scoring candidates are presented to the "reconcile" (verification) operator for selection. In preliminary tests, our current version of affiliation field reformatting successfully identifies the correct affiliation over 80% of the time when the affiliation is represented in the list. This success rate is expected to improve with parallel efforts to reduce OCR errors and the expansion of the list of affiliations from ongoing production data.

### 1.2 Article Titles and Authors

The reformatting of author and title fields is implemented by predefined rules. Based on journal title and field identification (author or title), the software selects a subset of rules from the inclusive set of all rules. The selected rule set and the OCR text are passed to the implementation algorithm. As each rule is applied, the OCR string is modified. Rules for title fields involve initial-letter capitalization and all-letter capitalization. Rules for author fields include characters used to delimit authors in a multiple-author list; tokens to be removed, such as Ph.D.; tokens to be converted, such as II to 2<sup>nd</sup>; and particles to be retained, such as "van." For example, Eric S. Van Bueron, Ph.D.

becomes Van Bueron ES. Our preliminary version of title and author reformatting correctly reformats more than 97% of the authors and titles from a test set of 1857 processed articles. We expect performance to improve with the addition of rules derived from production data.

## 2 Reformatting the Author field

Reformatting the author field uses *forward chaining*<sup>1</sup> rules based deduction. The reformat module can have many rules defined for a particular field. Each rule has a number of requirements among which are that it must

- Be associated with a specific ISSN number (Journal Title)
- Fall into one of eight categories. The categories are pre-defined in the reformat module and are required to help in our conflict resolution strategy, which in our case is *specificity ordering*. Whenever the conditions of one triggering rule is a superset of another rule, the superset rule takes precedence in that it deals with more specific situations. An example of this is shown later. The eight categories and examples are listed in Table 1 shown in Appendix A.

The example column in Table 1 shows the complete reformatted field. Note that a single rule or category does not necessarily complete the reformatting, but may need to be combined to achieve correct reformatting of the author field.

With the eight categories defined, the first step in using the reformat module for a given ISSN is to define which rules are appropriate for a particular ISSN (or journal title), since the printed format varies widely among journals. As an example, in one journal the authors appear as:

Glenn M Ford, MD, John Smith, PhD,  
and John Glover

This can be difficult to parse with a default set of rules, such as ', and' and ', ' so that other rules need to be defined. By defining, in the database, the rules for a specific Journal Title over a

specific period<sup>2</sup> of time we can customize the rules to work for unusual or specific cases.

The above example fails in the default rule set that only has ', and' and ', ' as the Author Delimiter because this would incorrectly identify 'MD' and 'PhD' as author names. To accommodate this journal (and others like it) a high priority rule trigger list was created for Author Delimiters such as ', MD', ', PhD', 'Mr.', 'Dr.', and other formal titles.

To avoid conflict among rules each word chain is passed through all the categories recursively until no more rules are triggered. As long as we have an antecedent with consequences we continue to process the word chain. Using the forwarding chaining method, when an if statement is observed to match an assertion, the antecedent (i.e., an if statement) is satisfied. When the entire set of "if statements" are satisfied, the rule is triggered. Each rule that is triggered establishes, in a working memory node, that it was executed. During conflict resolution the reformat module decides which rules take priority over others via specificity ordering. An example would be:

Reduce category executes on 'John Smith II' and makes this 'J S II'

Convert category executes 'John Smith II' and marks Smith as convert pre-word and 'II' to '2<sup>nd</sup>'.

Our conflict resolution specifies that the convert category is more specific than the reduce category, thus keeping the word 'Smith' and '2<sup>nd</sup>'. In addition, the pre-word convert flag in this particular example signals the conflict resolution manager to keep 'Smith', initialize 'J', and append '2<sup>nd</sup>'. This is possible because we have retained our original text and the converted text. The text did not change and an integrated rule has informed us that the word 'Smith' remained the same and by examining all words, we deduce that this is the last name.

Example Before/After:  
Before - John Smith II  
After - Smith J 2nd

<sup>1</sup> Forward Chaining is the logical construct in which the number of conclusions reached is small, but the number of ways to reach a particular conclusion is large.

<sup>2</sup> Journals often change formats over the years to accommodate new publishers or printers. Therefore the rules may need to change even though the Journal Title remains the same.

