Automatic Extraction of Bibliographic Information from Biomedical Online Journal Articles Using a String Matching Algorithm

Jongwoo Kim, Daniel X. Le, George R. Thoma
National Library of Medicine
8600 Rockville Pike, Bethesda, MD 20894, USA
{jongkim, danle, and gthoma}@mail.nih.gov

Abstract

A system has been developed to extract bibliographic data (grant numbers and databank accession numbers) from online biomedical journal articles for the National Library of Medicine's MEDLINE® database. Rule-based algorithms and a string matching algorithm are proposed to extract the bibliographic data from HTML-formatted articles. Experiments conducted with 411 medical articles from 73 journal issues show an accuracy exceeding 96%.

1. Introduction

The U.S. National Library of Medicine (NLM) creates and disseminates MEDLINE®, a bibliographic database of 14 million citations to the biomedical journal literature. The production of this database relies on different methods: the automatic extraction of bibliographic data from scanned (paper) journals, from online journals in HTML, PDF, and XML formats, as well as the reception of such data directly from journal publishers. This data is verified by operators and offered to expert indexers who add descriptive index terms, thereby completing citations for MEDLINE. Systems called MARS [1] and WebMARS [2] have been developed for the first two methods. The third method (publishers sending XML-tagged data to NLM) is the most desirable, but certain items are often not provided by the publishers, such as grant numbers and databank accession numbers, requiring the operators to manually search for these in the article and key them in. Grant numbers, usually appearing in the acknowledgment section of an article, identify the funding agency such as an institute at the U.S. National Institutes of Health. Databank accession numbers uniquely identify the genetic or protein sequence contents of well-known databanks such as GenBank, Swiss-Prot or EMBL. The manual entry of dozens of such numbers for every MEDLINE citation is a labor-intensive task.

To eliminate this manual step, a technique is developed to extract grant numbers and databank accession numbers from online articles (if available) using WebMARS, and insert these in the XML-tagged records sent in by publishers. WebMARS consists of several modules among which are an automated zoning module [2] that segments an article into several zones or regions of contiguous text, and an automated labeling module [3,4] that identifies these zones as *article title, author names, institutional affiliation, abstract, rubric, page numbers, email,* and other useful bibliographic data.

In this paper, a new version of the automated labeling module is proposed to label the text representing grant numbers and databank accession numbers by using rule-based algorithms and a new string matching algorithm.

Section 2 describes features used in the automated labeling module and Section 3 describes a string matching algorithm. Sections 4, 5, and 6 describe rule-based algorithms for the extraction of grant number, databank accession number, and US zip code. Experimental results and a summary are presented in Sections 7 and 8.

2. Features used in Automated Labeling Module

When research reported in an article is funded by one of the institutions of the U.S. Public Health Service (PHS), a number representing a grant from this funding source will appear in the article. These grant numbers usually appear in a sentence with other information such as organization names and/or support words, e.g., "support", "fund", "finance", etc. The automated labeling module employs the names of granting organizations, grant number formats, and other support words as features to recognize grant numbers.

Databank accession number is the registration number of a DNA sequence in one of several databanks. This number usually appears with other information such

as the name of a databank and/or words such as "deposit", "submit", etc. Databank names include "GenBank", "DDBJ", etc. The databank accession number appears in three formats and is composed of alphabetic character(s) followed by a five or six-digit number.

Unfortunately, authors often do not follow the established formats when writing grant numbers and databank accession numbers or they make errors in naming institutions and databanks. This makes it difficult for the automated labeling module to label grant numbers and databank accession numbers correctly, generating over/under labeling problems.

To overcome these problems, an (approximate) string matching algorithm is employed to identify grant organization and databank name features efficiently. Five rules have been created for grant numbers and three rules for databank accession numbers using combinations of the features.

In total, eight rules have been created for automated labeling using seven word list tables. Tables 2 and 3 at the end of the paper show some of these word lists.

