Progress Report

In the past 6 months, we have been working on matching a quality term (Q-term) to a PATO term. We have tried two approaches, and I will elaborate on them in the following sections.

# First Approach

In the following sections, unless specified otherwise, the similarity of two terms are calculated using perl WordNet::Similarity, and the PATO structure is accessed by Java OWLAccessor.

## Description

The first approach can be described as below:

1. Identify the category [FNA gloss category] term *C* that a Q-term *Q* belongs to. We call *C* **parent** of *Q*, and all other terms under *C* **siblings** of *Q*.

[Tried, then dropped the following step to expand PATO search space to its entirety]

1. Find the PATO match *A* of term *C* using manually matched results [Alyssa’s mapping results FNA-PATO]. Find all children terms of *A*. We call these children terms **cousins** of *Q*.
2. Find 5 terms most similar to *Q* under the category *C*. We call the 5 terms **closest siblings** of *Q*.
3. Compare each cousin (along with its PATO synonyms OR definition keywords, using maximum) of Q to closest siblings of Q, and select 5 cousins with highest
   1. Maximum similarity with the closest siblings.
   2. Weighted average similarity with the closest siblings.

The 5 cousins are recommended matches of Q.

## Limitation

The limitations of this approach are:

1. The PATO match A of category C could be a leaf node in PATO, which means it does not have any children.
2. Even though the PATO match A is an internal node, the best match of Q could be a term that is not A’s child. The above 2 limitation led to the drop of step 2 noted above.
3. If all siblings of Q are not similar to Q, accuracy of the result could be adversely affected.
4. We thought the category and siblings of a Q-term can provide information of the context of the Q-term. However, it turned out that the information of context is not adequate to distinguish two senses of a same term using this approach. [when use sibling terms from FNA gloss, they may still be rather remote to the target term; when use dictionary.com sibling terms, they may be too general-a mix of multiple senses. For example, dictionary.com provides synonyms for both “twisted(shape)” and “twisted(attitude)”, making “twisted(attitude)” the best matched sense]

## Solution & Source Code

To overcome limitation 1 and 2, we have expanded the cousin space to whole PATO terms, which means we have removed step 2. Run [approach1runner.pl](MaptoPATOCode/approach1runner.pl) to obtain results using the modified approach (after dropping step 2).

To overcome limitation 3, we decided to extract synonyms from dictionary.com as a term’s siblings. The automatic extraction of synonyms can be done by HTML::TreeBuilder and HTML::Scanning. Please see [extractSynonyms.pl](MaptoPATOCode/extractSynonyms.pl)::getSection. To obtain similarity results using synonyms from dictionary.com, change the sibling-selection logic in [approach1runner.pl](file:///C:\Documents%20and%20Settings\Hong%20Updates\Desktop\TODO\PATOMap\PATOMap\MaptoPATOCode\approach1runner.pl) (see [runner2.pl](MaptoPATOCode/runner2.pl) [stiff] and [runner3.pl](MaptoPATOCode/runner3.pl)[twisted] for examples).

## Results

[Mapping Result111205.xlsx](Results/Mapping%20Result111205.xlsx): Result of the original approach with limitations. The best matches generated by the program sometimes include antonyms. We thought of using other similarity module such as lesk to solve the problem.

[CompareResult\_LeskAndVector111207.xlsx](file:///C:\Users\Zilong%20Chang\Documents\WORK\PATOMap\Results\CompareResult_LeskAndVector111207.xlsx): Results for comparison of vector and lesk method using original approach. We have decided to use vector since basically lesk does not solve the problems of antonyms.

[folder vector\_wholesapce\_output](Results/vector_wholesapce_output): Results after expanding cousin space (search entire PATO), but before introducing synonyms from dictionary.com.

[twisted\_allsyn\_wa\_op](Results/twisted_allsyn_wa_op.txt): Matches of word “twisted” after expanding cousin space and introducing synonyms from dictionary.com.

[stiff\_allsyn\_wa\_op](Results/stiff_allsyn_wa_op.txt): Matches of word “stiff” after expanding cousin space and introducing synonyms from dictionary.com.

## Perl Packages Needed:

[downloadable from CPAN, using commandline command in Linux or ppm under windows]

WordNet::Similarity

WordNet::QueryData

WordNet::InfoContent

HTML::TreeBuilder

HTML::Scanning

# Second Approach

The second approach is inspired by the methodology proposed in this paper (Figure 2):

<http://www.tc.umn.edu/~liux0395/fp143-liu.pdf>

This approach instead uses vector space (file wordvectors.dat) generated from our own corpus (all fish PDFs).

## Description

1. Generate vector space using [genVectors.sh](file:///C:\Documents%20and%20Settings\Hong%20Updates\Desktop\TODO\PATOMap\PATOMap\MaptoPATOCode\genVectors.sh). genVectors.sh runs huge-count.pl and vector-input.pl, which requires Text::NSP and UMLS::Similarity. Replace the content of wordvectors.dat with the output of genVectors.sh, but retain wordvectors.dat format. The new wordvectors.dat will be used by WordNet::Similarity::Vector module to generate similarity measurements.
2. Use [vector.pm](MaptoPATOCode/vector.pm) (Zilong’s version) to replace the vector.pm module in WordNet::Similarity. The new vector.pm performs the following functions for approach2runner.pl
   1. Use context sentences (from [mysql database](MaptoPATOCode/fishsentences.sql)) of the Q-term Q to build vector VQ.
   2. Use combined definition keywords of a PATO term M and its parent (from [mysql database](MaptoPATOCode/patokeywords.sql)) to build vector VM in the vector space.
3. Run [approach2runner.pl](file:///C:\Documents%20and%20Settings\Hong%20Updates\Desktop\TODO\PATOMap\PATOMap\MaptoPATOCode\approach2runner.pl), which calls vector.pm
   1. Calculate the similarity based on the two vectors VM and VQ.
   2. Repeat step 3 and 4 for each PATO term.
   3. Select the PATO term with the highest similarity.

## Results

The results [output120306](Results/output120306) are not satisfactory. The reason is still under research. Have tried to expand corpus from character descriptions to entire PDFs, didn’t help. Tried use dictionary.com definition, then character descriptions (where a term appears) as context sentences. Expanded PATO definition keywords to include immediate parents’ def keywords. None of the above helped.

## Perl Packages Needed

From CPAN

WordNet::Similarity

WordNet::QueryData

WordNet::InfoContent

UMLS::Similarity

Text::NSP

HTML::TreeBuilder

HTML::Scanning