IceParser: An Incremental Finite-State Parser for Icelandic

Hrafn Loftsson & Eiríkur Rögnvaldsson 2007

Commentary: Matthew Whelpton

IceParser

- Overview
  - What is *IceParser*?
  - What does *IceParser* output?
  - Why is the input to *IceParser* challenging?
  - How well does *IceParser* work?
  - What kinds of errors does *IceParser* make?

What is *IceParser*?

- IceParser is
  - the first parser published for Icelandic
  - a shallow parser
  - which follows the constructive method of parsing
  - working incrementally on the input
  - using finite state transducers
  - implemented in Java

A shallow parser

- Full parsing
  - Complete parse tree
  - every constituent identified
  - from root to leaves
- Shallow parsing
  - Only major chunks of the sentence are analysed
  - in particular the constituent relations of the verb to other phrases is not made explicit

A shallow parser

- Shallow parsers are
  - sufficient in most cases for
    - information extraction
    - text summarisation
    - some kinds of grammar checking
  - well suited to handling
    - low quality and fragmented input
    - spoken language

the constructive method

- Grefenstette, 1996; Abney, 1997
  - introduces syntactic boundary tags into the input string
    - the man left
    - [NP the man NP] [VP left VP]
  - recognises the positioning of tags on the basis of lexico-syntactic patterns
    - a determiner followed by a noun are a noun phrase when followed by a verb

the constructive method

- finite state transducers
  - one transducer for each kind of boundary to be introduced (e.g. NP or VP)
- cascade
  - individual transducers are strung together
  - output from one is input to the next
- incremental
  - so the parsing is incremental, i.e. in steps
  - boundary tags are added in successive sweeps of the text

Java implementation

- Implemented in Java with JFlex
  - creates a Deterministic Finite-state Automaton (DFA)
  - extremely efficient
- JFlex rather than XFST because *IceParser* part of the NLP toolkit for Icelandic – all in Java

What does *IceParser* output?

- Input
  - POS tagged text
  - Icelandic Frequency Dictionary tagset
- Output
  - Shallow annotation schema (hand-made for IceParser)
    - Constituent structure annotation (Module 1: 14 transducers)
    - Syntactic function annotation (Module 2: 8 transducers)

Constituent Structure

- Constituents
  - Core phrases: AdvP, AP, NP, PP, VP
  - Clause building: CP (coordinating), SCP (subordinating)
  - Other: InjP (interjections), MWE (multi-word expressions, extra-syntactic collocations)
- Bottom-up analysis
  - most embedded constituents first
  - AdvP > AP > NP > ...
  - Generally adverbs modify adjectives and adjectives modify nouns
  - [NP [AP [AdvP very AdvP] good AP] teacher NP]

More on constituents

- Sequences of agreeing AP and NP
  - APs; NPs
- VP subclassified
  - VP = finite verb phrase
  - VPi = infinitival verb phrase
  - VPb = verb phrase with predicative complement (cf *vera* "be")
  - VPs = supine verb phrase
  - VPp = past participle verb phrase
  - VPg = present participle verb phrase
- Crucially the VP does NOT include complements and modifiers (shallow parsing), contrary to linguistic evidence!

Transducer Operation

- Transducer *Phrase_AdvP*
  - Pattern Recognition
    - Adv={WordSpaces}{AdvTag}
    - {WordSpaces} = word characters followed by whitespace
    - {AdvTag} = adverb POS tag
  - Action
    - Input: mjög aa
    - Output: [AdvP mjög aa AdvP]

Grammatical Functions

- Eight grammatical functions
  - *SUBJ - subject
  - *OBJ - object
  - *OBJAP – object in AP
  - *OBJNOM – nominative object
  - *IOBJ – indirect object
  - *QUAL – genitive qualifier
  - *COMP - complement
  - *TIMEX – temporal expression

More on grammatical functions

- The various subject, object and complement labels allow a "relative position indicator"
  - Where is the function-assigning predicate relative to the phrase
    - *SUBJ> - predicate follows subject
    - *SUBJ< - predicate precedes subject

Func_SUBJ transducer

- NomSubj={NPNom}|{NPsNom}
- VPorVPBe={VP}|{VPBe}
- SubjVerb=({NomSubj}{WS}+{VPorVPBe}|{DatSubj}{WS}+{VPDat}|{AccSubj}{WS}+{VPAcc}
- {VPDat} – recognises verbs that take dative subjects (listed as regular expressions)
- Output
  - {*SUBJ> [NP vagnstjórinn NP] *SUBJ>} [VP sá VP] {*OBJ< [NP mig NP] *OBJ<}
  - (driver-the saw me)

What´s the challenge? Icelandic!

