A Part-of-speech tagger for Irish using Finite-State Morphology and Constraint Grammar Disambiguation

E. Uí Dhonnchadha[1,2], J. Van Genabith[2]

[1]Centre for Language and Communication Studies
Trinity College, Dublin 2, Ireland
[2]National Centre for Language Technology
Dublin City University, Glasnevin, Dublin 9, Ireland

uidhonne@tcd.ie josef@computing.dcu.ie

Abstract

This paper describes the methodology used to develop a part-of-speech tagger for Irish, which is used to annotate a corpus of 30 million words of text with part-of-speech tags and lemmas. The tagger is evaluated using a manually disambiguated test corpus and it currently achieves 95% accuracy on unrestricted text. To our knowledge, this is the first part-of-speech tagger for Irish.

1. Introduction

This paper describes the methodology used to develop a part-of-speech tagger for Irish. The tagger is used to annotate a corpus of 30 million words of text (Kilgarriff *et al*, forthcoming) with part-of-speech tags and lemmas. The text is annotated using Parole morpho-syntactic tags (ITÉ, 2002; Calzolari *et al*, 1996) and XML Corpus Encoding Standard (XCES, 2002).

The tagger is evaluated using a manually disambiguated test corpus and it currently achieves 95% accuracy on unrestricted text. To our knowledge, this is the first part-of-speech tagger developed for Irish.

Section 2 presents the methodology used in our approach. Section 3 describes our test corpus and presents overall results. Section 4 provides a detailed error analysis and section 5 concludes with an outline of further work.

2. Methodology

We implement the part-of-speech tagger in three stages: tokenization, morphological analysis of tokens and part-of-speech disambiguation of the morphological analyses. Tokenization and morphological analysis are carried out using finite-state transducers following Beesley and Karttunen (2003). The morphological analysis for each token results, in most cases, in a number of possible analyses. Constraint Grammar (Tapanainen, 1996) rules are then used to choose the correct analysis given the context in which the token is found.

A rule-based approach is used throughout in order to exploit the rich inflectional morphology of Irish. This will also form the basis for future work such as chunking and parsing.

2.1. Tokenization

Irish text can be segmented, to a large extent, according to the white space between words. In most cases a token consists of a word or punctuation symbol as in (1). However, a word may consist of more than one token (2) and a token can consist of more than one word (3). Numbers, dates and abbreviations also require special consideration.

1) "*Iontach!", arsa Seán*. "Wonderful!", said Seán. : t_1=" t_2=Iontach t_3=! t_4=" t_5=, t_6=arsa t_7=Seán t_8=.

2) *M'athair* "my father" : t_1=M' t_2=athair
D'fhéach "looked" : t_1=D' t_2=fhéach
s'againne "this one of ours" : t_1=s' t_2=againne

3) *Tar éis* "after" : t_1= Tar éis
Cé is moite "except" : t_1= Cé is moite

2.2. Morphological Analysis

The stream of tokens (4) is processed by a sequence of morphological analysers and guessers (Uí Dhonnchadha and Van Genabith, 2005) as implemented in (Beesley and Karttunen, 2003).

4) *an maith leis úlla*? "does he like apples?"

Each token is analysed, without reference to context, and receives a number of possible analyses.

5) *"<an>"*
"an" Art Sg Def
"an" Part Vb Q
"is" Cop Pres Q
"is" Cop Pres Dep Q

6) "*<maith>*"
"maith" Adj Base
"maith" Adj Masc Com Sg
"maith" Adj Gen Weak Pl
"maith" Noun Fem Com Sg
"maith" Verb VTI Imper 2P Sg

7) "*<leis*">
"leis" Noun Fem Com Sg
"le" Pron Prep 3P Sg Masc
"le" Pron Simp

8) "<*úlla*>"
"úll" Noun Masc Com Pl
"úll" Noun Masc Voc Pl
9) "<*?*>"
"?" Punct Fin Q

2.3. Constraint Grammar Disambiguation

Constraint Grammar (CG) (Tapanainen, 1996) rules are used to process the output of the morphological analyzer. In the examples below we refer to example sentence (4). CG rules operate over all the tokens in a sentence, where each sentence is delimited by a punctuation mark (e.g. .!? etc.) or an XML tag (e.g. </s>). In CG, each sentence is described in terms of cohorts (5,6,7,8,9), readings ("*an*" has four readings, "*maith*" has five, etc.) and tags (e.g. Art Sg Def). A cohort consists of a token and all the possible readings for that token. Each reading includes a lemma, morphological tags and possibly grammatical function tags.

