A System for Automated Extraction of Metadata from Scanned Documents using Layout Recognition and String Pattern Search Models

Dharitri Misra, Siyuan Chen, George R. Thoma, National Library of Medicine, Bethesda, Maryland

Abstract

One of the most expensive aspects of archiving digital documents is the manual acquisition of context-sensitive metadata useful for the subsequent discovery of, and access to, the archived items. For certain types of textual documents, such as journal articles, pamphlets, official government records, etc., where the metadata is contained within the body of the documents, a cost effective method is to identify and extract the metadata in an automated way, applying machine learning and string pattern search techniques.

At the U. S. National Library of Medicine (NLM) we have developed an automated metadata extraction (AME) system that employs layout classification and recognition models with a metadata pattern search model for a text corpus with structured or semi-structured information. A combination of Support Vector Machine and Hidden Markov Model is used to create the layout recognition models from a training set of the corpus, following which a rule-based metadata search model is used to extract the embedded metadata by analyzing the string patterns within and surrounding each field in the recognized layouts.

In this paper, we describe the design of our AME system, with focus on the metadata search model. We present the extraction results for a historic collection from the Food and Drug Administration, and outline how the system may be adapted for similar collections. Finally, we discuss some ongoing enhancements to our AME system.

Introduction

An effective technique for extraction of metadata from homogeneous digitized collections, or heterogeneous collections with a small number of text layouts, is to automate the process through machine learning techniques by developing classification models for individual layouts. From the contents of a classified and segmented document, metadata is extracted by searching for designated string patterns using different techniques [1][2]. This metadata may then be used for discovering records of interest through standard text search, or by browsing/searching individual metadata fields, after the collection is archived.

There are several well-known learning models such as the Naïve Bayes model (NB), the Support Vector Machine (SVM), and the Hidden Markov Model (HMM) [3][4] used for recognizing document layouts through classifying textlines in individual pages of a collection. The model parameters are trained with a training set in which individual lines are classified manually. This procedure can be applied to different collections by providing corresponding training sets.

The procedure to retrieve individual metadata fields from the segmented text, generated using the layout models, is not always simple. Often, search patterns need to be identified and applied using programmed instructions to locate and extract the metadata fields for each collection [5]. For complex cases, this is not only cumbersome but also error-prone, requiring a high degree of manual intervention and frequent program updates. Search procedures for individual collections are difficult to generalize across collections, and programmatic approaches may not easily accommodate new search criteria. A generalized rule-based search technique, adaptable for individual collections, offers a promising alternative for classifiable documents.

As part of a Digital Preservation project, we have developed an Automated Metadata Extraction (AME) system to extract metadata from digitized collections by combining the automated layout classification technique with a rule-based search technique. We generate recognition models for different document layouts based on a combination of Support Vector Machine and Hidden Markov Model [6]; and then use a metadata search model with encoded search/extraction rules to retrieve the field values from various text segments (page header, item header, item body, etc.) identified by the recognition models.

The metadata search model is initially created by incorporating a set of search rules determined by manual analysis of sample document pages. This model is improved iteratively by applying it to extract metadata from sample batches and then observing the outputs, which include a list of the of metadata fields that could not be identified for each item in a batch.

In order to accommodate complex cases that cannot be easily expressed using our search rules, the AME system allows incorporating specialized logic as a supplement to rule-based extraction. In addition, it allows for collection-level post-processing of metadata. Finally, a GUI is provided for review and manual correction of extracted metadata.

In the following sections of this paper, we provide a description of our AME system, its workflow, and details of the metadata search model. We then present the results of metadata extraction for a collection of documents from the Food and Drug Administration. Finally we discuss the customization of the model for different collections, and ongoing enhancements to the AME system.

AME System Description

Our Java-based Automated Metadata Extraction system may be used in a stand-alone mode to extract, review and store metadata in XML format from OCR'ed texts of a scanned document collection. It may also be used as a library (Jar file), to be integrated with an application, where metadata for extracted items are provided in binary format through Java API calls. In the following sections, we discuss the stand-alone AME system.

The "OCR output" mentioned in the following sections refers to character-level features (such as geometric coordinates and font attributes) generated by the FineReader 8.0 OCR engine [7] from the scanned TIFF images of a corpus, and later formatted as a text file by a separate AME application. An "item" refers to an entity, such as an article or an official record, which may be archived and accessed as a unit.

