Automatic Thesaurus Generation for Minority Languages

Kevin Scannell
Saint Louis University

June 14, 2003

Project Overview

There are about 6800 languages spoken in the world. Counting generously, a modern computer operating system is available in perhaps only 25 of them.

My main goal is to add Irish to this list. I will describe some work I have done in this direction, mostly for text processing tools:

- Spell checker
- Monolingual thesaurus
- Grammar checker

I am also the team leader for the translation of GNU/Linux into Irish (actually, the only member of the team).

The Irish Language

Around 50,000 native speakers in Gaeltacht regions in the west of Ireland.

The first official language of Ireland (Article 8 of the Constitution); the language receives (for the moment) financial support from the government.

Taught throughout Ireland in the schools, but used only rarely outside the Gaeltacht.

Also important for the discussion below: major standardization in spelling in grammar in the 1940's and 1950's.

An Béal Bocht

The title of a famous book – meaning, literally, “the poor mouth”; the dictionaries define it as “persistent complaint of poverty”.

On the one hand, I have worked on this project with no funding and no special computational resources. I use a free operating system (Gentoo Linux) and free text processing tools (`sed`, `grep`, `ptx`, etc.)

My training and primary research area are not in natural language processing or linguistics; the lack of adequately trained experts in NLP is likely to be a serious problem for many minority languages.

The good news, on the other hand, is that Irish has an embarrassment of riches in terms of online dictionaries and texts which have been essential to my work. Surveyed in my paper.

Morphology

I wrote a small computer program called `morph-ga` in 1999-2000 which generates all inflected forms of a given Irish noun, verb, or adjective. It is written in C++.

Plural and genitive nouns and all verb tenses are formed by adding endings to the root. Irish words are also subject to several kinds of initial mutation.

Verbs are particularly complex; the current record holder is the verb *fuaimnigh*, (“pronounce”) which produces 87 unique forms:

Applications of `morph-ga`

- The obvious application is to spellchecking. Only a list of headwords with correct grammatical information needs to be stored, and all forms are guaranteed to appear in the final product.

- Effective corpus searching. Used locally, but also in searching the Internet with the help of the Google API (Google searches can be performed from within your own programs – I did this in Perl).

- "Spelling Standardizer": my database stores spelling variants alongside each headword. Can generate tables consisting of variants in one column and standard forms in the other. Used by one of the spellchecking packages I will describe below.

Lemmatization

The most important application of `morph-ga` is to what I call “naïve stemming”.

A trivial and inefficient approach to lemmatization is to store in memory every form of every headword in the database with information about the form (tense, number, etc.) and a pointer back to the headword.

The version I have implemented is only slightly more intelligent than this; it strips initial mutation off of a word appearing in a text and uses some simple heuristics for limiting the number of possible stems. Example worked out in my paper.

The same idea can be applied to words that are not in the database at all: a list of possible stems is created; these are run through `morph-ga` until a match is found. I can then decide manually whether to add the new headword to the database.

Lexicon building

Building a lexicon using the lemmatization approach of the previous slide requires the use of a corpus of non-trivial size. I have a suite of software tools using the Google API which searches for Irish texts on the Internet and downloads them automatically into the corpus.

There are also jobs which run nightly and download the full text of the various online Irish newspapers, discussion groups, etc.

Of course, this is not a scientific approach and the resulting corpus is not balanced or well-sampled. But at least it is pretty big: currently 5,333,310 words, 30+ megabytes of plain text.

Citations

The lemmatizer can be instructed to add citation information to the database when a word is found in a corpus text.

Citations can also be added easily by hand, which I have done for the standard (print) terminology dictionaries.

Citations are assigned two parameters: one is an “editorial weight” (so that apparent spelling problems coming from unedited text can be disposed of more easily) and an “authority weight” (e.g. pre-standard material has low authority even if well-edited).

Spellcheckers

The list of all forms of all headwords can be run through a final shell script which uses pattern matching to look for potential spelling problems.

The result is a clean list of 300,000 words (generated from almost 30,000 headwords) which is available for free and packaged for the standard Unix/Linux spellcheckers: aspell, ispell, and myspell.

Microsoft recently released an Irish spellchecker which appears to be much smaller (it reports as misspelled more than half of my list). No free access to the word list, so who knows for sure?

The aspell package has a nice feature: I wrote coarse phonetics for Irish which the package uses to generate improved suggestions for misspellings.

Thesaurus Generation

Basic strategy is to exploit existing English language thesauri to tell when two Irish words are semantically related.

In short, if two Irish words have (disambiguated) English translations which are near each other in an English thesaurus, then a “confidence parameter” is increased. When it passes a certain threshold, the words appear together in the resulting Irish thesaurus.

The major technical obstacles were the assignment of correct English translations (using limited online resources) and the disambiguation of these.

Results

The semantic equivalence classes generated by this scheme are clustered into around 1000 top-level categories, parallel to the categories found in the classical Roget's thesaurus.

The software automatically chooses representative Irish nouns for these classes, based on a combination (1) frequency (2) centrality (3) lack of ambiguity.

The final version is output as a rather large PDF file with internal hypertext links. It can also write itself as XML so I hope to make it available on the web soon.

Work in Progress

Grammar checker. Basic architecture: a sequence of pipes and filters which take a raw corpus text and add various XML markups.

I have turned the lemmatization software described earlier into a standalone part-of-speech tagger.

I am currently working on a rule-based disambiguation scheme for these tags.

This much will allow effective “local” grammar checking: looking for correct initial mutations, etc. via pattern matching in place of a full-scale parser.