To select the most likely morphological analysis for an ambiguous token, CG uses other cohorts within the sentence. A positional reference system is used whereby the cohort under consideration is at position 0, the following cohort is at position 1 and the previous cohort is at position –1, and so on.

There are two basic types of rule: 'select' and 'remove'. Select rules are used to choose the appropriate reading from a cohort based on the context to the left and/or right. If this is not possible, remove rules may delete impossible/unlikely readings from a cohort based on the context, and hopefully only the correct reading will remain.

(10) gives an example of a rule which selects noun reading(s) if the previous token is unambiguously (C=careful) an article. Example (11) shows a rule which removes the vocative case reading from "*úlla*" in (8), as it is not preceded by a vocative particle.

10) SELECT (Noun) IF (-1C (Art));
11) REMOVE (Noun Voc) IF (NOT -1 (Part Voc));

In cohorts (5) to (9) we cannot select readings straight away but we can remove readings. For instance, an interrogative verbal particle "*an*" cannot precede an imperative verbal form, therefore we can remove this reading, leaving just the possibility of article or copula. In the case of "*maith*", since a verb cannot directly follow an article or a copula this reading is removed, leaving noun and adjective readings. Further CG rules are applied until the analysis (12) is achieved.

12) *"<an>"*
"is" Cop Pres Q
"<*maith*>"
"maith" Adj Base
"<*leis*">
"le" Pron Prep 3P Sg Masc
"<*úlla*>"
"úll" Noun Masc Com Pl
"<*?*>"
"?" Punct Fin Q

In order to implement the tagger we coded over three hundred CG disambiguation rules based on the main syntactic patterns in Irish. Some tokens remain ambiguous as we conservatively leave a choice of readings rather than risk an incorrect selection.

3. Evaluation

Our evaluation focused on two aspects of the disambiguation process a) the percentage of ambiguous tokens remaining after processing and b) the accuracy of the disambiguation. In looking at ambiguous tokens our aim is to ascertain the cause of the ambiguity and to devise new rules, where possible, to eliminate the ambiguity. Currently 97% of tokens in our test corpus (see below) are disambiguated with regard to POS category. However, 7% of these tokens still contain some inflectional or lemma ambiguity.

3.1. Test Corpus

In order to assess the accuracy of the disambiguation process, we created a sub-corpus of 3,000 randomly selected sentences (70,000 tokens approx.) from the 30 million word corpus. This sub-corpus was then randomly distributed into two parts, with approximately two thirds for development and one third held-out for testing of CG based disambiguation rules. Each token in the development and test corpus was manually disambiguated.

3.2. Overall Analysis

The test corpus was automatically disambiguated by our CG tagger. The outcome was a 95% accuracy for part-of-speech category (increasing to 96% when ambiguous tokens are included). However if additional features are taken into consideration (e.g. case, number etc.) accuracy drops to 93%.

We evaluate each stage of the tagging process by examining the differences between the manually tagged text and the automatically tagged text. In this manner, we are able to determine whether the discrepancies arise at the tokenization, morphological analysis or POS disambiguation stages.

4. Error Analysis

4.1. Tokenization

Tokenization is assessed by comparing the alignment between the automatically tokenized output and the test corpus. Misalignments are due either to shortcomings in the tokenizer or malformed input. All of the tokenizer problems encountered related to instances where punctuation should not have been separated from the token, as in the case of contractions like *im'* meaning "*i mo*" (in my) etc. This was addressed by adding specific regular expressions to the tokenizer to deal with these cases.

Tokenizer problems:

- contractions, e.g. *im'* (*i mo* - in my) *a's* (*agus* - and), *'un* (*chun* - towards), *a'* (*an/ag* - the/at)
- numbers with trailing punctuation, e.g. *1995.*
- items which should stay together, e.g. *(iii), (B)*
- email & urls, e.g. *panceltic@eircom.net*, *www.oneworld.com*

- abbreviations *gCo.*, *Uimh.*, *CD-ROM*
- proper names with English genitive, e.g. *Madigan's*, *Pete's Pizzas*

A certain amount of malformed input will always be encountered when processing large amounts of unrestricted text. However, malformed strings in the development sub-corpus, were corrected in order to provide a good basis for rule induction at a later stage.