A detailed discussion of our string matching algorithm and rule-based algorithms are presented in the next sections.

3. String Matching Algorithm

There are several approximate string matching algorithms found in the literature [5-10]. These algorithms are designed to estimate similarity (edit distance) between two strings using operations such as insertion, deletion, substitution, and transposition. The edit distance between two strings, as calculated using these algorithms, is proportional to the number of operations. The more operations we use for matching two strings, the larger the edit distance (less similarity value) between the two strings. For strings such as grant organization and databank names, a small number of typographic errors results in a disproportionately large edit distance compared to the total length of the strings being compared. These existing algorithms are overly sensitive to such errors.

Therefore, we need a string matching algorithm to find any possible grant organization and databank names with/or without typographic errors in sentences. A new string matching algorithm, which is less sensitive to typographic errors than existing algorithms, is proposed for our experiment. We call the measure JongWoo Distance (JWD_{Gap}) from now on.

The formula for JWD_{Gap} is as follows:

$$JWD_{Gap}(T,S) = \underset{i=1 to P}{Max} D_{Gap}\{(T \wedge S)_i\},$$

$$D_{Gap}\{(T \wedge S)_i\} = \frac{N\{(T \wedge S)_i\}}{|T|}\left[1 - \alpha \cdot \frac{G\{(T \wedge S)_i\}}{N\{(T \wedge S)_i\}}\right],$$

where $\boldsymbol{T}$ is a target string, $|\boldsymbol{T}|$ is the length of $\boldsymbol{T}$, $\boldsymbol{S}$ is an input source string, $(\boldsymbol{T} \wedge \boldsymbol{S})_i$ is the ith matching candidate in string matching $\boldsymbol{T}$ and $\boldsymbol{S}$, $N(x_i)$ is the number of matched elements in x_i, $G(x_i)$ is the number of gaps in x_i, and P is the number of matching candidates in string matching $\boldsymbol{T}$ and $\boldsymbol{S}$. α ($0 < \alpha < 1$) is a parameter.

Assume that we are trying to find a target string $\boldsymbol{T}$= "abc" from input source string $\boldsymbol{S}$ = "dakceabhgc". The following matching results are shown in Table 1. Column is for target string $\boldsymbol{T}$ and row is for input source string $\boldsymbol{S}$. "O" means "match", and "x" and "*" mean "no match". The matching result is in the last row.

Table 1. Matching result of string *T* and *S*.

	S	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}
T		d	a	k	c	e	a	b	h	g	c
t_1	a	x	O	x	x	x	O	x	x	x	x
t_2	b	x	x	x	x	x	x	O	x	x	x
t_3	c	x	x	x	O	x	x	x	x	x	O
		*	a	*	c	*	a	b	*	*	c

In Table 1, there are two matching candidates (P=2); $(\boldsymbol{T} \wedge \boldsymbol{S})_1$="a*c" and $(\boldsymbol{T} \wedge \boldsymbol{S})_2$="ab**c". The proposed algorithm tries to choose the match that maximizes JWD_{Gap} value. In the case of $(\boldsymbol{T} \wedge \boldsymbol{S})_1$="a*c", $|T|$=3. $N\{(\boldsymbol{T} \wedge \boldsymbol{S})_1\}$=$N${"a*c"}=2 since "a" and "c" are matched. $G\{(\boldsymbol{T} \wedge \boldsymbol{S})_1\}$ = G{"a*c"}=1 since there is a gap between "a" and "c". In the case of $(\boldsymbol{T} \wedge \boldsymbol{S})_2$="ab**c", $N\{(\boldsymbol{T} \wedge \boldsymbol{S})_2\}$ = N{"ab**c"} = 3 and $G\{(\boldsymbol{T} \wedge \boldsymbol{S})_2\}$=$G${"ab**c"}=1. Therefore, when =0.5,

$$JWD_{Gapt}(T,S) = Max\left[D_{Gap}("a*c"), D_{Gap}("ab**c")\right]$$
$$= Max\left[\frac{2}{3}\{1 - 0.5 \cdot \frac{1}{3}\}, \frac{3}{3}\{1 - 0.5 \cdot \frac{1}{3}\}\right]$$
$$= Max[0.55, 0.83] = 0.83$$

