Using Digrams to Map Controlled Medical Vocabularies

Roberto A. Rocha, MD and Stanley M. Huff, MD
Department of Medical Informatics, University of Utah 84112

A program for matching between controlled medical vocabularies has been developed which adopts methods used in the domain of Information Retrieval. This program combines a stemmer based on fragments of words (digrams) with a similarity function. The proposed stemmer did not require any knowledge about word-formation rules and helped the identification of several kinds of word variants. The adopted similarity function assigned the highest score to the best candidate match in 99.0% of the cases.

INTRODUCTION

Several controlled medical vocabularies (CMVs) are currently available. However, they usually cover diverse domains with uneven scopes and objectives [1]. The absence of an accepted "standard" method for representing medical concepts, and the need to translate clinical data to existent CMVs has made computerized vocabulary mapping an active area of medical informatics research [2-7].

In our site, several projects require some form of vocabulary matching, ranging from the integration of clinical systems [8,9], to our participation in the Unified Medical Language System (UMLS) project [10,11]. In addition, our research in the area of medical data representation and controlled medical vocabularies has directed many of our efforts toward an automated vocabulary translation method [6,12].

The problem of automated mapping ("translating") between CMVs can be approached in two different ways. One method is known as lexical matching, or "string matching", where the goal is to try to identify similarities among the strings (words and phrases) used in both source and target vocabularies [5,7]. The other method is know as conceptual matching, where instead of comparing words and phrases, the process tries to identify similarities between concepts ("meanings") [3,4,6].

Ideally, both methods should be combined. For example, lexical matching methods could be used to refine the output of a conceptual matching application. In theory, conceptual matching should produce better results, but for small and well-defined domains its complexity may not be justifiable.

The methods applied to vocabulary matching are closely related to the methods utilized in the area of Information Retrieval (IR). IR systems look for similarities between queries and collections of documents [13], while vocabulary matching systems seek similarities between source vocabularies and target vocabularies. We could say, for instance, that each source term is a "query" against the "collection" of target terms. In addition, the use of "meanings" versus "strings" to retrieve information is an active area of research in IR [13,14].

Our intention in this paper is to describe how we adapted IR methods to the area of vocabulary translation. The experiments described here examine lexical matching as a potential technique to create mappings between vocabularies.

METHODS

Matching Algorithm

Initially, we adapted a stemming technique known as n-gram [15]. The n-gram method decomposes terms into sets of adjacent characters. These sets of characters can be of any length. Based on the experience of Adamson and Boreham [16], we decided to used pairs of adjacent characters, called "digrams" or 2-grams. Table 1 presents some examples of words with their respective digrams.

Table 1 - Examples of words and their digrams.

id	*word*	*unique digrams*
1	dyspnea	dy, ys, sp, pn, ne, ea
2	dyspneic	dy, ys, sp, pn, ne, ei, ic
3	dypsnea	dy, yp, ps, sn, ne, ea
4	dysp	dy, ys, sp
5	dyspepsia	dy, ys, sp, pe, ep, ps, si, ia

In addition to the digram method, we adopted a similarity scoring method known as "Dice's Similarity Coefficient" (D_{St}) [13]. D_{St} is defined as:

$$D_{st} = \frac{2 \times M}{S + T}$$

where S is the number of unique elements of the source term, T is the number of unique elements of the target term, and M is the number of unique elements common to both source and target.

Several variations in word morphology and orthography are known to decrease the efficiency of any lexical matching algorithm [13,15,16]. In Table

2, we present some examples of how digrams and Dice's coefficient help the identification of word variations, like the ones displayed in Table 1.

The combination of the digram stemming method and Dice's similarity coefficient was the matching algorithm used for this project.

Table 2 - Using Dice's coefficient to calculate similarities between words.
(Refer to Table 1 for decoding the ids)

Variant form	*Comparison (id vs. id)*	D_{st}
derivation	2 vs. 1	0.77
misspelling	3 vs. 1	0.50
truncation	4 vs. 1	0.67
truncation	4 vs. 5	0.54
(unrelated)	5 vs. 1	0.43

Matching Process

In addition to the matching algorithm, we implemented routines to "normalize" both source and target terms. These routines were applied as pre-matching processes. The normalization steps are summarized in Table 3.