Malformed input:
- word split into two, e.g. *fan- faidh*, *rach aidh*,
- two words joined, e.g. *séamach*, (*sé amach*), *céacu* (*cé acu*)

4.2. Morphological Analysis

Each token is analysed for lemma, part-of-speech category, and other morpho-syntactic features (tense, mood, gender, number etc.). Each token is assigned one or more analysis by one of the following methods:
- finding the token in the finite-state lexicon
- finding the root of the token in the lexicon plus standard prefixes or suffixes
- finding a compound made up of items found in the lexicon,
- predicting the token's characteristics based on inflectional or derivational affixes, capitalisation, vowel structure of the final syllable, or other distinctive characteristics.

On average 95% of tokens are found in the lexicon using one of the first three methods, and are associated with an appropriate lemma, part-of-speech category and morpho-syntactic features. The remainder are assigned a guessed analysis using predictive strategies (guessers). They tend to produce the correct POS and features, although the lemmas produced tend to be more problematic due to internal changes to morphemes. In a few cases more than one possible head is identified in a compound, and the correct one may not always be chosen (see Uí Dhonnchadha and Van Genabith, 2005).

Discrepancies between the manually tagged and automatically tagged text can arise at the morphological analysis stage when the analysis assigned manually is not provided by the morphological analyzer. Either the lexeme is in the lexicon but is missing the required analysis, or it is not in the lexicon and the guessed analysis is not the appropriate one for the context.

Examples of missing lexemes
- non-standard spelling of function words e.g. *fúithe* (*fúithi* - under her), *léithe* (*léi* - with her), *daofa* (*dóibh*/*díobh* - to them/from them)
- non-standard verb forms, e.g. *atáid*, *bhfuilimíd*, *bhfeacaíos*

Examples of missing analyses
- adj. as well as noun analysis required, e.g. *coiteann* (common)
- verbal noun as well as common noun required e.g. *iascaireacht* (fishing)

There are two ways of dealing with missing lexemes, a) by adding more items to the lexicon and b) by improving the guessers' strategy. The longer term objective is to increase the lexicons (which currently contain approx. 30K stems), but an examination of the types of items which are missing reveals that we can deal with some specific areas immediately. Apart from missing open class words (nouns, verbs and adjectives), and those caused by typographical errors, we find that many of the members of the functional class of conjugated prepositions (prepositional pronouns) have dialectal variants which can be added to the lexicon. In addition, a number of non-standard inflected verb forms not included in current reference grammars (Bráithre Críostaí, 1999), are now being generated by the morphological transducers.

Additional analyses are added for specific lexemes where necessary, but we do not say that all nouns can be adjectives or vice versa, as this would introduce a detrimental level of ambiguity.

Particular attention was paid to the effectiveness of the guessers as we can never expect to achieve 100% coverage with the morphological analysers alone. The order in which guessers are tried is important. In our implementation a sequence of guessers are tried until one succeeds, e.g. first we check if the unrecognised token could be a prefixed or suffixed lexeme, or if it has a verb inflectional suffix, or if it is a compound of two lexemes. If these fail we look at the last syllable of the token for clues as to its POS and morphological features.

A problem was noted where some common word-final morphemes e.g. *-each(a), -acht* as in (13) and (14) ,which are homonyms of lexemes *each* and *acht*, were being analysed as compound heads. A new guesser was introduced to test for such word-final morphemes before testing for compounds.

13) *truslógacht* (jumping) not *truslóg* (jump) + *acht* (act - of parliament)
14) *sheanmháthaireacha* (grandmothers) not *sheanmháthair* (grandmother) + *eacha* (horses)

Another issue which came to light is the fact that in some cases it is necessary to supply multiple guessed analyses. Previously only the most likely analysis was generated but there are a few suffixes where several analyses are equally likely, e.g. *-adh* could indicate either a verb form, a common noun or a verbal noun, also *-tí* could be a verbal or nominal suffix.

4.3. CG Disambiguation

CG disambiguation achieves overall 95% accuracy for part-of-speech category. In order to focus our efforts in improving performance, individual accuracy statistics were computed for each of twenty part-of-speech categories. Table 1 shows individual results (from the development sub-corpus) ranging from 99% to 61%. In order to improve results we focus our attention on the distinction between noun classes as they account for the greatest number of tokens proportionally, and also the copula and numerals.