I.e., "ab**c" is selected as the result of string matching $\boldsymbol{T}$ and $\boldsymbol{S}$ with $JWD_{\boldsymbol{Gap}}$=0.83. Since JWD_{Gap} becomes sensitive to $G(x_i)/N(x_i)$ when is close to 1, and less sensitive to $G(x_i)/N(x_i)$ when is close to 0, =0.5 is used in our experiment.

As shown in the above example, $JWD_{\boldsymbol{Gap}}$ is affected by the number of gaps ($G(x_i)$) and is not directly affected by the number of operations for string matching. It helps the proposed algorithm to find more possible candidates for grant organizations and databank names than existing algorithms.

4. Rule-based Algorithm for Grant Number

When research reported in an article is funded by one of the organizations at the U.S. Public Health Service (PHS) and a grant number is found in the article, the zone is labeled as grant number.

A typical sentence mentioning financial support is: "*This work was funded by grants from the National Institutes of Health (GM55026 to M.M.G. and GM62831 to E.C.L.).*" "GM55026" and "GM62831" are the grant numbers. "GM" stands for National Institute of General Medical Sciences (NIGMS).

Another example is "*This study was supported by contract NO1 AI-45252 from the National Institute of Allergy and Infectious Diseases and by grant MO1-RR08084 from the NCRR, NIH." "NO1 AI-45252*" and "MO1-RR08084" are grant numbers. "AI" and "RR" stand for the National Institute of Allergy and Infectious Diseases and National Center for Research Resources (NCRR), respectively

4.1 Features for Grant Number

There are seven institutions belonging to the U.S. Public Health Service (PHS): National Institutes of Health (NIH), Food and Drug Administration (FDA), Health Resources and Services Administration (HRSA), Centers for Disease Control and Prevention (CDC), Office of the Assistant Secretary of Health (OASH), Substance Abuse and Mental Health Services Administration (SAMHSA), and Agency for Health Care Policy and Research (AHCPR). Each of these institutions may also include several subdivisions such as institute, center, office, etc.

The official format for a grant number is as follows:

Application Type (one-digit) + Activity Code (three-digit) + Administering Organization (two-digit character) + Serial Number (five to six-digit number) + Grant Year (two-digit) + Other (four-digit).

Some authors frequently use two or three items in the official format to express grant numbers, though many use a simplified version as follows:

Administering Organization (two-digit character) + Serial Number (five to six-digit number)

Application Type identifies the type of grant application received and processed, Activity Code identifies a specific category of extramural activity, Administering Organization identifies subdivision, Serial Number is sequentially assigned by a subdivision, and Grant Year indicates the budget period of a project.

Each subdivision (e.g., an institute at NIH) belonging to the PHS has its own Administering Organization identified by a two-digit character. For example, National Library of Medicine (NLM), a subdivision of NIH, is an Administering Organization identified as "LM". Therefore, a research grant from NLM starts with "LM" followed by a five or six-digit number.

Grant numbers are usually mentioned together with the corresponding granting organization (institute) in the article, as shown in the two example sentences above. Therefore, pairs of {an institution name, a subdivision name, Administering Organization} are collected and saved in the GrantOrganizationList as shown in Table 2.

When authors acknowledge the financial support for their research, they usually use words such as "support", "fund", "grant", etc. as shown in the two examples. These words (most frequent ones) are also collected from several grant number zones and saved in a word list named SupportWordList as shown in Table 3.