Components

The AME System consists of four main components, whose functions are described below:

- **Layout Recognition Model Trainer:** Creates the Layout recognition model by generating a set of truth files from an OCR'ed training set for each layout. Manually assigned line classifications of the truth files are used to train the model, created by zoning and labeling the textlines using a combination of SVM and HMM.
- **Metadata Search Model Generator:** Generates a binary search model for a collection, incorporating the extraction rules for each metadata field in individual layouts in the form of a set of search rules and attributes. The input rules to this module may be provided as an XML file, or may be encoded into a Java source module.

  [Note that the search model may be updated any time during operations, but may require regressive testing for already-processed documents, if new documents require major changes to existing search rules.]
- **Metadata Extractor:** Extracts and stores metadata for individual items from OCR output, using a specified XML schema. It employs a Metadata Search Engine module to perform actual pattern search using the metadata search rules in the model. The Extractor also generates a searchable text file for each extracted item, and an accompanying statistics file indicating the metadata fields that are missing in each item.
- **Metadata Validator**: This GUI module is used by the curator or operator to review/edit metadata fields in a set of items, and save the validated data. It is also used in the initial phase of the process to identify additional search patterns and update the metadata search model.

Metadata Extraction Workflow

Metadata extraction for a corpus starts after the metadata recognition models are generated and an optimal metadata search model is created for that corpus. The OCR'ed pages, along with the original TIFF files, are submitted to the Metadata Extractor to identify individual items in the set, and then to extract their metadata. Each submission may consist of a few to several hundred sequential pages. The metadata extraction workflow for a submitted batch is illustrated in Figure 1.

Individual items within the batch are identified by classifying the text lines using the metadata recognition models. The textual data corresponding to each identified item is first corrected automatically for OCR errors, and then submitted to the Metadata Search Engine. A post-processing step is performed to apply any collection-specific logic or other cleanup for the extracted values. Associated information, such as the item's text (consisting of a part of a page to several pages), and the extraction statistics for the batch are also output at this point. Metadata review/validation may be performed as an optional manual step after all items are extracted, or the data may be stored on the disk for later review.

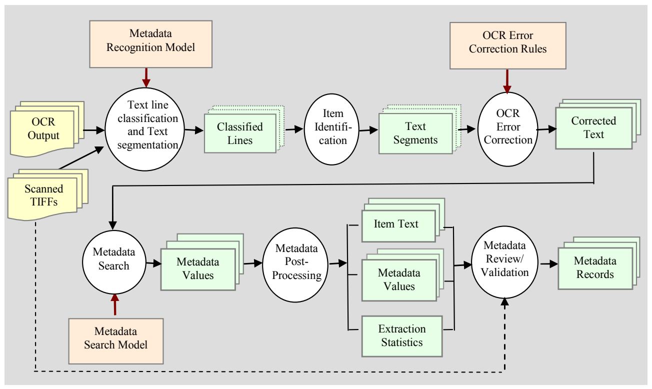


***Figure 1.** Metadata Extraction Processing Flow*

Metadata Search Model

The structure of the Metadata Search model is shown in Figure 2. The components, in reverse hierarchy, are: *SearchPattern*, *SearchRule*, and *ExtractionRule*, each of which is specified as a node in the XML document and instantiated as a Java object during processing. Each *ExtractionRule* applies to a specific metadata field in one or more layouts of the collection. A *SearchPattern* is an abstract class (shown as a dotted box); its derived classes are: *TaggedSearchPattern*, *LineClass-SearchPattern*, *TextSearchPattern,* and *DelimitedSearchPattern*, each of which corresponds to a specific "search type" attribute.

It may be noted that each *SearchRule* encapsulates only one *SearchPattern*, whereas an *ExtractionRule* may contain several *SearchRules*, which are processed according to their specified search order in the model.