Table 1 Tagging accuracy per POS category

POS Description	POS	No. of tokens	% Correct
Article	Td	2913	99
Adverb	R	833	98
Pronoun	P	3082	96
Preposition	Sp	7119	95
Conjunction	C	2706	94
Particle: verb.	Q	1600	94
Noun: comm.	Nc	11429	93
Verb	Vm	3762	92
Abbreviation	Y	142	89
Determiner	D	1365	88
Adjective	Aq	2110	87
Noun: subst.	Ns	303	87
Noun: proper	Np	1715	86
Noun: verbal	Nv	1718	80
Particle: oth.	U	497	80
Foreign	X	510	78
Copula	W	770	77
Interjection	I	15	73
Adj.: verbal	Av	331	68
Numeral	M	754	61

We also examined the most common types of POS ambiguity, presented in Table 2. The most common type of ambiguity is between two classes of noun. In tagging, we distinguish between common nouns (15) and verbal nouns (15) (most of which are de-verbal or de-agentive). Verbal nouns can occur in syntactic constructions where common nouns cannot (16). However, ambiguity arises from the fact that all verbal nouns can function as common nouns (18).

15) *an doras* (the door) - art + noun
16) *ag freastal* (serving) - prep (asp.) + verbal noun
17) **ag doras* (dooring) - prep (asp.) + noun
ag doras (at door) - prep (loc.) + noun
18) *an freastal* (the service) - art. + noun

Table 2 Ambiguous POS classes

Ambiguous Classes			tokens	%
N comm.	vs.	N verbal	283	11.5
N comm.	vs.	Adj	240	9.7
N comm.	vs.	Verb	179	7.2
N comm.	vs.	N proper	153	6.2
N prop	vs.	Foreign	122	4.9
N verbal	vs.	Verb	113	4.6
Prep.	vs.	Det. poss.	82	3.3
Prep.	vs.	V particle	76	3.0
Number	vs.	Prep	75	3.0
N comm.	vs.	Foreign	66	2.6
180 other types of ambiguity			1080	43.7
Total			2469	100.0

Noun-adjective ambiguity is the next most common type of discrepancy between the manually and automatically tagged texts. Analysis of the results in Table 2 is ongoing and will lead to refinement of the existing rules and the creation of new rules. Table 2 shows that the ten most common ambiguities account for over half of the differences encountered between the two corpora.

5. Further Work

Work has begun on automatically inducing new CG rules by processing the manually tagged corpus, following Samuelsson *et al*. (1996). Bigram and unigram frequency statistics are used to identify likely and unlikely tag combinations. Data on phrase patterns are also collected. This information is used to automatically create CG rules. These rules it is hoped will complement the hand-written CG rules and aid us in reaching our target of 97%-99% accuracy, comparable to state-of-the-art taggers for English and other technologically advanced languages.

6. References

Beesley K. & Karttunen L., 2003. *Finite State Morphology*. CSLI Publications: California.

Bráithre Críostaí, 1999. *Graiméar Gaeilge na mBráithre Críostaí.* 2nd. ed. Baile Átha Cliath: An Gúm

Calzolari, N., Monachini, M., (1996). *Multext*. http://www.lpl.univ-aix.fr/projects/multext/LEX/LEX1.html (accessed March 2006).

ITÉ, (2002). *Parole Common Morphosyntactic Tagset*. http://www.ite.ie/pos.htm (accessed March 2006).

Kilgarriff, A., Rundell, M. and Uí Dhonnchadha, E., (forthcoming). *Efficient corpus development for lexicography: building the New Corpus for Ireland*. In: Linguistics Resources and Evaluation Journal (LREJ).

Samuelsson, C., Tapanainen, P. and Voutilainen, A., 1996. *Inducing Constraint Grammars*, published in L. Miclet and C. de la Higuera (Eds.), Grammatical Inference: Learning Syntax from Sentences. Lecture Notes in Artificial Intelligence 1147. ICGI'96, Springer.

Tapanainen, P., 1996. *The Constraint Grammar Parser CG-2*. Publication No. 27, University of Helsinki.

Uí Dhonnchadha E., and Van Genabith, J., 2005. *Scaling an Irish FST morphology engine for use on unrestricted text*. Fifth International Workshop on Finite-State Methods in Natural Language Processing, Helsinki.

XCES, 2002. *Corpus Encoding Standard for XML*. http://www.cs.vassar.edu/XCES/ (accessed March 2006).