Table 3 - Steps used to normalize terms

Original term: *"Thyroid Function Study :Serum :Quantitative - TSH or Thyroid Stimulating Hormone - MIU/ML"*
Step 1 - Remove punctuation, special characters, and numbers: *"Thyroid Function Study Serum Quantitative TSH or Thyroid Stimulating Hormone MIU ML"*
Step 2 - Lower case all characters: *"thyroid function study serum quantitative tsh or thyroid stimulating hormone miu ml"*
Step 3 - Remove duplicate words: *"thyroid function study serum quantitative tsh or stimulating hormone miu ml"*
Step 4 - Remove stop words: *"thyroid function study serum quantitative tsh stimulating hormone miu ml"*
Step 5 - Sort words in ascending order: *"function hormone miu ml quantitative serum stimulating study thyroid tsh"*

Following the normalization process, a word index was created so that a given word pointed to one or more terms where it occurred. A unique word list was then created by extracting only distinct words from the word index. Finally, a "digram index" was obtained from the word list. The digram index was very similar to the word index, but having the words replaced by their respective digrams. Table 4 has examples of entries found in these ancillary files. All ancillary files were loaded into a relational database.

In order to observe the interactions between the digram method and Dice's similarity coefficient, three matching strategies were implemented: 1) *No-digram strategy*: the digram method was not used, only the original source term words were matched; 2) *Standard-digram strategy*: the digram method was used to identify words similar only to those present in the source term and not found in the target vocabulary; and 3) *Full-digram strategy*: the digram method was used to identify words similar to all those present in the source term.

Experiments

Two experiments were conducted to evaluate our approach to lexical matching. The first experiment was designed to determine how useful a similarity score like Dice's coefficient is in ranking the best candidate mappings. The matching strategy used for this first experiment was the standard-digram strategy.

Table 4 - Examples of the entries found in the target vocabulary indexes.

Normalized terms with numeric identifiers: *"chest pain / 25575", "chronic pain / 133595"*
Word index entries: *"chest / 25575 / 2", "chronic / 133595 / 2", "pain / 25575 / 2", "pain / 133595 / 2"*
Word list entries: *"chest / 1269", "chronic / 145", "pain / 489"*
Digram index entries: *"pa / 489 / 3", "ai / 489 / 3", "in / 489 / 3"*

For this first experiment, we obtained a large set of commonly used PTXT [17] codes to match against the UMLS Metathesaurus version 1.3 (Meta 1.3) [18]. PTXT is challenging to match because it has many peculiarities of a vocabulary supporting a complex information system. These peculiarities are usually format-related (i.e., abbreviations, truncations, misspellings, etc.), or content-related (i.e., daily-use clinical terminology, protocol-oriented terms, etc.).

The initial set of PTXT codes represented a variety of domains, including laboratory, radiology, discharge diagnosis, nurse charting, etc. This source file was called "PTXT-mixed". From PTXT-mixed we isolated codes corresponding to prescribed drugs, generating a second source file called "PTXT-drugs".

Considering the actual sources and coverage of Meta 1.3, we expected fewer format-related intricacies. However, because Meta 1.3 makes the distinction between strings and their underlying concepts, it can easily mislead a lexical matching method.

Also during this first experiment, we normalized the PTXT-drugs file using a "specialized" filtering routine. This special routine removed units, drug concentrations, and drug presentation forms, leaving only the chemical name and the brand name.

The second experiment was designed to evaluate whether digrams improved the recognition of variant forms, and we used all three matching strategies. The source terms were chest x-ray descriptions from the Iliad data dictionary [19]. The file with these terms was called "Iliad-cxr", and the target vocabulary was again Meta 1.3.