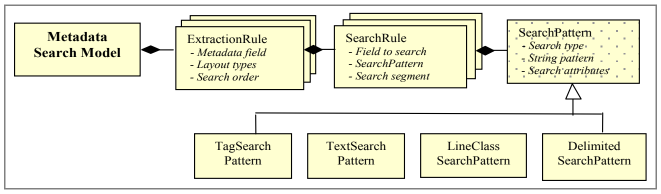


***Figure 2.** Metadata Search Model*

The XML structure of an ExtractionRule (with a single SearchRule and SearchPattern) is shown in the boxed text below. Uppercase notations of the attribute values indicate string or numeric constants, which are used by the Metadata Search Engine. The value "FIELD", for example, refers to one of the known metadata fields for the corpus.

SearchPattern Attributes

Table 1 shows the general attributes of a SearchPattern (shown as node attributes or child nodes in the XML representation) that are applied in locating and retrieving the metadata field in the specified section of the text. Note that the pattern assigned to a string refers to a Regular Expression represented as a Java String pattern [8].

Extraction Rule Structure

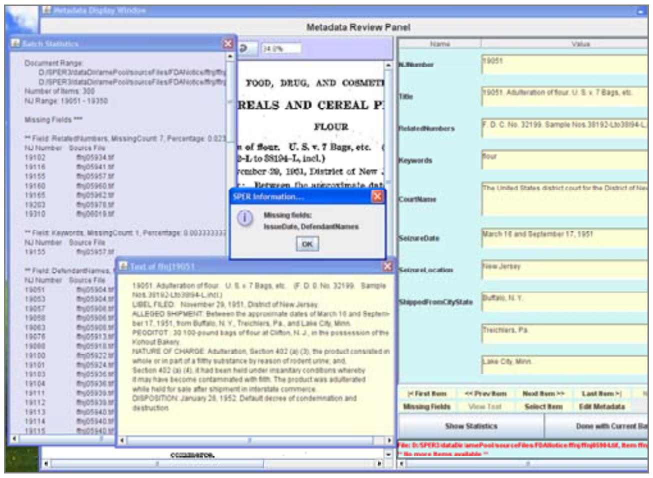
```
<ExtractionRule id="e1" field="FIELD"
          layouts="L1, L2"
          searchOrder="SGTYPE1, SGTYPE2" >
  <SearchRule id="sr1" field="FIELD"
      segment="SGTYPE1" seachPatternId="sp1">
    <SearchPattern id="sp1" searchType="SRCHTYPE"
           fieldType="FLDTYPE" multiValued="TRUE" >
        <patternsToMatch string="matchingPattern"/>
        <cueWord  string="cuePattern" />
        <beginPatterns string="bpattern1" />
        <bpUse skip="TRUE" />
        <endPatterns string="epattern1" />
        <epUse term="TERMTYPE" />
        <cleanupPatterns errorPattern="errorP1"
              correctPattern="correct1" />
    </SearchPattern>
  </SearchRule>
</ExtractionRule>
```

Table 1 – General Attributes of a SearchPattern

Attribute	Meaning
fieldType	Type of field being searched (Text, Date, City/State, number…)
multiValued	one or more values to search for
patternsToMatch	Actual pattern(s) that should match the search
cueWord	Pattern whose occurrence indicates the presence of a search pattern in the text segment
beginPatterns	One or more patterns indicating the beginning of a segment for doing a match
bpUse	skip=TRUE means the search for the field should start after skipping the begin pattern
endPatterns	Pattern(s) to terminate or follow the matched text
epUse	Indicates how the end pattern should be used to delimit the matched text (end/split/discard etc.)
cleanupPatterns	Additional (error) patterns, if any, and their substitution values, to be used for a specific field search

Iterative Improvements to the Search Model

It may not always be practical to determine all the search rules and patterns for each metadata field before metadata extraction starts. Hence, the AME system enables an operator to update the search model through analysis of the missing fields in the extracted items using a graphical display window. A visual comparison of the scanned image text, OCR interpreted text and metadata values of an item, as shown in Figure 3, reveals missing search patterns that should be added to the model, as well as additional OCR corrections patterns. The manually updated model may be re-used for the same set of documents until most errors are removed; final corrections to the metadata may be applied manually if necessary.



