Concept Extraction, Concept Semantic Categorization and Assertion Classification from Clinical Data

Rocio Guillen, PhD, Charles Cowart, MS

California State University San Marcos, San Marcos, CA

Abstract

This paper reports on Concept Extraction task and the Assertion task of the 2010 i2b2/VA Challenge Evaluation. We worked on these tasks independently, that is the output from the concept extraction task was not inputted to the assertion task. We used a classical natural language processing approach consisting of tokenization and part-of-speech tagging, noun-phrase and adjective phrase identification. To assign a semantic category we created three dictionaries: 1) medical problems, 2) treatments, and 3) tests from different UMLS knowledge sources. We then matched syntactic categories with each of the entries in the dictionaries to determine a candidate semantic category. In the assertion task, medical problems were classified using the Naive-Bayes component of Weka . The set of concepts was pre-processed to be inputted to Weka.

1. Introduction

The goal of the 2010 i2b2/VA Challenge Evaluation was to study three main challenges: 1) extraction of medical problems, tests and treatments; 2) classification of assertions made on medical problems; and 3) relations between medical problems, tests and treatments. The data for this challenge included discharge summaries from Partners HealthCare and from Beth Israel Deaconess Medical Center, as well as discharge summaries and progress notes from University of Pittsburgh Medical Center. All records have been fully de-identified and manually annotated for concept, assertion, and relation information.

A detailed description of the challenge can be found at the i2b2 website [1]. We worked on tasks 1 and 2, and submitted results for task 2. In the following sections we describe our system that is based on a hybrid approach that combines dictionary look-up, pattern matching and machine learning techniques.

2. Concept Extraction and Semantic Categorization

Extracting medical concepts from (NPs) and adjective phrases (APs) required of the following steps: tokenization, part-of-speech tagging, and the identification proper as a concept [2,4,5,6,7].

Tokenization was straight forward because the records were already pre-processed, in a sense, which facilitated the task.

Part-of speech tagging requires of a dictionary. We created an ad-hoc dictionary, using the syntactic information for each lexicon entry in the UMLS SPECIALIST Lexicon [3] that includes many biomedical terms. Medical records were processed to assign parts-of-speech to the tokens formerly

Identification of concepts was performed with shallow parsing. We created simple rules to identify noun phrases and adjective phrases following the guidelines for concepts specified by the organizers of the task. Analysis of the noun phrases and adjectives phrases thus obtained showed that accuracy was low with respect to the marked data provided as testing set. Further work and analysis of these rules is ongoing.

Semantic Categorization included the creation of three glossaries terms, namely medical diseases, treatments, and tests. We used various UMLS knowledge sources for this purpose. We extracted the NPs and APs for further analysis to classify them into one of three categories: 1. medical problems, 2. treatments and 3. tests. Pattern matching and dictionary-lookup was performed to assign a semantic category to the extracted concepts. We observed that there was more

than one candidate category for some concepts. We are working on concepts and their context to calculate frequencies and probabilities to assign weights to semantic categories and select the one with the highest score.

3. Assertion Classification

We use machine learning techniques for assigning assertions to marked concepts in clinical data. Unlike the task of identifying concepts in unmarked data, this problem appeared relatively straightforward in that the end result is limited to one of six possible assertion values per concept/sentence pair. The assertion values were predefined by the organizers of the task as 1. present (patient has concept), 2. absent (patient does not have concept), 3. possible (patient may have concept), 4. conditional (patient has concept under specific conditions), 5. hypothetical (patient may develop concept in the future), and 6. associated with someone else (problem exists but not for the patient in question).

We considered that it should be possible to perform the classification without having to fully ascertain the meaning of each sentence. Given a small fixed set of possibilities, a basic metric to measure success would be whether or not our algorithm performed better than a purely random assignment of assertions. Given a set of six possibilities, a purely random assignment of assertion values would correspond to a 16.67% success rate.

Given the initial test data however, we determined that not all assertion values occur with equal frequency; the "present" value in particular occurred with roughly 67% frequency. Thus, a simple process assigning "present" to each input could potentially be correct some 70% of the time. Our goal then was to demonstrate that our algorithm could achieve more than 70% accuracy.

Our first objective was to create a mapping

of known assertions to their corresponding full-text sentences from the initial set of given training data. This was accomplished using a rather straightforward Perl script. An additional script would subsequently group the test data into each of the six assertion value types, and perform additional pre-processing as well to normalize the input.

Part of the pre-processing included converting letters in each sentence to lower case and various punctuation characters were converted to upper-case keywords such as PERIOD, SEMI (for semi-colon), etc. The concept itself was replaced with the keyword CONCEPT. Each keyword was assigned its own leading and trailing space and all whitespace found within the sentence was converted into one space value. In addition, leading space was added to both the beginning and end of a sentence.

Our initial assessment of the training data was that capitalization would not provide reliable evidence of a proper noun since its use was entirely voluntary by the Medical Doctor; thus it was eliminated as a source of variation. This would later prove to be valuable as 'markup' of the text could be written in uppercase and easily detectable by a script or human eye.

We determined that unlike most English, the test data represented 'notes' meant to be understood by a trained professional in an implied context. Rather than create one or more ambiguous parse trees for each sentence, for our initial attempt we considered it would be more straightforward to operate on a working copy of the data 'in-place'. In this sense, we are to some extent generating the most probable parsing of the sentence, but rather than use a trained support-vector-machine to generate the parsing we used human-generated regular-expressions in ways that we felt would produce the most likely parsing of the sentence.