There are many foreign institution names similar to those of the organizations in Table 2. These foreign institutions usually put their country names before or after the institution names. Therefore, 180 country names (excluding USA) are collected and saved in CountryNameList table as shown in Table 3. This table is used to distinguish the institutes/institutions in Table 2 from other organizations in foreign countries.

4.2 Rules for Grant Number

Since there are several ways of expressing financial support in research articles, we globally categorize them into three. Three rules are therefore created to extract the grant numbers.

Rule 1:
If (SupportWordList does exist and
GrantOrganizationList does exist and
CountryNameList does not exist and
Grant Number does exist and
Grant Number is not ZipCode)
The Zone is labeled as Grant Number.

Rule 2:
If (SupportWordList does exist and
CountryNameList does not exist and
Grant Number does exist and
Grant Number is not ZipCode)
The Zone is labeled as Grant Number.

Rule 3:
If (GrantOrganizationList does exist and
CountryNameList does not exist and
Grant Number does exist and
Grant Number is not ZipCode)
The Zone is labeled as Grant Number.

Rules 1, 2, and 3 are made under the assumption that article authors write grant numbers in the correct formats. Unfortunately, we find that many authors do not write grant numbers correctly. For example, "*This work is supported by funds from the National Institutes of Health (Grant RO112686 to A.I.B.), the Alzheimer Association, and the National Health and Medical Research Council.*" "RO112686" identifies a grant from National Institutes of Health. However, the author missed the "Administering Organization" identifier in the grant number.

Therefore, we added two more rules to handle such cases. When a word is larger than four digits and at least three of these digits are composed of Arabic numbers, we define the word as a "candidate for grant number".

Rule 4:
If (SupportWordList does exist and
GrantOrganizationList does exist and
CountryNameList does not exist and
Candidate for Grant Number does exist and
Candidate for Grant Number is not ZipCode)
The Zone is labeled as Grant Number.

Rule 5:
If (GrantOrganizationList does exist and
CountryNameList does not exist and
Candidate for Grant Number does exist and
Candidate for Grant Number is not ZipCode)
The Zone is labeled as Grant Number.

Rule 1 labels a zone as grant number when the zone has words belonging to SupportWordList, has words belonging to GrantOrganizationList, does not have words belonging to CountryNameList, and has words with formats of grant number and the words are not zip codes. The other rules have similar meanings.

5. Rule-based Algorithm for Databank Accession Number

Databank accession number is the registration number of a DNA sequence in any of several databanks. The following is a common type of sentence mentioning databank accession numbers: "*The nucleotide sequence(s) reported in this paper has been submitted to the GenBank™/EMBL Data Bank with accession number(s) Z72390.*" Therefore, "Z72390" is the databank accession number. Another example is "*The sequence data were submitted to the National Center for Biotechnology Information and were registered with accession numbers AY569561 through AY569566.*" "AY569561", "AY569566", and the intermediate numbers in this range are databank accession numbers.

5.1 Features for Databank Accession Number

There are three formats for databank accession numbers.

One-digit Alphabetic Character	+ Five-digit Number
Two-digit Alphabetic Character	+ Six-digit Number
Three-digit Alphabetic Character	+ Five-digit Number

DNA sequences appear in eleven databanks such as "GENBANK", "EMBL", "DDBJ", "Swiss-Prot", etc. These databank names are collected and saved in the DatabankNameList table. Since the words "deposit", "submit", and "access" are frequently used in zones with databank accession numbers, two other word lists, DepositWordList and AccessionWordList, are also made for databank accession number as shown in Table 3.

5.2 Rules for Databank Accession Number

There are several ways of expressing databank accession number in journal articles. We globally categorize them into three ways and create three rules to extract these.

Rule 1:
If (SubmitWordList does exist and
DatabankNameList does exist and
AccessionWordList does exist and
A format of Databank Number does exist and
A format of Databank Number is not ZipCode)
The Zone is labeled as Databank Accession Number.