Knowing the differences in granularity between Iliad and Meta [6], we modified Dice's coefficient to handle one-to-many matches. In this case, we added a new search condition to select all terms from Meta 1.3 having the total number of words identical to the number of shared words, i.e., forcing T and M to be equal (see Dice's formula).

The output of the first and the second experiments was reviewed by the authors. A simple tool was used to display the source term and the candidate target terms. For the first experiment, only a *single best match* was selected, usually disregarding modifiers or explicit contexts found in the PTXT terms. For the second experiment, either *a single best match or a combination of matches* was selected. Also during the second experiment, all candidate Meta 1.3 terms were presented to the reviewer with their respective semantic types, helping the identification of the most appropriate concepts and not simply the matching string forms.

RESULTS

First Experiment

The results of the first experiment are presented in two parts: matching PTXT-mixed to Meta 1.3, and matching PTXT-drugs to Meta 1.3.

Matching PTXT-mixed to Meta 1.3: The file PTXT-mixed had 2,671 entries. Normalizing PTXT-mixed, we obtained 2,530 unique terms.

All the terms in English from the Meta 1.3 were used as the target vocabulary, corresponding to 255,742 entries. After normalization, we obtained 200,730 unique terms. From these unique terms, we generated the word index with 578,526 entries, the word list with 91,029 entries, and the digram index with 823,649 entries.

The time required to identify all candidate mappings for a given source term was variable. When extensive searches against the digram index were necessary, the matching process usually took a couple of minutes. However, searches against the word list and the word index usually took just a few seconds.

After completing the matching process, an average of 170.85 candidate target terms per source term were obtained. The results of the manual review, grouped by Dice's coefficient, are summarized in Table 5.

Out of the 831 matches obtained, in 823 (99.0%) cases the matched term had the highest Dice's coefficient, and in 8 (1.0%) cases, the matched term did not have the highest Dice's score.

Table 5 - Summary of the review of the first experiment (PTXT-mixed to Meta 1.3).

Dice's coefficient range	*Match (%)*	*No Match (%)*
Perfect score: 1.0	219 (26.3)	129 (7.0)
High score: from 0.8 to 0.99	42 (5.1)	58 (3.1)
Medium score: from 0.6 to 0.79	161 (19.4)	518 (28.2)
Low score: from 0.0 to 0.59	409 (49.2)	1135 (61.7)
Total	831 (31.1)	1840 (68.9)

Matching PTXT-drugs to Meta 1.3: The file called PTXT-drugs had 1,142 entries. Normalizing PTXT-drugs with the *standard method* generated the same number of unique terms. Normalizing it with the *special method* generated 970 unique terms.

After running the matching process, for those terms normalized with the special routine, we obtained an average of 36.35 candidate target terms per source term. The results of the manual review, grouped by Dice's coefficient, are summarized in Table 6. Table 7 summarizes the performance of Dice's coefficient for the matched terms.

Second Experiment

The file called Iliad-cxr had 238 entries. After normalizing these terms, we obtained the same number of unique terms. The results of the matching processes are summarized in Table 8.

DISCUSSION

The results of our first experiment demonstrate the usefulness of Dice's similarity coefficient, despite its simplicity. Although we identified only 31.1% of matches between PTXT-mixed and Meta 1.3, Dice's

coefficient ranked the best match with the highest score on 99.0% of the cases. Similar performance in ranking the best match was observed between PTXT-drugs and Meta 1.3 (99.2%, 98.7%).

Table 6 - Summary of the review of the first experiment (PTXT-drugs to Meta 1.3).