Document Range:
D:/SPER3/dataDir/amePool/sourceFiles/FDANotice/ffnj/ffnj
D:/SPER3/dataDir/amePool/sourceFiles/FDANotice/ffnj/ffnj
Number of Items: 300
NJ Range: 19051 - 19350

Missing Fields ***

** Field: RelatedNumbers, MissingCount: 7, Percentage: 0.023

NJ Number	Source File
19102	ffnj05934.tif
19116	ffnj05941.tif
19155	ffnj05957.tif
19160	ffnj05960.tif
19165	ffnj05962.tif
19203	ffnj05978.tif
19310	ffnj06019.tif

** Field: Keywords, MissingCount: 1, Percentage: 0.003333333

NJ Number	Source File
19155	ffnj05957.tif

** Field: DefendantNames,

NJ Number	Source File
19051	ffnj05904.tif
19053	ffnj05904.tif
19057	ffnj05906.tif
19058	ffnj05906.tif
19063	ffnj05908.tif
19076	ffnj05913.tif
19088	ffnj05918.tif
19100	ffnj05922.tif
19101	ffnj05924.tif
19103	ffnj05936.tif
19104	ffnj05936.tif
19111	ffnj05939.tif
19112	ffnj05939.tif
19113	ffnj05940.tif
19114	ffnj05940.tif
19115	ffnj05940.tif

FOOD, DRUG, AND COSMETI

REALS AND CEREAL P

FLOUR

n of flour. U. S. v. 7 Bags, etc. (
2-L to 38194-L, incl.)
vember 29, 1951, District of New
: Between the approximate dat

commerce.

Missing fields:
IssueDate, DefendantNames

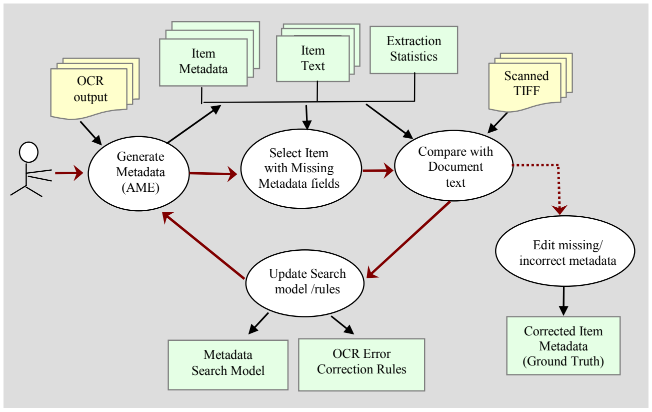
19051. Adulteration of flour. U. S. v. 7 Bags, etc. (F. D. C. No. 32199. Sample Nos. 38192-L to 38194-L, incl.)
LIBEL FILED: November 29, 1951, District of New Jersey.
ALLEGED SHIPMENT: Between the approximate dates of March 16 and September 17, 1951, from Buffalo, N. Y., Treichlers, Pa., and Lake City, Minn.
PRODUCT: 30 100-pound bags of flour at Clifton, N. J., in the possession of the Kohout Bakery.
NATURE OF CHARGE: Adulteration, Section 402 (a) (3), the product consisted in whole or in part of a filthy substance by reason of rodent urine; and, Section 402 (a) (4), it had been held under insanitary conditions whereby it may have become contaminated with filth. The product was adulterated while held for sale after shipment in interstate commerce.
DISPOSITION: January 28, 1952. Default decree of condemnation and destruction.

Name	Value
NJNumber	19051
Title	19051. Adulteration of flour. U. S. v. 7 Bags, etc.
RelatedNumbers	F. D. C. No. 32199. Sample Nos. 38192-L to 38194-L
Keywords	flour
CourtName	The United States district court for the District of Ne
SeizureDate	March 16 and September 17, 1951
SeizureLocation	New Jersey
ShippedFromCityState	Buffalo, N. Y.
	Treichlers, Pa.
	Lake City, Minn.

File: D:/SPER3/dataDir/amePool/sourceFiles/FDANotice/ffnj/ffnj05904.tif, Item ffn
** No more Items available **

Figure 3. Metadata display and correction screen

The validated metadata records are then stored as the *ground truth* for those pages for regression testing, and may also be made available to archive the corresponding items. This iterative process is shown graphically in Figure 4.