The conflict resolution strategy at the category level is that of specificity ordering. There is also a conflict resolution strategy within a given category: priority list rule ordering. Rules within a given category are assigned a priority level to avoid conflicts. An example of this is the following:

Glenn Ford, John Smith, and David Wells

We have the following Author Delimiter rules defined

' and ', and'

However, the ',' is assigned priority 1, and the ', and' is assigned higher priority 2. If we did not give a higher priority to ', and' we could end up with 'and' as part of the author name or create a null value.

In our latest ground truth testing of the author reformat rules system we tested 1857 authors from OCR data. Of those 1857, 41 were reformatted incorrectly, for a 97.29% correction rate. Of those 41, all 41 were missing rules defined for a given case. An example of a missing rule is given in the case of an author field that reads:

Glenn M. Ford, Jr., John Smith.

By adding the rule [' , Jr. ' Author Delimiter priority 2] to our test set, with just a new rule created and no changes in code required, we achieved 100% correct reformatting in the test set.

### 3 Reformatting the Affiliation field

The reformatting strategy for the affiliation field is quite different from the above. The OCR data for an affiliation field could contain many affiliations, since each author may have a different affiliation. This data is often difficult to reformat. One reason is that only the affiliation of the first author is to be retained, in line with MEDLINE conventions. Another reason is that the desired data is spread out over the entire field and not contiguous. For example, in a 30 word affiliation zone, we may only want to retain words 1-8, 12-14, and word 30. Our method is to do probability matching to historical data of ~130,000 unique affiliations collected to date.

The first step is to read all these unique affiliations into memory and create a Ternary Search Tree [1, 4] for each affiliation. We then create a soundex word list [2, 3] for each affiliation.

When a zone is identified at the labeling stage as an affiliation field, the OCR data is first processed through a partial-matching algorithm. Low confidence characters are replaced with wildcards.

Example: Uniuersity. The 'u' is actually a 'v' but the OCR engine assigned it as a 'u' with a low confidence level. The partial match algorithm replaces the 'u' with a '.' signifying that this character is a wildcard, and that any word in our search tree that has the pattern Uni<any letter>ersity is considered to be a match.

The first step is to determine if a word in the affiliation zone matches one in the affiliation list. Ignoring implemented performance optimizations<sup>3</sup> we perform a partial word match for all the words in the OCR list and build up a chain of those words that do match. We also track distances between chains.

Consider the example of trying to find the affiliation "Department of Computer Science, University of Maryland" in the affiliation list. The OCR input string looks like: "Department of Computer Science, Department of Engineering, University of Maryland, Department of Computer Science, Johns Hopkins University."

Since only the first affiliation is to be retained, there is considerable data that is irrelevant. The problem is to retrieve just the data needed. By word chaining we can find chains of words that exist in both the OCR text and in an affiliation zone and then use these to derive weighted probabilities.

In this example there is a chain of 4 words that match, followed by 3 that do not match, followed by 3 more that match, and finally 7 that do not. Our probability algorithms compute chain word matches and distances between chained words.

---

<sup>3</sup> Optimizations such as: if the first word does not exist in the affiliation listing entry 1, go to entry 2 instead of looking at every OCR word.

The next step in our process reverses the partial word match. The ~130,000 affiliations are matched to the OCR affiliation.

Using the same example, "Department of Computer Science, University of Maryland" has 7 words and all 7 occur in our OCR word list. It is likely there is another affiliation entry that looks like "Department of Computer Science, University of Delaware". This would give a high match of 6/7 words. By comparing and weighting word matches from OCR to Corrected Affiliation and Corrected Affiliation to OCR, and using information such as the number of words matched, total number of words, chain of words matched, and chain of words unmatched, we arrive at a probability between 0 and 1. Note that partial matching is used to help cover OCR errors that would ruin a literal string pattern matching as the affiliation field is often in a smaller font and might incur higher than normal OCR error rates.

In addition to a partial match search algorithm, a soundex algorithm is used with the addition of OCR substitution. For the example in which "Uniuersity" has the 'u' as low confidence, a substitution table developed lists common OCR errors where a u == v == y. All three letters are substituted in the low confidence 'u' position, and if a word matches with a soundex hash it counts as a match.