Rule 2:
If (DatabankNameList does exist and
AccessionWordList does exist and
A format of Databank Number does exist and
A format of Databank Number is not ZipCode)
The Zone is labeled as Databank Accession Number.

Rule 3:
If (DatabankNameList does exist and
A format of Databank Number does exist and
A format of Databank Number is not ZipCode)
The Zone is labeled as Databank Accession Number.

Rule 1 labels a zone as databank accession number when the zone has words belonging to SubmitWordList, DatabankNameList, and AccessionWordList, and has words with formats of databank accession numbers, and the words are not zip codes. The other rules have similar meanings.

6. Rule-based Algorithm for Zip Code

Since the formats of US zip codes are very similar to the formats of grant numbers and databank accession numbers, a rule-based algorithm for zip code is also developed to increase labeling accuracy.

6.1 Features for US Zip Code

The US zip codes appear in two formats. For example, MD 20894 and Maryland 20894.

Full state name + Five-digit Number
Two-digit abbreviated state name + Five-digit Number

There are too many combinations to save pairs of (state name, five-digit number). Fortunately, we find that the first two digits of a zip code are the same or similar in the same state and there are some two-digit numbers in a given state. Therefore, a list for US zip code (USZipCodeList) is made as shown in Table 3. It contains pairs of (a two-character state name, the first two digits of the zip code) and (a full state name, the first two digits of the zip code).

In the case of Maryland, for example, every zip code starts with 20, 21, or 26. Therefore, the pairs (Maryland", "20"), ("MD", "20"), (Maryland", "21"), ("MD", "21"), (Maryland", "26"), and ("MD", "26") are saved in the table.

6.2 Rules for US Zip Code

There are two ways to write US zip codes as mentioned previously. These two ways can be expressed in one rule as follows:

Rule 1:
If (State name with a five-digit number does exist and
(the state name, first two digits of the number) is
in the USZipCodeList)
The Zone is labeled as Zip Code.

7. Experimental Results

We used 0.6 as a threshold for JWD_{Gap} in this experiment. I.e., we assumed that there was an institute name/databank name ($\boldsymbol{T}$) in a sentence ($\boldsymbol{S}$) when $JWD_{Gap}(\boldsymbol{T},\boldsymbol{S}) \bullet 0.6$. Since $0 < < 1$, $=0.5$ was selected in our experiment for $JWD_{\boldsymbol{Gap}}$.

For our experiment, we selected 411 articles from 73 different journals, and Table 4 shows the experimental results. There were eleven errors in grant number zones and three errors in databank accession number zones. In total, fourteen errors were found and all these were due to over-labeling (False Positive error).

The fourth row in Table 4 gives the overall results in terms of Precision, Recall, and F-Measure.

Most errors occurring in grant number were generated by Rules 4 and 5, the rules that were made for labeling grant numbers with wrong formats. From the point of view of processing time and accuracy, since under-labeling errors are more serious than errors due to over-labeling, Rules 4 and 5 will be retained in our module.

We conclude that the accuracy of the labeling module was above 96.96% for all three measures used.

8. Summary

This paper describes an automated labeling module using rule-based algorithms and a string matching algorithm to label bibliographic information (grant number and databank accession number) in HTML-formatted articles. Experiments conducted for 411 journal articles show above 96.96% labeling accuracy based on the label field in all the three measures.

Since the current labeling module only uses zone-level (local) information to label a zone, it generates over-labeling errors.

Future work will use article-level (global) information in the labeling module to minimize over-labeling errors. The module will also be extended to label other important bibliographic information. Machine learning algorithms will also be adapted to change rules automatically.

9. Acknowledgment

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

10. References

[1] G.R. Thoma, D.X. Le "Automating data entry for online biomedical databases", *Proc. 14th National Conference on Integrated Online Library Systems IOLS'99*", Medford, NJ, May 1999, pp. 121-128.