OCR output
Item Metadata
Item Text
Extraction Statistics
Scanned TIFF
Generate Metadata (AME)
Select Item with Missing Metadata fields
Compare with Document text
Edit missing/ incorrect metadata
Update Search model /rules
Metadata Search Model
OCR Error Correction Rules
Corrected Item Metadata (Ground Truth)

Figure 4. Search model update from analysis of missing field value

Correction of OCR Errors

Characters generated by OCR from scanned TIFF images are often recognized incorrectly, especially for older documents. If an error occurs within a pattern specified in the search, the search would either fail or the results would be unreliable. The AME system presently addresses this by using a collection-specific Text Editor, which replaces frequently misinterpreted search words and patterns with their actual values, using built-in substitution patterns. The corrections may be customized to the level of individual metadata fields. (For example: The word *III.* may be an abbreviation for the state *Illinois,* or the number 111, or an error depending upon a *location*, a *numerical* field, or neither)

Test case – the FDANJ Collection

NLM has acquired a collection of medico-legal documents from the Food and Drug Administration, referred to as FDA Notices of Judgment (or FDANJ for short). It consists of about 70,000 published notices of judgment (NJ) from court cases involving products seized under authority of the 1906 Pure Food and Drug Act. The NJs are resources in themselves but also lead users to the over 2,000 linear foot collection of evidence files used to prosecute each case. Our goal is to create a digital library for browsing the collection as well as searching the collection's metadata and full-text.

The FDANJ collection comprises more than 41,000 pages, grouped under four different categories: Foods and Drugs (FDNJ), Drugs and Devices (DDNJ), Foods (FFNJ), and Cosmetics (CSNJ), with approximately 15,000, 22,000, 4,600, and 150 document pages respectively. These documents, published between the years 1906 to 1964 vary not only in their layouts, but also in their style and level of details within each category. For example, an NJ may span four or five lines in one set to tens of pages in another set. Figure 5 shows the four typical layout styles exhibited in these documents.

FDANJ Text Layouts

Figure 5. Typical layout styles of FDANJ documents

Metadata Fields and Search Rules

There are eleven metadata fields, shown in column 2 of Table 2, that are to be extracted from each NJ. Fields 5 and 7-11 are multi-valued, occurring more than once within a text segment. Depending upon the style, certain metadata fields may not be present in an NJ.

Table 2 also shows the number of extraction rules and search rules identified for individual fields of this collection.

Table 2 – Metadata Fields with Number of Extraction and Search Rules

	Field Name	Data Type	# of Extr. Rules	Text Segments	# of Search Rules
1	*Case Number*	Number	3	Case Header	(1, 1, 2)
2	*Title*		4	Case Header	(1, 1, 2, 2)
3	*Evidence Numbers*	Text	2	Case Header, Page Header	(1, 1)
4	*Issue Date*	Date	1	Page Header	1
5	*Product keywords*	Text	4	Case Title, Case Body	(5, 4, 3, 1)
6	*Defendant Names*	Text	4	Case Body, Case Title	(11, 13, 13, 8)
7	*Adjudicating Court Name*	State	2	Case Body	(4, 1)
8	*Seizure Location*	City/State	4	Case Body	(8, 8, 5, 5)
9	*Seizure Date*	Date	2	Case Body	(10, 3)
10	*Shipped From*	City/State	4	Case Body	(4, 2, 5, 6)
11	*Shipped Into*	City/State	2	Case Body	(5, 2)

Note that a single *ExtractionRule* may apply to more than one layout; also that the actual number of search patterns in a *Search Rule* may be effectively more than one, since a *SearchPattern* object may specify multiple Regular Expressions as *CueWord*, *BeginPatterns* and *EndPatterns* attributes.

A search is conducted for a metadata field, in sequential order w.r.t. the *SearchRule*s, until a match is found using its *SearchPattern*. For multivalued fields, all values are extracted using the same search pattern.

Examples of Search Pattern

The following are typical examples of begin and end patterns used in finding the Defendant Names (where patternOfDate and monthPattern are String constants for date and month, and "\\W" refers to any non-word character in a Regular Expression).

begin pattern	"in\\W+possession\\W+of\\W+"
end pattern	"appeared|alleging|respectively|and|[0-9]+|having"

begin pattern
end pattern
	"against\\W+(the\\W+)?(said\\W+)?
	"alleging|for\\W+\\w+\\W+violations?|under charging|in"

begin pattern *end pattern* where *patternOfDate* and *monthPattern*	"On\\W+" + **patternOfDate** “claimants*|having|filed|which|appeared|sold" "\\(" + **monthPattern** + "\\)\\W+" + "[0-9]{1,2}\\W+" + "[0-9]{4}\\W+" "(January|February|March|April|May|June|July|August|September||October|November|December)"