In our ground truth testing with affiliation zones, if the OCR affiliation exists in our affiliation list of 130,000 entries, the probability that the affiliation match is the correct one is 88%. The affiliation reformat module picks the top 5 candidates which are presented to the reconcile operator who can choose the correct one in the 5, or pick the nearest match and type in any missing data, usually a room number, zip code, or an email address.

## Appendix A

Table 1: Categories of Author Reformat Rules

Category	Description	Example
Particle Name	Many names contain "particles" forming an integral part of the family name and possibly bearing significance to the family. A particle is retained as part of the reformatted author name.	<i>Etienne du Vivier</i> becomes <i>du Vivier E</i> , where 'du' is a particle and is retained as is and preceding the last name Vivier. The first name is initialized.
Compound	Compound family names are preserved in the form given and are often difficult to detect. We	<i>L.G. Huis in 't Veld</i> becomes <i>Huis in 't Veld LG</i>

## 4 Reformatting the Article Title field

The title field uses the same principles as in the author rules system, but requires very few rules or categories. Of the 8 categories mentioned in the author reformat section, only 3 are used: Uppercase, Lowercase and First Letter Upper.

## 5 Current Work

Current research focuses on the correct detection of superscripts in both the author and affiliation fields to help improve reformatting algorithms. With this information available, correct affiliation matching is expected to reach the middle 90 percent range.

## 6 Summary

This paper has described the field reformatting stage in the automated data entry process being designed at the National Library of Medicine. The rules and rule categories applicable to reformatting the author, title and affiliation fields have been given.

## References

- [1] Bentley JL, Sedgewick R. Fast algorithms for sorting and searching strings. Proc. 8<sup>th</sup> Annual ACM-SIAM Symposium on Discrete Algorithms. Jan 1997.
- [2] Baase S. Computer Algorithms, Addison-Wesley, 1988, pp 242-4.
- [3] Hall PAV, Dowling GR. Approximate string matching. ACM Computing Surveys (1980).12:381-402.
- [4] Bentley J., Sedgewick B. Ternary Search Trees. Dr. Dobb's Journal, April 1998, pp 20-25

	use a mix of rules to deduce it as a compound name. Most compound names use a hyphen. Those that don't can often use particle name rules to help preserve the compound name.	<i>H.G. Huigbregtse-Meyerink</i> becomes <i>HuigBregtse-Meyerink</i> <i>HG</i>
Convert	Convert is a broad category that deals with general requirements to convert one pattern of text with another.	James A. Smith IV becomes Smith JA 4 <sup>th</sup>
Religious	Religious titles include Mother, Sister, Father, Brother. Names with surnames are handled differently from those that have no surnames.	Surname example: <i>Sister Mary Hilda Miley</i> becomes <i>Miley MH</i>  No-Surname example: <i>Sister May Hilda</i> becomes <i>Mary Hilda Sister</i> For translated articles, e.g., from the French, <i>Soeur</i> becomes <i>Sister</i> .
Reduce	Reduction rules cover the elimination of text with a single author name. It also handles the Reduction of a person's given name and marking of the Surname if present.	<i>Mr. John Smith</i> becomes <i>Smith J</i>  <i>John Smith MD</i> becomes <i>Smith J</i>
Lowercase	Some fields present all data uppercase. This rule simply converts to lower case all text that is uppercase.	JOHN SMITH becomes <i>Smith J</i>
First Letter Upper	Title and Author at times will require that the first letter of a specific word be uppercased, depending on other rules.	JOHN SMITH becomes <i>Smith J</i>
Author Delimiter	Many articles are by multiple authors who contributed to the paper, such as this one. This rule takes an OCR stream of text and creates a word list, a chain of words, and delimits where a particular author begins and ends in the complete chain of words.	Example 1: <i>Glenn M Ford, John Smith</i> becomes: <i>Ford GM</i> <i>Smith J</i> (, is the delimiter here)  Example 2: <i>Glenn M. Ford, John Smith, and Susan O'Malley</i> becomes: <i>Ford GM</i> <i>Smith J</i> <i>O'Malley S</i> (', and' is the trigger, which must precede in priority ';' as a triggered rule)