[2] D.X. Le, L.Q. Tran, et. al., "Automated Medical Citation Records Creation for Web-Based On-Line Journals," *14th IEEE Symposium on Computer-Based Medical Systems*, Bethesda, MD, July 2001, pp. 315-320.

[3] J. Kim, D. Le, and G. Thoma, **"**Automated labeling of bibliographic data extracted from biomedical online journals," *Proc. SPIE Electronic Imaging*, Vol. 5010, January, 2003, pp. 47-56.

[4] J. Kim, D. Le, and G. Thoma, "Automated Labeling of Biomedical Online Journal Articles**,"** *Proc. 9th World Multiconference on Systemics, Cybernetics and Informatics*, July, Orlando, FL, Vol. 3, 2005. pp. 406-411.

[5] T.H. Cormen, C.E. Leiserson, R.L. Revest and C. Stein, "Introduction to Algorithms", Second Edition, MIT Press and McGraw-Hill, 1001, pp. 906-932.

[6] V.I. Levenstein, "Binary Codes Capable of Correcting Deletions, Insertion, and Reversals." *Sov. Phys. Dokl.*, Vol. 6, 1996, pp. 707-710.

[7] R.A. Wagner and M.J.Fisher, "The String-to-String Correction Problem", *Journal of ACM*, Vol. 21, 1974, pp. 168-178.

[8] D. Sankoff and J. Kruskal, *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*, *Addison-Wesley*, Reading, MA.

[9] G. Das, R. Fleisher, L. Gasieniek, D. Gunopulos, and J. Kärkäinen, "Episode Matching", *Proc. 8th Annual Symposium on Combinatorial Pattern Matching*,Vol. 1264, 1997, Berlin, pp. 12-27.

[10] S. Needleman and C. Wunsch, "A General Method Applicable to the Search for Similarities in the Amino Acid Sequences of Two Proteins", *J. Mol. Biol*. 48, 1970, pp. 444-453.

Table 2. Granting Organizations that are part of U.S. Public Heath Service

Organization Name	**Subdivision**	**Administering Organization Identifier**
National Institutes of Health (NIH)	National Library of Medicine (NLM)	LM
Health Resources and Services Administration (HRSA)	Division of Disadvantaged Assistance	MB
Food and Drug Administration (FDA)	Center for Biological Evaluation and Research	BA
Centers for Disease Control and Prevention (CDC)	National Center for Injury Prevention and Control	CE
Office of the Assistant Secretary of Health (OASH)	Office of Family Planning	FP
Substance Abuse and Mental Health Services Administration (SAMHSA)	Office of the Administrator	OA
Agency for Health Care Policy and Research (AHCPR)	Agency for Health Care Policy and Research	HS

Table 3. Word list tables used in the Automated Labeling Module.

Table Name	**Words in the Table**
SupportWordList	support, fund, grant, finance, etc.
CountryNameList	Korea, Canada, Mexico, England, China, France, Germany, etc.
DatabankNameList	GenBank, Embl, Ddbj, Swiss-Prot, CSD, GDB, HGML, OMIN, PDB, PIR, PRFSEQDB
DepositWordList	Submit, deposit, register, etc.
AccessionWordList	Accession, access, etc.
USZipCodeList	("MD","20"), ("Maryland","20"), ("MD","21"), ("Maryland","21"), ("MD","26"), ("Maryland","26"), ("MI","48"), ("Michigan","48"), ("MI","49"), ("Michigan","49")

Table 4. Test results for the Automated Labeling Module. TP (True Positive), FP (False Negative)

Label	**Number**	**Correct (TP)**	**Error (FP)**	**Precision (%)**	**Recall (%)**	**F-Measure (%)**
Grant	328	317	11	96.65	100.00	98.30
Databank	133	130	3	97.74	100.00	98.86
Total Zones	**461**	**447**	**14**	**96.96**	**100.00**	**98.46**