Regular-expressions would prove to be

valuable in quickly replacing variations in text with standard keywords. By replacing text with keywords, sentences began to take on greater and greater regularity. Keywords could in turn be merged or replaced with other keywords when suitable. Where a parsing may be likely but perhaps not conclusive, the keywords were left alone as it was expected that this would be accounted for in the machine-learning classifier.

Adding leading and trailing space to the sentence proved invaluable in this regard, as multiple versions of each regular expression were not needed to account for keywords at the beginning and end as well as the middle of a sentence. In all we used over 400 regular-expressions and identified 169 features for use in the machine-learning classifier.

Simultaneously, we performed an N-gram generation test to determine if reliable features could be identified without this process. We took each sentence in turn and built from it a list of all single words, then two adjacent words, and so on up through 10 words or the end of the sentence was reached. We then generated a unique list from this data and scanned the entire dataset for occurrences of each N-gram in each subset of training data (subset by assertion type). What we found was no N-gram occurred with greater than 3-10% frequency in any of the assertion types.

Our initial attempt at classification was carried using the Naïve Bayes Algorithm [8], because it is straightforward and it is well known for generating good results.

A Naïve Bayesian Classifier is an algorithm that uses the Bayes' Theorem to separate or 'classify' a set of inputs, based upon a prior set of pre-classified inputs. Given a set of examples for each of the six assertion types, each example containing the presence of one or more features, a complete list of observed features can be determined and each example can be represented as a row in a

'features matrix'. Within this features matrix, the presence or absence of a feature in each example is represented as either 'yes' or 'no'.

Relative to other probabilistic classifiers, the Naïve Bayesian Classifier is straightforward and assumes that each feature occurs independently of each other and contributes independently to the probability of the assertion type occurring. This proved to be of particular value to our classification process, as feature dependence would otherwise require a dataset that scales exponentially with the number of features. Given that we did not know at the time how many features we may define, and our final declaration of 169 features, our need for example data representing each and every possible combination of features would have far outstripped the data available to us. Moreover, it is worth noting that our iterative approach of merging features where appropriate into more unique and specific features may contribute to encapsulating some of the relatedness of the prior features.

We grouped our keyword-laden sentences by assertion type and reduced them each into a group of keywords. We then scripted a process to take this data, along with the list of unique features found, and produce an ARFF format file used by the Weka machine-learning software library [9]. The ARFF file specified each of the 169 features, with a possible value of either yes or no, representing their presence or absence in a sentence. It also contained keywords from each of the roughly 2800 sentences in the original training set.

Training the Weka implementation of the Naïve Bayes Classifier is a relatively straightforward process where the previously mentioned ARFF file is used as input and a Weka 'model' formatted file is obtained as output. This model is subsequently used along with test input to produce output indicating the most probable

assertion for each test input as determined by Weka and the model. A Perl script was used to map the results to the original test input.

The initial training set of 2887 example concepts marked as “problems” and released by i2b2 was used as our primary training set. The subsequent release of the full 11968 assertion examples by i2b2 became our initial test set. The initial 2887 example assertions proved to be 90% effective in correctly determining the assertion type of the full 11968 example assertions (of which the set of 2887 examples was a subset). Repeating this process using the entire 11,968 sentence assertions training set we were able to correctly classify the i2b2 test set of 18,550 assertions 86.7% of the time, based on the ground truth set that was subsequently released.

In both cases the number of incorrectly-identified cases was spread fairly evenly throughout all six assertion types (between 8% and 12%); implying that we had not taken advantage either knowingly or unknowingly of the dominance of one type of assertion or another to generate our results.

A closer examination of the sentences incorrectly tagged showed a dominance of failures where the nature of the concept had not been taken into account. For example, allergies by their very nature are conditional with the presence of an allergen. However, our algorithm would classify the sentence, “Patient has allergies.” as ‘concept present’ rather than ‘concept conditional’.

4. References

1. i2b2 NLP Challenge website. https://www.i2b2.org/NLP/Relations/

2. D. Jurafsky D. and J.H. Martin. Speech and Language Processing. Second Edition. Pearson. 2009.

3. Unified Medical Language System (UMLS) website. http://www.nlm.nih.gov/research/umls/

4. N. Bennett, Q. He, K. Powell, and B. Schatz (1999) , Extracting Noun Phrases for All of MEDLINE. *AMIA '99 (American Medical Informatics Assoc) Annual Conf*, Washington, DC, Nov, 671-675.

5. Y. Huang, H.J. Lowe, D. Klein and R.J. Cucina. (2005), Improved Identification of Noun Phrases in Clinical Radiology Reports Using a High-Performance Statistical Natural Language Parser Augmented with the UMLS Specialist Lexicon. *Journal of the American Medical Informatics Association. 12(3): 275-285. May-June 2005.*

6. Q. Li and Y.F. Brook Wu Identifying important concepts from medical documents. (2006). J*ournal of Biomedical Informatics. 39(6):668-679. December 2006.*

7. S. M. Meystre and P. J. Haug. Natural Language Processing to Extract Medical Problems from Electronic Clinical Documents: Performance Evaluation. (2006). J*ournal of Biomedical Informatics. 39(6):589-599. December 2006.*

8. T. Mitchell. Machine Learning. McGraw Hill 1997.

9. *http://*www.cs.waikato.ac.nz/ml/**weka/**