IDA2: Intelligent Discovery of Acronyms and Abbreviations

By Adam Mallen with advising from Dr. Craig Struble and Dr. Lenwood Heath

A project for the 2009 summer REU program for the MSCS dept. of Marquette University

Introduction

The purpose of IDA2 is to develop a database which stores abbreviations and acronyms (short forms) and the associated definitions (long forms) found in Medline abstracts. In addition to this dictionary of abbreviation and definition pairs, the database contains references to all the Medline abstracts which include a given short form/long form pair.

Medline is a collection of about 18 million biomedical publication abstracts available publicly through PubMed [1].

Abbreviation Finder

To find abbreviations and the associated expanded forms we implemented the Schwartz and Hearst algorithm for identifying abbreviation definitions in biomedical text [2]. There are two main steps our program performs on each block of abstract text:

1) Identify short form and long form candidates

Abbreviation candidates are found by matching one of two patterns:

(i) long form '(' short form ')'

e.g. Following a baseline examination, including assessments of clinical attachment level (CAL), careful instruction was given.

(ii) short form '(' long form ')'

e.g. Following a baseline examination, including assessments of CAL (clinical attachment level), careful instruction was given.

Long form candidates have no more than min(|A|+5, |A|*2) words, where |A| represents the number of characters in the short form.

2) Identify the correct long form

The correct long form is found in the following steps:

- Start at the end of both the short form and long form candidates
- Move right to left trying to match the characters in the short form to characters in the long form candidate
- Characters from the short form match to characters anywhere in the long form candidate
- The first character of a short form must match the first character of a word in the long form candidate.

For example, take the short form/long form candidate pair

<'CAL', 'following a baseline examination, including assessments of clinical attachment level'>

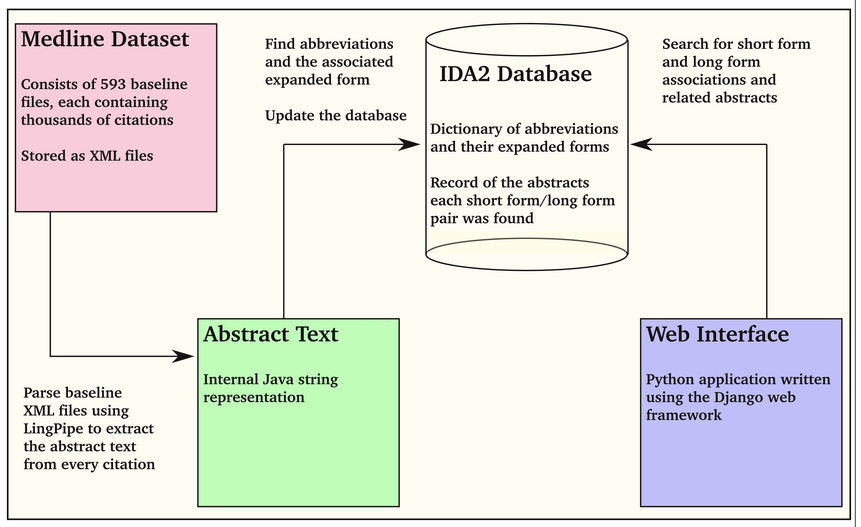
The character matches are shown using capial letters.

CAL would match to Clinical attAchment leveL.

The resulting correct short form/long form pair would be

<CAL, clinical attachment level>

System Design



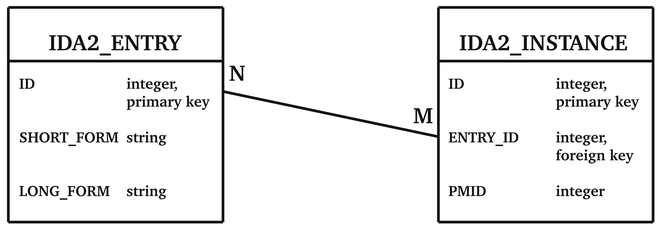
IDA2 Database

Our driver program constructs a MYSQL database in three steps:

- Parses Medline citations with LingPipe to extract the abstracts [3]
- Runs the Schwartz and Hearst abbreviation finder algorithm
- Connected to and populated the database using the JDBC Java package.

We accelerated processing by using the BISTRO Condor pool to run up to 40 of these programs at once, populating the database in parallel. This allows processing of the entire Medline baseline over night instead of taking days.

The following is the schema used by the IDA2 database.



The IDA2_ENTRY table acts as a dictionary which keeps a record of all the short form/long form pairs found in the Medline abstracts.

The IDA2_INSTANCE table keeps a record of each instance where the abbreviation finder discovered a short form/long form pair.

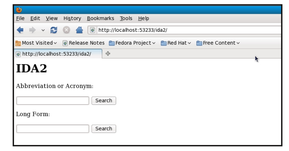
Results

Total short form/long form pairs:	1,497,702
Total abstracts containing abbreviations:	4,126,655
Pairs with only one abstract:	1,116,530
Average abstracts per pair:	5.4732
Max abstracts per pair:	30,900
Total unique short forms:	365,792
Average long forms per short form:	4.0944
Max long forms per short form:	4375

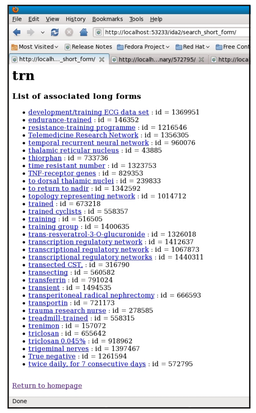
Web Interface

To interact with the IDA2 database we developed a front end user web interface using the Django Python web framework [4]. This web interface allows users to search for all related long forms for a given a short form, all related short forms for a given long form, and links to all Medline abstracts which contain a given short form/long form pair.

The IDA2 homepage



Search results for the short form 'TRN'



Abstracts containing the short form 'TR' and long form 'thalamic reticular nucleus

Future Work

There are two major tasks that are useful next steps for this project.

1) Clustering different expanded forms into one long form

For example, the following short form/long form pairs are treated as distinct pairs in the IDA2 database, but refer to the same concept.

<AD, Alzheimer disease>, <AD, Alzheimer's disease>, <AD, Alzheimer type dementia>, and <AD, Alzheimer dementia>

2) Disambiguating global abbreviations

Global abbreviations are ambiguous because they are not accompanied with the related expanded form. Machine learning algorithms can be learned on the IDA2 database and used to predict the long form for an ambiguous short form.

References

[1] PubMed: http://www.ncbi.nlm.nih.gov/pubmed/

[2] Schwartz and Hearst, "A Simple Algorithm for Identifying Abbreviation Definitions in Biomedical Text" Pacific Symposium on Biocomputing, 2003

[3] LingPipe: http://alias-i.com/lingpipe/

[4] Django: http://www.djangoproject.com/