Jacy: an implemented HPSG grammar of Japanese

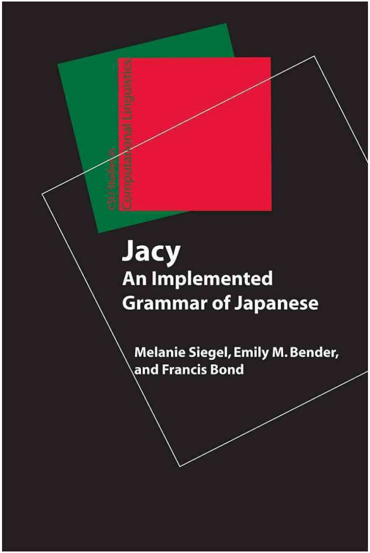
David **Moeljadi** and Takayuki **Kuribayashi**
and many more
Division of Linguistics and Multilingual Studies,
Nanyang Technological University, Singapore

The 25th International Conference on Head-Driven Phrase Structure Grammar
University of Tokyo, Komaba Campus

2 July 2018

Jacy demo: Outline

1. Introduction
   - Motivation
   - History and applications
   - Deep Linguistic Processing with HPSG Initiative (DELPH-IN)
   - Grammar engineering
   - The current state
     - Covered phenomena
     - Coverage and evaluation
     - Corpus/Treebank
2. Phenomena *DEMO
   - Argument scrambling and omission
   - -reru / -rareru verbal endings
3. Treebanking *DEMO
4. Japanese-English machine translation *DEMO
5. Conclusions and future work



Siegel, Melanie, Emily M. Bender, and Francis Bond (2016) *Jacy: an implemented grammar of Japanese*. Stanford: CSLI Publications.

Motivation

- **Applications** that rely on deep linguistic processing, such as message extraction systems, machine translation and dialogue understanding systems are becoming **feasible**
- **Requirement** for rich and highly precise information, well-defined output structures
- **Requirement** for robustness: wide coverage, large and extensible lexica, interfaces to preprocessing
- **Requirement** for extensibility to multiple languages
- **Requirement** for efficient processing
- The JACY Japanese HPSG has been developed for and used in real-world applications that require the handling of peripheral phenomena

History of the JACY grammar: Project context

- 1998-2000
  - **Verbmobil**: Machine translation of application-oriented spoken dialogues (`http://verbmobil.dfki.de/`)
- 2001-2002
  - Co-operation with YY Technologies (CA, USA): Automatic email response (Co-operation with Stephan Oepen, Ulrich Callmeier, Monique Sugimoto, Atsuko Shimada, Dan Flickinger) (`http://www.dfki.de/~siegel/jacy/jacy.html`)
- 2002-2004
  - EU project **DeepThought**: Hybrid and shallow methods for knowledge-intensive information extraction (`http://www.project-deepthought.net`)
- Lexeed project at Nippon Telegraph and Telephone Corporation: Ontology extraction, **Hinoki** treebank
- Japanese-English machine translation project with the LOGON initiative: open-source semantic transfer-based machine translation — **JaEn**

Deep Linguistic Processing with HPSG Initiative (DELPH-IN)

- a research collaboration between linguists and computer scientists
- builds and develops open source grammar, tools for grammar development and NLP applications using HPSG and MRS
    - Head-Driven Phrase Structure Grammar (**HPSG**; Pollard and Ivan A Sag, 1994; Ivan A. Sag, Wasow, and Emily M. Bender, 2003): feature structures, type hierarchy, efficient processing
    - Minimal Recursion Semantics (**MRS**; Copestake et al., 2005): flat semantic formalism, works well with typed feature structures, structures are underspecified for scopal information (compact representation of ambiguities)
- 18-22 June 2018: The 14th **Annual DELPH-IN Summit**, hosted by Berthold Crysmann (Laboratoire de linguistique formelle, CNRS & U Paris Diderot)
- **wiki** page: `http://moin.delph-in.net/FrontPage`
- **DELPH-IN discourse** (Q&A): `https://delphinqa.ling.washington.edu/`

The Development Tools

- The Linguistic Knowledge Builder (**LKB**) (Copestake, 2002): grammar development system
- Platform for Experimentation with efficient HPSG processing Techniques (**PET**) (Callmeier, 2000): a very efficient HPSG parser, for processing
- Answer Constraint Engine (**ACE**) (Packard, 2013): an efficient processor for DELPH-IN HPSG grammars
- ITSDB or **[incr tsdb()]** (pronounced *tee ess dee bee plus plus*) (Oepen and Daniel Flickinger, 1998): a tool for testing, profiling the performance of the grammar (analyzing the coverage and performance), tracking changes, and annotating treebanks
- Full Forest Treebanker (**FFTB**) (Packard, 2014): a treebanking tool for DELPH-IN grammars, allowing the selection of an arbitrary tree from the "full forest" without enumerating/unpacking all analyses in the parsing stage