Search/Extraction Results

We processed more than 1200 document pages exhibiting all four layouts from the FFNJ category, grouped into 12 batches – each batch containing up to 250 pages. The metadata extraction model was initially developed by creating search patterns in collaboration with the curator of the collection. It was improved iteratively using the techniques discussed earlier, until an optimal stage was reached. After a set of batches was processed, a summary file was generated pertaining to each batch, which indicates the total number of NJs identified and processed, along with the percentage of misses for each field within the batch. These results and other findings for four test batches are shown in Table 3, and discussed below.

Table 3 –Percentage of Metadata Extraction Errors per field in Test Batches

	Batch 1		Batch 2		Batch 3		Batch 4		Average	
# of Pages	50		66		64		128		308	
# of NJs	200		200		200		300		900	
NJ Range	2101 – 2300		6201 – 6400		10001 - 10200		19051 – 19350			
Case Number	-	**3.0**	-	**3.5**	-	**1.5**	-	**1.0**	-	**2.1**
Title	1.0	**2.0**	0.5	**3.5**	-	**3.0**	-	**2.6**	0.33	**2.8**
Evidence Numbers	1.5	**2.5**	-	**-**	-	**-**	1.7	**1.0**	0.8	**0.9**
Issue Date	-	**-**	-	**-**	100	**-**	-	**-**	22	**-**
Defendant Names	-	**1.5**	0.5	**10**	1.5	**7.0**	12.6	**6.0**	4.7	**6.1**
Product Keywords	1.0	**2.0**	-	**-**	-	**2.0**	-	**-**	0.2	**0.9**
Adj. Court Names	2.5	**3.5**	2.5	**3.5**	1.5	**3.0**	3.0	**2.3**	2.5	**3.1**
Seizure Location	3.0	**4.0**	0.5	**4.5**	3.5	**4.0**	7.3	**8.6**	4.0	**5.6**
Seizure Date	-	**0.5**	1.5	**2.5**	1.5	**6.5**	2.3	**2.0**	1.6	**2.8**
Shipped From	6.0	**4.5**	3.0	**8.0**	3.0	**6.0**	5.6	**6.0**	4.5	**6.1**
Shipped Into	3.0	**4.5**	0.5	**8.0**	3.5	**5.5**	7.3	**8.6**	4.0	**6.9**

Analysis of Results

Table 3 presents metadata extraction results from four sample batches in the FFNJ category, consisting of approximately 300 pages and 900 cases. The last column of the table shows the results averaged over these four batches.

The value in the left hand cell for each Batch indicates the percentage of cases for which a field could not be found by the metadata search engine. (A '-' indicates that the field is present in every case.) The value in the right hand side cell (with grey background) shows the actual percentage of erroneous or missing values for those fields. The discrepancy is explained as follows:

a) Some metadata fields are optional, or may not be present in documents with a certain style, resulting in a false “miss.” For the *Issue Date* field for Batch 3, the date was missing in the header of the first page of the set, so it is marked as absent for all cases in this batch.

b) A false match may lead to an erroneous or partially correct value, especially for multi-valued items such as *Seizure Date*. Such matches would be revealed only after comparison of the extracted values with the actual data.

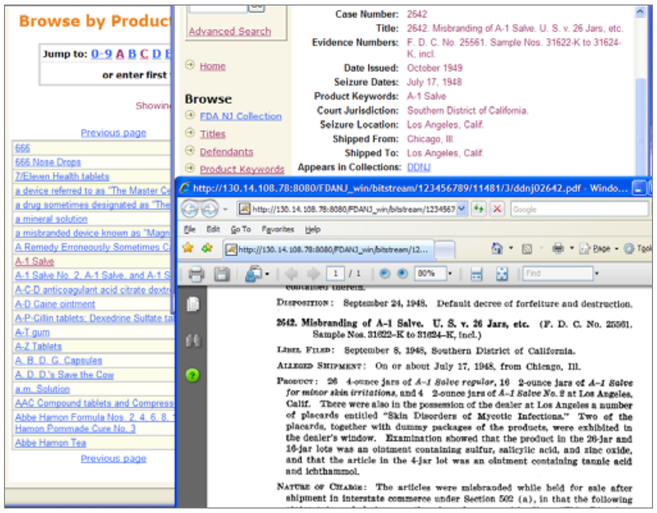
c) A number of errors result due to failure in identifying tags or cue words in various search patterns because of random OCR errors in those texts, which could not be corrected using our simple OCR text editor.

d) Occasionally, though infrequently, a line is classified incorrectly by the layout model.