- Morphologically rich (inflection)
  - Nouns: 3 genders, 4 cases, 2 numbers
    - Suffixed definite article: 3 genders, 4 cases, 2 numbers
  - Adjectives: strong/weak, 3 degrees, 3 genders, 4 cases, 2 numbers
  - Verbs: 3 persons, 2 moods, 2 tenses, 2 voices (and more)
- Icelandic Frequency Dictionary tagset
  - Approx 660 tags
- Relatively free word order at the main phrasal level
  - Subject and object noun phrases
  - Preposition phrases
  - Adverbial phrases

How well does *IceParser* work?

- Gold Standard
  - Icelandic Frequency Dictionary
    - 509 sentences (8281 tokens)
  - Manually annotated by two experts using the *IceParser* annotation scheme
- F-measure = 2 * precision * recall/(precision + recall)
- Scoring program
  - *Evalb* (Sekine and Collins, 1997)

Type	Parser (Tagger)	F-measure
Constituent Tagging	*IceParser* IFD	96.7%
	IceParser IceTagger	91.9% c. -5% [tagger 91.1%]
	Knutsson et al. (2003) Swedish	88.7%
	Kokkinakis and Johansson-Kokkinakis (1999) Swedish	93.3%
Function Tagging	*IceParser* IFD	84.3%
	IceParser IceTagger	75.3% c. -10%
	Müller (2004) German	82.5%

Errors

- One AdvP instead of two
  - [PP um [NP það NP] PP] [VP vissi VP] [NP stelpan NP] [AdvP ekki þá AdvP]
  - (about that knew girl not then)
  - "ekki" [not] applies independently to the preceding sentence, not to "þá" [then]
  - [PP um [NP það NP] PP] [VP vissi VP] [NP stelpan NP] [AdvP ekki AdvP] [AdvP þá AdvP]

Errors

- Definite noun – adjective inversion
  - [NP árin NP] [AP gullnu AP]
  - (years-the golden)
  - [NP árin [AP gullnu AP] NP]
- IceParser doesn´t have a transducer pattern for post-nominal adjectives

Errors

- Overapplication of NP grouping based on case agreement (Phrase_NPs Transducer)
  - [AP sterkur AP] [VPb var VPb] [NPs [NP hann NP] [CP og CP] [NP íþróttamaður NP] NPs] [AP ágætur AP]
  - (strong was he and athlete fine)
  - Inversion of first clause means two clausal subjects side by side and therefore grouped.
  - [AP sterkur AP] [VPb var VPb] [NP hann NP] [CP og CP] [NP íþróttamaður [AP ágætur AP] NP]

Errors

- Subject postposing
  - [VPb er VPb] [AdvP ekki AdvP] [VPi að koma VPi] {*SUBJ [NP matur NP] *SUBJ}?
  - (is not to come food?)
  - IceParser correctly identifies the subject but not where its predicate is - [VPb er VPb].
  - This is because the infinitive (VPi) intervenes.
  - [VPb er VPb] [AdvP ekki AdvP] [VPi að koma VPi] {*SUBJ< [NP matur NP] *SUBJ<}?

Flow-on Errors

- Errors in the phrase structure module can cause additional errors in the syntactic function module.
  - [CP og CP] [VP tóku VP] [NP [AP [AdvP fram AdvP] eigin AP] dósir NP]
  - (and took out own cans)
  - [CP og CP] [VP tóku VP] [AdvP fram AdvP] [NP [AP eigin AP] dósir NP]
  - The original error produces an incorrect object labeling
    - {*OBJ< [NP [AP [AdvP fram AdvP] eigin AP] dósir NP] *OBJ<}
    - The object doesn´t in fact include “fram”

IceParser

- Summary
  - What is *IceParser*?
    - incremental finite state parser for Icelandic
  - What does *IceParser* output?
    - text tagged with constituent and function boundary labels
  - Why is the input to *IceParser* challenging?
    - because Icelandic is a b$%#(#rd!
  - How well does *IceParser* work?
    - competitively well!
  - What kinds of errors does *IceParser* make?
    - see above!