Multilingual grammar development

- English Resource Grammar (**ERG**) (Dan Flickinger, 2000; Dan Flickinger, 2011)
- **Jacy** (Siegel, Emily M Bender, and Bond, 2016)
- **Zhong** (Fan, Song, and Bond, 2015), for Chinese languages (Mandarin, Cantonese, ...)
- Indonesian Resource Grammar (**INDRA**) (Moeljadi, Bond, and Song, 2015), for Indonesian
- ...
- **The LinGO Grammar Matrix** (Emily M. Bender, Dan Flickinger, and Oepen, 2002) (Emily M. Bender, Drellishak, et al., 2010): a web-based questionnaire for writing new DELPH-IN grammars

Other tools

- **delphin-viz**: DELPH-IN data structure visualizations and demo interface
  `http://delph-in.github.io/delphin-viz/demo/`
- **Demophin**: a DELPH-IN web demo
  `http://chimpanzee.ling.washington.edu/demophin/jacy/`
- **PyDelphin**: a set of Python libraries for the processing of DELPH-IN data
  `https://github.com/delph-in/pydelphin`
- **typediff**: a tool to investigate and compare phenomena in one grammar (e.g. JACY) with those in other DELPH-IN grammars (e.g. ERG)
  `https://github.com/ned2/typediff`
- Linguistic Type Data-Base (**LTDB**): a documentation containing linguistic description of lexical types, usage examples and distribution based on the grammar and treebanks, typed feature structure definitions of the lexical types
  `https://github.com/fcbond/ltdb`
  `http://compling.hss.ntu.edu.sg/ltdb/Jacy_1301/`

Grammar engineering

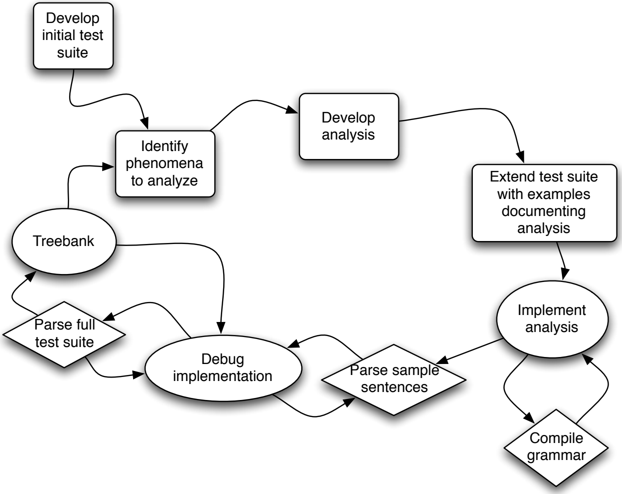


Figure: Grammar Development Cycle (Emily M. Bender, Dan Flickinger, and Oepen, 2011)

Grammar engineering

- Grammar engineering courses:
  `http://moin.delph-in.net/TeachingCourses`
- Grammar engineering FAQ:
  `http://moin.delph-in.net/GrammarEngineeringFaq`
- Feature Geometry FAQ:
  `http://moin.delph-in.net/GeFaqFeatureGeometry` (see also the cheat sheet)

Installation

- Install subversion
  ```
  sudo apt install subversion
  ```
- Install logon (see `LogonInstallation` page)
  ```
  svn checkout http://svn.emmtee.net/trunk logon
  ```
- Install Emacs
  ```
  sudo apt install emacs
  ```
- Install git
  ```
  sudo apt install git
  ```
- Install JACY
  ```
  git clone https://github.com/delph-in/jacy.git
  ```
- Install ACE
  `http://sweaglesw.org/linguistics/ace/`

The current state: grammar size

Year	2000	2001	2002	2003	2005	2008	2009	2015
Rules	27	50	51	54	47	81	86	137
Lexemes	3,399	5,369	5,681	5,147	35,220	30,898	56,944	56,914
Types	1,246	1,709	1,736	1,889	2,204	2,185	2,324	2,473

Table: Change in grammar size over time

Covered phenomena

- Verbs and adjectives
  - Inflectional and derivational rules
  - Auxiliary constructions
  - Passive constructions
  - Causative
- Nominal structures
  - Names and named entities
  - Pronouns (demonstrative, locative, personal, reflexive)
  - Nominalizers
  - Temporal nouns
  - Noun modification (relative clause)
  - Numeral classifiers
- Particles
- Adverbs
- Interrogatives
- Demonstratives
- Honorifics

Test suites and coverage

- A **test suite** is a curated collection of test items (sometimes including both grammatical an ungrammatical examples) meant to test specific properties of a grammar
  - '**mrs**': a small set of sentences, originally in English, that are meant to cover some of the basic semantic phenomena (argument structure, quantification, negation, modification etc.)
    `http://moin.delph-in.net/MatrixMrsTestSuite`
  - 'vanilla': a collection of phenomena that are specific to Japanese
  - etc.