It is seen from the test cases in Table 3 that the actual error in extracting metadata for any field using the current model remains less than 10 percent, which is also confirmed from the results of other FFNJ test batches. This we regard as satisfactory, although the results should improve with better OCR correction algorithms.

Using the Extracted Metadata for Resource Discovery

Figure 6 illustrates the Web-based retrieval of an FDNJ record in an archive for the specific product “A1-Salve” by browsing through the “Product Keywords” metadata field.



Browse by Product

Jump to: 0-9 A B C D E

or enter first

Showing

Previous page

666
666 Nose Drops
7/Eleven Health tablets
a device referred to as "The Master C
a drug sometimes designated as "The
a mineral solution
a misbranded device known as "Magn
A Remedy Erroneously Sometimes C
A-1 Salve
A-1 Salve No. 2, A-1 Salve, and A-1 S
A-C-D anticoagulant acid citrate dextr
A-D Caine ointment
A-P-Cillin tablets; Dexedrine Sulfate ta
A-T gum
A-Z Tablets
A. B. D. G. Capsules
A. D. D.'s Save the Cow
a.m. Solution
AAC Compound tablets and Compress
Abbe Hamon Formula Nos. 2, 4, 6, 8,
Hamon Pommade Cure No. 3
Abbe Hamon Tea

Previous page

Advanced Search

Home

Browse
FDA NJ Collection
Titles
Defendants
Product Keywords

Case Number:	2642
Title:	2642. Misbranding of A-1 Salve. U. S. v. 26 Jars, etc.
Evidence Numbers:	F. D. C. No. 25561. Sample Nos. 31622-K to 31624-K, incl.
Date Issued:	October 1949
Seizure Dates:	July 17, 1948
Product Keywords:	A-1 Salve
Court Jurisdiction:	Southern District of California.
Seizure Location:	Los Angeles, Calif.
Shipped From:	Chicago, Ill.
Shipped To:	Los Angeles, Calif.
Appears in Collections:	DDNJ

http://130.14.108.78:8080/FDANJ_win/bitstream/123456789/11481/3/ddnj02642.pdf - Windo...

http://130.14.108.78:8080/FDANJ_win/bitstream/1234567
Google
File Edit Go To Favorites Help
http://130.14.108.78:8080/FDANJ_win/bitstream/12...
Page
Tools
1 / 1
80%
Find

contained therein.

DISPOSITION: September 24, 1948. Default decree of forfeiture and destruction.

2642. Misbranding of A-1 Salve. U. S. v. 26 Jars, etc. (F. D. C. No. 25561. Sample Nos. 31622-K to 31624-K, incl.)

LIBEL FILED: September 8, 1948, Southern District of California.

ALLEGED SHIPMENT: On or about July 17, 1948, from Chicago, Ill.

PRODUCT: 26 4-ounce jars of *A-1 Salve regular*, 16 2-ounce jars of *A-1 Salve for minor skin irritations*, and 4 2-ounce jars of *A-1 Salve No. 2* at Los Angeles, Calif. There were also in the possession of the dealer at Los Angeles a number of placards entitled "Skin Disorders of Mycotic Infections." Two of the placards, together with dummy packages of the products, were exhibited in the dealer's window. Examination showed that the product in the 26-jar and 16-jar lots was an ointment containing sulfur, salicylic acid, and zinc oxide, and that the article in the 4-jar lot was an ointment containing tannic acid and ichthammol.