	PTXT-drugs (standard filter)		*PTXT-drugs (special filter)*	
Dice's coefficient range	*Match (%)*	*No Match (%)*	*Match (%)*	*No Match (%)*
Perfect score: 1.0	14 (2.6)	4 (0.6)	264 (28.7)	56 (25.0)
High score: from 0.8 to 0.99	17 (3.2)	3 (0.5)	54 (5.9)	9 (4.0)
Medium score: from 0.6 to 0.79	101 (19.0)	59 (9.7)	491 (53.5)	90 (40.2)
Low score: from 0.0 to 0.59	400 (75.2)	544 (89.2)	109 (11.9)	69 (30.8)
Total	532 (46.6)	610 (53.4)	918 (80.4)	224 (19.6)

Dice's coefficient was in some cases mislead by the normalization process, and by the digram method. High scores end up being assigned to terms that did not match, ranging from 0.5% to 25.0%. Using less generic filters and limiting the domains of the target vocabulary, we will certainly improve precision.

Table 7 - Matched terms and the value of their Dice's coefficients (PTXT-drugs to Meta 1.3).

	Highest D_{st} (%)	*Not highest D_{st} (%)*
standard filter	528 (99.2%)	4 (0.8%)
specialized filter	906 (98.7%)	12 (1.3%)

In addition to ranking unrelated concepts as good candidates for a match, Dice's coefficient was also responsible for hiding candidate matching terms. This effect was obvious when the percentage of matches between PTXT-drugs and Meta 1.3 almost doubled (from 46.6% to 80.4%) after we applied the special filter. This filter improved the precision of the matching process, reducing the average number of candidate target terms per source term from 170.85 to 36.35. The normalization process also helped to reduce the redundancy of both source and target vocabularies. However, the practice of using "aggressive" filters may not be indicated when format-related details are important.

The second experiment has demonstrated that the digram method can improve the recall of the matching process. We observed an increase in the number of candidate target terms per source term, from 39.70 to 54.67. The full-digram strategy produced a slightly higher average number of target terms per match, reflecting the improvement in recall.

The differences in granularity between Iliad-cxr and Meta 1.3, combined with the adaptation of Dice's coefficient to handle one-to-many matches, produced low average Dice's coefficients. However, despite this effort, many "modifiers" present in the Iliad vocabulary were not available in the Meta 1.3 vocabulary, making almost all matches incomplete. These problems were reflected in the performance of the full-digram strategy, which identified only four additional concepts not revealed by the other two strategies (56 versus 54 and 50).

Reviewing the candidate Meta 1.3 terms with their semantic types attached, helped the identification of important deficiencies of the lexical matching process. Fifty-six concepts were correctly identified because they were either nonambiguous (such as disease names and body parts), or because only a single meaning of the string was present. In other cases, the opposite occurred, i.e., the exact same string was present, but with an inappropriate meaning. For instance, words like "opacity", "abnormality", and "inflation" were perfect string matches, but their meanings in Meta 1.3 did not correspond to their meanings in the Iliad-cxr terms.

Overall, the combination of a n-gram stemmer with a similarity coefficient was a good choice for a general purpose lexical matching tool. The digram algorithm did not require any knowledge about word-formation rules, and did not rely upon the existence of affix dictionaries. An additional benefit of digram

	No-digram strategy	Standard-digram strategy	Full-digram strategy
1. Average number of candidate target terms per source term	39.70	45.17	54.67
2. Average Dice's coefficient of the candidate target terms	0.34	0.34	0.35
3. Average number of target terms per match	3.17	3.17	3.41
4. Total number of concepts identified	50	54	56
5. Total number of concepts identified only by this method	0	2	4

Table 8 - Summary of the results of the second experiment.

stemmers is their applicability in detecting spelling problems [13], and their usefulness in multilingual environments. Dice's coefficient was very simple to implement and has shown its potential as well.

The lexical matching system described here was successful, and it will help the maintenance of our local systems [12]. Future plans include a study to compare this method with InterMatch [6].

Acknowledgment

Roberto A. Rocha is supported by a scholarship from the National Council for Scientific and Technological Development (CNPq), Secretary for Science and Technology, Brazil. This project was partially supported by grant number 1 R03 HS 08053-01 from the Agency for Health Care Policy and Research.

Reference

[1] Evans, DA, Cimino, JJ, Hersh, WR, Huff, SM, Bell, DS (for the Canon Group). Toward a Medical-concept Representation Language. *JAMIA* 1(3):207-17, 1994.