Type	Test Suite	Total	Parsed as is		Handling unknowns	
		# Sents	# Sents	Cover (%)	# Sents	Cover (%)
Functional	mrs	135	126	93	127	94
	vanilla	120	105	87	105	87
	kinou1	1500	1321	88	1328	88
	kinou2	1099	918	83	940	85
	kinou3	1116	866	77	883	79
Natural	tanaka/tc-003	1500	1145	76	1172	78
	tanaka/tc-004	1500	1136	75	1173	78
	tanaka/tc-005	1500	1114	74	1145	76
	haikingu	104	34	32	66	63

The Hinoki Treebank

- The Lexeed corpus
  - at Nippon Telegraph and Telephone Corporation (NTT)
  - 53,600 dictionary definition sentences and 36,000 example sentences
- The Tanaka corpus
  - at the Japanese National Institute of Information and Communications Technologies (NICT)
  - 15,000 example sentences

Table: Hinoki manual annotation result

	Type	Number	%
Good	Single Good Tree	7,809	52.1
	Multiple Good Trees	679	4.5
Bad	No Good Trees	1,604	10.7
	No Parse Found	2,826	18.8
	Resource Limitation	2,082	14.0
	Total	15,000	100

JACY: a Japanese open-source HPSG

- JACY is an open-source HPSG grammar for Japanese (MIT license)
- probably the most distributed grammar development, developed by researchers in different continents (unlike ERG)
- JACY homepage:
  `http://moin.delph-in.net/JacyTop`
- Grammar sources (MIT license):
  `https://github.com/delph-in/jacy`
- On-line documentation, linguistic type database (LTDB):
  `http://compling.hss.ntu.edu.sg/ltdb/Jacy_1301/`
- Demo page:
  `http://delph-in.github.io/delphin-viz/demo`
  `http://chimpanzee.ling.washington.edu/demophin/jacy/`
- DELPH-IN mailing list to ask questions
  `https://delphinqa.ling.washington.edu/`

Some Japanese phenomena in JACY

- Argument scrambling and omission
- -*reru* / -*rareru* verbal endings
- ...

Verbal arguments scramble

Argument order is free, but arguments can not appear after the verb

(1)	フランシス	が	田中	に	ボール	を	渡す
	Furanshisu	ga	Tanaka	ni	bo-ru	wo	watasu
	Francis	NOM	Tanaka	DAT	ball	ACC	hand

”Francis hands Tanaka a ball”

(2)	田中	に	フランシス	が	ボール	を	渡す
	Tanaka	ni	Furanshisu	ga	bo-ru	wo	watasu
	Francis	NOM	Tanaka	DAT	ball	ACC	hand

(3)	ボール	を	田中	に	フランシス	が	渡す
	bo-ru	wo	tanaka	ni	Furanshisu	ga	watasu
	ball	ACC	Tanaka	DAT	Francis	NOM	hand

(4)	* フランシス	が	渡す	田中	に	ボール	を
	Furanshisu	ga	watasu	Tanaka	ni	bo-ru	wo
	Francis	NOM	hand	Tanaka	DAT	ball	ACC

Verbal arguments omission

Verbal arguments are frequently omitted even if it is the subject

(5)	フランシス	が	ボール	を	渡す
	Furanshisu	ga	bo-ru	wo	watasu
	Francis	NOM	ball	ACC	hand

“Francis hands a ball”

(6)	田中	に	フランシス	が	渡す
	Tanaka	ni	Furanshisu	ga	watasu
	Tanaka	DAT	Francis	NOM	hand

“Francis hands to Tanaka”

(7)	田中	に	ボール	を	渡す
	Tanaka	ni	bo-ru	wo	watasu
	Tanaka	DAT	ball	ACC	hand

“Hand Tanaka a ball”

れる (reru)/られる (rareru)

(8)	食べ	られる
	tabe	rareru
	eat	PASS

(9)	話さ	れる
	hanasa	reru
	speak	PASS

The verbal endings **れる** (reru) and **られる** (rareru) can be used for:

- passive
  - simple
  - adversative
- honorification
- potential

(1) Indicative vs Simple passive

Simple passive is available for transitive/ditransitive verbs and promotes an object to the subject

(10)

田中	が	ご飯	を	食べ	た
Tanaka	ga	gohan	wo	tabe	ta
Tanaka	NOM	gohan	ACC	eat	PAST

"Tanaka ate the rice"

(11)

ご飯	が	田中	に	食べ	られ	た
gohan	ga	Tanaka	ni	tabe	rare	ta
Tanaka	NOM	gohan	DAT	eat	PASS	PAST

"the rice was eaten by Tanaka"

(2) Adversative passive

The passive forms of intransitive verbs and transitive verbs and almost always indicates the event is unfavorable for the subject

(12)

子供	が	親	に	死な	れ	た
kodomo	ga	oya	ni	shina	re	ta
child	NOM	parent	DAT	die	PASS	PAST

passive expression for "the child lost his parent"

(13)

フランシス	が	ご飯	を	田中	に	食べ	られ	た
Furanshisu	ga	gohan	wo	Tanaka	ni	tabe	rare	ta
Francis	NOM	gohan	ACC	Tanaka	DAT	eat	PASS	PAST

"Francis's rice was eaten by Tanaka"

(3) Honorification

(14)	先生	が	ご飯	を	食べ	られ	た
	sensei	ga	gohan	wo	tabe	rarer	ta
	teacher	NOM	rice	ACC	eat	HON	PAST

“The teacher ate the rice”

(4) Potential

(15)	彼	が	ドリアン	を	食べ	られる
	kare	ga	dorian	wo	tabe	rareru
	3SG	NOM	durian	ACC	eat	POT

“He can eat durian”

Full Forest TreeBanker (FFTB)

- A *treebank* is a syntactically annotated corpus of sentences with parse trees
- Full Forest Treebanker (FFTB) (Packard, 2014): a tool for treebanking with DELPH-IN grammars that allows the users to select manually a tree from the "full forest" of possible trees without listing or specifying all analyses in the parsing stage and store it into database for statistical ranking of candidate parses, transfers, and translations
- grammar-based corpus annotation
- test-suite format: `http://compling.hss.ntu.edu.sg/courses/hg7021/testsuites.html`
- DEMO: FFTB with 'mrs' test-suite

Japanese-English machine translation

- Semantic-transfer-based Japanese-to-English machine translation system, built using the LOGON infrastructure
  `https://github.com/delph-in/JaEn`
- The system consists of the two HPSG grammars and one transfer grammar
  - JACY used to parse the Japanese input
  - ERG used for the generation of the English output
  - transfer grammar which transfers the MRS representation produced by JACY into an MRS representation that ERG can generate from

Japanese-English machine translation

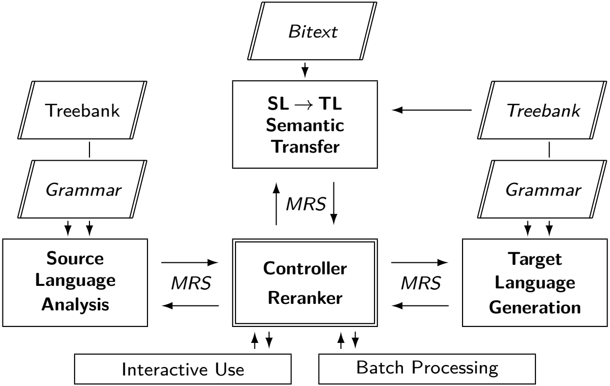


Figure: Architecture of the JaEn MT system.

JaEn DEMO

(16)

雨	が	降る
ame	ga	furu
rain	NOM	fall

“It rains.”

(17)

雨	が	降っ	た
ame	ga	fur	ta
rain	NOM	fall	PAST

“It rained.”

(18)

日本	の	ケーキ	が	あっ	た
nihon	no	keeki	ga	ar	ta
Japan	ADN	cake	NOM	exist	PAST

“There was/were Japanese cake(s).”