NATURE OF CHARGE: The articles were misbranded while held for sale after shipment in interstate commerce under Section 502 (a), in that the following

Figure 6. Access to an FDANJ document by browsing a metadata field

Adaption to Other Collections

The Metadata Extraction system has been designed with collection level portability in mind. To customize for a particular collection, the followings are needed:

- A metadata search model representing various search rules. (The attributes of various elements in the model should cover searches of reasonable complexity.)
- An OCR text editor (or Lexicon) to correct domain-specific errors in the OCR text.
- An XML schema to format and output the metadata records.
- A collection-specific derived class of the Metadata Extractor to allow for processing of complex cases which are difficult to extract through pattern search rules, and also for post-processing operations.
- A set of other parameters for the collection (for example: input, output file paths) in the form of a Java Properties file.

Further Enhancements

There are several areas in which further work is ongoing or planned to make the AME system a robust tool to extract metadata from other semi-structured documents with relative ease.

- Better OCR error correction algorithm: We have developed a reliable algorithm for correcting suspected words through use of Lexicons [9]. This would be integrated with the AME system in future.
- Further automation in updating the search model and comparison of extracted data with ground truth.
- Processing of other collections to test the search model and the metadata extractor in handling different types of data.

Summary

We have developed an automated metadata extraction system using layout recognition and metadata search models. The rule-based search model, incorporating search rules as string patterns (represented as Regular Expressions) has been applied to a semi-structured text corpus from the Food and Drug Administration. It was successful in extracting embedded metadata with more than 90 percent accuracy and indicating where a search failed. The system is designed for easy customizing to other collections with similar characteristics.

Acknowledgment

The authors acknowledge the collaboration of John Rees, the curator of the Archives and Modern Manuscript program at NLM's History of Medicine Division, in helping with the development of the layout models and the metadata search model for the FDANJ collection, and on other aspects of this work.

This research was supported by the Intramural Research Program of the National Institutes of Health, National Library of Medicine, and Lister Hill National Center for Biomedical Communications.

References

[1] Flynn P, Zhou L, Maly K, Zeil S, Zubair M: Automated Template-Based Metadata Extraction Architecture, Proc. ICADL 2007. Eds: D.H.-L. Goh et al. Berlin Heidelberg: Springer-Verlag: LNCS 4822: 327–336

[2] Heidorn P, Wei: Automatic Metadata Extraction from Museum Specimen Labels, Proc. Int'l Conf. on Dublin Core and Metadata Applications 2008: 57-68

[3] Cortes C., Vapnik V.: Support-vector Network. Machine Learning. Vol. 20, pages 273-297, (1995)

[4] Rabiner, L. R., Juang, B. H.: Fundamentals of Speech Recognition. Englewood Cliffs, NJ: Prentice-Hall. (1993).

[5] Misra D, Mao S, Rees J, Thoma, G.R.: Archiving a Historic Medico-legal Collection: Automation and Workflow Customization, Proc. IS&T Archiving Conference, Washington DC, pg 157-161. (2007).

[6] Thoma GR, Mao S, Misra D, Rees J Design of a Digital Library for Early 20th century Medico-legal Documents Proc. ECDL 2006. Eds: Gonzalo J et al. Berlin: Springer-Verlag; LNCS 4172: 147-57.

[7] ABBYY FineReader OCR Engine 8.0. http://www.abbyy.com

[8] java.util.regex.Pattern in http://java.sun.com/j2se/1.4.2/docs/api/

[9] Chen S, Misra D, Thoma GR, Efficient OCR Word Validation Using Partial Format Derivation and Language Model (in press).

Author Biography

Dharitri Misra is Lead Consultant at Aquilent, Inc. and is a researcher at the U.S. National Library of Medicine. Her work involves developing experiments and tools to help in the preservation of digital resources, with automated extraction of metadata from text documents. She earned her M.S. and Ph.D. degrees in Physics from the University of Maryland.

Siyuan Chen is a postdoctoral fellow at the U.S. National Library of Medicine. He earned his M.S. and Ph.D. degrees in Electrical Engineering from the State University of New York at Buffalo. His research interests include handwriting recognition, optical character recognition, pattern recognition, and machine learning.

George R. Thoma is a Branch Chief at an R&D division of the U.S. National Library of Medicine. He directs R&D programs in document image analysis, biomedical image processing, animated virtual books, and related areas. He earned a B.S. from Swarthmore College, and the M.S. and Ph.D. from the University of Pennsylvania, all in Electrical Engineering. Dr. Thoma is a Fellow of the SPIE, the International Society for Optical Engineering.