[2] Wingert F. Medical Linguistics: Automated Indexing into SNOMED. *CRC Critical Reviews in Medical Informatics* 1(4): 335-403, 1987.

[3] Masarie FE, Miller RA, Bouhaddou O, Giuse NB, Warner, HR. An Interlingua for Electronic Interchange of Medical Information: Using Frames to Map between Clinical Vocabularies. *Computers and Biomedical Research* 24: 379-400, 1991.

[4] Cimino JJ, Barnett GO. Automated Translation Between Medical Terminologies Using Semantic Definitions. *MD Computing* 7(2): 104-109, 1990.

[5] Sherertz DD, Tuttle MS, Blois MS, Erlbaum MS. Intervocabulary Mapping within the UMLS: The Role of Lexical Matching. *Proc. of the 12th Symposium on Comp. Applic. in Medical Care*, 201-206, 1988.

[6] Rocha RA, Rocha BHSC, Huff SM. Automated Translation Between Medical Vocabularies Using a Frame-Based Interlingua. *Proc. of the 17th Symposium on Comp. Applic. in Medical Care*, 690-694, 1993.

[7] Sherertz DD, Tuttle MS, Olson NE, Erlbaum MS, Nelson SJ. Lexical Mapping in the UMLS Metathesaurus. *Proc. of the 13th Symposium on Comp. Applic. in Medical Care*, 494-499, 1989.

[8] Gibson, R, Haug, P. Linking the Computerized Severity Index (CSI) to Coded Patient Findings in the HELP System Patient Database. *Proc. of the 17th Symposium on Comp. Applic. in Medical Care*, 673-77, 1993.

[9] Wong ET, Pryor TA, Huff SM, Haug PJ, Warner, HR. Interfacing a Stand-Alone Diagnostic Expert System with a Hospital Information System. *Computers and Biomedical Research* 27: 116-129, 1994.

[10] Huff SM, Warner HR. A Comparison of Meta-1 and HELP Terms: Implications for Clinical Data. *Proc. of the 14th Symposium on Comp. Applic. in Medical Care*, 166-169, 1990.

[11] Bouhaddou O, Warner H, Huff S, Bray B. Sorenson D, Dougherty N. Evaluating How the UML Meta1.1 Covers Disease Information Contained in a Diagnostic Expert System (Iliad). *Abstract presented at the 1993 AMIA Spring Congress*, St. Louis, May 9-12.

[12] Rocha RA, Huff SM, Haug PJ. Implementing a Controlled Medical Vocabulary Server. *Abstract presented at the 1994 AMIA Spring Congress*, San Francisco, May 4-7.

[13] Salton, G. *Automatic Text Processing - The Transformation, Analysis, and Retrieval of Information by Computer.* Addision-Wesley Publishing Co., 1989.

[14] Yang Y, Chute CG. Words or Concepts: the Features of Indexing Units and their Optimal Use in Information Retrieval. *Proc. of the 17th Symposium on Comp. Applic. in Medical Care*, 685-689, 1993.

[15] Frakes WB, Baeza-Yates R, eds. *Information Retrieval - Data Structures & Algorithms.* Prentice Hall, 1992.

[16] Adamson GW, Borcham J. The Use of an Association Measure Based on Character Structure to Identify Semantically Related Pairs of Words and Document Titles. *Information Storage and Retrieval*, 10(7-8):253-260, 1974.

[17] Kuperman GJ, Gardner RM, Pryor TA. *HELP: A Dynamic Hospital Information System.* Springer-Verlag, 1991.

[18] Lindberg, DAB, Humphreys, BL, McCray, AT. The Unified Medical Language System. *Methods of Information in Medicine*, 32(4):281-91, 1993

[19] Warner HR, Haug P, Lincoln M, Warner H Jr, Sorenson D, Fan C. Iliad as an Expert Consultant to Teach Differential Diagnosis. *Proc. of the 12th Symposium on Comp. Applic. in Medical Care*, 371-376, 1988.