Conclusions and Future Work

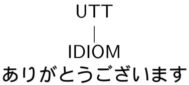
- JACY
  - a broad-coverage Japanese computational grammar
  - uses the framework of Head-driven Phrase Structure Grammar (HPSG) with Minimal Recursion Semantics (MRS)
  - encodes precise morphological, syntactic, semantic, and pragmatic information in feature structures
  - has been developed within many different research projects
  - is being developed in a multilingual context, where much value is placed on parallel and consistent semantic representations
- Future Work
  - will be further adapted to other domains: the newspapers (including the grammar of headline text) and general text such as Wikipedia
  - revise analyses
  - integration with Japanese Wordnet
  - update the treebank

Acknowledgments

- Some slides borrow from Melanie Siegel's presentation slides (`http://www.delph-in.net/jacy/jacy.pdf`)

(19) a. ありがとう ございます
arigatou gozai masu
“Thank you”

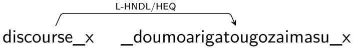
b.



c.

$$\begin{bmatrix} \textit{mrs} & & \\ \text{TOP} & \boxed{0}\ h & \\ \text{INDEX} & \boxed{2}\ i & \\ \text{RELS} & \left\langle \begin{bmatrix} \textit{discourse\_x\_rel} & \\ \text{LBL} & \boxed{4}\ h \\ \text{ARG0} & \boxed{5}\ e \\ \text{L-HNDL} & \boxed{6}\ h \\ \text{R-HNDL} & \boxed{7}\ h \end{bmatrix}, \begin{bmatrix} \textit{\_doumoarigatougozaimasu\_x\_rel} & \\ \text{LBL} & \boxed{6}\ h \\ \text{ARG0} & \boxed{8}\ e \end{bmatrix} \right\rangle & \\ \text{HCONS} & \left\langle \begin{bmatrix} \textit{qeq} & \\ \text{HARG} & \boxed{0}\ h \\ \text{LARG} & \boxed{1}\ h \end{bmatrix} \right\rangle & \end{bmatrix}$$

d.



References I

Emily M. Bender, Scott Drellishak, et al. "Grammar customization". In: *Research on Language and Computation*. Netherlands: Springer, 2010, pp. 23–72.

Emily M. Bender, Dan Flickinger, and Stephan Oepen. "Grammar Engineering and Linguistic Hypothesis Testing: Computational Support for Complexity in Syntactic Analysis". In: *Language from a Cognitive Perspective: Grammar, Usage and Processing*. Stanford: CSLI Publications, 2011, pp. 5–29.

Emily M. Bender, Dan Flickinger, and Stephan Oepen. "The grammar matrix: an open-source starter-kit for the rapid development of cross-linguistically consistent broad- coverage precision grammars". In: *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*. Taipei, 2002, pp. 8–14.

Ulrich Callmeier. "PET - a platform for experimentation with efficient HPSG processing techniques". In: 6.1 (2000), pp. 99–107.

Ann Copestake. *Implementing Typed Feature Structure Grammars*. Stanford: CSLI Publications, 2002.

Ann Copestake et al. "Minimal Recursion Semantics: An Introduction". In: *Research on Language and Computation* 3.4 (2005), pp. 281–332.

References II

Zhenzhen Fan, Sanghoun Song, and Francis Bond. “Building Zhong [|], a Chinese HPSG shared-grammar”. In: (2015).

Dan Flickinger. “Accuracy v. Robustness in Grammar Engineering”. In: *Language from a Cognitive Perspective: Grammar, Usage and Processing*. Ed. by Emily M. Bender and Jennifer E. Arnold. Stanford, CA: CSLI Publications, 2011, pp. 31–50.

Dan Flickinger. “On Building a More Efficient Grammar by Exploiting Types”. In: 6.1 (2000), pp. 15–28.

David Moeljadi, Francis Bond, and Sanghoun Song. “Building an HPSG-based Indonesian Resource Grammar (INDRA)”. In: *Proceedings of the GEAF Workshop, ACL 2015*. 2015, pp. 9–16. URL: http://aclweb.org/anthology/W/W15/W15-3302.pdf.

Stephan Oepen and Daniel Flickinger. “Towards systematic grammar profiling: Test suite technology ten years after”. In: 12.4 (1998), pp. 411–436.

Woodley Packard. *ACE, the Answer Constraint Engine*. 2013. URL: http://sweaglesw.org/linguistics/ace/ (visited on 04/21/2015).

References III

Woodley Packard. *FFTB: the full forest treebanker*. Dec. 2014. URL: http://moin.delph-in.net/FftbTop (visited on 04/24/2015).

Carl Pollard and Ivan A Sag. *Head-driven phrase structure grammar*. University of Chicago Press, 1994.

Ivan A. Sag, Thomas Wasow, and Emily M. Bender. *Syntactic Theory: A Formal Introduction*. 2nd ed. Stanford: CSLI Publications, 2003.

Melanie Siegel, Emily M Bender, and Francis Bond. *Jacy: An implemented grammar of Japanese*. CSLI Publications, 2016.